




Hybrid Fast Sweeping Methods for Anisotropic Eikonal Equation in Two-Dimensional Tilted Transversely Isotropic Media

Guangnan Huang¹ · Songting Luo² 

Received: 12 June 2019 / Revised: 2 July 2020 / Accepted: 7 July 2020
© Springer Science+Business Media, LLC, part of Springer Nature 2020

Abstract

We present hybrid fast sweeping methods for computing first-arrival traveltimes of the qP, qSV and qSH waves in two-dimensional tilted transversely isotropic media, based on solving the anisotropic eikonal equation. A factorization approach is applied to resolve the source singularity near the point source, which leads to a factored anisotropic eikonal equation whose solutions can be computed with high accuracy. The proposed methods solve the factored equation in a neighborhood of the point source with the size of the neighborhood independent of the mesh, and solve the original equation outside the neighborhood. The methods enjoy all the appealing features, such as efficiency, accuracy and convergence, of the usual fast sweeping method. Furthermore, the “super-convergence” property of the first-order fast sweeping method, i.e., both its numerical solution and gradient are first-order accurate, allows us to design a second-order fast sweeping method based on a linear discontinuous Galerkin formulation. As a post-processing procedure of the first-order method, the second-order method reduces the local degrees of freedom from three to one in the linear discontinuous Galerkin formulation, which implies a simple local updating formula, hence an efficient second-order scheme. Numerical experiments are presented to demonstrate the proposed methods.

Keywords TTI eikonal equation · Hybrid fast sweeping method · Source singularity · Factorization approach · Super-convergence

Mathematics Subject Classification 65N06 · 86-08

GH is partially supported by National Natural Science Foundation of China (41504095). SL is partially supported by NSF DMS 1418908 and 1719907.

✉ Songting Luo
luos@iastate.edu
Guangnan Huang
bobking2@126.com

¹ Department of Geophysics, East China University of Technology, Nanchang 330013, China

² Department of Mathematics, Iowa State University, Ames, IA 50011, USA

1 Introduction

Anisotropy has been widely observed and investigated by geophysicists in the crust, the upper mantle and the inner core of the Earth [40,60,62]. For example, the inner core is a complex asymmetric structure with strongly anisotropic property [18], and the anisotropic strength of rock in the upper mantle depends on the percentage of different minerals. The kinematic and dynamic features of the seismic wave are very different when it propagates in isotropic and anisotropic media. In isotropic media, there are only compressional wave and shear wave. The phase velocity is equal to the group velocity for compressional and shear waves. While in anisotropic media, which according to the orientational angle of the symmetric axis can be divided into vertical transversely isotropic (VTI), horizontal transversely isotropic (HTI) and tilted transversely isotropic (TTI) media, there are three different kinds of wavemodes, i.e., one quasi-compressional wave (qP wave) and two quasi-shear waves (qSV and qSH waves). Each wavemode propagates with its own wavespeed and polarization direction. Unlike in isotropic media, the phase and group velocities of each wavemode are not equivalent to each other anymore. And both velocities depend on not only the elastic moduli parameters, but also the seismic wave direction [12].

The anisotropy makes it challenging to compute the first-arrival traveltimes of the seismic wave efficiently. Many Eulerian eikonal solvers have been developed to compute the first-arrival traveltimes. They were first introduced to compute the traveltimes in isotropic media; for instance see [8,24,31,43,50,53,54,57,68,69]. And some of them were extended to deal with the eikonal equation in anisotropic media; for instance see [1,16,20,30,32,48,49]. Among all the Eulerian methods, the fast marching method (FMM) and the fast sweeping method (FSM) are two appealing Eulerian eikonal solvers; see [7,15,17,28,29,31,37,46,47,50,52,54,56–59,65–67,74–76] and references therein. These methods have been applied in various applications due to their efficiency. Both methods use upwinding finite difference approximation to discretize the eikonal equation and enforce a causality condition to pick the correct solution from all possible solutions of the discretized equation that corresponds to the first-arrival traveltimes. The FMM advances the wavefront step by step and uses a heap-sort algorithm to update the solution for points in the wavefront until there are no more points to update. The FSM uses Gauss–Seidel iteration with alternating sweeping orderings to solve the coupled system of nonlinear equations until convergence. In general, the FMM has complexity $O(N \log N)$ and the FSM has complexity $O(N)$, with N the number of simulation points. However, they may suffer from the source singularity due to the non-differentiability of the solution at point sources [45]. The source singularity induces large errors at the source which will further spread to the whole domain and make the traveltimes inaccurate. Without any treatment of the source singularity, such methods, even high-order methods, have only $O(h \log h)$ order of convergence with h the mesh size, which poses a problem to calculate some quantities involving derivatives of the traveltimes, such as take-off angles and geometric spreading factors [5]. Different techniques have been proposed to treat the source singularity [21,23,31,42,44,51,57,71,72]. Among them, the factorization approach proposed in [21,36,37,39,42,72] is the most promising technique that can treat the source singularity efficiently.

The factorization approach was first proposed for the isotropic eikonal equation and anisotropic eikonal equation with weak anisotropy [21,26,36,37,39,42,72]. The eikonal is factored into two factors. One factor is a known function that captures the singularity around the source such that the other factor is smooth near the source. The smooth factor satisfies a modified/factored equation that can be solved efficiently with high accuracy. Hence the

eikonal is recovered with high accuracy. The factorization technique has been successfully applied in solving the isotropic eikonal equation, the eikonal equation in the tilted elliptic anisotropic (TEA) media, and the anisotropic eikonal equation for the qP wave in TTI media based on a simplified formulation, with point-source conditions. For example, Luo et al. [35–37,39] extended this factorization method to higher-order schemes to calculate first-arrival traveltimes and amplitudes; Treister and Haber [63,64] used the first- and second-order finite-difference schemes in the fast marching method to solve the factored eikonal equation; Luo and Qian [37,38] gave a systematic procedure to obtain analytical approximations for the known factor that captures the source singularity, and extended the factorization approach to eikonal equation in the TEA media; following this approach, Waheed et al. [6], Tavakoli et al. [61], and Waheed and Alkhalifah [5] proposed an iterative factored eikonal solver for computing the first-arrival traveltimes of the qP wave in TTI media with a simplified formulation of the anisotropic eikonal equation; and Bouteiller et al. [70] extended the factorization approach to a high-order method in the framework of discontinuous Galerkin method by transforming the simplified anisotropic eikonal equation into a time-dependent equation.

In this work, in order to compute the first-arrival traveltimes for all the three wavemodes, i.e., the qP, qSV and qSH waves, in general TTI media, we intend to solve the anisotropic eikonal equation in the media without simplification. We further investigate the fast sweeping methods for the anisotropic eikonal equation in two-dimensional (2-D) TTI media. The factorization approach will be applied to resolve the source singularity near the source, where a factored anisotropic eikonal equation will be solved in a neighborhood of the source point. The size of this neighborhood is independent of the mesh size, and outside this neighborhood the original anisotropic eikonal equation will be solved. This hybrid approach was first proposed in [38] for solving the isotropic eikonal equation, aiming to improve the efficiency compared to solving only the factored equation in the whole domain. Here we extend it to solve the anisotropic eikonal equation in general TTI media, which aims to improve the efficiency of the fast sweeping method recently developed for solving the equation [25]. The local solver of the fast sweeping method at each grid point requires solving a highly nonlinear equation for updating the traveltimes, for which we design a systematic way to find the solutions [25]. Firstly, the interval that contains all possible solutions of the nonlinear equation is determined based on Fermat's principle. Secondly, this interval is partitioned into subintervals such that each subinterval contains exactly one solution, and the partitioning points for this interval can be pre-determined such that its information can be repeatedly used in the iterations. Thirdly, efficient root-finding algorithms, such as the improved Illinois algorithms [22], are applied to find the solution from each subinterval. And finally, a causality condition is enforced to determine whether a solution is a valid candidate to update the traveltimes, and the minimum of all candidates will be chosen to update the traveltimes. The proposed method will enjoy all the desired properties of the usual fast sweeping method [25,75], such as the efficiency, accuracy and convergence due to its monotonicity. Moreover, we observe that the numerical solution will enjoy the so-called "super-convergence" property that has been observed in cases with isotropic eikonal equation and TEA eikonal equation [3,34,35], i.e., both the numerical solution and its numerical gradient are first-order accurate if the solution is smooth. This property allows us to design a second-order fast sweeping method that is based on a discontinuous Galerkin (DG) formulation for the anisotropic eikonal equation. The DG formulation was first used in [11,33,35,73] to design fast sweeping methods for the isotropic eikonal equation. We will extend it to the anisotropic eikonal equation in general 2-D TTI media, where the DG formulation based on linear polynomial approximation will be derived for the factored anisotropic eikonal equation in the fixed neighborhood of the point source, and for the original anisotropic eikonal equation outside of this neighborhood. The linear DG

formulation has three local degrees of freedom to determine the solution and its gradient, which requires to solve a system of three nonlinear equations. And a sophisticated local solver has to be designed to solve this nonlinear system to obtain a proper solution that corresponds to the first-arrival traveltime. With the “super-convergence” property, we can reduce the degrees of freedom to one by fixing the gradient of the solution with that computed by the first-order method. Consequently, the only degree of freedom is the solution, which can be computed through a simple local updating formula. With the “super-convergence” property, the complexity of the method is significantly reduced compared to the original linear DG formulation [33,73]. The resulting method will be a second-order accurate fast sweeping method.

The rest of the paper is organized as follows. The anisotropic eikonal equation for general TTI media, the factorization techniques, and the factored anisotropic eikonal equation are introduced in Sect. 2. The numerical schemes in the framework of the fast sweeping method for computing the first-arrival traveltime of the three wavemodes are presented in Sect. 3. A few anisotropic models, including benchmark models from geophysics, are used in the numerical experiments to verify the accuracy and efficiency of the proposed methods, which is summarized in Sect. 4. Conclusive remarks will be given at the end.

2 Anisotropic Eikonal Equation

According to Hooke’s law, stress σ_{ij} is related to strain e_{kl} by a stiffness tensor C_{ijkl} ,

$$\sigma_{ij} = \sum_{kl} C_{ijkl} e_{kl}.$$

By the symmetry of the stiffness tensor, the elastic wave equation without body force is

$$\rho \frac{\partial^2 U_j}{\partial t^2} = \sum_{i,k,l} \frac{\partial}{\partial x_i} \left(C_{ijkl} \frac{\partial U_k}{\partial x_l} \right),$$

where ρ is the mass density and $\mathbf{U} = \{U_i\}$ is the displacement vector. High-frequency approximation to the elastic wave equation leads to the Christoffel equation [9,10],

$$\left(\sum_{i,l} a_{ijkl} p_i p_l - \delta_{jk} \right) U_k = 0, \quad (1)$$

where $a_{ijkl} = C_{ijkl}/\rho$ are the density-normalized parameters, δ_{jk} is the Kronecker delta, $\mathbf{p} = \{p_i\} = \nabla T$ is the slowness vector, and T is the traveltime or the phase of the mode. The Christoffel equation implies the eikonal equation for the phase T ,

$$\det \left(\sum_{i,l} a_{ijkl} p_i p_l - \delta_{jk} \right) = 0. \quad (2)$$

By introducing $\mathbf{p} = \mathbf{n}/v$, where \mathbf{n} is the unit normal vector to the wavefront and v is the normal or phase velocity of the wavefront, one can derive the anisotropic eikonal equation,

$$v|\nabla T| = 1, \quad \det \left(\sum_{i,l} a_{ijkl} n_i n_l - v^2 \delta_{jk} \right) = 0. \quad (3)$$

A general tilted transversely isotropic (TTI) medium can be defined by five elastic moduli parameters (by using Voigt's notation) $\{C_{11}(\mathbf{x}), C_{13}(\mathbf{x}), C_{33}(\mathbf{x}), C_{44}(\mathbf{x}), C_{66}(\mathbf{x})\}$, the mass density $\rho(\mathbf{x})$, and the orientational angle $\theta_0(\mathbf{x})$ of the symmetric axis of the medium [9, 10, 77]. θ_0 is formed by the vertical axis and the symmetric axis of the medium. For the three wavemodes in the medium, i.e., qP, qSV and qSH waves, by introducing $v_m = v\rho$ ($m = 1, 2, 3$) that correspond to the three wavemodes, the anisotropic eikonal equation can be recast as

$$v_m |\nabla T| = \rho, \quad (m = 1, 2, 3), \quad (4)$$

where v_m ($m = 1, 2, 3$) are defined as [4, 14, 77],

$$\begin{aligned} v_{1,2} &= \sqrt{\mathcal{M} \pm \sqrt{\mathcal{M}^2 - \mathcal{N}}}, \\ v_3 &= \sqrt{C_{44} + (C_{66} - C_{44}) \sin^2 \zeta}, \end{aligned} \quad (5)$$

with

$$\begin{aligned} \mathcal{M} &= \frac{\mathcal{Q}_1 + \mathcal{Q}_2}{2}, \\ \mathcal{N} &= \mathcal{Q}_1 \mathcal{Q}_2 - \mathcal{Q}_3, \\ \mathcal{Q}_1 &= C_{44} \cos^2 \zeta + C_{11} \sin^2 \zeta, \\ \mathcal{Q}_2 &= C_{33} \cos^2 \zeta + C_{44} \sin^2 \zeta, \\ \mathcal{Q}_3 &= \frac{(C_{13} + C_{44})^2 \sin^2 2\zeta}{4}. \end{aligned}$$

Here, the angle ζ is formed by the phase slowness direction \mathbf{n} and the symmetric axis of the medium, with $\zeta = \theta - \theta_0$ and θ the orientational angle of \mathbf{n} from the vertical axis.

We intend to solve the anisotropic eikonal equation (4) in 2-D TTI media, in the sense of viscosity solution [13], as we want to compute first-arrival traveltimes of the three wavemodes. In 2-D, θ is defined as

$$\theta = \cos^{-1} \left(\frac{T_z}{\sqrt{T_x^2 + T_z^2}} \right),$$

with $(T_x, T_z) \equiv \nabla T$. Clearly, v_m ($m = 1, 2, 3$) are functions of ∇T , i.e., $v_m = v_m(T_x, T_z)$, and can also be viewed as functions of θ , i.e., $v_m = v_m(\theta)$.

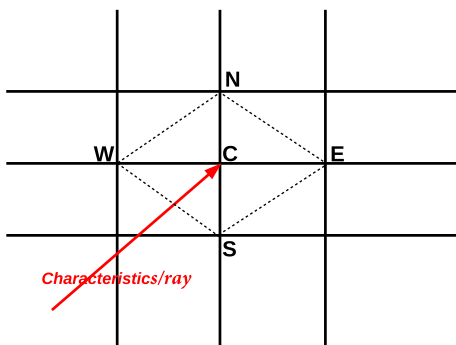
One of the main difficulties for solving the anisotropic eikonal equation (4) with a point-source condition is the source singularity if high accuracy is required [44]. The source singularity induces large errors for any Eulerian eikonal solvers, including high-order methods, near the source, and the error will spread to the whole domain. As a result, the numerical solution exhibits $O(h \log h)$ order of convergence with h the mesh size. In order to resolve this issue, we use the factorization approach [21, 36, 37, 39, 42, 72]. We assume the traveltime T is decomposed as

$$T = T_0 + \tau, \quad (6)$$

where T_0 is a known factor that captures the source singularity, and τ is a smooth factor near the source that serves as the smooth correction. τ satisfies the factored anisotropic eikonal equation

$$v_m |\nabla T_0 + \nabla \tau| = \rho, \quad (m = 1, 2, 3), \quad (7)$$

Fig. 1 Local mesh of an interior point C with four neighboring points E , W , N , and S



where v_m is defined in (5) with

$$\theta = \cos^{-1} \left(\frac{T_{0z} + \tau_z}{\sqrt{(T_{0x} + \tau_x)^2 + (T_{0z} + \tau_z)^2}} \right),$$

with $(T_{0x}, T_{0z}) \equiv \nabla T_0$ and $(\tau_x, \tau_z) \equiv \nabla \tau$. Since τ is smooth near the source, the factored Eq. (7) can be solved efficiently for τ with high accuracy. Hence T can be recovered with high accuracy through (6).

For numerically solving Eqs. (4) and (7), we present the hybrid fast sweeping methods.

3 Hybrid Fast Sweeping Methods

We first present the method for Eqs. (4) and (7) with first-order accuracy, and then extend it to second-order accuracy with linear discontinuous Galerkin formulations. We assume the domain is $\Omega = [x_{\min}, x_{\max}] \times [z_{\min}, z_{\max}]$, which is covered by a uniform mesh (x_i, z_j) with $x_i = x_{\min} + i \Delta x$, ($i = 0, 1, 2, \dots, N_x$), $z_j = z_{\min} + j \Delta z$, ($j = 0, 1, 2, \dots, N_z$), $\Delta x = (x_{\max} - x_{\min})/N_x$, and $\Delta z = (z_{\max} - z_{\min})/N_z$. And we denote $f_{ij} \equiv f(x_i, z_j)$ for any function f .

3.1 First-Order Hybrid FSM

The local solver of the FSM at each grid point requires to discretize the equation with upwinding finite difference approximations and a causality condition to choose valid candidates for updating the value at the grid point.

Figure 1 shows an interior grid point C with four neighboring points W , E , N and S . The anisotropic eikonal equation (4) needs to be discretized on the four triangles: $\triangle CEN$, $\triangle CNW$, $\triangle CWS$ and $\triangle CSE$. For example on $\triangle CWS$, it is discretized as

$$H(P, Q) \equiv v_m(P, Q)|(P, Q) - \rho(C)| = 0, \quad (8)$$

with

$$P = T_x(C) \approx \frac{T(C) - T(W)}{\Delta x}, \quad Q = T_z(C) \approx \frac{T(C) - T(S)}{\Delta z}.$$

This discretized equation (8) must be solved for $T(C)$, given the two neighboring values $T(W)$ and $T(S)$, and it may have multiple or no solutions for $T(C)$:

- If there are solutions for $T(C)$, each of them is required to satisfy a **causality condition**, i.e., the characteristics/ray passing through C falls in ΔCWS (Fig. 1). If a solution satisfies the causality condition, it become a candidate for updating $T(C)$. The characteristics/ray is defined as (H_P, H_Q) with

$$H_P \equiv \frac{\partial H}{\partial P} = \frac{P}{\sqrt{P^2 + Q^2}} v_m - \frac{Q}{\sqrt{P^2 + Q^2}} \frac{dv_m}{d\theta},$$

$$H_Q \equiv \frac{\partial H}{\partial Q} = \frac{Q}{\sqrt{P^2 + Q^2}} v_m + \frac{P}{\sqrt{P^2 + Q^2}} \frac{dv_m}{d\theta},$$

where P , Q , θ and v_m are computed with the solution $T(C)$. For ΔCWS , the causality condition requires $H_P \geq 0$ and $H_Q \geq 0$.

Among all candidates that satisfy the causality condition, we choose the minimum one as the possible update for $T(C)$ from ΔCWS , corresponding to the first-arrival traveltime.

- If there are no solutions for $T(C)$ or no solutions for $T(C)$ that satisfy the causality condition, we assume the characteristics/ray falls on either the edge \overline{WC} or the edge \overline{SC} , and the possible update for $T(C)$ from ΔCWS is given as

$$\min \left\{ T(W) + \frac{\Delta x}{U_m^{WC}/\rho(C)}, T(S) + \frac{\Delta z}{U_m^{SC}/\rho(C)} \right\},$$

where $U_m^{WC}/\rho(C)$ and $U_m^{SC}/\rho(C)$ ($m = 1, 2, 3$) are the group velocity of the wave along edge \overline{WC} and \overline{SC} , respectively. The definition and computation of U_m^{WC} and U_m^{SC} will be discussed in Sect. 3.1.2.

This process is repeated along all the four triangles. Each of the triangles will provide a possible update for $T(C)$, and the minimum one from the four triangles is chosen to update $T(C)$, which corresponds to the first-arrival traveltime.

In the local solver, there are two key issues: (i) the discretized equation (8), which is highly nonlinear in $T(C)$, needs to be solved; and (ii) U_m along a given edge or characteristics/ray, such as U_m^{WC} and U_m^{SC} , is required.

3.1.1 Solving Discretized Equation (8)

This discretized equation (8) is highly nonlinear in $T(C)$, and it may have multiple solutions. For finding its roots, we proceed as follows [25]:

- Determine the interval that contains all possible solutions,
- Partition the interval into subintervals such that each subinterval contains at most one solution,
- Apply efficient root-finding algorithms, such as the improved Illinois algorithms [22], on each subinterval to find the solution in it, hence all the solutions in the interval.

The interval that contains all possible solutions for $T(C)$, for example for ΔCWS , is given as

$$I^{CWS} = \left[\min\{T(W), T(S)\}, \min \left\{ T(W) + \frac{\Delta x}{U_m^{WC}/\rho(C)}, T(S) + \frac{\Delta z}{U_m^{SC}/\rho(C)} \right\} \right],$$

following the Fermat's principle.

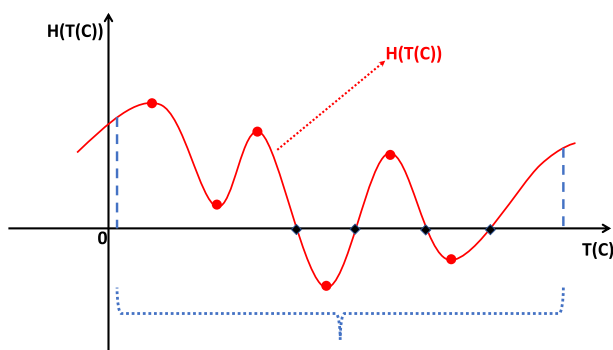


Fig. 2 Demonstration of partitioning I^{CWS} into subintervals with extreme points of $H(T(C))$: interval between two vertical dashed lines is I^{CWS} , red dots are extreme points of $H(T(C))$, and black diamonds are zeros of $H(T(C))$ (Color figure online)

For partitioning I^{CWS} , we treat H in (8) as a function of $T(C)$,

$$\begin{aligned} H(T(C)) &\equiv H\left(\frac{T(C) - T(W)}{\Delta x}, \frac{T(C) - T(S)}{\Delta z}\right) \\ &= v_m\left(\frac{T(C) - T(W)}{\Delta x}, \frac{T(C) - T(S)}{\Delta z}\right) \sqrt{\left(\frac{T(C) - T(W)}{\Delta x}\right)^2 + \left(\frac{T(C) - T(S)}{\Delta z}\right)^2} - \rho(C). \end{aligned}$$

In order to find zeros $T(C)$ of $H(T(C))$ on I^{CWS} , we will use the extreme points of $H(T(C))$ to partition I^{CWS} into subintervals such that each subinterval contains at most one zero, referring to Fig. 2. One can find its extreme points by solving

$$\frac{dH}{dT(C)} = \frac{\partial H}{\partial P} \frac{\partial P}{\partial T(C)} + \frac{\partial H}{\partial Q} \frac{\partial Q}{\partial T(C)} = 0,$$

for critical points, which leads to

$$\left(\frac{P}{\sqrt{P^2 + Q^2}} v_m - \frac{Q}{\sqrt{P^2 + Q^2}} \frac{dv_m}{d\theta}\right) \frac{1}{\Delta x} + \left(\frac{Q}{\sqrt{P^2 + Q^2}} v_m + \frac{P}{\sqrt{P^2 + Q^2}} \frac{dv_m}{d\theta}\right) \frac{1}{\Delta z} = 0.$$

With the definition of θ , the above equation can be recast as

$$\left(\sin(\theta) v_m(\theta) - \cos(\theta) \frac{dv_m(\theta)}{d\theta}\right) \frac{1}{\Delta x} + \left(\cos(\theta) v_m(\theta) + \sin(\theta) \frac{dv_m(\theta)}{d\theta}\right) \frac{1}{\Delta z} = 0. \quad (9)$$

Note that it can be viewed as an equation in θ , i.e., $F(\theta) = 0$, with $F(\theta)$ the left-hand side of (9). We can first apply a root-finding algorithm, such as the false position method and its variants, to compute its roots, without loss of generality, denoted as θ_k for $k = 1, \dots, K$, and then use $\{\theta_k\}$ to find extreme points of H , denoted as $\{T_k\}$, through the following relation based on the definition of θ ,

$$\tan(\theta_k) = \frac{(T_k - T(W))/\Delta x}{(T_k - T(S))/\Delta z}, \quad k = 1, \dots, K.$$

Moreover, the roots $\{\theta_k\}$ of (9) can be pre-determined, and repeatedly used to locate $\{T_k\}$ during the iterations in the local solver. Once the partitioning points are determined and the interval I^{CWS} is partitioned into subintervals (Fig. 2), we can apply appropriate root-finding

algorithms to find zeros of $H(T(C))$ subinterval by subinterval, for example, the improved Illinois algorithms [22] are used. Hence, we find possible candidates for updating $T(C)$ at point C , and this process is repeated for all the four triangles related to C .

Remark 1 For each point C , we can save the roots of Eq. (9) for all the four triangles associated with C . Then for all the points on the mesh, we have a table of θ_k 's, which is the reference table that is repeatedly read and used to compute the partitioning points T_k 's in the local solver.

3.1.2 Computing U_m along a Characteristic/Ray Direction

U_m ($m = 1, 2, 3$) are in general defined by [4, 14, 77],

$$U_m = \sqrt{v_m^2 + \left(\frac{dv_m}{d\theta}\right)^2}, \quad m = 1, 2, 3, \quad (10)$$

with

$$\begin{aligned} \frac{dv_{1,2}}{d\theta} &= \frac{1}{2v_{1,2}} \left(\frac{d\mathcal{M}}{d\theta} \pm \frac{\mathcal{M} \frac{d\mathcal{M}}{d\theta} - \frac{1}{2} \frac{d\mathcal{N}}{d\theta}}{\sqrt{\mathcal{M}^2 - \mathcal{N}^2}} \right), \\ \frac{dv_3}{d\theta} &= \frac{C_{66} - C_{44}}{2v_3} \sin 2\zeta, \\ \frac{d\mathcal{M}}{d\theta} &= \frac{C_{11} - C_{33}}{\sin 2\zeta}, \\ \frac{d\mathcal{N}}{d\theta} &= [\mathcal{Q}_1(C_{44} - C_{33}) + \mathcal{Q}_2(C_{11} - C_{44})] \sin 2\zeta - \frac{(C_{44} + C_{13})^2}{2} \sin 4\zeta. \end{aligned}$$

Clearly, U_m ($m = 1, 2, 3$) are functions of θ that is the orientational angle of the slowness vector instead of the orientational angle of the characteristics/ray. Therefore, in order to find U_m along a given characteristics/ray, such as U_m^{WC} and U_m^{SC} , we will need to find θ along the characteristic/ray direction first, and then compute it with (10). To achieve the goal, we follow the procedure in [10] for computing U_m with a given characteristic/ray direction.

According to Cerveny [10], the slowness vector $\mathbf{p}_m \equiv \mathbf{n}/v = \rho/v_m(\sin \theta, \cos \theta)$ and the group velocity vector $\mathbf{U}_m \equiv U_m/\rho(N_1, N_3)$ satisfy the following relation,

$$\mathbf{p}_m \cdot \mathbf{U}_m = 1,$$

where (N_1, N_3) is the characteristic/ray direction. This relation implies the following equation,

$$G(\theta) \equiv \sin(\theta)N_1 + \cos(\theta)N_3 - \frac{v_m(\theta)}{U_m(\theta)} = 0, \quad (11)$$

which will be used to compute U_m as well as v_m along a given characteristic/ray direction (N_1, N_3) . It is also an equation in θ , for which we can apply a root-finding algorithm, such as the false position method and its variants, to compute its roots, and use the roots to compute U_m and v_m . And among all possible roots, we will choose the one that gives the minimum U_m . Moreover, its roots can be pre-determined and repeatedly used in the local solver.

Remark 2 For each point C , the roots of Eq. (11) are pre-computed along the four edges connecting C , as well as the corresponding v_m and U_m . For all the points on the mesh, we have a reference table that consists of v_m , U_m and associated θ along four edges of each

point. This table is repeatedly read and used in the local solver if the update of the numerical solution must follow one of the edges.

The discretized equation (8) at all the grid points are coupled together as a system of nonlinear equations. This system can be solved by Gauss–Seidel iterations with alternative orderings, where the local solver at each point is described as above. This is the fast sweeping method [75].

We can apply the same procedure to the factored anisotropic eikonal equation (7). On ΔCWS , the factored equation is discretized as

$$\bar{H}(p, q) \equiv v_m(T_{0x}(C) + p, T_{0z}(C) + q) |(T_{0x}(C) + p, T_{0z}(C) + q)| - \rho(C) = 0, \quad (12)$$

with

$$p = \tau_x(C) \approx \frac{\tau(C) - \tau(W)}{\Delta x}, \quad q = \tau_z(C) \approx \frac{\tau(C) - \tau(S)}{\Delta z}.$$

This equation must be solved for $\tau(C)$, given the two neighboring values $\tau(W)$ and $\tau(S)$:

- If there are solutions for $\tau(C)$, each of them is also required to satisfy the causality condition with the characteristics/ray defined as (\bar{H}_p, \bar{H}_q) ,

$$\begin{aligned} \bar{H}_p &\equiv \frac{\partial \bar{H}}{\partial p} = \frac{T_{0x} + p}{\sqrt{(T_{0x} + p)^2 + (T_{0z} + q)^2}} v_m - \frac{T_{0z} + q}{\sqrt{(T_{0x} + p)^2 + (T_{0z} + q)^2}} \frac{dv_m}{d\theta}, \\ \bar{H}_q &\equiv \frac{\partial \bar{H}}{\partial q} = \frac{T_{0z} + q}{\sqrt{(T_{0x} + p)^2 + (T_{0z} + q)^2}} v_m + \frac{T_{0x} + p}{\sqrt{(T_{0x} + p)^2 + (T_{0z} + q)^2}} \frac{dv_m}{d\theta}. \end{aligned}$$

On ΔCWS , the causality condition is equivalent to $\bar{H}_p \geq 0$ and $\bar{H}_q \geq 0$.

Among all candidates that satisfy the causality condition, we choose the minimum one as the possible update for $\tau(C)$ from ΔCWS , corresponding to the first-arrival traveltime.

- If there are no solutions for $\tau(C)$ or no solutions for $\tau(C)$ that satisfy the causality condition, we assume the characteristics/ray falls on either the edge \overline{WC} or the edge \overline{SC} , and the possible update for $\tau(C)$ from ΔCWS is given as

$$\min \left\{ \tau(W) + \frac{\Delta x}{U_m^{WC}/\rho(C)} - (T_{0x}(C)\Delta x), \tau(S) + \frac{\Delta z}{U_m^{SC}/\rho(C)} - (T_{0z}(C)\Delta z) \right\}.$$

This process is repeated along all the four triangles. Each of the triangles will provide a possible update for $\tau(C)$, and the minimum one from the four triangles is chosen to update $\tau(C)$, which corresponds to the first-arrival traveltime.

In the local solver, U_m along a given edge or characteristics/ray, such as U_m^{WC} and U_m^{SC} , will be computed exactly as in Sect. 3.1.2, and the discretized equation (12) will be solved following the procedure in Sect. 3.1.1. Furthermore, T_0 , T_{0x} and T_{0z} are needed in the local solver, for which we present the algorithm in Sect. 3.1.4.

3.1.3 Solving Discretized Equation (12)

The interval that contains all possible solutions for $\tau(C)$ in (12), for example on ΔCWS , is given as

$$\begin{aligned} J^{CWS} &= [\min\{T_0(W) + \tau(W), T_0(S) + \tau(S)\} - T_0(C), \\ &\quad \min \left\{ \tau(W) + \frac{\Delta x}{U_m^{WC}/\rho(C)} - (T_{0x}(C)\Delta x), \tau(S) + \frac{\Delta z}{U_m^{SC}/\rho(C)} - (T_{0z}(C)\Delta z) \right\}] \end{aligned}$$

following the Fermat's principle.

For partitioning J^{CWS} , we treat \bar{H} in (12) as a function of $\tau(C)$ and find its extreme points by solving

$$\frac{d\bar{H}}{d\tau(C)} = \frac{\partial \bar{H}}{\partial p} \frac{\partial p}{\partial \tau(C)} + \frac{\partial \bar{H}}{\partial q} \frac{\partial q}{\partial \tau(C)} = 0,$$

for critical points, which leads to

$$\left(\frac{T_{0x} + p}{\sqrt{(T_{0x} + p)^2 + (T_{0z} + q)^2}} v_m - \frac{T_{0z} + q}{\sqrt{(T_{0x} + p)^2 + (T_{0z} + q)^2}} \frac{dv_m}{d\theta} \right) \frac{1}{\Delta x} + \left(\frac{T_{0z} + q}{\sqrt{(T_{0x} + p)^2 + (T_{0z} + q)^2}} v_m + \frac{T_{0x} + p}{\sqrt{(T_{0x} + p)^2 + (T_{0z} + q)^2}} \frac{dv_m}{d\theta} \right) \frac{1}{\Delta z} = 0,$$

and can be recast as Eq. (9). Therefore, its solutions, θ_k for $k = 1, \dots, K$, can be used to find extreme points of \bar{H} , denoted as τ_k for $k = 1, \dots, K$, through the following relation,

$$\tan(\theta_k) = \frac{(\tau_k - \tau(W))/\Delta x + T_{0x}(C)}{(\tau_k - \tau(S))/\Delta z + T_{0z}(C)}.$$

Once the partitioning points are determined, we can apply efficient root-finding algorithms, such as the improved Illinois algorithms [22], to find the roots of (12) subinterval by subinterval.

3.1.4 Computing T_0 , T_{0x} and T_{0z}

We will choose T_0 to be the solution of the anisotropic eikonal equation (4) with constant coefficients. Assume the point source is \mathbf{x}_0 , we choose T_0 to satisfy

$$v_m |\nabla T_0(\mathbf{x})| = \rho(\mathbf{x}_0), \quad (13)$$

where the elastic moduli parameters are given by $\{C_{11}(\mathbf{x}_0), C_{13}(\mathbf{x}_0), C_{33}(\mathbf{x}_0), C_{44}(\mathbf{x}_0), C_{66}(\mathbf{x}_0)\}$, and the tilted angle is $\theta_0(\mathbf{x}_0)$. Then we have

$$T_0(\mathbf{x}) = \frac{|\mathbf{x} - \mathbf{x}_0|}{U_m / \rho(\mathbf{x}_0)},$$

where $U_m / \rho(\mathbf{x}_0)$ is the group velocity along the characteristic/ray direction $(\mathbf{x} - \mathbf{x}_0) / |\mathbf{x} - \mathbf{x}_0|$. U_m can be computed as in Sect. 3.1.2, as well as v_m , hence T_0 can be computed.

Furthermore, from Eq. (13), we have

$$T_{0x} = \frac{\rho(\mathbf{x}_0)}{v_m} \frac{T_{0x}}{\sqrt{T_{0x}^2 + T_{0z}^2}} = \frac{\rho(\mathbf{x}_0)}{v_m} \sin(\theta),$$

$$T_{0z} = \frac{\rho(\mathbf{x}_0)}{v_m} \frac{T_{0z}}{\sqrt{T_{0x}^2 + T_{0z}^2}} = \frac{\rho(\mathbf{x}_0)}{v_m} \cos(\theta).$$

Hence we have T_{0x} and T_{0z} .

With the local solvers described above, we can solve the factored anisotropic eikonal equation (7) in a neighborhood of the source \mathbf{x}_0 , denoted as Ω_0 , and solve the original anisotropic eikonal equation (4) outside this neighborhood, denoted as $\Omega_0^c \equiv \Omega - \Omega_0$. The size of Ω_0 is chosen to be independent of the mesh as the mesh size approaches zero.

The first-order hybrid fast sweeping method is summarized as follows.

Algorithm 1 (First-Order Hybrid FSM)

1. *Initialization: assign exact/approximate values at grid points according to given boundary condition, which will be fixed during iterations, and assign large positive values for all other grid points.*
2. *Gauss–Seidel iteration: sweep the mesh with four alternating orderings iteratively:*

$$\begin{aligned} (1) \quad & i = 0 : N_x, j = 0 : N_z, \quad (2) \quad i = 0 : N_x, j = N_z : 0, \\ (3) \quad & i = N_x : 0, j = 0 : N_z, \quad (4) \quad i = N_x : 0, j = N_z : 0. \end{aligned}$$

At each grid point (x_i, z_j) ,

- if it is in Ω_0 , use the local solver for the factored anisotropic eikonal equation (7) to compute $\tau(C)$, and $T(C) = \tau(C) + T_0(C)$.
 - if it is in Ω_0^c , use the local solver for the anisotropic eikonal equation (4) to compute $T(C)$, and $\tau(C) = T(C) - T_0(C)$ if necessary.
3. *Termination: if the l_1 norm difference of the solutions between two successive iterations is smaller than the specified accuracy requirement or the number of iterations exceeds the prescribed maximum number of iterations, terminate the iteration.*

In Algorithm 1, the source singularity is resolved with the factorization approach. Consequently, the method will have clean first-order accuracy, which will be demonstrated in Sect. 4. And since the causality condition is imposed, the method is monotone [38,46,47,75] such that its numerical solution will converge to the desired viscosity solution corresponding to the first-arrival traveltimes [2].

Furthermore, the numerical solution has the so-called “super-convergence” property that has been observed previously for isotropic eikonal equation, i.e., its numerical gradient is first-order accurate too. This property can be used to design an efficient second-order fast sweeping method based on a linear DG formulation [11,33,35,73]. We will apply the similar DG formulation to design a second-order FSM for the anisotropic eikonal equation.

3.2 Second-Order Hybrid FSM

For each grid point (x_i, z_j) , the cell associated with it as the cell center is $\{I_{ij}\} \equiv I_i \times I_j$ with $I_i = [x_{i-1/2}, x_{i+1/2}]$ and $I_j = [z_{j-1/2}, z_{j+1/2}]$. And we define the piecewise linear polynomial approximation space as

$$V_h^1 = \{v : v|_{I_{ij}} \in P^1(I_{ij}), \forall i, j\},$$

where $P^1(I_{ij})$ denotes linear polynomials on I_{ij} .

3.2.1 Linear DG Formulation for (4)

Following the formulation in [11,33,35,73], a numerical scheme based on the DG formulation for (4) is: finding $T_h \in V_h^1$ such that $\forall w_h \in V_h^1$,

$$\begin{aligned}
& \int_{I_{ij}} v_m |\nabla T_h| w_h(x, z) dx dz \\
& + \alpha_{eij} \int_{I_j} [T_h](x_{i+1/2}, z) w_h(x_{i+1/2}^-, z) dz + \alpha_{wij} \int_{I_j} [T_h](x_{i-1/2}, z) w_h(x_{i-1/2}^+, z) dz \\
& + \alpha_{nij} \int_{I_i} [T_h](x, z_{j+1/2}) w_h(x, z_{j+1/2}^-, z) dx + \alpha_{sij} \int_{I_i} [T_h](x, z_{j-1/2}) w_h(x, z_{j-1/2}^+, z) dx \\
& = \int_{I_{ij}} \rho(x, z) w_h(x, z) dx dz.
\end{aligned} \tag{14}$$

$[T_h]$ denotes the jump of T_h across the cell boundary, and

$$\begin{aligned}
w_h(x_{i+1/2}^-, z) &= \lim_{x \rightarrow x_{i+1/2}^-} w_h(x, z), \quad w_h(x_{i-1/2}^+, z) = \lim_{x \rightarrow x_{i-1/2}^+} w_h(x, z), \\
w_h(x, z_{j+1/2}^-) &= \lim_{z \rightarrow z_{j+1/2}^-} w_h(x, z), \quad w_h(x, z_{j-1/2}^+) = \lim_{z \rightarrow z_{j-1/2}^+} w_h(x, z).
\end{aligned}$$

$\{\alpha_{eij}, \alpha_{wij}\}$ are local constants that approximate $\partial H / \partial T_x$, and $\{\alpha_{nij}, \alpha_{sij}\}$ are local constants that approximate $\partial H / \partial T_z$. The local constants must be chosen wisely such that the causality is enforced correctly.

The linear polynomial T_h is represented as $T_h|_{I_{ij}} = \bar{T}_{ij} + \xi_{ij} X_i + \eta_{ij} Z_j$ with $X_i = (x - x_i) / \Delta x$ and $Z_j = (z - z_j) / \Delta z$. The unknowns on I_{ij} are \bar{T}_{ij} , ξ_{ij} , and η_{ij} , which are approximations of T , $\Delta x T_x$, and $\Delta z T_z$ at (x_i, z_j) , respectively.

In order to enforce the causality properly [11,33,35,73], we choose the local constants as

$$\begin{aligned}
\alpha_{eij} &= \min \left\{ 0, \frac{P_{ij}}{\sqrt{P_{ij}^2 + Q_{ij}^2}} v_m - \frac{Q_{ij}}{\sqrt{P_{ij}^2 + Q_{ij}^2}} \frac{dv_m}{d\theta} \right\}, \\
\alpha_{wij} &= \max \left\{ 0, \frac{P_{ij}}{\sqrt{P_{ij}^2 + Q_{ij}^2}} v_m - \frac{Q_{ij}}{\sqrt{P_{ij}^2 + Q_{ij}^2}} \frac{dv_m}{d\theta} \right\}, \\
\alpha_{nij} &= \min \left\{ 0, \frac{Q_{ij}}{\sqrt{P_{ij}^2 + Q_{ij}^2}} v_m + \frac{P_{ij}}{\sqrt{P_{ij}^2 + Q_{ij}^2}} \frac{dv_m}{d\theta} \right\}, \\
\alpha_{sij} &= \max \left\{ 0, \frac{Q_{ij}}{\sqrt{P_{ij}^2 + Q_{ij}^2}} v_m + \frac{P_{ij}}{\sqrt{P_{ij}^2 + Q_{ij}^2}} \frac{dv_m}{d\theta} \right\},
\end{aligned} \tag{15}$$

with $P_{ij} = \xi_{ij} / \Delta x$, $Q_{ij} = \eta_{ij} / \Delta z$, and $\left\{ v_m, \frac{dv_m}{d\theta} \right\}$ computed with (P_{ij}, Q_{ij}) .

Given cell I_{ij} , by choosing $w_h = 1$, and approximating the integrals in (14) simply by mid-point rule, we can derive a nonlinear equation on $\{\bar{T}_{ij}, \xi_{ij}, \eta_{ij}\}$,

$$v_m \sqrt{(\xi_{ij} / \Delta x)^2 + (\eta_{ij} / \Delta z)^2} \Delta x \Delta z + \gamma_{ij} \bar{T}_{ij} + \beta_{ij} \xi_{ij} + \lambda_{ij} \eta_{ij} = R_{ij}, \tag{16}$$

where

$$\begin{aligned}\gamma_{ij} &= (\alpha_{wij} - \alpha_{eij})\Delta z + (\alpha_{sij} - \alpha_{nij})\Delta x, \\ \beta_{ij} &= -\frac{(\alpha_{wij} + \alpha_{eij})\Delta z}{2}, \\ \lambda_{ij} &= -\frac{(\alpha_{sij} + \alpha_{nij})\Delta x}{2}, \\ R_{ij} &= \rho_{ij}\Delta x\Delta z - \alpha_{eij}(\bar{T}_{i+1,j} - \xi_{i+1,j}/2)\Delta z + \alpha_{wij}(\bar{T}_{i-1,j} + \xi_{i-1,j}/2)\Delta z \\ &\quad - \alpha_{nij}(\bar{T}_{i,j+1} - \eta_{i,j+1}/2)\Delta x + \alpha_{sij}(\bar{T}_{i,j-1} + \eta_{i,j-1}/2)\Delta x.\end{aligned}$$

We may further choose $w_h = X_i$ and Z_j to derive two more equations on $\{\bar{T}_{ij}, \xi_{ij}, \eta_{ij}\}$ such that a system of three nonlinear equations on $\{\bar{T}_{ij}, \xi_{ij}, \eta_{ij}\}$ can be solved to determine them. Here, we utilize the “super-convergence” property [3,34,35]. If the solution T is smooth, the numerical solutions obtained by Algorithm 1, denoted as $\{T^1, T_x^1, T_z^1\}$, are first-order accurate both in its value and gradient, i.e.,

$$|T^1 - T| = O(\Delta x, \Delta z), \quad |T_x^1 - T_x| = O(\Delta x, \Delta z), \quad |T_z^1 - T_z| = O(\Delta x, \Delta z).$$

With this property, we can fix ξ_{ij} and η_{ij} in (16) as

$$\xi_{ij} = T_{xij}^1 \Delta x, \quad \eta_{ij} = T_{zij}^1 \Delta z,$$

which implies a simple updating formula for \bar{T}_{ij} ,

$$\bar{T}_{ij} = \frac{R_{ij} - v_m \sqrt{(\xi_{ij}/\Delta x)^2 + (\eta_{ij}/\Delta z)^2} \Delta x \Delta z - \beta_{ij} \xi_{ij} - \lambda_{ij} \eta_{ij}}{\gamma_{ij}}. \quad (17)$$

3.2.2 Linear DG Formulation for (7)

The same linear DG formulation can be applied for (7): finding $\tau_h \in V_h^1$ such that $\forall w_h \in V_h^1$,

$$\begin{aligned}& \int_{I_{ij}} v_m |\nabla \tau_h + \nabla T_0| w_h(x, z) dx dz \\ & + \alpha_{eij} \int_{I_j} [\tau_h](x_{i+1/2}, z) w_h(x_{i+1/2}^-, z) dz + \alpha_{wij} \int_{I_j} [\tau_h](x_{i-1/2}, z) w_h(x_{i-1/2}^+, z) dz \\ & + \alpha_{nij} \int_{I_i} [\tau_h](x, z_{j+1/2}) w_h(x, z_{j+1/2}^-) dx + \alpha_{sij} \int_{I_i} [\tau_h](x, z_{j-1/2}) w_h(x, z_{j-1/2}^+) dx \\ & = \int_{I_{ij}} \rho(x, z) w_h(x, z) dx dz.\end{aligned} \quad (18)$$

The linear polynomial τ_h is represented as $\tau_h|_{I_{ij}} = \bar{\tau}_{ij} + \mu_{ij}X_i + v_{ij}Z_j$. The unknowns on I_{ij} are $\bar{\tau}_{ij}$, μ_{ij} , and v_{ij} , which are approximations of τ , $\Delta x \tau_x$, and $\Delta z \tau_z$ at (x_i, z_j) , respectively.

In order to enforce the causality properly, we choose the local constants as in (15) with

$$P_{ij} = \mu_{ij}/\Delta x + T_{0xij}, \quad Q_{ij} = v_{ij}/\Delta z + T_{0zij}.$$

Given cell I_{ij} , by choosing $w_h = 1$, and approximating the integrals in (18) simply by mid-point rule, we can derive a nonlinear equation on $\{\bar{\tau}_{ij}, \mu_{ij}, v_{ij}\}$,

$$v_m \sqrt{(\mu_{ij}/\Delta x + T_{0xij})^2 + (v_{ij}/\Delta z + T_{0zij})^2} \Delta x \Delta z + \gamma_{ij} \bar{\tau}_{ij} + \beta_{ij} \mu_{ij} + \lambda_{ij} v_{ij} = R_{ij}, \quad (19)$$

where

$$\begin{aligned} \gamma_{ij} &= (\alpha_{wij} - \alpha_{eij}) \Delta z + (\alpha_{sij} - \alpha_{nij}) \Delta x, \\ \beta_{ij} &= -\frac{(\alpha_{wij} + \alpha_{eij}) \Delta z}{2}, \\ \lambda_{ij} &= -\frac{(\alpha_{sij} + \alpha_{nij}) \Delta x}{2}, \\ R_{ij} &= \rho_{ij} \Delta x \Delta z - \alpha_{eij} (\bar{\tau}_{i+1,j} - \mu_{i+1,j}/2) \Delta z + \alpha_{wij} (\bar{\tau}_{i-1,j} + \mu_{i-1,j}/2) \Delta z \\ &\quad - \alpha_{nij} (\bar{\tau}_{i,j+1} - v_{i,j+1}/2) \Delta x + \alpha_{sij} (\bar{\tau}_{i,j-1} + v_{i,j-1}/2) \Delta x. \end{aligned}$$

We may further choose $w_h = X_i$ and Z_j to derive two more equations on $\{\bar{\tau}_{ij}, \mu_{ij}, v_{ij}\}$ such that a system of three nonlinear equations on $\{\bar{\tau}_{ij}, \mu_{ij}, v_{ij}\}$ can be solved to determine them. Instead, we utilize the same “super-convergence” property. If the solution τ is smooth, the numerical solutions obtained by Algorithm 1, denoted as $\{\tau^1, \tau_x^1, \tau_z^1\}$, are first-order accurate both in its value and gradient, i.e.,

$$|\tau^1 - \tau| = O(\Delta x, \Delta z), |\tau_x^1 - \tau_x| = O(\Delta x, \Delta z), |\tau_z^1 - \tau_z| = O(\Delta x, \Delta z).$$

With this property, we can fix μ_{ij} and v_{ij} in (19) as

$$\mu_{ij} = \tau_{xij}^1 \Delta x, \quad v_{ij} = \tau_{zij}^1 \Delta z,$$

which implies a simple updating formula for $\bar{\tau}_{ij}$,

$$\bar{\tau}_{ij} = \frac{R_{ij} - v_m \sqrt{(\mu_{ij}/\Delta x + T_{0xij})^2 + (v_{ij}/\Delta z + T_{0zij})^2} \Delta x \Delta z - \beta_{ij} \mu_{ij} - \lambda_{ij} v_{ij}}{\gamma_{ij}}. \quad (20)$$

With the linear DG formulation, the “super-convergence” property, and the updating formulas (17) and (20), we have a second-order hybrid fast sweeping method as follows.

Algorithm 2 (Second-Order Hybrid FSM)

1. *Initialization:* assign exact/approximate values at grid points according to given boundary condition, which will be fixed during iterations, and assign large positive values for all other grid points.
2. *Preprocessing:* Algorithm 1 is performed to obtain $\{T^1, T_x^1, T_z^1\}$ and $\{\tau^1, \tau_x^1, \tau_z^1\}$.
3. *Gauss–Seidel iteration:* sweep the mesh with four alternating orderings iteratively:

$$\begin{aligned} (1) \quad & i = 0 : N_x, j = 0 : N_z, \quad (2) \quad i = 0 : N_x, j = N_z : 0, \\ (3) \quad & i = N_x : 0, j = 0 : N_z, \quad (4) \quad i = N_x : 0, j = N_z : 0. \end{aligned}$$

At each grid point (x_i, z_j) ,

- if it is in Ω_0 , use the updating formula (20) to compute τ_{ij} , and $T_{ij} = \tau_{ij} + T_{0ij}$.
- if it is in Ω_0^c , use the updating formula (17) to compute T_{ij} , and $\tau_{ij} = T_{ij} - T_{0ij}$ if necessary.

4. *Termination:* if the l_1 norm difference of the solutions between two successive iterations is smaller than the specified accuracy requirement or the number of iterations exceeds the prescribed maximum number of iterations, terminate the iteration.

Algorithm 2 serves as a post-processing procedure for Algorithm 1. With a minimum amount of extra work, Algorithm 2 achieves second-order accuracy.

Both Algorithms 1 and 2 are efficient in a sense that the complexity is $O(N)$ with N the number of grid points. And the iteration number in the Gauss–Seidel iterations is independent of the mesh size as it approaches zero. In both algorithms, the solutions of the two Eqs. (9) and (11) can be pre-determined (with parallelization if necessary), and repeatedly used in the iterations.

To demonstrate Algorithms 1 and 2, numerical experiments are presented in Sect. 4.

4 Numerical Examples

We present several numerical experiments to demonstrate Algorithms 1 and 2. To show comparisons, we also present numerical solutions computed by the Lax–Friedrichs (LxF) fast sweeping methods for Hamilton–Jacobi type equations [28,38], which are described in “Appendix” as Algorithm 3 with central finite difference approximations and Algorithm 4 with high-order essentially nonoscillatory (ENO) finite difference approximations [41] or weighted ENO (WENO) approximations [27,74]. For notational simplicity, we denote the traveltimes computed by Algorithms 1, 2, 3 and 4 as T^1 , T^2 , T^3 and T^4 , respectively. The numerical experiments are performed with Matlab on a Laptop with Mac OSX.

Example 1 We test the accuracy of the proposed methods using the following Sinusoidal velocity model:

- ρ is given as

$$\rho(x, z) = \frac{1}{(1 + 0.2 \sin(0.5\pi z) \sin(3\pi(x + 0.05)))},$$

- the elastic moduli parameters are given by

$$[C_{11}, C_{13}, C_{33}, C_{44}, C_{66}] \equiv [9.08, 2.98, 7.54, 2.27, 3.84],$$

- the computational domain is $[-0.5, 0.5] \times [0, 1]$,
- the tilted angle θ_0 of the symmetric axis is chosen from $\{0, \pi/3, \pi/2\}$,
- the source point is $(x_0, y_0) = (0, 0.35)$.
- Ω_0 is centered at the source with size 0.15.

Tables 1, 3 and 5 show the l_1 errors for T_x^1 , T_z^1 , T^1 , T^2 and T^4 of all the three wavemodes. For T^4 , the LxF scheme with second-order ENO approximations is used, where τ is assigned to be zeros in a neighborhood of the source point of size $2 \max\{\Delta x, \Delta z\}$. The reference solutions are obtained by Algorithm 2 on a refined mesh 1281×1281 . We observe first-order accuracy for T_x^1 , T_z^1 , and T^1 , and second-order accuracy for T^2 and T^4 . For the numerical experiments, both Algorithms 1 and 2 take 5 iterations to converge to the desired tolerance 10^{-6} as the mesh size decreases. Tables 2, 4 and 6 show the comparisons in terms of CPU times for Algorithms 1, 2, and 4. For Algorithm 2, we record the CPU time as it serves as the post-processing of Algorithm 1.

Example 2 We apply the methods to compute the first-arrival traveltimes of the three wave-modes with the benchmark Hess VTI model [55]. A window of the model from grid point 1 to grid point 1811 with spacing (0.03, 0.03) (km) is chosen as the numerical experiment. Figure 3 shows the five elastic moduli parameters and mass density ρ . The tilted angle is

Table 1 Sinusoidal velocity model: l_1 errors for qP wave are recorded. $\theta_0 = 0, \pi/3, \pi/2$

Mesh	41×41	81×81	161×161	321×321	641×641
qP wave					
Tilted angle $\theta_0 \equiv 0$; VTI					
$T_x^1 l_1$ error	3.300E-3	1.655E-3	8.066E-4	3.548E-4	1.202E-4
conv. order	–	0.995	1.036	1.184	1.561
$T_z^1 l_1$ error	2.850E-3	1.539E-3	7.639E-4	3.419E-4	1.176E-4
conv. order	–	0.888	1.010	1.159	1.539
$T^1 l_1$ error	1.437E-3	8.309E-4	4.444E-4	2.295E-4	1.166E-4
conv. order	–	0.790	0.902	0.953	0.976
$T^2 l_1$ error	1.294E-4	4.184E-5	1.239E-5	3.414E-6	8.008E-7
conv. order	–	1.628	1.755	1.859	2.091
$T^4 l_1$ error	9.894E-4	2.493E-4	6.182E-5	1.617E-5	
conv. order	–	1.988	2.011	1.934	
Tilted angle $\theta_0 \equiv \pi/3$; TTI					
$T_x^1 l_1$ error	3.230E-3	1.542E-3	7.323E-4	3.196E-4	1.083E-4
conv. order	–	1.066	1.074	1.196	1.561
$T_z^1 l_1$ error	2.618E-3	1.283E-3	6.194E-4	2.771E-4	9.542E-5
conv. order	–	1.028	1.050	1.160	1.538
$T^1 l_1$ error	1.686E-3	9.672E-4	5.155E-4	2.658E-4	1.349E-4
conv. order	–	0.801	0.907	0.955	0.978
$T^2 l_1$ error	1.038E-4	3.602E-5	1.159E-5	3.551E-6	9.923E-7
conv. order	–	1.526	1.635	1.706	1.839
$T^4 l_1$ error	9.050E-4	2.171E-4	5.413E-5	1.426E-5	
conv. order	–	2.059	2.003	1.924	
Tilted angle $\theta_0 \equiv \pi/2$; HTI					
$T_x^1 l_1$ error	3.312E-3	1.648E-3	7.956E-4	3.496E-4	1.184E-4
conv. order	–	1.006	1.050	1.186	1.562
$T_z^1 l_1$ error	2.561E-3	1.324E-3	6.589E-4	2.960E-4	1.017E-4
conv. order	–	0.951	1.006	1.154	1.541
$T^1 l_1$ error	1.535E-3	8.803E-4	4.687E-4	2.415E-4	1.226E-4
conv. order	–	0.802	0.909	0.956	0.978
$T^2 l_1$ error	1.278E-4	4.119E-5	1.221E-5	3.421E-6	8.856E-7
conv. order	–	1.633	1.754	1.835	1.949
$T^4 l_1$ error	9.894E-4	2.493E-4	6.182E-5	1.617E-5	
conv. order	–	1.988	2.011	1.934	

$\theta_0 \equiv 0$. Figure 4 shows the contour plots for all the three wavemodes computed by the proposed methods. For T^4 , the LxF scheme with third-order WENO approximations is used, where τ is assigned to be zeros in a neighborhood of the source point of size $2 \max\{\Delta x, \Delta z\}$. Ω_0 is centered at the source with size 4.

Table 2 Sinusoidal velocity model: CPU time (in seconds) of Algorithm 1, 2 and 4 for qP wave are recorded. $\theta_0 = 0, \pi/3, \pi/2$

CPU time/mesh	41×41	81×81	161×161	321×321
qP wave				
Tilted angle $\theta_0 \equiv 0$; VTI				
Alg. 1	5.818	1.513E1	4.607E1	1.683E2
Alg. 2	0.088	0.141	0.289	0.730
Alg. 4	7.590	1.883E1	6.306E1	4.013E2
Tilted angle $\theta_0 \equiv \pi/3$; TTI				
Alg. 1	5.113	1.298E1	4.252E1	1.611E2
Alg. 2	0.060	0.135	0.282	0.684
Alg. 4	2.122	1.818E1	6.133E1	4.100E2
Tilted angle $\theta_0 \equiv \pi/2$; HTI				
Alg. 1	5.818	1.513E1	4.607E1	1.683E2
Alg. 2	0.088	0.141	0.289	0.730
Alg. 4	7.590	1.883E1	6.306E1	4.013E2

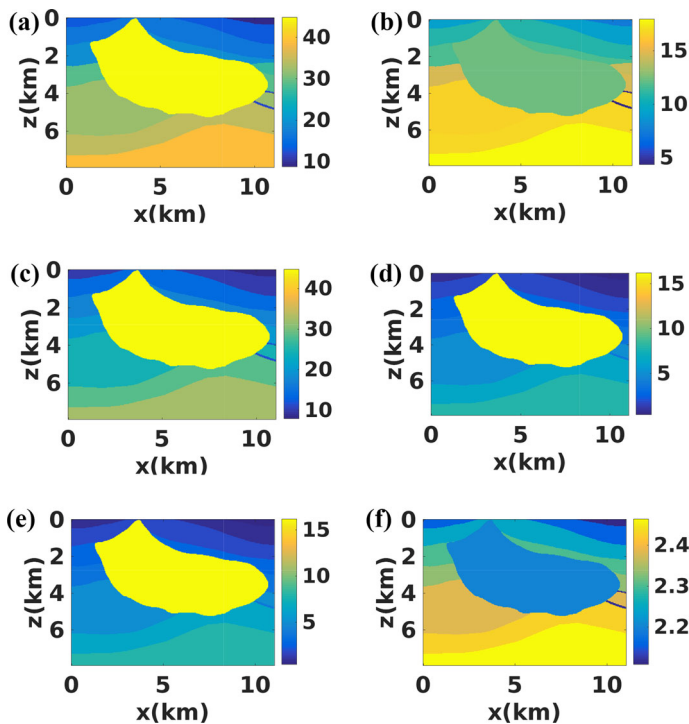


Fig. 3 A window from Hess VTI model: **a–f** C_{11} , C_{13} , C_{33} , C_{44} , C_{66} , and ρ . Mesh: 363×261 with $(\Delta x, \Delta z) = (0.03, 0.03)$ (km) (Color figure online)

Example 3 we apply the methods to compute the first-arrival traveltime of the three wave-modes with the benchmark BP TTI model [19]. The five elastic moduli parameters and the tilted angle θ_0 are given as in Fig. 5, and the mass density is $\rho \equiv 1$. The size of the model is

Table 3 Sinusoidal velocity model: l_1 errors for qSV wave are recorded. $\theta_0 = 0, \pi/3, \pi/2$

Mesh	41×41	81×81	161×161	321×321	641×641
qSV wave					
Tilted angle $\theta_0 \equiv 0$; VTI					
$T_x^1 l_1$ error	1.047E-2	5.524E-3	2.764E-3	1.250E-3	4.330E-4
conv. order	–	0.922	0.998	1.144	1.529
$T_z^1 l_1$ error	9.692E-3	5.079E-3	2.531E-3	1.141E-3	3.943E-4
conv. order	–	0.932	1.004	1.149	1.532
$T^1 l_1$ error	3.526E-3	2.092E-3	1.140E-3	5.967E-4	3.054E-4
conv. order	–	0.753	0.875	0.933	0.966
$T^2 l_1$ error	2.721E-4	1.302E-4	5.332E-5	1.880E-5	5.131E-6
conv. order	–	1.063	1.287	1.503	1.873
$T^4 l_1$ error	1.475E-3	3.823E-4	1.045E-4	2.954E-5	
conv. order	–	1.947	1.871	1.822	
Tilted angle $\theta_0 \equiv \pi/3$; TTI					
$T_x^1 l_1$ error	9.341E-3	4.756E-3	2.323E-3	1.028E-3	3.502E-4
conv. order	–	0.973	1.033	1.176	1.553
$T_z^1 l_1$ error	8.445E-3	4.342E-3	2.136E-3	9.539E-4	3.267E-4
conv. order	–	0.959	1.023	1.163	1.545
$T^1 l_1$ error	2.738E-3	1.601E-3	8.615E-4	4.460E-4	2.269E-4
conv. order	–	0.774	0.894	0.949	0.974
$T^2 l_1$ error	3.138E-4	1.111E-4	3.549E-5	1.043E-5	2.497E-6
conv. order	–	1.497	1.646	1.766	2.062
$T^4 l_1$ error	2.247E-3	6.729E-4	1.912E-4	5.375E-5	
conv. order	–	1.739	1.815	1.830	
Tilted angle $\theta_0 \equiv \pi/2$; HTI					
$T_x^1 l_1$ error	1.023E-2	5.374E-3	2.689E-3	1.217E-3	4.217E-4
conv. order	–	0.928	0.998	1.143	1.529
$T_z^1 l_1$ error	9.675E-3	5.131E-3	2.591E-3	1.180E-3	4.111E-4
conv. order	–	0.915	0.985	1.134	1.521
$T^1 l_1$ error	3.435E-3	2.047E-3	1.119E-3	5.862E-4	3.001E-4
conv. order	–	0.746	0.871	0.932	0.965
$T^2 l_1$ error	2.339E-4	1.078E-4	4.326E-5	1.532E-5	4.309E-6
conv. order	–	1.117	1.317	1.497	1.829
$T^4 l_1$ error	1.427E-3	3.769E-4	1.061E-4	3.057E-5	
conv. order	–	1.920	1.828	1.795	

1801 \times 12596 with spacing (0.00625, 0.01) (km). We choose three windows from the model as three numerical experiments (indicated by solid lines in Fig. 5):

- window 1: from grid point 501 to grid point 2201,
- window 2: from grid point 4501 to grid point 6001,
- window 3: from grid point 7251 to grid point 8751.

Table 4 Sinusoidal velocity model: CPU time (in seconds) of Algorithm 1, 2 and 4 for qSV wave are recorded. $\theta_0 = 0, \pi/3, \pi/2$

CPU time/mesh	41×41	81×81	161×161	321×321
qSV wave				
Tilted angle $\theta_0 \equiv 0$; VTI				
Alg. 1	6.339	1.389E1	4.970E1	1.638E2
Alg. 2	0.096	0.098	0.303	0.780
Alg. 4	1.807	8.801	6.509E1	4.056E2
Tilted angle $\theta_0 \equiv \pi/3$; TTI				
Alg. 1	6.044	1.543E1	4.607E1	1.624E2
Alg. 2	0.107	0.090	0.279	0.739
Alg. 4	7.953	8.449	6.726E1	4.073E2
Tilted angle $\theta_0 \equiv \pi/2$; HTI				
Alg. 1	5.489	1.508E1	4.652E1	1.755E2
Alg. 2	0.110	0.140	0.273	0.787
Alg. 4	8.017	1.895E1	5.581E1	4.121E2

For each window, a subset of the data is chosen in the numerical experiments with $(\Delta x, \Delta z) = (0.05, 0.05)$ (km). Figures 6, 7 and 8 show the contour plots for all the three wavemodes computed by the proposed methods. For T^4 , the LxF scheme with third-order WENO approximations is used, where τ is assigned to be zeros in a neighborhood of the source point of size $2 \max\{\Delta x, \Delta z\}$. Ω_0 is centered at the source with size 4.

4.1 Discussion of Numerical Experiments

Example 1 verifies the accuracy and efficiency of the proposed methods. Examples 2 and 3 with benchmark models from geophysics verify the feasibility for computing first-arrival traveltimes of the three wavemodes in general 2-D TTI media. In the local solvers of Algorithms 1 and 2, the partitioning points (i.e., $\{\theta_k\}$), the group velocities along edges of the mesh, and $\{T_0, \nabla T_0\}$ can be pre-computed easily in parallel. And they can be repeatedly used during the iterations of the methods.

For the factorization approach to resolve the source singularities, besides the additive decomposition (6), we can also adopt the multiplicative decomposition [21,36,37,39,42,72] as $T = \tau T_0$. For simplicity, we choose the additive decomposition in current application because the equations in θ for determining the partitioning points (i.e., $\{\theta_k\}$) are the same for the anisotropic eikonal equation and factored anisotropic eikonal equation. If the multiplicative decomposition is used, a different equation in θ for determining the partitioning points (i.e., $\{\theta_k\}$) must be solved for the factored anisotropic eikonal equation [25].

5 Conclusion

We present both first-order and second-order hybrid fast sweeping methods for computing the first-arrival traveltimes of the qP, qSV and qSH waves in the two-dimensional tilted transversely isotropic media, based on solving the anisotropic eikonal equation numerically. The methods solve a factored anisotropic eikonal equation after applying a factorization approach to resolve the source singularity near the source in a neighborhood of the source,

Table 5 Sinusoidal velocity model: l_1 errors for qSH wave are recorded. $\theta_0 = 0, \pi/3, \pi/2$

Mesh	41×41	81×81	161×161	321×321	641×641
qSH wave					
Tilted angle $\theta_0 \equiv 0$; VTI					
$T_x^1 l_1$ error	5.753E-3	2.928E-3	1.415E-3	6.224E-4	2.112E-4
conv. order	–	0.974	1.049	1.184	1.559
$T_z^1 l_1$ error	5.460E-3	2.888E-3	1.419E-3	6.321E-4	2.168E-4
conv. order	–	0.918	1.025	1.166	1.543
$T^1 l_1$ error	2.361E-3	1.387E-3	7.484E-4	3.882E-4	1.977E-4
conv. order	–	0.767	0.890	0.947	0.973
$T^2 l_1$ error	2.107E-4	6.947E-5	2.108E-5	5.974E-6	1.411E-6
conv. order	–	1.600	1.720	1.819	2.081
$T^4 l_1$ error	1.465E-3	3.618E-4	9.002E-5	2.444E-5	
conv. order	–	2.017	2.006	1.881	
Tilted angle $\theta_0 \equiv \pi/3$; TTI					
$T_x^1 l_1$ error	6.037E-3	2.893E-3	1.353E-3	5.839E-4	1.965E-4
conv. order	–	1.061	1.096	1.212	1.571
$T_z^1 l_1$ error	4.935E-3	2.340E-3	1.120E-3	4.936E-4	1.689E-4
conv. order	–	1.076	1.063	1.182	1.547
$T^1 l_1$ error	2.946E-3	1.681E-3	8.940E-4	4.605E-4	2.336E-4
conv. order	–	0.809	0.910	0.957	0.979
$T^2 l_1$ error	1.723E-4	5.857E-5	1.878E-5	5.805E-6	1.664E-6
conv. order	–	1.556	1.640	1.693	1.802
$T^4 l_1$ error	1.667E-3	4.038E-4	9.809E-5	2.728E-5	
conv. order	–	2.045	2.041	1.846	
Tilted angle $\theta_0 \equiv \pi/2$; HTI					
$T_x^1 l_1$ error	6.064E-3	2.917E-3	1.361E-3	5.898E-4	1.989E-4
conv. order	–	1.055	1.099	1.206	1.568
$T_z^1 l_1$ error	4.550E-3	2.219E-3	1.054E-3	4.623E-4	1.573E-4
conv. order	–	1.035	1.074	1.188	1.555
$T^1 l_1$ error	2.853E-3	1.632E-3	8.682E-4	4.470E-4	2.268E-4
conv. order	–	0.805	0.910	0.957	0.978
$T^2 l_1$ error	1.932E-4	6.392E-5	1.992E-5	6.442E-6	2.111E-6
conv. order	–	1.595	1.682	1.628	1.609
$T^4 l_1$ error	1.715E-3	4.110E-4	1.004E-4	2.629E-5	
conv. order	–	2.060	2.033	1.933	

and solve the original anisotropic eikonal equation outside of the neighborhood of the source. Consequently, the traveltimes computed by the first-order method has clean first-order accuracy. Furthermore, its gradient is also first-order accurate, which is the so-called “super-convergence” property and is utilized to design a second-order fast sweeping method based on a linear discontinuous Galerkin formulation. For the first-order method, a systematic approach is designed to solve the nonlinear discretized equation in the local solver at each grid point.

Table 6 Sinusoidal velocity model: CPU time (in seconds) of Algorithm 1, 2 and 4 for qSH wave are recorded. $\theta_0 = 0, \pi/3, \pi/2$

CPU time/mesh	41×41	81×81	161×161	321×321
qSH wave				
Tilted angle $\theta_0 \equiv 0$; VTI				
Alg. 1	5.825	1.454E1	5.131E1	1.691E2
Alg. 2	0.105	0.132	0.261	0.740
Alg. 4	7.837	1.949E1	6.479E1	4.184E2
Tilted angle $\theta_0 \equiv \pi/3$; TTI				
Alg. 1	5.591	1.475E1	4.540E1	1.665E2
Alg. 2	0.110	0.151	0.225	0.784
Alg. 4	7.537	1.951E1	6.731E1	4.258E2
Tilted angle $\theta_0 \equiv \pi/2$; HTI				
Alg. 1	5.930	1.548E1	4.557E1	1.591E2
Alg. 2	0.087	0.142	0.277	0.721
Alg. 4	7.786	1.937E1	6.614E1	4.273E2

With the appropriate enforcement of the causality condition, the method enjoys all the desired properties of the usual fast sweeping method, i.e., efficiency, accuracy and convergence guaranteed by monotonicity. The second-order method, which serves as a post-processing procedure of the first-order method, is also efficient and accurate. Moreover, necessary ingredients in both methods, such as the group velocities along given characteristic/ray directions and the extreme points of the discretized Hamiltonian, can be pre-determined and repeatedly used in the local solver. Numerical experiments verify the efficiency of the proposed methods. As future work, the techniques will be extended to three-dimensional cases, where we will investigate how to design a similar systematic procedure for solving the nonlinear equation of the local solver efficiently. Moreover, the Lax–Friedrichs scheme presented in “Appendix” will be extended to higher-order schemes.

The datasets during and/or analysed during the current study are available from the corresponding author on reasonable request.

Appendix: Lax–Friedrichs Fast Sweeping Method

Following [28,38], the Lax–Friedrichs (LxF) scheme for the anisotropic eikonal equation (4) at each grid point (x_i, z_j) is

$$H^{LF}(P_{ij}, Q_{ij}) \equiv v_m(P_{ij}, Q_{ij}) \sqrt{P_{ij}^2 + Q_{ij}^2} - \rho_{ij} - \alpha_x \frac{T_{i+1,j} - 2T_{ij} + T_{i-1,j}}{2\Delta x} - \alpha_z \frac{T_{i,j+1} - 2T_{ij} + T_{i,j-1}}{2\Delta z} = 0, \quad (21)$$

with $P_{ij} = (T_{i+1,j} - T_{i-1,j})/(2\Delta x)$ and $Q_{ij} = (T_{i,j+1} - T_{i,j-1})/(2\Delta z)$, where α_x and α_z are constants chosen to satisfy $\partial H^{LF}/\partial T_{ij} \geq 0$, and $\partial H^{LF}/\partial \{T_{i-1,j}, T_{i+1,j}, T_{i,j-1}, T_{i,j+1}\} \leq 0$.

From (21), we can derive the local updating formula for T_{ij} as

$$T_{ij} = \frac{\rho_{ij} - v_m(P_{ij}, Q_{ij}) \sqrt{P_{ij}^2 + Q_{ij}^2} + \alpha_x \frac{T_{i+1,j} + T_{i-1,j}}{2\Delta x} + \alpha_z \frac{T_{i,j+1} + T_{i,j-1}}{2\Delta z}}{\alpha_x/\Delta x + \alpha_z/\Delta z}. \quad (22)$$

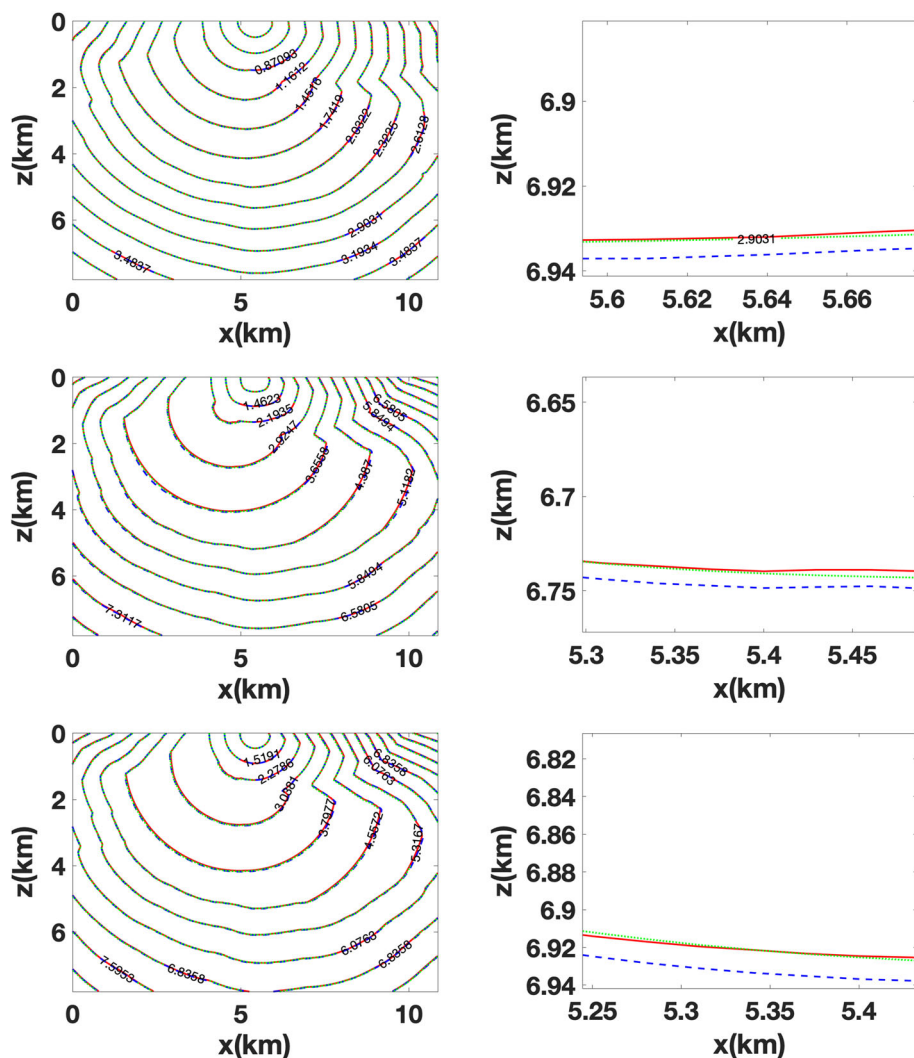


Fig. 4 Contour plots for the Hess VTI model: Top–Middle–Bottom: qP, qSV, and qSH waves. Left figure shows a zoom-in region of Right figure. Red-curve: Algorithm 2; Blue-dashed: Algorithm 1; and Green-dotted: Algorithm 4 (Color figure online)

Similarly, the LxF scheme for the factored anisotropic eikonal equation at (x_i, z_j) is

$$\begin{aligned} \bar{H}^{LF}(p_{ij}, q_{ij}) \equiv & \\ & v_m(p_{ij} + T_{0xij}, q_{ij} + T_{0zij}) \sqrt{(p_{ij} + T_{0xij})^2 + (q_{ij} + T_{0zij})^2 - \rho_{ij}} \\ & - \alpha_x \frac{\tau_{i+1,j} - 2\tau_{ij} + \tau_{i-1,j}}{2\Delta x} - \alpha_z \frac{\tau_{i,j+1} - 2\tau_{ij} + \tau_{i,j-1}}{2\Delta z} = 0, \end{aligned} \quad (23)$$

with $p_{ij} = (\tau_{i+1,j} - \tau_{i-1,j})/(2\Delta x)$ and $q_{ij} = (\tau_{i,j+1} - \tau_{i,j-1})/(2\Delta z)$, where α_x and α_z are constants chosen to satisfy $\partial \bar{H}^{LF} / \partial \tau_{ij} \geq 0$, and $\partial \bar{H}^{LF} / \partial \{\tau_{i-1,j}, \tau_{i+1,j}, \tau_{i,j-1}, \tau_{i,j+1}\} \leq 0$.

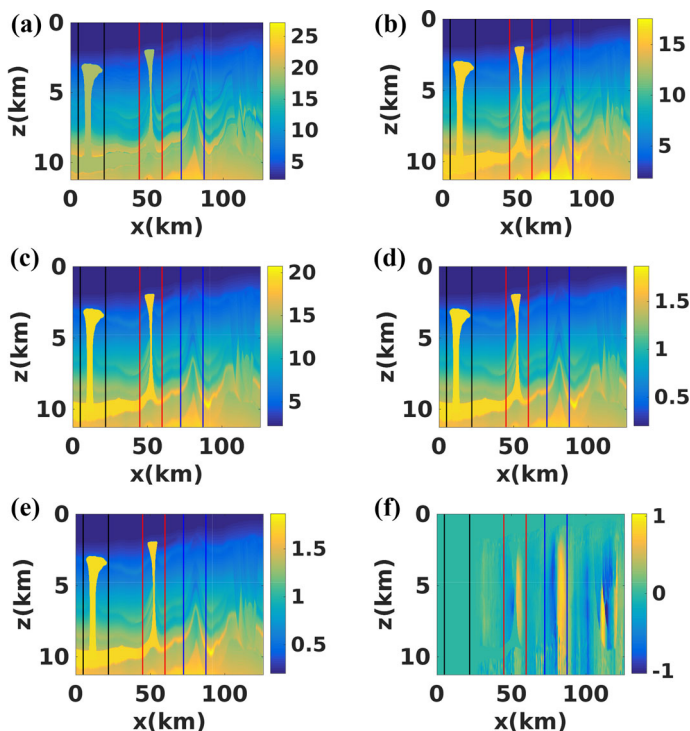


Fig. 5 BP TTI model: **a–f** C_{11} , C_{13} , C_{33} , C_{44} , C_{66} , and θ_0 . Three windows (indicated by solid vertical lines) are chosen as numerical experiments. Mesh: 341×226 (window 1), 301×226 (window 2), and 301×226 (window 3), with $(\Delta x, \Delta z) = (0.05, 0.05)$ (km)

From (23), we can derive the local updating formula for τ_{ij} as

$$\tau_{ij} = \frac{\rho_{ij} - v_m(p_{ij} + T_{0xij}, q_{ij} + T_{0zij})\sqrt{(p_{ij} + T_{0xij})^2 + (q_{ij} + T_{0zij})^2} + \alpha_x \frac{\tau_{j+1,j} + \tau_{j-1,j}}{2\Delta x} + \alpha_z \frac{\tau_{i,j+1} + \tau_{i,j-1}}{2\Delta z}}{\alpha_x / \Delta z + \alpha_z / \Delta z} \quad (24)$$

With the LxF schemes and the local updating formulas, we summarize the first-order hybrid LxF fast sweeping method.

Algorithm 3 (First-Order LxF FSM)

1. *Initialization: assign exact/approximate values at grid points according to given boundary condition, which will be fixed during iterations, and assign large positive values for all other grid points.*
2. *Gauss–Seidel iteration: sweep the mesh with four alternating orderings iteratively:*

- (1) $i = 0 : N_x, j = 0 : N_z$, (2) $i = 0 : N_x, j = N_z : 0$,
- (3) $i = N_x : 0, j = 0 : N_z$, (4) $i = N_x : 0, j = N_z : 0$.

At each grid point (x_i, z_j) ,

- if it is in Ω_0 , use the updating formula (24) to compute τ_{ij} , and $T_{ij} = \tau_{ij} + T_{0ij}$.

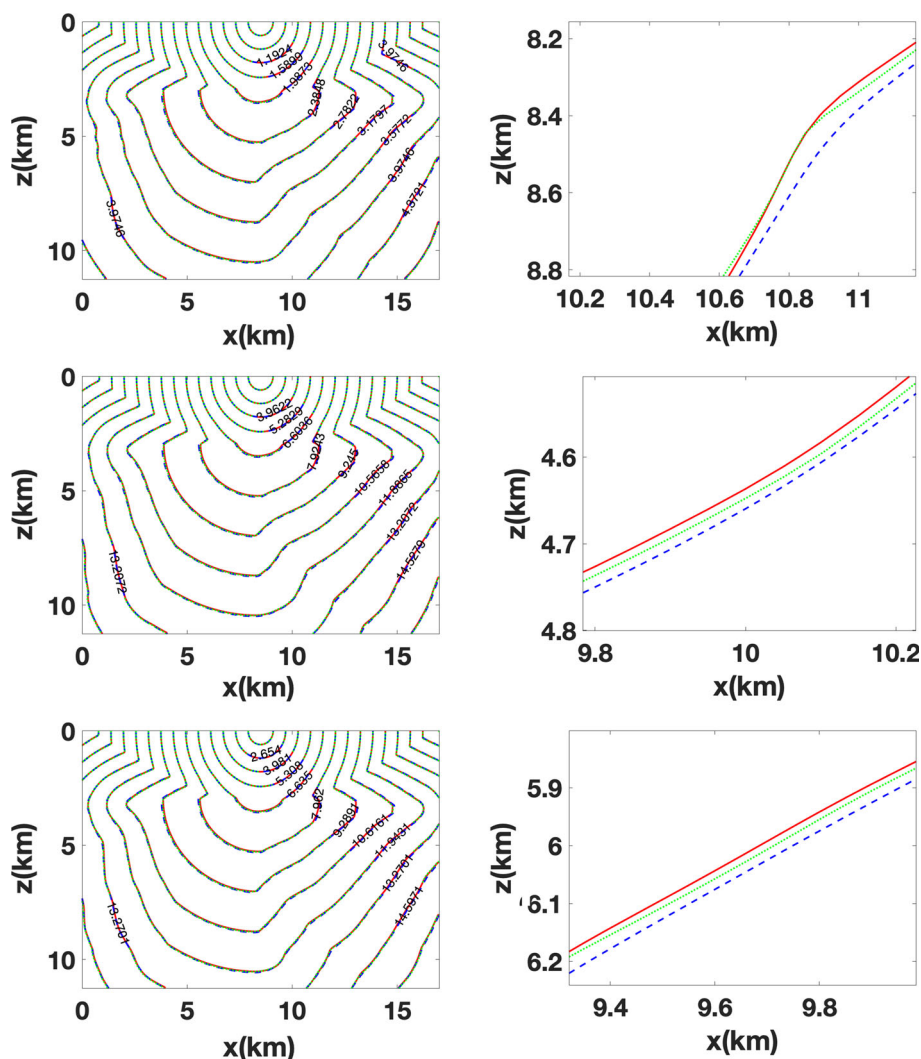


Fig. 6 Contour plots for window 1 of the BP TTI model: Top–Middle–Bottom: qP, qSV, and qSH waves. Left figure shows a zoom-in region of Right figure. Red-curve: Algorithm 2; Blue-dashed: Algorithm 1; and Green-dotted: Algorithm 4 (Color figure online)

- if it is in Ω_0^c , use the updating formula (22) to compute T_{ij} , and $\tau_{ij} = T_{ij} - T_{0ij}$ if necessary.
- 3. *Termination*: if the l_1 norm difference of the solutions between two successive iterations is smaller than the specified accuracy requirement or the number of iterations exceeds the prescribed maximum number of iterations, terminate the iteration.

We can improve the accuracy using high-order approximations of the derivatives, such as the high-order essentially nonoscillatory (ENO) finite difference approximations [41], or high-order Weighted ENO (WENO) finite difference approximations [27,74].

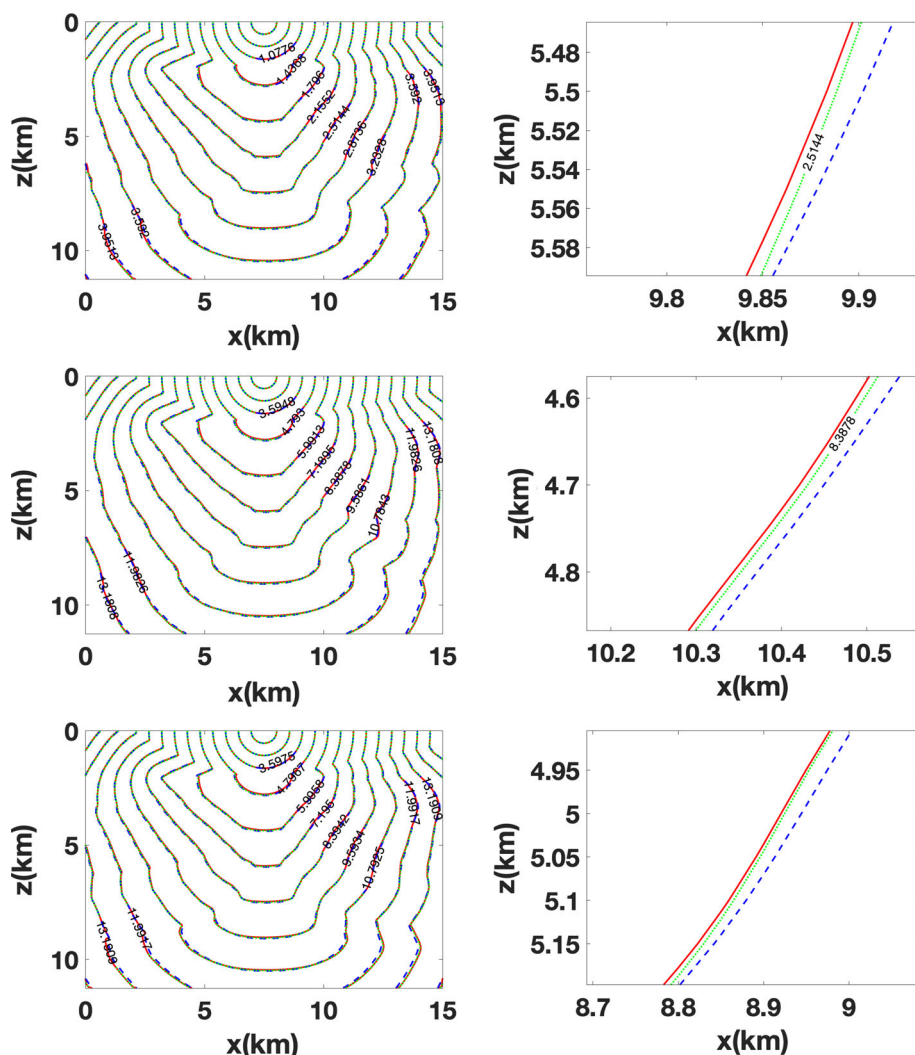


Fig. 7 Contour plots for window 2 of the BP TTI model: Top–Middle–Bottom: qP, qSV, and qSH waves. Left figure shows a zoom-in region of Right figure. Red-curve: Algorithm 2; Blue-dashed: Algorithm 1; and Green-dotted: Algorithm 4 (Color figure online)

We replace $T_{i\pm 1,j}$, $T_{i,j\pm 1}$ in (22) with

$$T_{i\pm 1,j} = T_{ij} \pm \Delta x T_{xij}^{\pm}, \quad T_{i,j\pm 1} = T_{ij} \pm \Delta z T_{zij}^{\pm},$$

where T_{xij}^{\pm} (and T_{zij}^{\pm}) are high-order ENO or WENO approximations of the right and left derivative of T in the x (and z) direction at (x_i, z_j) , respectively. Then we have the local updating formula for T_{ij} ,

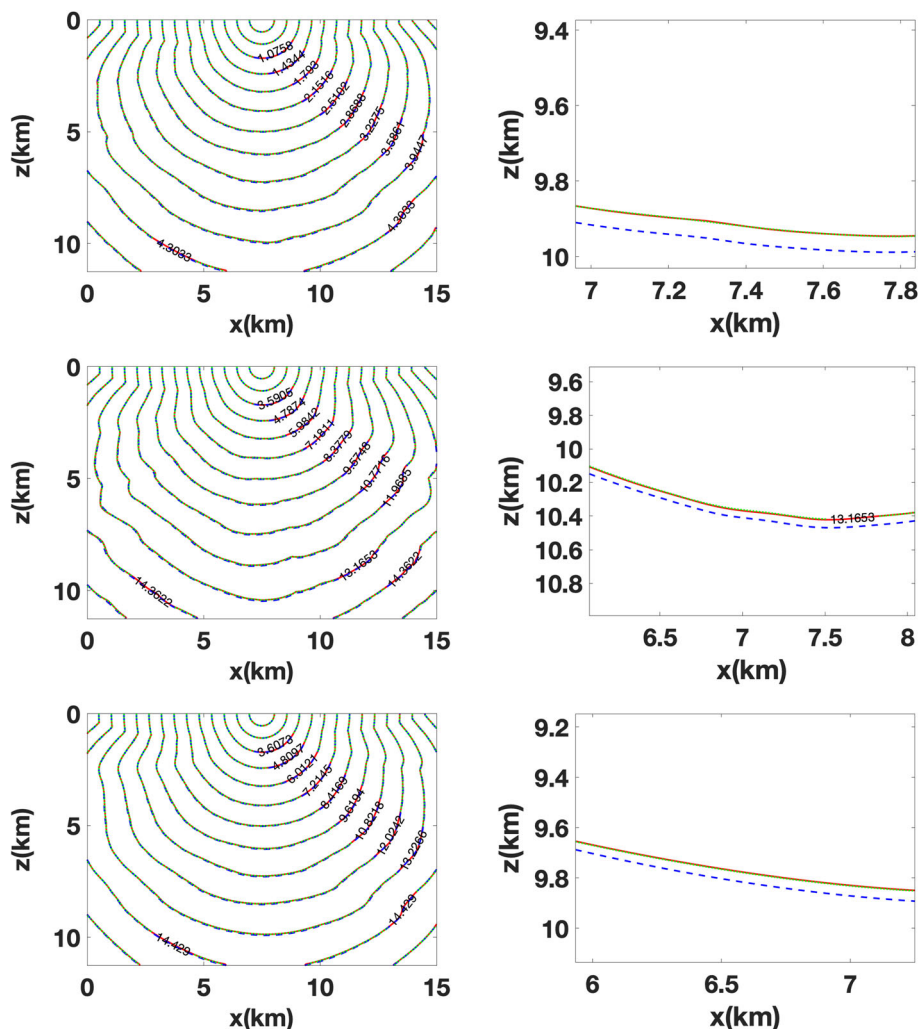


Fig. 8 Contour plots for window 3 of the BP TTI model: Top–Middle–Bottom: qP, qSV, and qSH waves. Left figure shows a zoom-in region of Right figure. Red-curve: Algorithm 2; Blue-dashed: Algorithm 1; and Green-dotted: Algorithm 4 (Color figure online)

$$T_{ij}^{new} = \frac{\rho_{ij} - v_m(P_{ij}, Q_{ij}) \sqrt{P_{ij}^2 + Q_{ij}^2} + \alpha_x \frac{2T_{ij}^{old} + \Delta x(T_{xij}^+ - T_{xij}^-)}{2\Delta x} + \alpha_z \frac{2T_{ij}^{old} + \Delta z(T_{zij}^+ - T_{zij}^-)}{2\Delta z}}{\alpha_x / \Delta z + \alpha_z / \Delta z}, \quad (25)$$

with $P_{ij} = \frac{T_{xij}^+ + T_{xij}^-}{2}$ and $Q_{ij} = \frac{T_{zij}^+ + T_{zij}^-}{2}$. Here T_{ij}^{old} and T_{ij}^{new} are the current old value and newly updated value at (x_i, z_j) , respectively.

Similarly, we replace $\tau_{i\pm 1, j}$, $\tau_{i, j\pm 1}$ in (24) with

$$\tau_{i\pm 1, j} = \tau_{ij} \pm \Delta x \tau_{xij}^{\pm}, \quad \tau_{i, j\pm 1} = \tau_{ij} \pm \Delta z \tau_{zij}^{\pm},$$

where τ_{xij}^{\pm} (and τ_{zij}^{\pm}) are high-order ENO or WENO approximations of the right and left derivative of τ in the x (and z) direction at (x_i, z_j) , respectively. Then we have the local updating formula for τ_{ij} ,

$$\tau_{ij}^{new} = \frac{\rho_{ij} - v_m(p_{ij} + T_{0xij}, q_{ij} + T_{0zij})\sqrt{(p_{ij} + T_{0xij})^2 + (q_{ij} + T_{0zij})^2} + \alpha_x \frac{2\tau_{ij}^{old} + \Delta x(\tau_{xij}^+ - \tau_{xij}^-)}{2\Delta x} + \alpha_z \frac{2\tau_{ij}^{old} + \Delta z(\tau_{zij}^+ - \tau_{zij}^-)}{2\Delta z}}{\alpha_x/\Delta z + \alpha_z/\Delta x}, \quad (26)$$

with $p_{ij} = \frac{\tau_{xij}^+ + \tau_{xij}^-}{2}$ and $q_{ij} = \frac{\tau_{zij}^+ + \tau_{zij}^-}{2}$. Here τ_{ij}^{old} and τ_{ij}^{new} are the current old value and newly updated value at (x_i, z_j) , respectively.

With the high-order ENO or WENO approximations, we have a high-order hybrid LxF fast sweeping method, which is the same as Algorithm 3, except that at each grid (x_i, z_j) , we use the new local updating formulas (25) and (26). We refer to the high-order LxF method with high-order ENO or WENO approximations as Algorithm 4, and demonstrate it with second-order ENO approximations and third-order WENO approximations in the numerical experiments.

References

- Alkhalifah, T.: Traveltime computation with the linearized eikonal equation for anisotropic media. *Geophys. Prospect.* **50**, 373–382 (2002)
- Barles, G., Souganidis, P.E.: Convergence of approximation schemes for fully nonlinear second order equations. *Asymptot. Anal.* **4**, 271–283 (1991)
- Benamou, J.D., Luo, S., Zhao, H.-K.: A compact upwind second order scheme for the eikonal equation. *J. Comput. Math.* **28**, 489–516 (2010)
- Berryman, J.G.: Long-wave elastic anisotropy in transversely isotropic media. *Geophysics* **44**(5), 896–917 (1979)
- Bin Waheed, U., Alkhalifah, T.: A fast sweeping algorithm for accurate solution of the tilted transversely isotropic eikonal equation using factorization. *Geophysics* **82**(6), WB1–WB8 (2017)
- Bin Waheed, U., Yarman, C.E., Flagg, G.: An iterative fast sweeping based eikonal solver for tilted orthorhombic media. *SEG Technical Program Expanded Abstracts 2014*, pp. 480–485 (2014)
- Boué, M., Dupuis, P.: Markov chain approximations for deterministic control problems with affine dynamics and quadratic cost in the control. *SIAM J. Numer. Anal.* **36**(3), 667–695 (1999)
- Cao, S., Greenhalgh, S.: Calculation of the seismic first-break time field and its ray path distribution using a minimum traveltime tree algorithm. *Geophys. J. Int.* **114**(3), 593–600 (1993)
- Cerveny, V.: Seismic rays and ray intensities in inhomogeneous anisotropic media. *Geophys. J. R. Astron. Soc.* **29**, 1–13 (1972)
- Cerveny, V.: *Seismic Ray Theory*. Cambridge University Press, Cambridge (2001)
- Cheng, Y., Shu, C.-W.: A discontinuous Galerkin finite element method for directly solving the Hamilton–Jacobi equations. *J. Comput. Phys.* **223**(1), 398–415 (2007)
- Crampin, S.: A review of wave motion in anisotropic and cracked elastic-media. *Wave Motion* **3**(4), 343–391 (1981)
- Crandall, M.G., Lions, P.-L.: Viscosity solutions of Hamilton–Jacobi equations. *Trans. Am. Math. Soc.* **277**, 1–42 (1983)
- Daley, P.F., Hron, F.: Reflection and transmission coefficients for transversely isotropic media. *Bull. Seismol. Soc. Am.* **67**(3), 661–675 (1977)
- Danielsson, P.: Euclidean distance mapping. *Comput. Graph. Image Process.* **14**, 227–248 (1980)
- Dellinger, J., Symes, W.W.: Anisotropic finite-difference traveltimes using a Hamilton–Jacobi solver. In: 67th Ann. Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, Soc. Expl. Geophys., Tulsa, OK, pp. 1786–1789 (1997)
- Dijkstra, E.W.: A note on two problems in connection with graphs. *Numer. Math.* **1**, 269–271 (1959)

18. Dziewonski, A.M., Anderson, D.L.: Preliminary reference earth model. *Phys. Earth Planet. Inter.* **25**(4), 297–356 (1981)
19. EAGE: 2007 BP Anisotropic Velocity Benchmark (2007). https://wiki.seg.org/wiki/2007_BP_Anisotropic_Velocity_Benchmark
20. Eaton, D.: Finite difference traveltime calculation for anisotropic media. *Geophys. J. Int.* **114**, 273–280 (1993)
21. Fomel, S., Luo, S., Zhao, H.-K.: Fast sweeping method for the factored eikonal equation. *J. Comput. Phys.* **228**(17), 6440–6455 (2009)
22. Ford, J.A.: Improved algorithms of illinois-type for the numerical solution of nonlinear equations (1995)
23. Han, S., Zhang, J., Zhang, W.: Calculating qP-wave traveltimes in 2-D TTI media by high-order fast sweeping methods with a numerical quartic equation solver. *Geophys. J. Int.* **210**(3), 1560–1569 (2017)
24. Hole, J., Zelt, B.C.: 3-D finite-difference reflection traveltimes. *Geophys. J. Int.* **121**, 427–434 (1995)
25. Huang, G., Luo, S., Deng, J., Vavryčuk, V.: Traveltime calculations for qP-, qSV- and qSH- waves in two-dimensional tilted transversely isotropic media. *J. Geophys. Res. Solid Earth*, 125, e2019JB018868. <https://doi.org/10.1029/2019JB018868>
26. Jacobs, M., Luo, S.: Asymptotic solutions for high frequency Helmholtz equations in anisotropic media with Hankel functions. *J. Sci. Comput.* **80**, 808–833 (2019). <https://doi.org/10.1007/s10915-019-00957-8>
27. Jiang, G.-S., Peng, D.: Weighted ENO schemes for Hamilton–Jacobi equations. *SIAM J. Sci. Comput.* **21**, 2126–2143 (2000)
28. Kao, C.Y., Osher, S., Qian, J.: Lax–Friedrichs sweeping schemes for static Hamilton–Jacobi equations. *J. Comput. Phys.* **196**, 367–391 (2004)
29. Kao, C.Y., Osher, S., Tsai, Y.H.: Fast sweeping method for static Hamilton–Jacobi equations. *SIAM J. Numer. Anal.* **42**, 2612–2632 (2005)
30. Kim, S.: Eikonal solvers for anisotropic traveltimes. In: 69th Ann. Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts, Soc. Expl. Geophys., Tulsa, OK, pp. 1875–1878 (1999)
31. Kim, S., Cook, R.: 3-D traveltime computation using second-order ENO scheme. *Geophysics* **64**, 1867–1876 (1999)
32. Lecomte, I.: Finite difference calculation of first traveltimes in anisotropic media. *Geophys. J. Int.* **113**, 318–342 (1993)
33. Li, F., Shu, C.-W., Zhang, Y.-T., Zhao, H.: Second order discontinuous fast sweeping method for eikonal equations. *J. Comput. Phys.* **227**(17), 8191–8208 (2008)
34. Luo, S.: Numerical Methods for Static Hamilton–Jacobi Equations. University of California, Irvine (2009). Ph.D Thesis
35. Luo, S.: A uniformly second order fast sweeping method for eikonal equations. *J. Comput. Phys.* **241**, 104–117 (2013)
36. Luo, S., Qian, J.: Factored singularities and high-order Lax–Friedrichs sweeping schemes for point-source traveltimes and amplitudes. *J. Comput. Phys.* **230**, 4742–4755 (2011)
37. Luo, S., Qian, J.: Fast sweeping methods for factored anisotropic eikonal equations: multiplicative and additive factors. *J. Sci. Comput.* **52**, 360–382 (2012)
38. Luo, S., Qian, J., Burrige, R.: High-order factorization based high-order hybrid fast sweeping methods for point-source eikonal equations. *SIAM J. Numer. Anal.* **52**(1), 23–44 (2014)
39. Luo, S., Qian, J., Zhao, H.-K.: Higher-order schemes for 3-D first-arrival traveltimes and amplitudes. *Geophysics* **77**, T47–T56 (2012)
40. Orcutt, J., Shearer, P.: Anisotropy in the oceanic lithosphere—theory and observations from the Ngendei seismic refraction experiment in the south-west Pacific. *Geophys. J. Int.* **80**(2), 493–526 (1985)
41. Osher, S., Shu, C.-W.: High-order essentially nonoscillatory schemes for Hamilton–Jacobi equations. *SIAM J. Numer. Anal.* **28**(4), 907–922 (1991)
42. Pica, A.: Fast and accurate finite-difference solutions of the 3D eikonal equation parametrized in celerity. In: 67th Ann. Internat. Mtg. Soc. of Expl. Geophys., pp. 1774–1777 (1997)
43. Podvin, P., Lecomte, I.: Finite difference computation of traveltimes in very contrasted velocity models: a massively parallel approach and its associated tools. *Geophys. J. Int.* **105**, 271–284 (1991)
44. Qian, J., Symes, W.W.: Adaptive finite difference method for traveltime and amplitude. *Geophysics* **67**, 167–176 (2002)
45. Qian, J., Symes, W.W.: An adaptive finite-difference method for traveltimes and amplitudes. *Geophysics* **67**, 167–176 (2002)
46. Qian, J., Zhang, Y.-T., Zhao, H.-K.: A fast sweeping methods for static convex Hamilton–Jacobi equations. *J. Sci. Comput.* **31**(1/2), 237–271 (2007)
47. Qian, J., Zhang, Y.-T., Zhao, H.-K.: Fast sweeping methods for eikonal equations on triangulated meshes. *SIAM J. Numer. Anal.* **45**, 83–107 (2007)

48. Qian, J., Symes, W.W.: Paraxial eikonal solvers for anisotropic quasi-p travel times. *J. Comput. Phys.* **173**(1), 256–278 (2001)
49. Qian, J., Symes, W.W.: Finite-difference quasi-P traveltimes for anisotropic media. *Geophysics* **67**(1), 147–155 (2002)
50. Qin, F., Luo, Y., Olsen, K.B., Cai, W., Schuster, G.T.: Finite difference solution of the eikonal equation along expanding wavefronts. *Geophysics* **57**, 478–487 (1992)
51. Rawlinson, N., Hauser, J., Sambridge, M.: Seismic ray tracing and wavefront tracking in laterally heterogeneous media. *Adv. Geophys.* **49**, 203–273 (2008)
52. Rouy, E., Tourin, A.: A viscosity solutions approach to shape-from-shading. *SIAM J. Numer. Anal.* **29**, 867–884 (1992)
53. Schneider Jr., W.A.: Robust and efficient upwind finite-difference traveltime calculations in three dimensions. *Geophysics* **60**, 1108–1117 (1995)
54. Schneider Jr., W.A., Ranzinger, K., Balch, A., Kruse, C.: A dynamic programming approach to first arrival traveltime computation in media with arbitrarily distributed velocities. *Geophysics* **57**, 39–50 (1992)
55. SEG: Hess VTI migration benchmark (2006). https://wiki.seg.org/wiki/Hess_VTI_migration_benchmark
56. Sethian, J.A.: A fast marching level set method for monotonically advancing fronts. *Proc. Natl. Acad. Sci.* **93**(4), 1591–1595 (1996)
57. Sethian, J.A., Popovici, A.M.: 3-D traveltime computation using the fast marching method. *Geophysics* **64**, 516–523 (1999)
58. Sethian, J.A., Vladimirsky, A.: Ordered upwind methods for static Hamilton–Jacobi equations. *Proc. Natl. Acad. Sci.* **98**, 11069–11074 (2001)
59. Sethian, J.A., Vladimirsky, A.: Ordered upwind methods for static Hamilton–Jacobi equations: theory and algorithms. *SIAM J. Numer. Anal.* **41**, 325–363 (2003)
60. Song, X., Richards, P.G.: Seismological evidence for differential rotation of the Earth’s inner core. *Nature* **382**, 221–224 (1996)
61. Tavakoli, F.B., Ribodetti, A., Virieux, J., Operto, S.: An iterative factored eikonal solver for TTI media. SEG Technical Program Expanded Abstracts 2015, pp. 3576–3581 (2015)
62. Tanimoto, T., Anderson, D.L.: Lateral heterogeneity and azimuthal anisotropy of the upper mantle: Love and Rayleigh waves 100–250 sec. *J. Geophys. Res.* **90**(B2), 1842–1858 (1985)
63. Treister, E., Haber, E.: A fast marching algorithm for the factored eikonal equation. *J. Comput. Phys.* **324**, 210–225 (2016)
64. Treister, E., Haber, E.: A multigrid solver to the Helmholtz equation with a point source based on travel time and amplitude. *Numer. Linear Algebra Appl.* **26**, e2206 (2019)
65. Tsai, Y.-H., Cheng, L.-T., Osher, S., Zhao, H.-K.: Fast sweeping algorithms for a class of Hamilton–Jacobi equations. *SIAM J. Numer. Anal.* **41**, 673–694 (2003)
66. Tsai, Y.R.: Rapid and accurate computation of the distance function using grids. *J. Comput. Phys.* **178**, 175–195 (2002)
67. Tsitsiklis, J.N.: Efficient algorithms for globally optimal trajectories. *IEEE Trans. Autom. Control* **40**, 1528–1538 (1995)
68. van Trier, J., Symes, W.W.: Upwind finite-difference calculations of traveltimes. *Geophysics* **56**, 812–821 (1991)
69. Vidale, J.: Finite-difference calculation of travel times. *Bull. Seismol. Soc. Am.* **78**, 2062–2076 (1988)
70. Virieux, J., Le Bouteiller, P., Métivier, L., Benjemma, M.: An accurate discontinuous Galerkin method for solving point-source eikonal equation in 2-D heterogeneous anisotropic media. *Geophys. J. Int.* **212**(3), 1498–1522 (2017)
71. Waheed, U.B., Yarman, C.E., Flagg, G.: An iterative, fast-sweeping-based eikonal solver for 3d tilted anisotropic media. *Geophysics* **80**(3), C49–C58 (2015)
72. Zhang, L., Rector, J.W., Hoversten, G.M.: Eikonal solver in the celerity domain. *Geophys. J. Int.* **162**, 1–8 (2005)
73. Zhang, Y.-T., Chen, S., Li, F., Zhao, H.-K., Shu, C.-W.: Uniformly accurate discontinuous Galerkin fast sweeping methods for eikonal equations. *SIAM J. Sci. Comput.* **33**(4), 1873–1896 (2011)
74. Zhang, Y.-T., Zhao, H.-K., Qian, J.: High order fast sweeping methods for static Hamilton–Jacobi equations. *J. Sci. Comput.* **29**, 25–56 (2006)
75. Zhao, H.-K.: A fast sweeping method for eikonal equations. *Math. Comput.* **74**, 603–627 (2005)
76. Zhao, H.-K.: Parallel implementations of the fast sweeping method. *J. Comput. Math.* **25**, 421–429 (2007)
77. Zhou, B., Greenhalgh, S.: Raypath and traveltime computations for 2d transversely isotropic media with dipping symmetry axes. *Explor. Geophys.* **37**(2), 150–159 (2006)