A Level Set Method for Three-dimensional Paraxial Geometrical Optics with Multiple Sources

Shingyu Leung* Jianliang Qian† Stanley Osher‡

July 7, 2004

Abstract

We apply the level set method to compute the three dimensional multivalued geometrical optics term in the paraxial formulation. The paraxial formulation is obtained from the 3-D stationary eikonal equation 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. The multivalued traveltime and amplitude-related quantity are obtained from solving advection equations with source terms. We derive an amplitude formula in the reduced phase space which is very convenient to use in the level set framework. By using a semi-Lagrangian method in the paraxial formulation, the method has $O(N^2)$ rather than $O(N^4)$ memory storage requirement in the five dimensional phase space, where N is the number of mesh points along one direction. Although the computational complexity is still $O(MN^4)$, where M is the number of steps in the ODE solver for the semi-Lagrangian scheme, this disadvantage is largely overcome by the fact that up to $O(N^2)$ multiple sources can be treated simultaneously. Three dimensional numerical examples demonstrate the efficiency and accuracy of the method.

1 Introduction

Consider the geometrical optics (high frequency) approximation for 3-D acoustic wave equations. According to the Debye procedure, the leading order term defining the geometrical optics term consists of two functions, one being the eikonal satisfying the eikonal equation,

$$|\nabla \tau| = \frac{1}{c}, \tag{1}$$

*Department of Mathematics, UCLA, Los Angeles, CA 90095-1555. Email: **syleung@math.ucla.edu**
†Department of Mathematics, UCLA, Los Angeles, CA 90095-1555. Email: **qian@math.ucla.edu**
‡Department of Mathematics, UCLA, Los Angeles, CA 90095-1555. Email: **sjo@math.ucla.edu**

and the other being the amplitude solving the transport equation,

$$\nabla \cdot (A^2 \nabla \tau) \quad = \quad 0, \tag{2}$$

where τ and A are the eikonal, a.k.a., traveltime in many applications, and the amplitude function respectively; c is the given wave velocity in an acoustic medium. These two equations appear in a variety of applications: high resolution seismic imaging [15, 4, 24, 30], underwater acoustics, semi-classical limit for quantum mechanics [22], and optical instruments, to name just a few.

As a first order scalar nonlinear PDE, equation (1) usually does not admit a global, smooth solution in the physical space. By viewing the gradient components as independent functions of some parameter, for a given non-characteristic boundary condition the method of characteristics yields a smooth solution for equation (1) in the phase space; but once projected to the physical space, the solution usually is multivalued. The concept of viscosity solution singles out a unique, physically relevant weak solution among these multivalued branches of solutions, so that a continuous, global solution is well defined in this class. However, in many applications, the multivalued solution is necessary and also physically relevant. In seismics, the later arrival traveltimes which do not correspond to the viscosity solution of the eikonal equation may carry more significant energy than the viscosity solution-based first arrivals do [12, 24].

Naturally, one may use the method of characteristics to derive a set of ODEs to compute traveltimes and amplitudes in the phase space, which essentially yields everything. However, the major disadvantage of the approach is that it lacks control of resolution of the solution in the physical space. Certainly, the shortcoming can be overcome to some extent by extra bookkeeping of data structures [40]. On the other hand, one may look for a PDE framework to compute these multivalued solutions in the phase space; this results in the so-called Eulerian geometrical optics [3, 8]. In the past decade, there are a lot of efforts in this direction: domain decomposition along caustic curves [2], the slowness matching method [37, 38], methods based on kinetic formulation [13, 16, 14], dynamical surface extension methods [31, 33], and Liouville equation based methods, such as the the segment projection method [9], the vectorial level set method [25, 27, 18, 5, 6, 29, 28, 17], the method based on escape parameters [11].

Extend the traveltime function $\tau{=}\tau(x, y, z)$ defined in a bounded spatial domain Ω to $T{=}T(x, y, z, \theta, \phi)$ defined in the reduced phase space via the slowness vector, equation (12). Consider its t-wavefront in the reduced phase space:

$$T(x, y, z, \theta, \psi) \quad = \quad t, \tag{3}$$

which consists of all the wavefronts starting from all the sources on the boundary $\partial\Omega$ and all the take-off directions pointing into the domain Ω and reaching the spatial location (x, y, z) with arrival angles (θ, ψ).

Now differentiate this identity w.r.t. t and use the ray tracing system (13),

$$\mathbf{w} \cdot \nabla_{x,y,z,\theta,\psi} T \quad = \quad 1, \tag{4}$$

where $\mathbf{w}$ denotes the right hand side of the system (13).

As pointed out in [5], the main issue is to identify those (θ, ψ) such that the slowness vector satisfies

$$\mathbf{p}(\theta, \psi) \quad = \quad \nabla_{x,y,z} T; \tag{5}$$

then the eikonal equation is satisfied locally.

In the slowness matching method developed in [37, 38], this condition is enforced by solving many point source problems, equation (19), directly in the physical space, which amounts to constructing many local fundamental solutions for the eikonal equation and identifying those slownesses satisfying (5) by the slowness matching condition. This method is highly efficient if traveltimes from multiple sources are desired as argued carefully in [38]. In fact it is the only method so far that stays in the physical space and at the same time resolves multivalued solutions.

To use the Liouville equation based phase space formulation more efficiently, we developed the level set method for the two-dimensional paraxial multivalued geometrical optics in [29, 28]. In this work, we continue to develop efficient level set methods for three dimensional multivalued geometrical optics in the paraxial formulation. In this case, the full Liouville equation is five dimensional; by using the paraxial assumption, we essentially reduce the problem by one dimension and equation (4) becomes

$$T_z + \mathbf{u} \cdot \nabla_{x,y,\theta,\psi} T = \frac{1}{c \sin\theta \cos\psi}\,; \tag{6}$$

see Section 3.2 for further details. Our framework provides multivalued geometrical optics terms for multiple sources simultaneously in that we are able to make use of the information from not only the zero level set but also all the nonzero level sets. By using a global semi-Lagrangian method to solve level set equations, the physical space variables are directly linked to the phase variables so that the condition (5) can be resolved efficiently and the computational memory requirement is reduced from $O(N^4)$ to $O(N^2)$, where N is the number of mesh points along one direction. In comparison to the usual finite difference discretization of the level set equations which requires $O(N^4)$ memory storage, this saving is very significant in the computational space of five dimensions. Although the method proposed here has the computational complexity $O(MN^4)$, where M is the number of steps in the ODE solver for the semi-Lagrangian scheme, it can handle up to N^2 multiple sources simultaneously; therefore, overall it is still very efficient if the geometrical optics terms for multiple sources are needed as in seismics [38].

In the method based on escape parameters [11], to identify those (θ, ψ) such that the condition (5) holds, the authors made use of the fact that the source locations and takeoff angles are constant along the rays; therefore, five Liouville equations are used to advect these constants as tags for the rays so that the traveltime T at (x, y, z, θ, ψ) can be distinguished by checking the tags at (x, y, z, θ, ψ). Then postprocessing is used to solve for (θ, ψ) satisfying (5) at (x, y, z) for specified source locations. This approach is efficient if traveltimes from up to N^2 multiple sources are desired by using the similar argument as in [38]. In terms of the paraxial formulation (6), the computational complexity and memory requirement of this approach are both of $O(N^5)$.

Our approach shares some similarities with [11], but their formulation is only for static HJ equations, and ours can be viewed for "artificial time" dependent HJ equa-

tions in terms of z-dependent paraxial formulation. In [11], the Liouville equations are solved by combining a local semi-Lagrangian and Dijkstra-like fast marching method, and the resulting computational memory requirement is $O(N^5)$ in the five dimensional reduced phase space. In this work, we explore a global semi-Lagrangian approach to solve the paraxial Liouville equations so that high order ODE solvers can be used right away, and the resulting method in our setup has the advantage that the computational memory requirement is only of $O(N^2)$.

In the five dimensional reduced phase space, the condition (5) defines an object of co-dimension 2; therefore the object can be resolved by the intersection of zero level sets of three functions defined in (x, y, z, θ, ψ) space. This is commonly used level set method, first proposed to compute multivalued phases in the high frequency asymptotics for acoustic wave equations in [25], 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 [27, 6], where the multivalued solutions for a class of steady Hamilton-Jacobi equations were computed and illustrated as self-intersecting wavefronts as well. It was also extended to compute multivalued wavefronts and multivalued phases in the high frequency asymptotics for the Schrödinger equation in [5], 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 time dependent and steady Hamilton-Jacobi equations in [18], where a level set formulation for handling the gradient of the solution was 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. The multivalued solution was provided by adding more Liouville equations and independent variables without increasing the formal complexity in [5, 20]. In [17], the level set method is applied to computing multivalued physical observables for the semi-classical limit of the Schrödinger equation.

The advantage of following only zero level sets is that local level set methods can be designed so that the memory requirement and computational complexity are optimal to some extent [1, 26]. The PDE based local level set method, first proposed in [26] for single level set motions and generalized in [25, 28] for vectorial level set motions, is one of the possible approaches to reduce the computational complexity from $O(N^5)$ to $O(N^3 LogN)$ in the paraxial formulation; but its memory requirement is still of $O(N^4)$. The tree-based local level set method also only follows the zero level set by putting more nodes near the zero level set [34, 23]; this reduces the computational complexity from $O(N^5)$ to $O(N^3 LogN)$, and the memory requirement $O(N^4)$ to $O(N^2 LogN)$ on average. The semi-Lagrangian method proposed in [23], which in turn is an extension of [34], is based on short time evolution of an Euler step, so that the zero level set can be tracked locally using a dyadic tree structure. Since such local level set methods follow only zero level sets, they can handle only wave propagation with a single source.

Given multiple sources, we can certainly initialize the level set function so that their initial zero level sets represent those multiple sources. With this initialization, one can still get all the multi-arrival rays from multiple sources. However, on the later times, one cannot distinguish rays from different sources because *all* rays are represented by the same intersection of zero level sets essentially. To treat multiple sources in the

framework of using only zero level sets, it is possible to use the idea proposed in [5] by augmenting the reduced phase space with one more dimension such that the solutions for multiple sources are viewed as graphs in the augmented phase space. This idea is worth exploring further.

If limited to a single source, then our approach can also be viewed in the framework of [5]. Thus the work presented here serves as a link between the work in [25, 27, 5] and that in [11].

The rest of the paper is organized as follows: Section 2 presents the paraxial formulation for the 3-D eikonal equation; Section 3 presents the level set formulation for multivalued wavefronts, traveltimes and amplitudes; Section 4 gives implementation details for the level set method; Section 5 demonstrates the accuracy of the proposed semi-Lagrangian level set method with extensive numerical examples; Section 6 concludes the paper.

2 3D Paraxial Formulation for Eikonal Equation

Consider a point source condition for the 3-D eikonal equation defined in an open, bounded domain $\Omega \subset \mathbf{R}^3$. To emphasize the point source condition the eikonal equation is rewritten as follows,

$$|\nabla_{\mathbf{x}} \tau(\mathbf{x}, \mathbf{x}_s)| = \frac{1}{c(\mathbf{x})}. \tag{7}$$

$$\lim_{\mathbf{x}\to\mathbf{x}_s} \frac{\tau(\mathbf{x}, \mathbf{x}_s)}{|\mathbf{x} - \mathbf{x}_s|} = \frac{1}{c(\mathbf{x}_s)}, \ \tau \geq 0, \tag{8}$$

where $\mathbf{x}_s$ is the given source point.

To apply the method of characteristics, we first parameterize the 3-D unit vectors by spherical coordinates. Points on a unit sphere, away from the x-axis, can be uniquely represented by the following rotated spherical coordinates,

$$\begin{aligned} x &= \cos\theta \\ y &= \sin\theta \sin\psi \\ z &= \sin\theta \cos\psi\,, \end{aligned} \tag{9}$$

where $\theta \in (0, \pi)$ is the angle between the point and the positive x-axis, and $\psi \in [-\pi, \pi)$ is the angle between the positive z-axis and the projection of the point onto the y-z plane; Figure 1 shows the standard and rotated spherical coordinates. Then the slowness vector $\nabla\tau = \mathbf{p}$ can be represented as

$$p_1 = \frac{\cos\theta}{c} \tag{10}$$

$$p_2 = \frac{\sin\theta \sin\psi}{c} \tag{11}$$

$$p_3 = \frac{\sin\theta \cos\psi}{c} \tag{12}$$

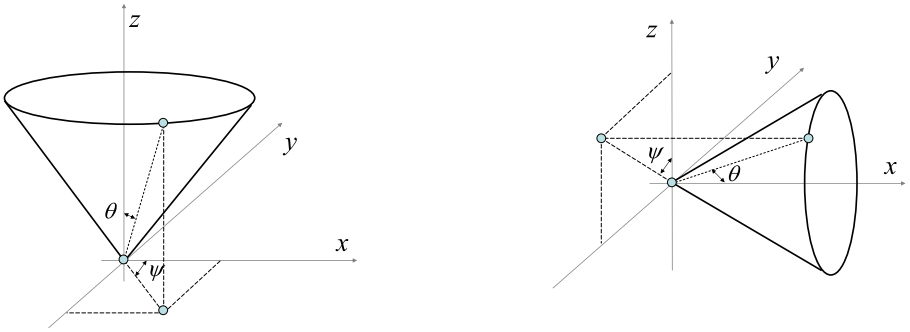

Figure 1: Spherical coordinates and rotated Spherical coordinates

By using the above parameterization we have the following ray tracing system

$$\begin{aligned}
\frac{dx}{dt} &= c\cos\theta \\
\frac{dy}{dt} &= c\sin\theta\sin\psi \\
\frac{dz}{dt} &= c\sin\theta\cos\psi \\
\frac{d\theta}{dt} &= \sin\theta\frac{\partial c}{\partial x} - \cos\theta\left(\cos\psi\frac{\partial c}{\partial z} + \sin\psi\frac{\partial c}{\partial y}\right) \\
\frac{d\psi}{dt} &= \frac{1}{\sin\theta}\left(\sin\psi\frac{\partial c}{\partial z} - \cos\psi\frac{\partial c}{\partial y}\right)
\end{aligned} \tag{13}$$

with initial conditions

$$\begin{aligned}
x|_{t=0} &= x_s \\
y|_{t=0} &= y_s \\
z|_{t=0} &= z_s \\
\theta|_{t=0} &= \theta_s \\
\psi|_{t=0} &= \psi_s
\end{aligned} \tag{14}$$

where $\mathbf{x}_s = (x_s, y_s, z_s)$ and θ_s and ψ_s vary from 0 to π and from $-\pi$ to π respectively. One can compare this formulation with the one from [19, 25]. We use this rotated spherical coordinates rather than the standard spherical coordinates system because points on the z-axis can now be uniquely represented. If the spherical coordinates was used, for non-constant media ($c \neq$ const), the ray tracing system would not be well-posed even for the ray travels vertically upward. Although points on the x-axis in this rotated coordinate system may still seem to cause problem, these points are actually out of our computational domain according to the paraxial assumption which will be discussed below.

Next we extend the arrival-time function $\tau(x, y, z)$ to the reduced phase space $\{(x, y, z, \theta, \psi)\}$, and denoted as $T(x, y, z, \theta, \psi)$, and consider the t-wavefront expanding

from the source point:

$$T(x, y, z, \theta, \psi) \quad = \quad t. \tag{15}$$

Differentiating this identity with respect to t, we have

$$\frac{dx}{dt}T_x + \frac{dy}{dt}T_y + \frac{dz}{dt}T_z + \frac{d\theta}{dt}T_\theta + \frac{d\psi}{dt}T_\psi = 1 \tag{16}$$

with the boundary condition

$$T(x_s, y_s, z_s, \theta_s, \psi_s) = 0\,, \tag{17}$$

for $0 \le \theta_s \le 2\pi$ and $-\pi \le \psi_s \le \pi$.

Since equation (16) is a linear advection equation, one may be tempted to solve it directly with the condition (17). However, for a given $(x, y, z) \ne (x_s, y_s, z_s)$, $T(x, y, z, \cdot, \cdot)$ is not necessarily well defined for every θ and ψ. In other words, equations (16) and (17) are not well-posed. To obtain an well-posed problem, we will assume the paraxial condition and use the level set formulation.

In some applications, for example wave propagation in reflection seismics [7], the arrival-times of interest are carried by the so-called sub-horizontal rays [15, 36, 30], where sub-horizontal means "oriented in the positive z-direction". A convenient characterization for sub-horizontal rays is that

$$\frac{dz}{dt} \ge c \sin\theta_{\max} \cos\psi_{\max} > 0 \tag{18}$$

for some $\theta_{\max}$ and $\psi_{\max}$ with $\pi/2 < \theta_{\max} < \pi$ and $0 < \psi_{\max} < \pi/2$. This inequality holds for rays making angles θ and ψ satisfying $|\theta| \le \theta_{\max} < \pi$ and $|\psi| \le \psi_{\max} < \pi/2$.

The traveltimes corresponding to these sub-horizontal rays satisfy the following paraxial eikonal equation

$$\frac{\partial\tau}{\partial z} = \quad = \quad \sqrt{\max\left(\frac{1}{c^2} - \left(\frac{\partial\tau}{\partial x}\right)^2 - \left(\frac{\partial\tau}{\partial y}\right)^2, \frac{\sin^2\theta_{\max}\cos^2\psi_{\max}}{c^2}\right)}. \tag{19}$$

To be specific, consider

$$\Omega \quad = \quad \{(x, y, z) : x_{\min} \le x \le x_{\max}, y_{\min} \le y \le y_{\max}, 0 \le z \le z_{\max}\} \tag{20}$$

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

$$\begin{aligned} x_z &= \frac{1}{\cos\psi\tan\theta} \\ y_z &= \tan\psi \\ \theta_z &= \frac{c_x}{c\cos\psi} - \frac{c_z + c_y\tan\psi}{c\tan\theta} \\ \psi_z &= \frac{c_z\tan\psi - c_y}{c\sin^2\theta}\,, \end{aligned} \tag{21}$$

with $x \in [x_{\min}, x_{\max}]$, $y \in [y_{\min}, y_{\max}]$, $\theta \in [\epsilon_\theta, \pi - \epsilon_\theta]$ and $\psi \in [\epsilon_\psi - \pi/2, \pi/2 - \epsilon_\psi]$.

3 Level Set Formulation

3.1 Representation of a Point Source

We first assume that a single source is located at the origin. Therefore, rays from this source can now be represented as the intersection of zero level sets of two level set functions, $\phi^1(z;x,y,\theta,\psi)$ and $\phi^2(z;x,y,\theta,\psi)$.

Differentiating the zero level set of these functions with respect to z, we get the following level set equations which govern the motion of the corresponding zero level sets,

$$\phi^m_z + \frac{dx}{dz}\phi^m_x + \frac{dy}{dz}\phi^m_y + \frac{d\theta}{dz}\phi^m_\theta + \frac{d\psi}{dz}\phi^m_\psi = 0 \tag{22}$$

for $m = 1, 2$. Or, these equations can be rewritten as

$$\phi^m_z + \mathbf{u}\cdot\nabla_{x,y,\theta,\psi}\phi^m = 0 \tag{23}$$

for $m = 1, 2$ where the velocity field $\mathbf{u} = (u^1, u^2, u^3, u^4)$ is given by the ray tracing system (21).

On $z = 0$, we initialize the level set functions by

$$\phi^1(0;x,y,\theta,\psi) = x \ \text{ and } \ \phi^2(0;x,y,\theta,\psi) = y\,. \tag{24}$$

3.2 Arrival-time

Arrival-time can be computed by inverting the third equation in (13) locally. This gives

$$T_z + \mathbf{u}\cdot\nabla_{x,y,\theta,\psi}T = \frac{1}{c\sin\theta\cos\psi}\,, \tag{25}$$

where $T = T(z;x,y,\theta,\psi)$.

To obtain the multivalued arrival-time on $z = z^*$, we first solve equations (23) and (25) up to $z = z^*$. We then compute the intersection of the zero level sets, denoted by

$$\Sigma_{\mathbf{0}} = \{(x,y,\theta,\psi) : \phi^1(z^*;x,y,\theta,\psi) = \phi^2(z^*;x,y,\theta,\psi) = 0\} \subset \mathbb{R}^4\,. \tag{26}$$

The arrival-times at $(z^*;x,y)$ can be determined by projecting $T(z^*;\Sigma_{\mathbf{0}})$ onto the x-y plane.

3.3 Representation of Multiple Sources

However, we have noticed that the level set functions contain much more information than what we can have used in the above algorithm since not only the zero level set but also the non-zero level sets are also useful. Rays emanating from a source in the phase space are not necessarily represented by the intersection of two *zero* level sets only. We can define rays from a source at a location (x_s, y_s) by the intersection of $\{\phi^1 = x_s\}$ and $\{\phi^1 = y_s\}$. Under the same velocity field, given by $\mathbf{u}$, to find all rays on $z = z^*$ from this source (x_s, y_s), one only needs to determine the set

$$\Sigma_{\mathbf{x}_s} = \{(x,y,\theta,\psi) : \phi^1(z^*;x,y,\theta,\psi) - x_s = \phi^2(z^*;x,y,\theta,\psi) - y_s = 0\}\,, \tag{27}$$

rather than the one defined by equation (26). In other words, using the initial conditions (24), a point $(x_i, y_j, \theta_k, \psi_l)$ on $z = z^*$ can be connected to the point

$$(\phi^1(z^*; x_i, y_j, \theta_k, \psi_l), \phi^2(z^*; x_i, y_j, \theta_k, \psi_l), \theta_0, \psi_0)$$

on $z = 0$ through the characteristic curve of (23) for some initial conditions (θ_0, ψ_0). More importantly, the initial conditions (24) can be reinterpreted as follows. All points, not only (x_i, y_j), but also $\forall x \in [x_{\min}, x_{\max}]$ and $\forall y \in [y_{\min}, y_{\max}]$, can be treated as locations of sources. Therefore, we are able to determine multi-arrival rays from multiple sources.

The representation of point sources discussed above is of course not the only way that we can represent rays from multiple sources. One can actually define $\Sigma_{\mathbf{0}}$ to denote all rays from *multiple* sources. The only thing we need to modify is the initial conditions. $\phi^1(0; x, y, \theta, \psi)$ and $\phi^2(0; x, y, \theta, \psi)$ need to be independent of the angles θ and ψ, and $\{\phi^1(0; x, y) = 0\} \cap \{\phi^2(0; x, y) = 0\}$ at locations of those sources. With this formulation, one can still get all the multi-arrival rays from multiple sources. But, on the level $z = z^*$, one cannot distinguish rays from different sources because all rays are represented by the same intersection of zero level sets essentially. However, using the formulation proposed here, we can separate the rays from different sources by using only one set of level set functions and making use of all the available level sets.

3.4 Amplitude

The amplitude of the ray can also be computed using the current formulation. Defining $\tilde{A} = \tilde{A}(z; x, y)$, we have

$$\tilde{A}(z; x, y) = \frac{c}{4\pi\sqrt{c_0}} \sqrt{\sin\tilde{\Theta} \left| \frac{\partial(\tilde{T}, \tilde{\Theta}, \tilde{\Psi})}{\partial(x, y, z)} \right|} \tag{28}$$

where $\tilde{T} = \tilde{T}(z; x, y)$, $\tilde{\Theta} = \tilde{\Theta}(z; x, y)$ and $\tilde{\Psi} = \tilde{\Psi}(z; x, y)$ are the arrival-time, take-off angles of θ and ψ from the source located at $\mathbf{x} = \mathbf{x}_s$, respectively. Following the approach in [28], we first extend all these functions into the phase space, denoted as T, Θ and Ψ, respectively. Using (23) and (25), we obtain

$$A(z; x, y, \theta, \psi) = \frac{1}{4\pi} \sqrt{\frac{c \sin\Theta}{c_0 \sin\theta \cos\psi}} \sqrt{\frac{\Delta_1}{\Delta_2}}, \tag{29}$$

where Θ and θ are the takeoff angle and the arrival angle respectively, and Δ_1 and Δ_2 are the Jacobians of the transformation given by

$$\Delta_1 = \begin{vmatrix} \phi^1_x & \phi^2_x & \Theta_x & \Psi_x \\ \phi^1_y & \phi^2_y & \Theta_y & \Psi_y \\ \phi^1_\theta & \phi^2_\theta & \Theta_\theta & \Psi_\theta \\ \phi^1_\psi & \phi^2_\psi & \Theta_\psi & \Psi_\psi \end{vmatrix} \tag{30}$$

and

$$\Delta_2 = \begin{vmatrix} \phi^1_\theta & \phi^2_\theta \\ \phi^1_\psi & \phi^2_\psi \end{vmatrix}. \tag{31}$$

See the appendix for derivation of these formulae.

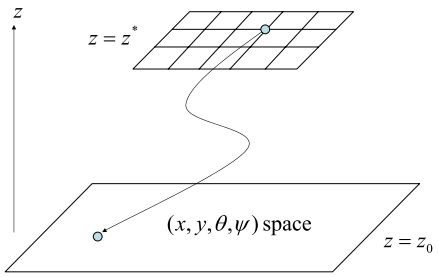

Figure 2: Semi-Lagrangian method to solve the advection equations.

4 Numerical Method

4.1 Level Set Equations

One way to solve equations (23) and (25) is to use, for example, RK3 in the z-direction and WENO5 upwind scheme in the x-y-θ-ψ space [21, 32]. This is a typical Eulerian approach, where grid points are fixed in space. Computational complexity of this method is therefore $O(N^5 LogN)$. This approach is not efficient in this high dimensional space. One reason is that, if we want to compute the multivalued arrivals from multiple sources, we probably do not want to use some localized level set methods, like the one proposed in [26]. In this case, keep tracking of multiple layers of level set functions with reinitializations and extensions will take a large portion of computational time.

Another potential difficulty is the limitation from the CFL condition by solving these hyperbolic equations. For this Eulerian approach, the step, Δz, for each z-direction marching is of $O(\min(\Delta x, \Delta y, \Delta\theta, \Delta\psi))$. This is acceptable for lower dimensional computations, like those in [29, 28] where the dimension involved is only 1+2 (time-like direction plus space directions). However, for the current problem, it is unreasonable to spend days of computations in solving these linear advection equations.

In this paper, we implement a semi-Lagrangian method [35, 10] to determine the values of ϕ^m at $z = z^*$ for $m = 1, 2$. Solving the level set equations with the method of characteristics, we have

$$\phi^m = \text{ const} \tag{32}$$

for $m = 1, 2$ along the characteristics given by (21). Following the idea of semi-Lagrangian methods, we trace back in the z-direction until $z = z_0$, i.e. we solve

$$\frac{d(\hat{x}, \hat{y}, \hat{\theta}, \hat{\psi})}{dz} = \mathbf{u} \tag{33}$$

for $(\hat{x}, \hat{y}, \hat{\theta}, \hat{\psi})|_{z=z_0}$ with "initial" conditions $(\hat{x}, \hat{y}, \hat{\theta}, \hat{\psi})|_{z=z^*} = (x, y, \theta, \psi)$ at the current point, as shown in Figure 2. Then the level set values are assigned as

$$\begin{aligned} \phi^1(z^*; x, y, \theta, \psi) &= \hat{x}|_{z=z_0} \\ \phi^2(z^*; x, y, \theta, \psi) &= \hat{y}|_{z=z_0}\,. \end{aligned} \tag{34}$$

Numerically, the above ODE system is solved using RK3, and the step size is independent of the number of grid points used in the computational domain. By using this semi-Lagrangian approach, the computational complexity drops to $O(MN^4)$, where M is the number of iterations in the z-direction and N is the number of grid points in each of x-y-θ-ψ direction. Different from the finite difference Eulerian approach, the factor M is independent of N and is chosen mainly for the purpose of *accuracy*.

A simpler case is to consider only one source on $z = z_0$. Then the order of complexity of using the global level set method approach as in [29] can still be reduced by a factor of N^2 to $O(N^3 LogN)$ if a local level set method, for example the one from [26], is applied. However, the memory requirement, which will be addressed later in Section 4.3, may still make the local level set method difficult to be implemented. Moreover, to compute arrival solutions from N^2 sources individually, one may need to solve N^2 times localized level set equations which makes the overall computational complexity back to $O(N^5 LogN)$.

4.2 Arrival-time Equation

For the arrival-time equation, we have

$$\frac{DT}{Dz} = \frac{1}{c\sin\theta\cos\psi}\,, \tag{35}$$

with $\frac{D}{Dz}$ is the material derivative given by

$$\frac{D}{Dz} = \frac{\partial}{\partial z} + \mathbf{u}\cdot\nabla = \frac{\partial}{\partial z} + x_z\frac{\partial}{\partial x} + y_z\frac{\partial}{\partial y} + \theta_z\frac{\partial}{\partial\theta} + \psi_z\frac{\partial}{\partial\psi}\,. \tag{36}$$

Therefore, we have

$$T = \int_\Gamma \frac{ds}{c\sin\theta\cos\psi} \tag{37}$$

where $T(0; x, y, \theta, \psi) = 0$ by the reciprocity and Γ is the characteristic given by the system (33). Again, RK3 is used to integrate the arrival-time of rays.

4.3 Multivalued Arrival-times

After we solve for ϕ^m (m=1,2) and T on the grid points at the time level $z = z^*$, we need to compute the intersection of the level surface $\{\phi^1 = x_s\}\cap\{\phi^2 = y_s\}$. To simplify this computation, we discretize the $\theta - \psi$ space for each point (x_i, y_j) in the following way. We first use rectangular grids in the $\theta - \psi$ space, giving (θ_k, ψ_l). One more grid point, denoted by $(\theta_{k+1/2}, \psi_{l+1/2})$, will then be added to the center of each cell, as seen in Figure 3. Therefore, each grid cell, vertices at (θ_k, ψ_l), (θ_{k+1}, ψ_l), (θ_k, ψ_{l+1}) and $(\theta_{k+1}, \psi_{l+1})$, will be sub-divided into 4 triangles, denoted by $\mathcal{T}_N$, $\mathcal{T}_E$, $\mathcal{T}_S$ and $\mathcal{T}_W$. ϕ^m (m=1,2) and T will be computed at all points $(x_i, y_j, \theta_k, \psi_l)$ and $(x_i, y_j, \theta_{k+1/2}, \psi_{l+1/2})$. On each triangle $\mathcal{T}_{(.)}$, ϕ^1 and ϕ^2 are interpolated linearly. Intersection of the level curves of ϕ^1 and ϕ^2 in each of the triangles $\mathcal{T}_{(.)}$, $\{\phi^1 = x_s\}|_{\mathcal{T}_{(.)}} \cap \{\phi^2 = y_s\}|_{\mathcal{T}_{(.)}}$, is computed, if any. The arrival-time at this intersection point will be interpolated linearly using the values of T at the vertices of the triangle $\mathcal{T}_{(.)}$.

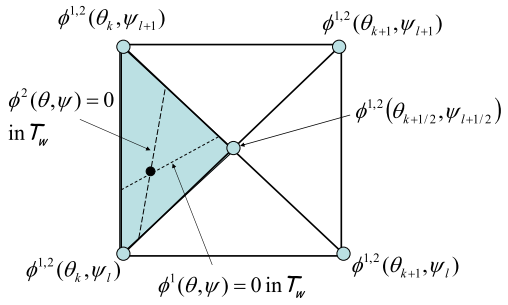

Figure 3: Determining the intersection of level set functions.

We emphasize that we need to solve each of the level set equations (23) and the arrival-time equation (25) only once, even if we care about more than one source on $z = z_0$. It is the intersection of the level surfaces that we need to repeat for each of the sources.

Another advantage of using this semi-Lagrangian method concerns the memory requirement. In the original level set method, where the level set equations are usually solved using WENO5-LF method, values at grid points are coupled together through the ENO-type reconstruction [29, 28]. Therefore, for the current 3D problem, one may need to assign the memory in the following arrays:

```
REAL PHI1(MAXM,MAXM,MAXN,MAXN), PHI2(MAXM,MAXM,MAXN,MAXN)
```

where `MAXM` is the number of grids in the x and y directions, and `MAXN` is the number of grids in the θ and ψ directions. The memory requirement would be $O(N^4)$.

In the Lagrangian or semi-Lagrangian approach, on the other hand, grid points in the $x - y$ space are independent of each other throughout all processes above. Level set function values at any two points $(x_i, y_j, \cdot, \cdot)$ and $(x_{i'}, y_{j'}, \cdot, \cdot)$, with $i \neq i'$ or $j \neq j'$, are determined independently through solving a system of ODE with different initial conditions. Although points in the θ-ψ space are dependent upon each other through the processes in determining $\Sigma_{\mathbf{x}_s}$ and $T(z^*; \Sigma_{\mathbf{x}_s})$, numerically, the memory allocation can still be reduced to

```
REAL PHI1(MAXN,MAXN), PHI2(MAXN,MAXN)
```

for each fixed point (x, y), where `MAXN` is the number of grids in the θ and ψ directions, therefore the memory requirement is $O(N^2)$.

This reduction in the memory requirement may not be significant in the formulation as in [29, 28]. However, it is important in the current calculations in the 1+4 dimensions (time-like direction plus $x-y-\theta-\psi$ directions). Parallel computation is also possible, although it is not implemented at this moment.

4.4 Amplitude

We notice that the takeoff angles Θ and Ψ are the by-products in the above computations of ϕ^1 and ϕ^2. We can simply set

$$\begin{aligned}\Theta(z^*;x,y,\theta,\psi) &= \hat{\theta}(z=z_0)\\ \Psi(z^*;x,y,\theta,\psi) &= \hat{\psi}(z=z_0)\,.\end{aligned} \tag{38}$$

However, in computing Δ_1, instead of numerically differentiating these four functions with respect to x, y, θ and ψ respectively, we notice that this quantity satisfies the equation

$$\frac{D\Delta_1}{Dz} = -(\nabla\cdot\mathbf{u})\Delta_1 \tag{39}$$

where $\Delta_1(0;x,y,\theta,\psi)=1$ by the reciprocity principle and

$$\nabla\cdot\mathbf{u} = \frac{(1+\cos^2\psi)c_z + (\tan\psi\cos^2\psi)c_y}{c\sin^2\theta\cos^2\psi}\,. \tag{40}$$

In turn we solve this equation by introducing $\tilde{\Delta}_1 = \log(\Delta_1)$. Then, similar to the computation of the arrival-time, we integrate

$$\tilde{\Delta}_1 = \int_\Gamma -(\nabla\cdot\mathbf{u})\,ds \tag{41}$$

with $\tilde{\Delta}_1(0;x,y,\theta,\psi)=0$ using RK3. Finally, we have

$$\Delta_1 = \exp(\tilde{\Delta}_1)\,. \tag{42}$$

For the quantity Δ_2, we can simply use the linear interpolants of ϕ^1 and ϕ^2 in the triangle $\mathcal{T}_{(.)}$ when we determining $\Sigma_{\mathbf{x}_s}$. Assuming that those linear interpolants are

$$\begin{aligned}\phi^1_{\mathcal{T}} &= a_{11}\theta + a_{12}\psi + b_1\\ \phi^2_{\mathcal{T}} &= a_{21}\theta + a_{22}\psi + b_2\,,\end{aligned} \tag{43}$$

we have $\Delta_2 = |a_{11}a_{22} - a_{12}a_{21}|$ defined on the triangle $\mathcal{T}_{(.)}$.

To compute $A(z^*;\Sigma_{\mathbf{x}_s})$, we first determine the quantity

$$\alpha = \frac{\Delta_1\sin\Theta}{\cos\psi\sin\theta} \tag{44}$$

at each grid point. Then, we use linear interpolation to find its value at the intersection $\{\phi^1 = x_s\}|_{\mathcal{T}_{(.)}} \cap \{\phi^2 = y_s\}|_{\mathcal{T}_{(.)}}$. Finally, we have

$$A(z^*;\Sigma_{\mathbf{x}_s}) = \frac{1}{4\pi}\sqrt{\frac{c}{c_s}}\sqrt{\frac{(\alpha)_{\Sigma_{\mathbf{x}_s}}}{(\Delta_2)_{\mathcal{T}_{(.)}}}}\,. \tag{45}$$

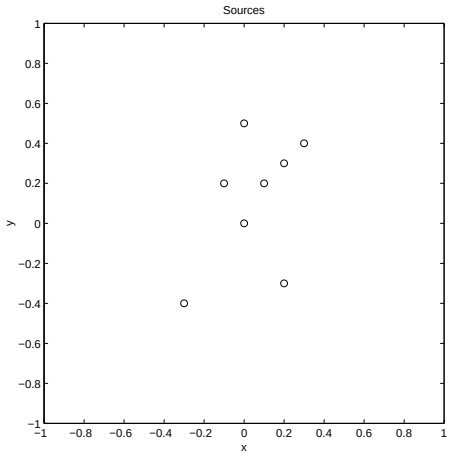

Figure 4: Location of sources on the level of $z = 0$.

5 Numerical Examples

In the following numerical examples, the computational domain is chosen to be

$$\Omega = \{(x, y, \theta, \psi) : x \in [-1, 1], y \in [-1, 1], \theta \in [\pi/20, 19\pi/20], \psi \in [-9\pi/20, 9\pi/20]\}\,. \tag{46}$$

Multiple sources are located on z=0. Their coordinates $(x, y)_s$ (s=1, 2, $\cdots$, 8) are given by $(0, 0)$, $(0.1, 0.2)$, $(0.2, 0.3)$, $(0.3, 0.4)$, $(0, 0.5)$, $(-0.1, 0.2)$, $(-0.3, -0.4)$ and $(0.2, -0.3)$, as shown in Figure 4. It should be emphasized that the number of sources can be up to $O(N^2)$ and their locations can be arbitrary.

5.1 Example 1: Constant Velocity Model

The first example is the constant velocity model where the velocity field is given by $c \equiv 1$. Exact solution of arrivaltimes is given by

$$T(z; x, y) = \sqrt{z^2 + x^2 + y^2}\,. \tag{47}$$

The sub-figures on the first row of Figure 5 show the computed solution. Convergence rate of the proposed method is also shown on the same row. We performed computations using N=$10n$ for $n = 2$ to 5, where N is the number of mesh points in every one of the x, y, θ and ψ directions. The x-axis on the graph represents the logarithm value of $\Delta x \Delta y$, and the y-axis is the logarithm value of the error of the solution measured in the L_1-norm and the L_∞-norm. The numerical rates of convergence, twice of the slopes of the least-squares fitted lines, are 1.90 and 1.86 in terms of two different norms.

The sub-figures in the second row shows some results for the amplitude. To have accurate results, we use 4 times number of grids in the θ-ψ space as that in the x-y space. We also plot the computed solution (in circles) together with the exact solution

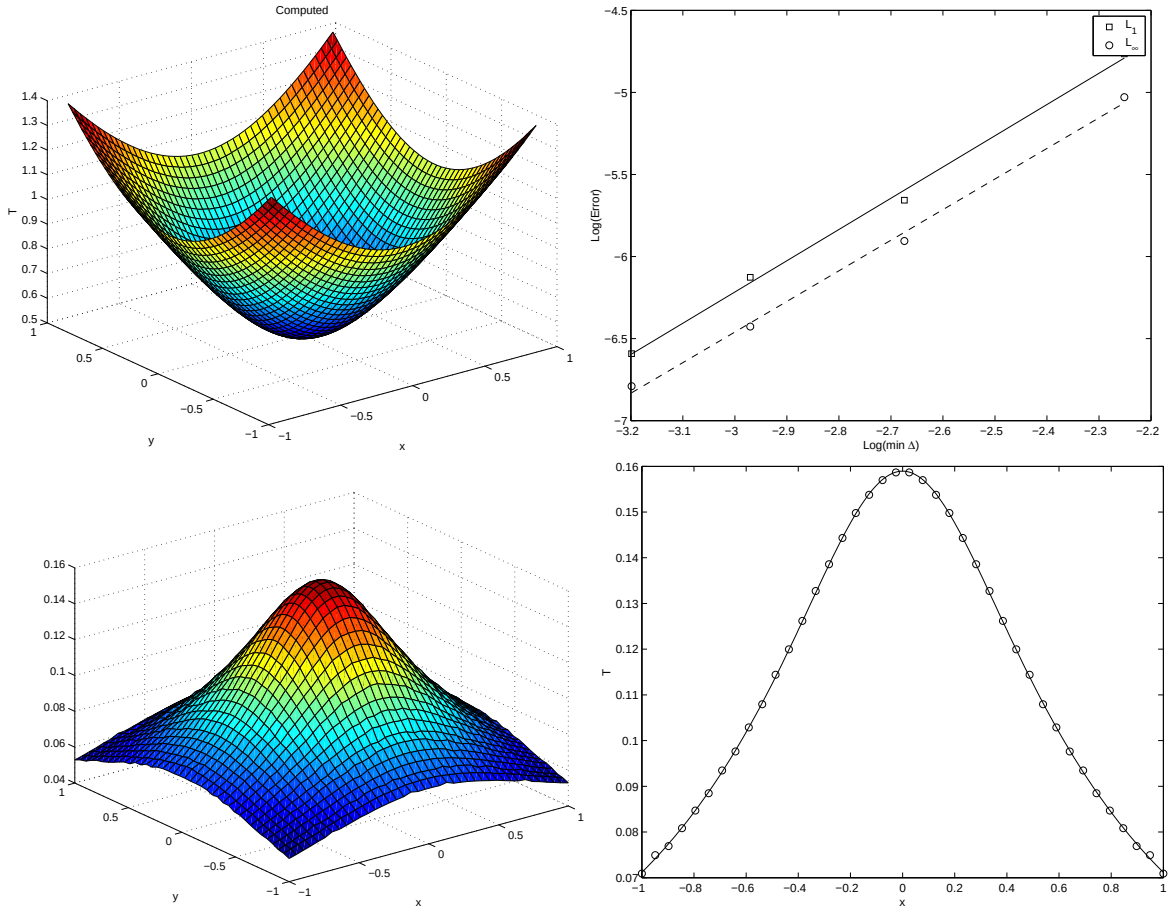

Figure 5: (Example 1) Computed arrivaltime using $1 \leq i, j, k, l \leq 50$ grids in the phase space at $z = 0.5$ and the corresponding convergence test result. Second row shows the amplitude of the arrival rays at the same z using $1 \leq i, j \leq 40$ and $1 \leq k, l \leq 200$. Last figure shows the cross section at $y = 0$ together with the exact solution.

(in solid line), given by

$$A(z;x,y) = \frac{1}{4\pi R} \tag{48}$$

where R is the distance from the source.

5.2 Example 2: Waveguide Model

This velocity model is the same as the one in [29, 28] where

$$c(x,y,z) = 1.1 - \exp(-0.5x^2)\,. \tag{49}$$

Figure 6 shows the multivalued arrivaltimes at z=1.2 with sources located at $(x,y)_s$ (s =1,2,$\cdots$,8), respectively. As the source varies, the traveltime varies for a ray to reach a specific location. There are three sheets of traveltime surfaces at z=1.2 for those given sources. When two sheets connect to each other in the phase space, they connect along the caustic curves, which are shown in the figures; this can also be seen more clearly in Figure 7.

In this case, c_y=c_z=0 and this implies

$$u^3 = \frac{c_x}{c\cos\psi} \text{ and } u^4 = 0\,. \tag{50}$$

Considering rays shooting out from the origin with $\psi = 0$ and using the paraxial ray tracing system (21), we get $\psi(z) \equiv 0$. This means that all the rays with $\psi(z=0) = 0$ from the origin will stay on the cross section $y = 0$ (one can compare our solution with that in [29, 28]). We compare the solution of this cross section with the one from the ray tracing method, as shown in the first sub-figure of Figure 7. The solution from the ray tracing method is plotted in the solid line, while the circles represent the solution from the current formulation. They match with each other very well. In all the sub-figures, we can see that the caustics develop and the traveltime becomes triple valued around caustics.

Figures 8 and 9 show the cross sections of the solution on $y = \mp 1$, respectively.

To look at the solutions more closely, we concentrate on the solution for the source located at $(x,y)_s = (0,0)$. More cross sections of the multivalued arrivaltime are plotted on Figure 10. The locations of the slides are $y = y_j$ for $j = 6, 11, 16, 26, 31, 36$. This corresponds to $y = (j-1)/40$. One can imagine that the leftmost slice is shifted to the left to the right and scaled according to the distance to the origin, since the velocity is a function of the x variable only.

Figure 11 shows the surface plots of the arrivaltime solutions according to the arrival orders. If the eikonal equation is solved directly, one would get the viscosity solution, which corresponds to the first arrivaltime, i.e. the solution on the first sub-figure.

Amplitudes of the arrival rays are also calculated. Figures 12 to 17 show the amplitude solutions corresponding to Figures 6 to 9.

In the calculation of the amplitude, one needs to calculate the Jacobian Δ_2. This quantity can be zero which corresponds to the location of *caustics*. Near caustics, the usual asymptotic expansion of the wave field is not valid anymore. This reflects in the fact that the amplitude blows up, as seen clearly in these figures.

In Figure 17, we can see that the amplitude becomes infinity along the caustic lines in the sub-figures of amplitudes corresponding to the second and third arrivals.

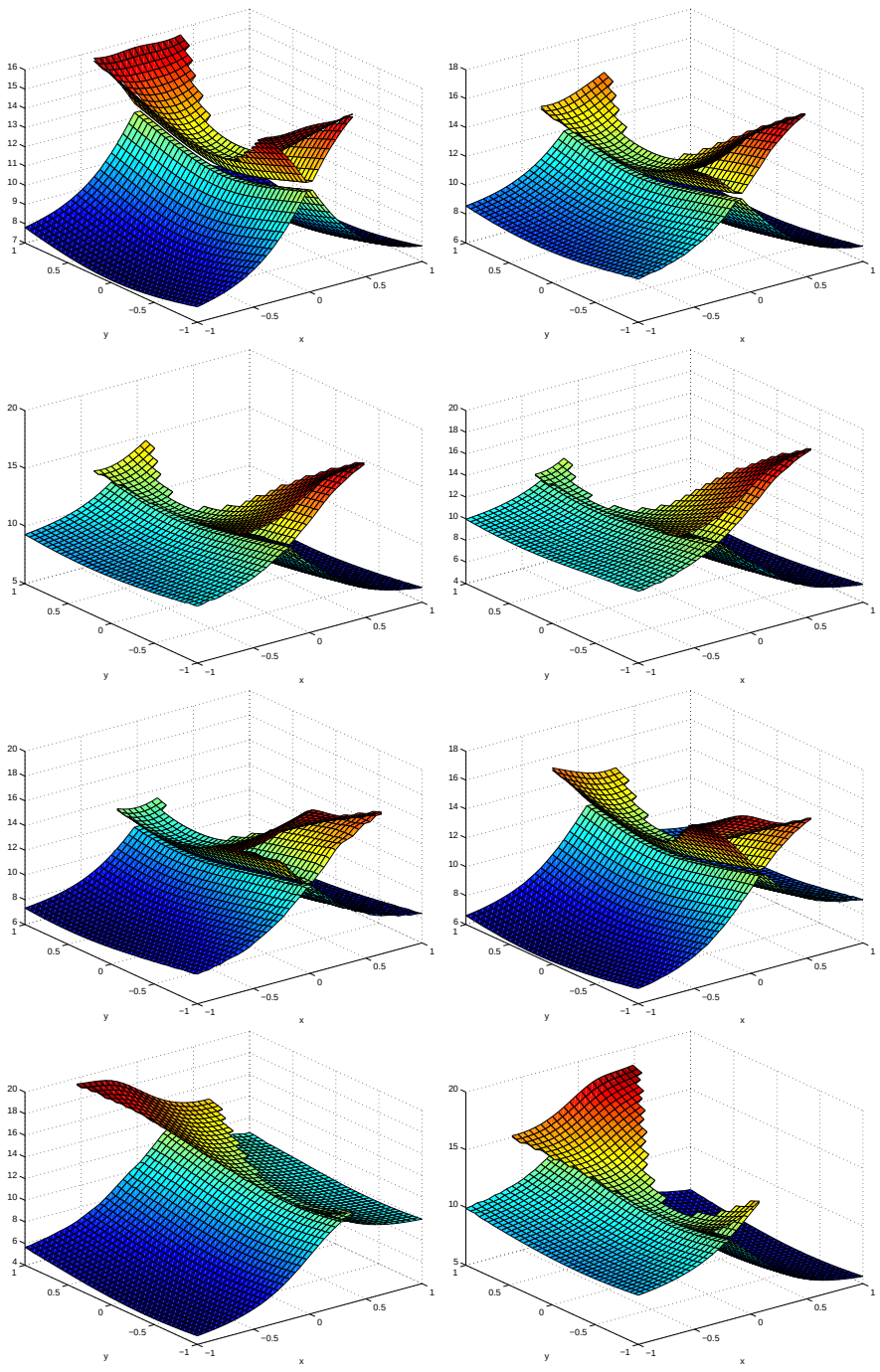

Figure 6: (Example 2) Arrivaltimes at $z = 1.2$ using $1 \leq i, j \leq 41$ and $1 \leq k, l \leq 201$ in the phase space with sources located at $(x, y)_s = (x, y)_{1,2,\cdots,8}$ respectively.

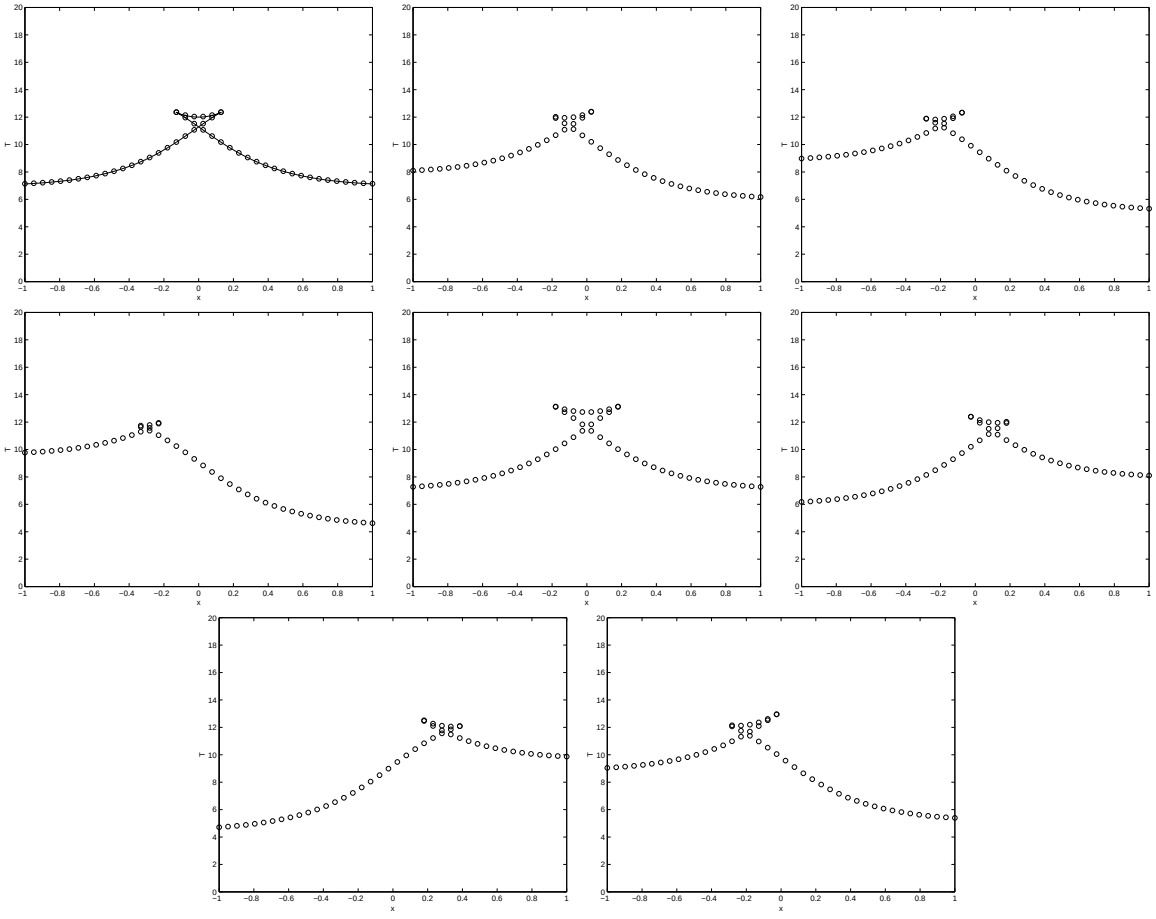

Figure 7: (Example 2) Arrivaltimes at $z = 1.2$ on the cross section $y = 0$ using $1 \leq i, j \leq 41$ and $1 \leq k, l \leq 201$ in the phase space with sources located at $(x, y)_s = (x, y)_{1,2,\cdots,8}$ respectively.

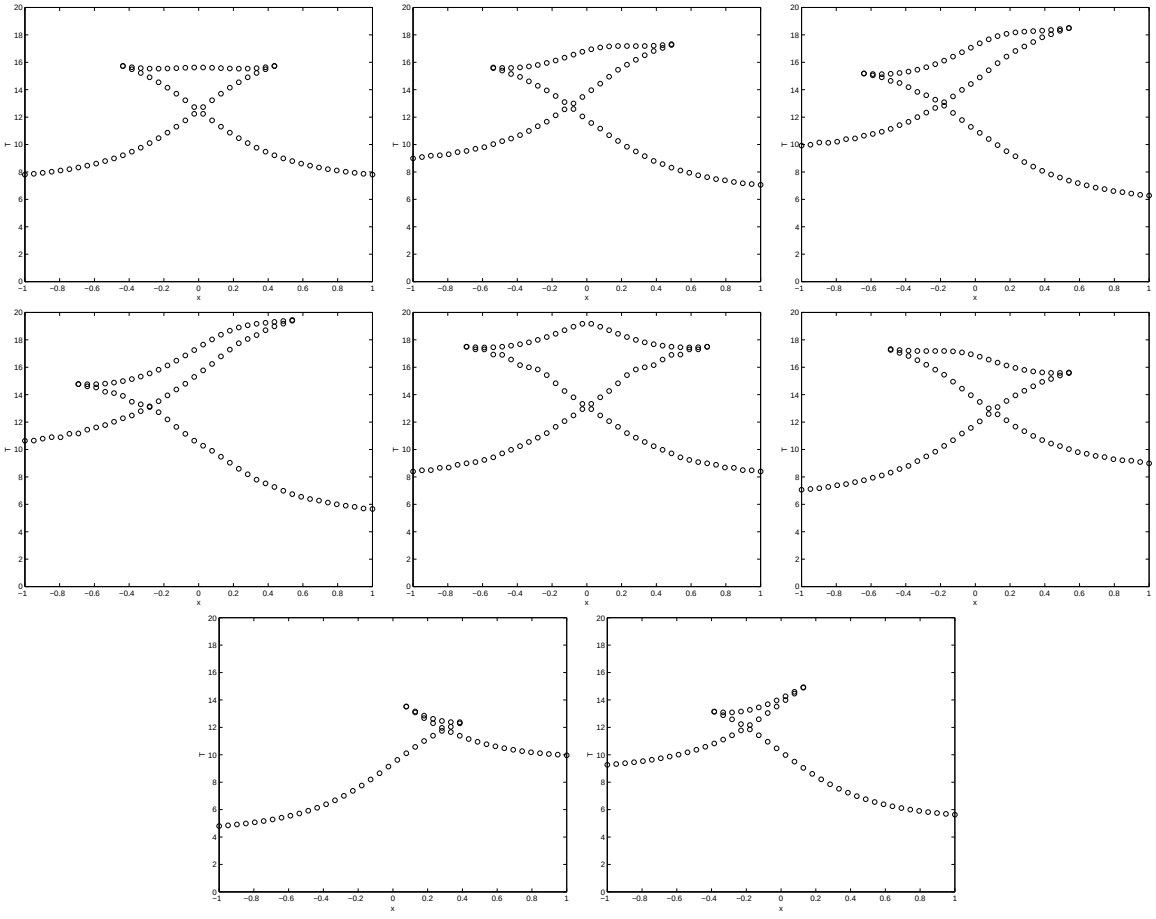

Figure 8: (Example 2) Arrivaltimes at $z = 1.2$ on the cross section $y = -1$ using $1 \leq i, j \leq 41$ and $1 \leq k, l \leq 201$ in the phase space with sources located at $(x, y)_s = (x, y)_{1,2,\cdots,8}$ respectively.

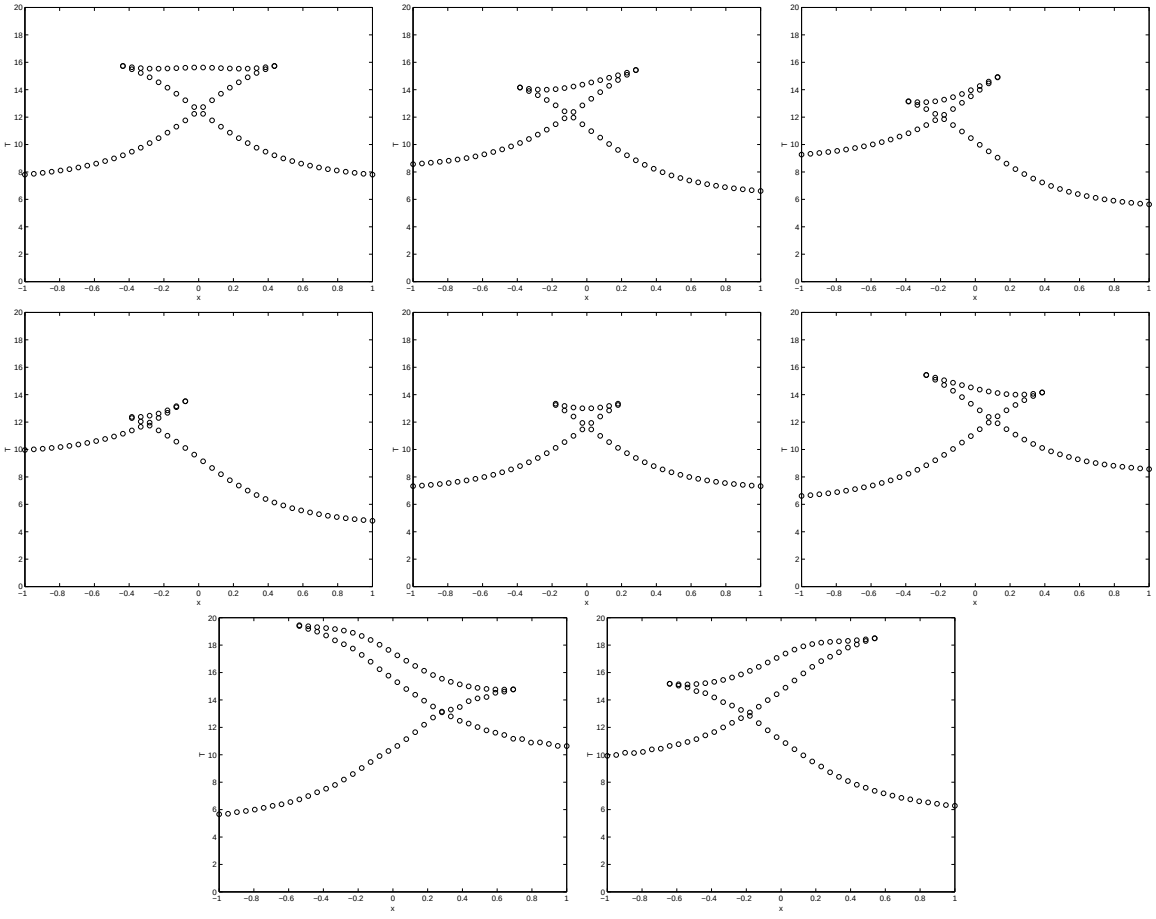

Figure 9: (Example 2) Arrivaltimes at $z = 1.2$ on the cross section $y = 1$ using $1 \leq i, j \leq 41$ and $1 \leq k, l \leq 201$ in the phase space with sources located at $(x, y)_s = (x, y)_{1,2,\cdots,8}$ respectively.

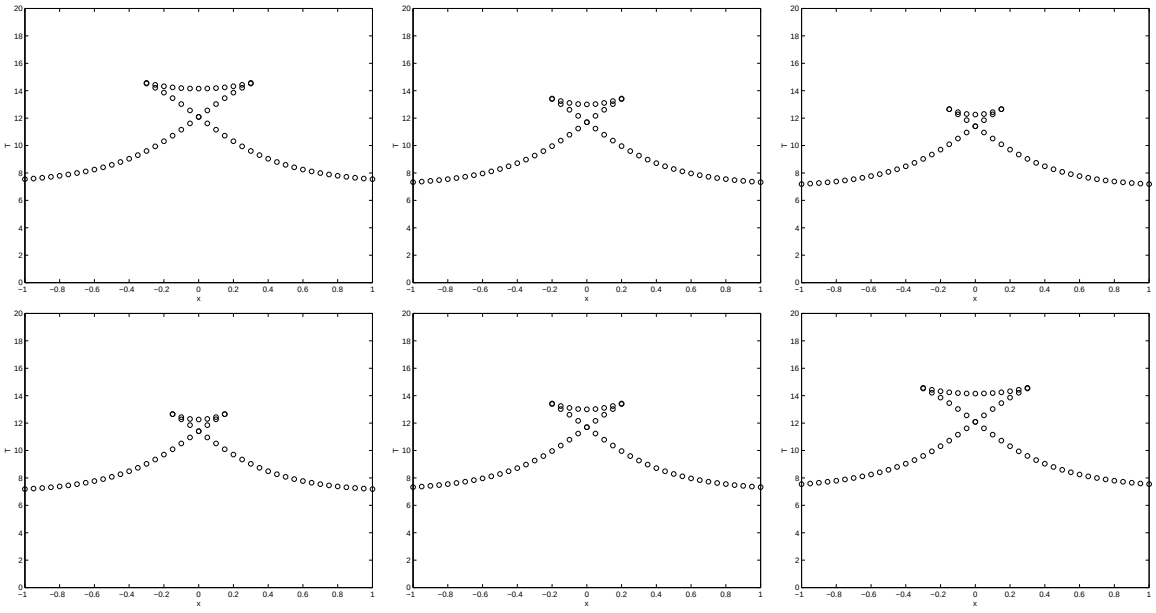

Figure 10: (Example 2) Arrivaltimes at $z = 1.2$ on different cross sections $y = y_j$ for $j = 6, 11, 16, 26, 31, 36$ using $1 \leq i, j \leq 41$ and $1 \leq k, l \leq 201$ in the phase space with sources located at $(x, y)_s = (0, 0)$.

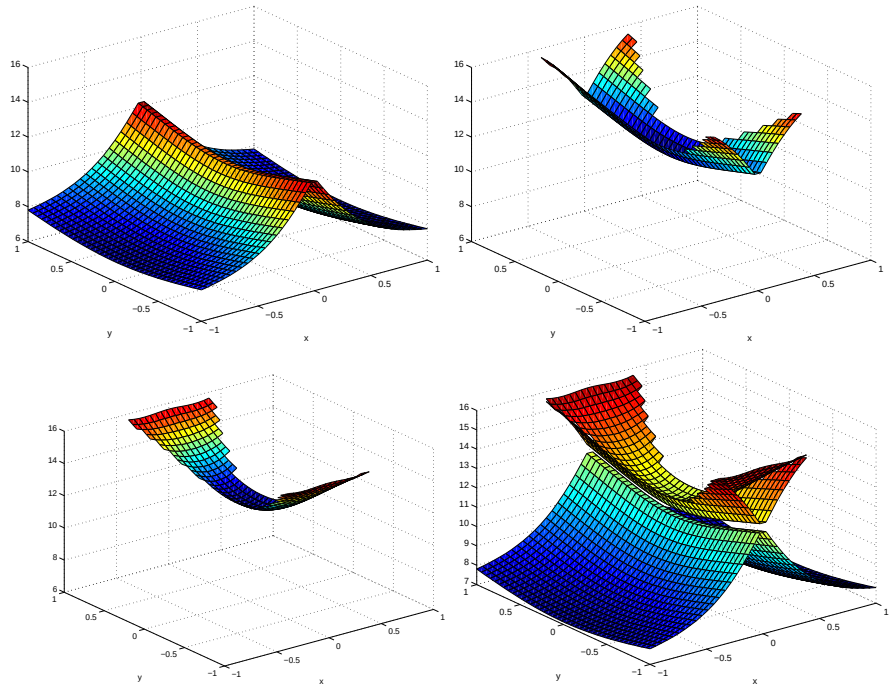

Figure 11: (Example 2) First, second and third arrivaltimes at $z = 1.2$ using $1 \leq i, j \leq 41$ and $1 \leq k, l \leq 201$ in the phase space with sources located at $(x, y)_s = (0, 0)$.

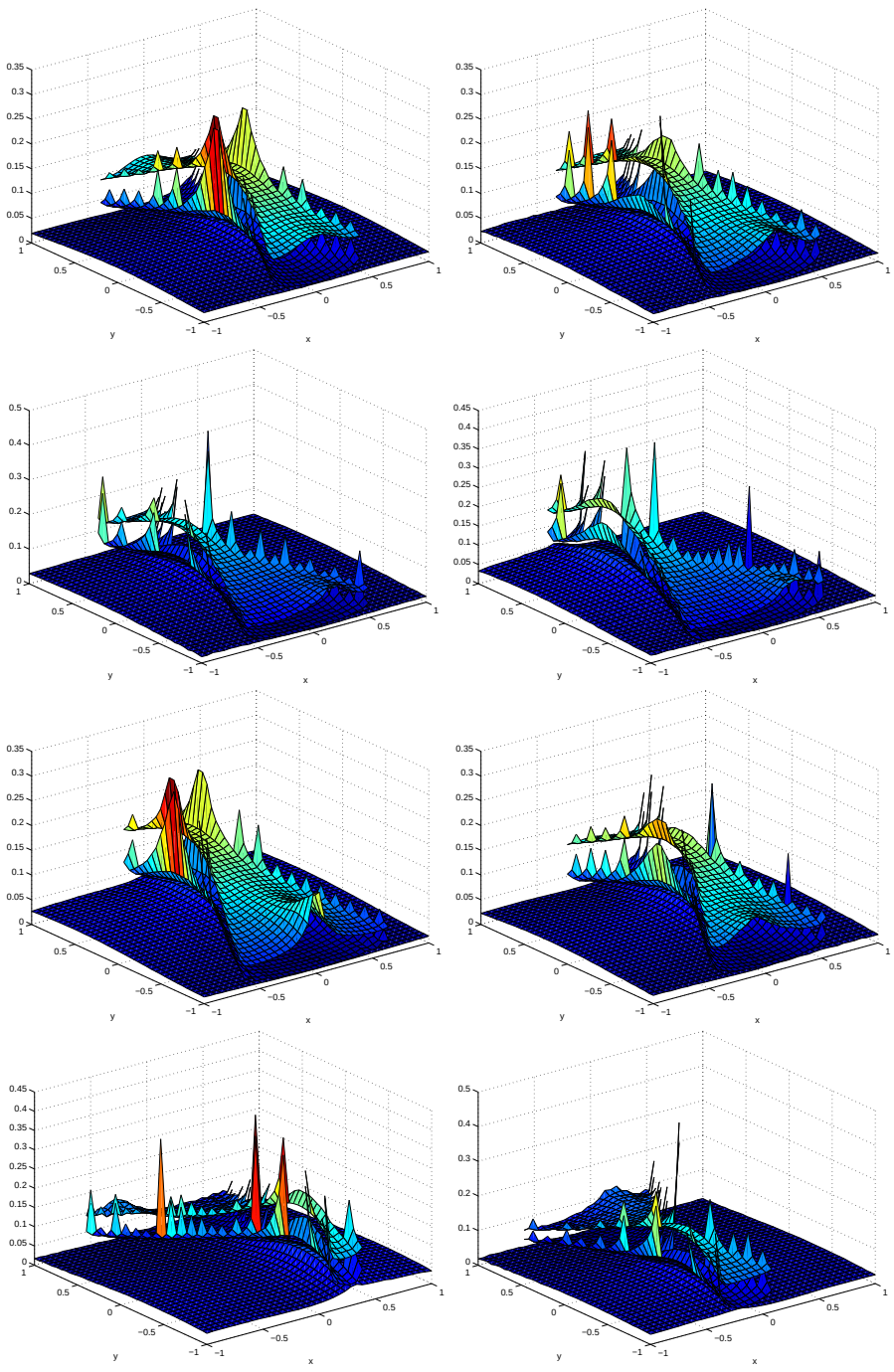

Figure 12: (Example 2) Amplitudes at $z = 1.2$ using $1 \leq i, j \leq 41$ and $1 \leq k, l \leq 201$ in the phase space with sources located at $(x, y)_s = (x, y)_{1,2,\cdots,8}$ respectively .

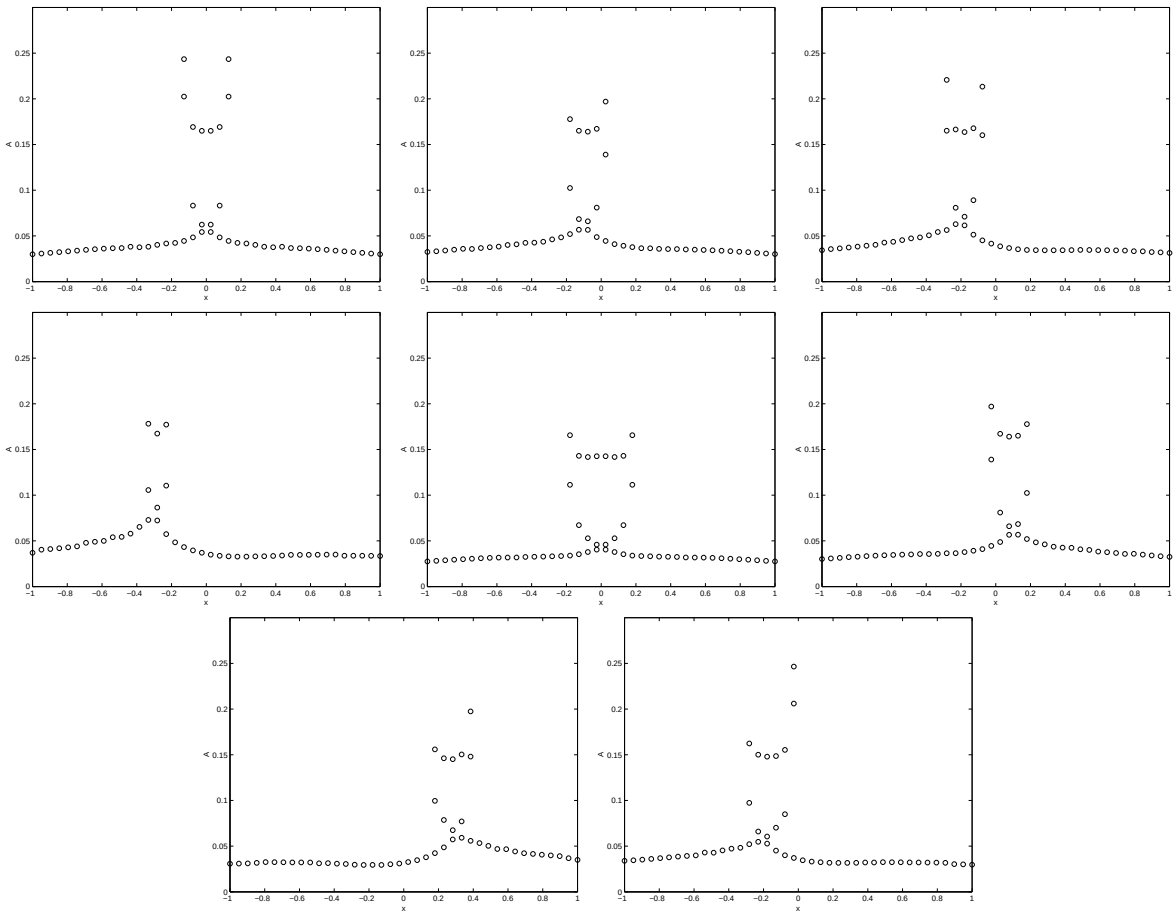

Figure 13: (Example 2) Amplitudes at $z = 1.2$ on the cross section $y = 0$ using $1 \leq i, j \leq 41$ and $1 \leq k, l \leq 201$ in the phase space with sources located at $(x, y)_s = (x, y)_{1,2,\cdots,8}$ respectively.

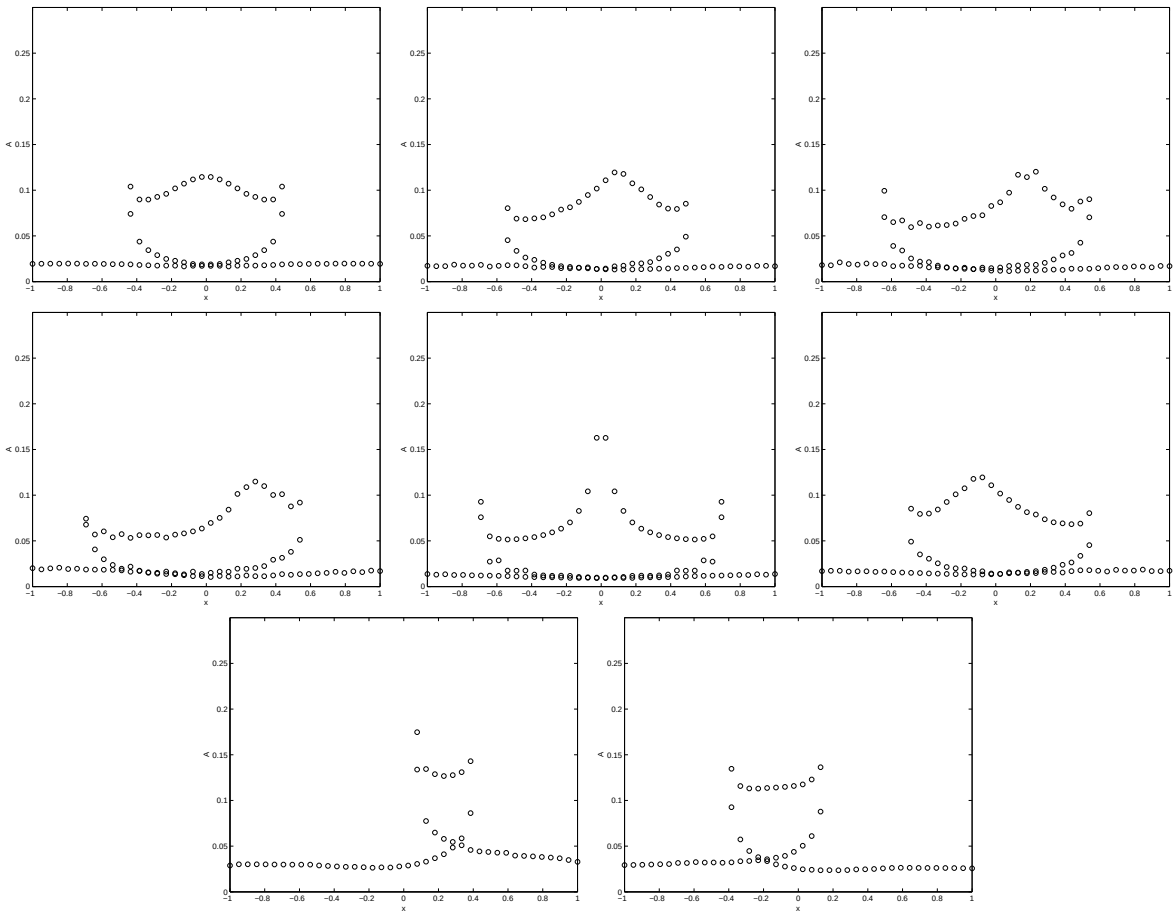

Figure 14: (Example 2) Amplitudes at $z = 1.2$ on the cross section $y = -1$ using $1 \leq i, j \leq 41$ and $1 \leq k, l \leq 201$ in the phase space with sources located at $(x, y)_s = (x, y)_{1,2,\cdots,8}$ respectively.

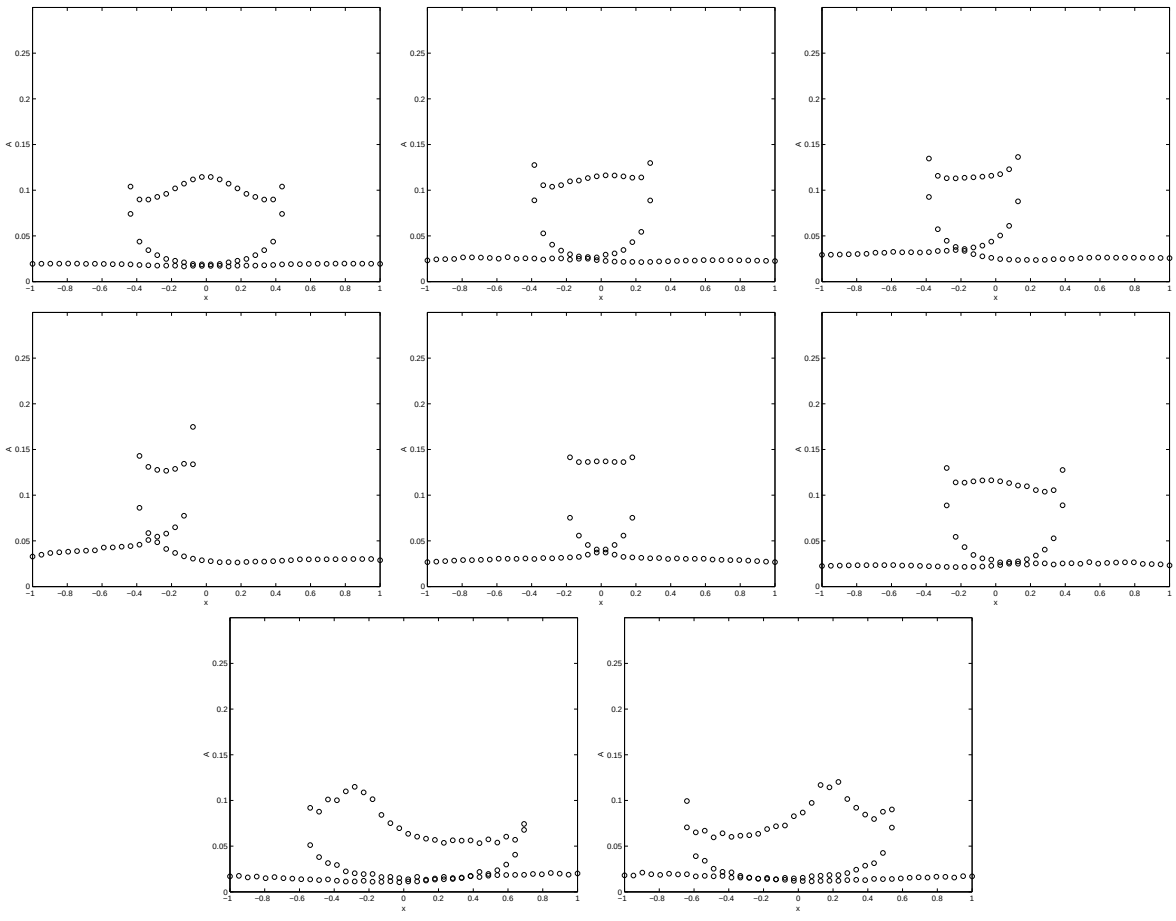

Figure 15: (Example 2) Amplitudes at $z = 1.2$ on the cross section $y = 1$ using $1 \leq i, j \leq 41$ and $1 \leq k, l \leq 201$ in the phase space with sources located at $(x, y)_s = (x, y)_{1,2,\cdots,8}$ respectively.

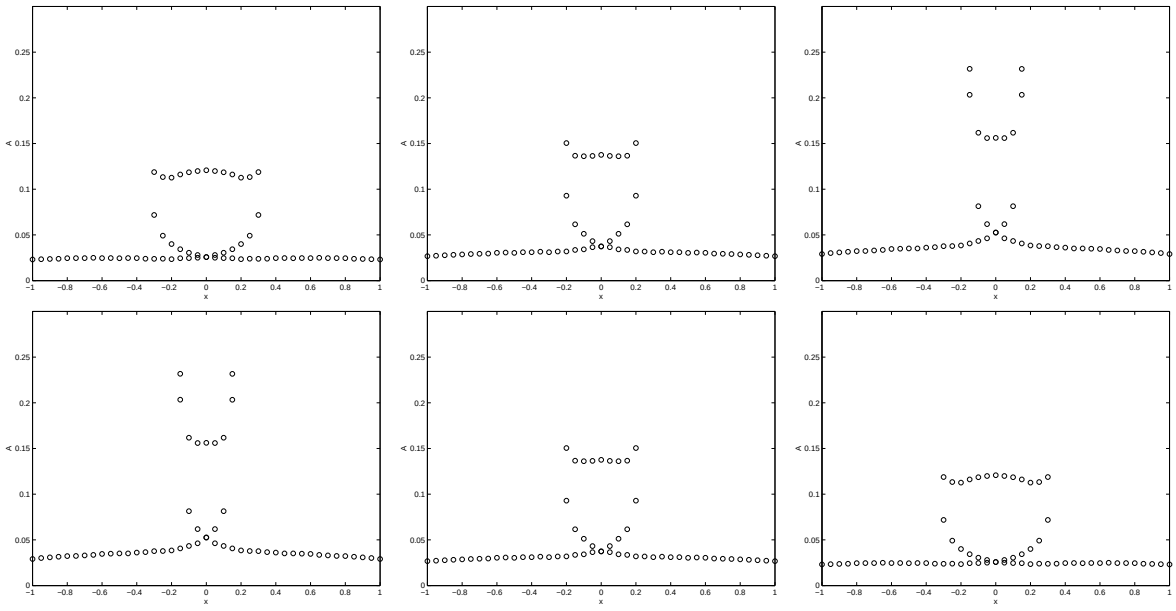

Figure 16: (Example 2) Amplitudes at $z = 1.2$ on different cross sections $y = y_j$ for $j = 6, 11, 16, 26, , 31, 36$ using $1 \leq i, j \leq 41$ and $1 \leq k, l \leq 201$ in the phase space with sources located at $(x, y)_s = (0, 0)$.

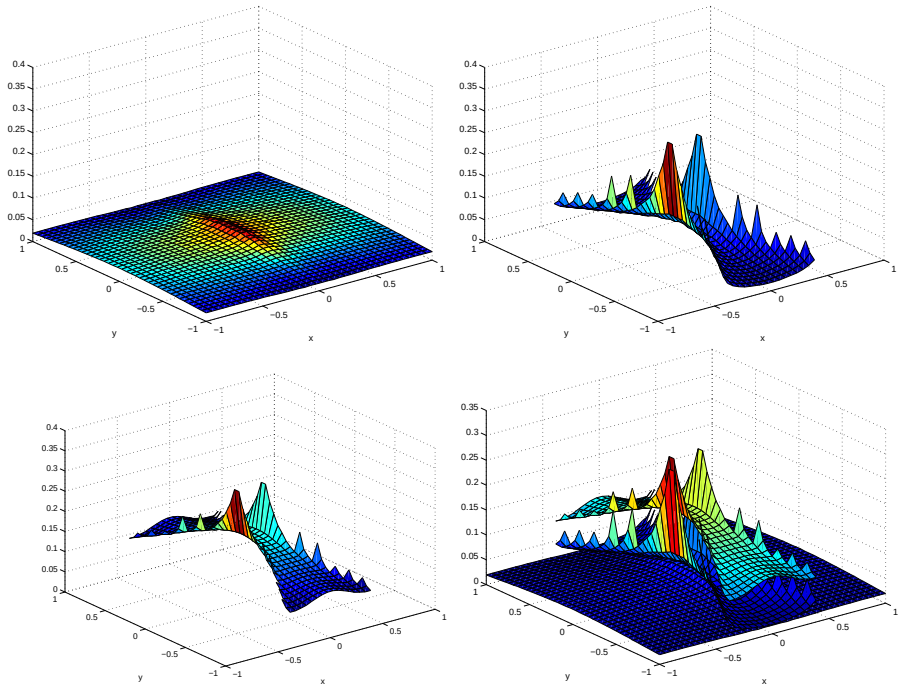

Figure 17: (Example 2) First, second and third arrival amplitudes at $z = 1.2$ using $1 \leq i, j \leq 41$ and $1 \leq k, l \leq 200$ in the phase space with sources located at $(x, y)_s = (0, 0)$.

5.3 Example 3: Vinje's Gaussian Model

This velocity model comes from [39] where

$$c(x,y,z) = 3 - 1.75\exp\left(-\frac{x^2+y^2+(z-0.75)^2}{0.5^2}\right). \tag{51}$$

Figure 18 shows some cross section plots of this velocity model. As seen from the figure, the velocity c is low in the region around (0,0,0.75).

Figure 19 shows the multivalued arrivaltimes at z=1.5 with sources located at $(x,y)_s$ (s =1,2,$\cdots$,8), respectively. As the source varies, the traveltime varies for a ray to reach a specific location. There are three sheets of traveltime surfaces at z=1.5 for those given sources. When two sheets connect to each other in the phase space, they connect along the caustic curves, which are shown in the figures; this can also be seen more clearly in Figure 20.

In this case, although u^4 cannot be simplified to 0, rays from the origin with initial $\psi = 0$ are still staying on the cross section $y = 0$. From the last equation in equation (21), if $\psi(z = 0) = 0$ and $y = 0$, which implies $c_y = 0$, we can obtain $\psi(z) \equiv 0$. Therefore, we can still compare our solution with the one obtained from ray tracing method on the cross section $y = 0$, as shown in Figure 20. Again, the solid line represents the solution using the ray tracing method. The solutions by the method presented here are plotted using circles. Two solutions match with each other very well. In all the sub-figures, we can see that the caustics develop and the traveltime becomes triple valued around caustics.

Figures 21 and 22 show the cross sections of the solution on $y = \mp 1$, respectively.

To look at the solutions more closely, we concentrate on the solution for the source located at $(x,y)_s = (0,0)$. More cross sections of the multivalued arrivaltime are plotted on Figure 23. The locations of the slides are $y = y_j$ for $j = 6, 11, 16, 26, 31, 36$. This corresponds to $y = j/40$. One can see the obvious symmetry since the velocity has rotational invariance for z fixed.

Figure 24 shows the surface plots of the arrivaltime solutions according to the arrival orders. If the eikonal equation is solved directly, one would get the viscosity solution, which corresponds to the first arrivaltime, i.e. the solution on the first sub-figure.

Amplitudes of the arrival rays are also calculated. Figures 25 to 30 show the amplitude solutions corresponding to Figures 19 to 22.

In Figure 30, we can see that the amplitude becomes infinity along the caustic lines in the sub-figures of amplitudes corresponding to the second and third arrivals.

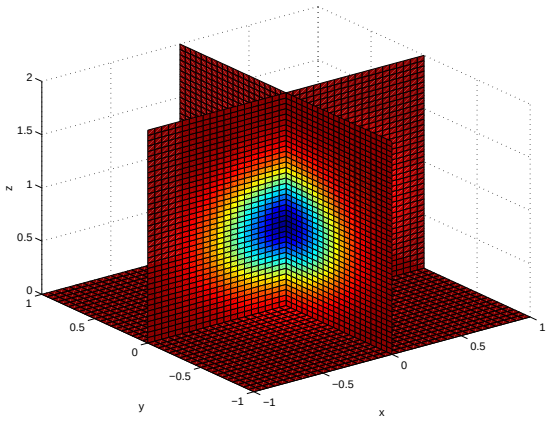

Figure 18: (Example 3) Cross sections of the velocity model.

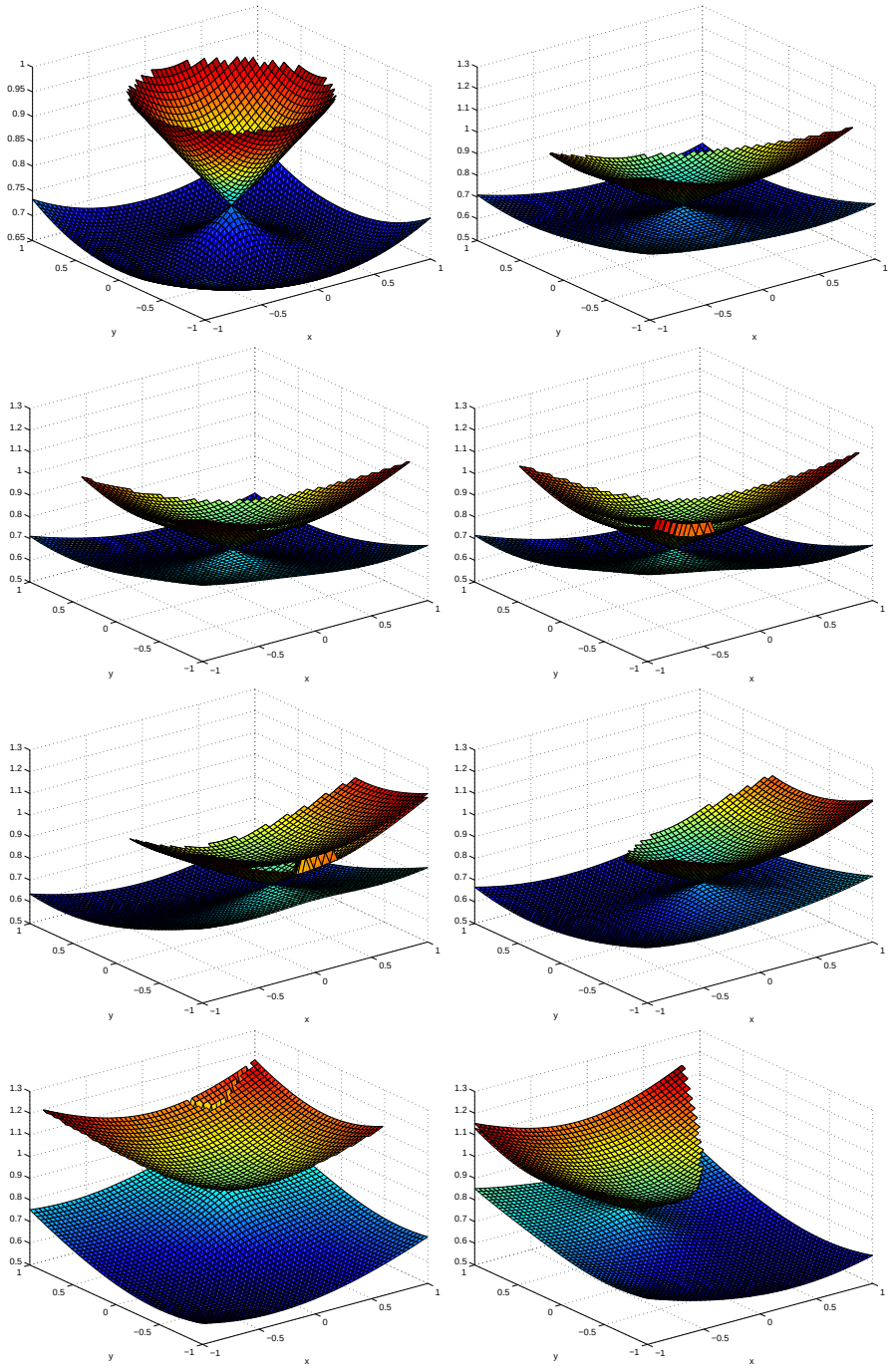

Figure 19: (Example 3) Arrivaltimes at $z = 1.5$ using $1 \leq i, j \leq 51$ and $1 \leq k, l \leq 401$ in the phase space with sources located at $(x, y)_s = (x, y)_{1,2,\cdots,8}$ respectively.

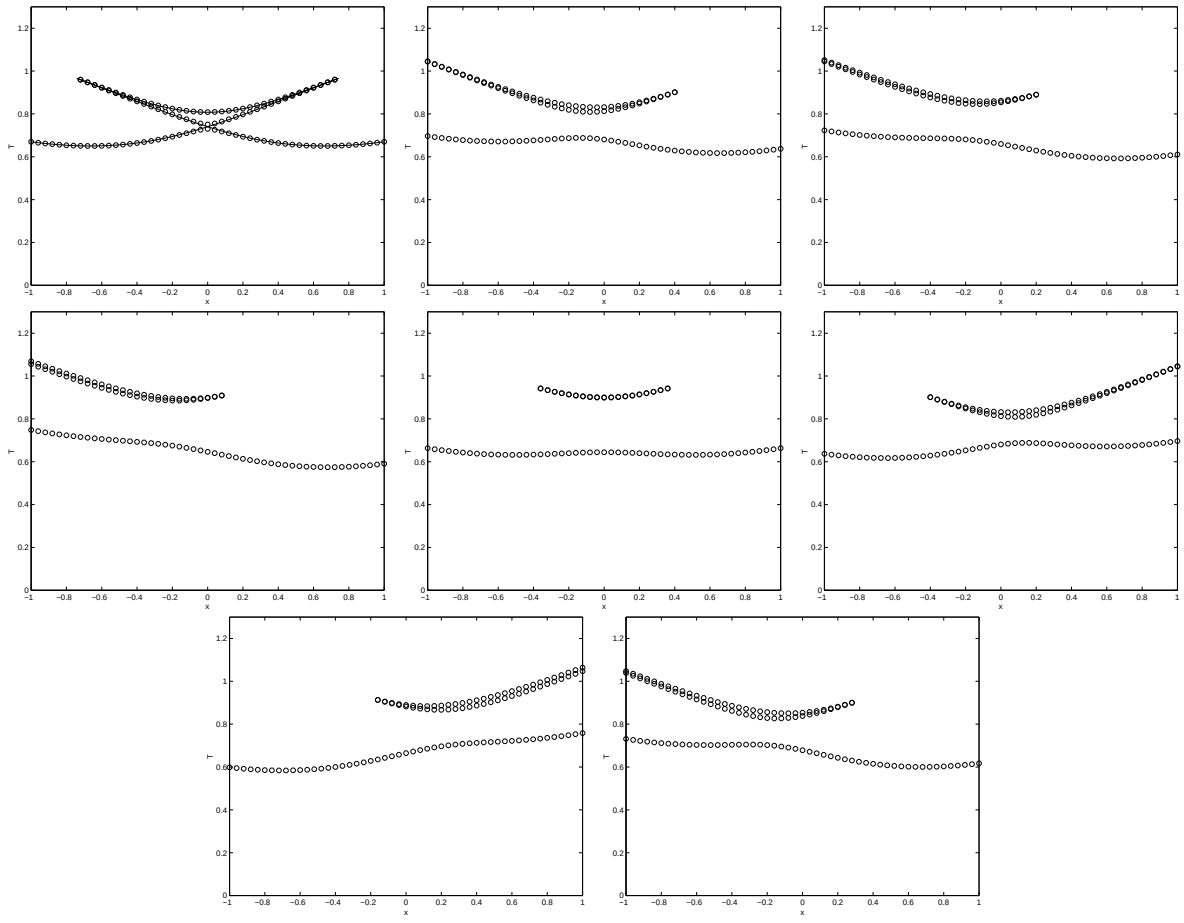

Figure 20: (Example 3) Arrivaltimes at $z = 1.5$ on the cross section $y = 0$ using $1 \leq i, j \leq 51$ and $1 \leq k, l \leq 401$ in the phase space with sources located at $(x, y)_s = (x, y)_{1,2,\cdots,8}$ respectively.

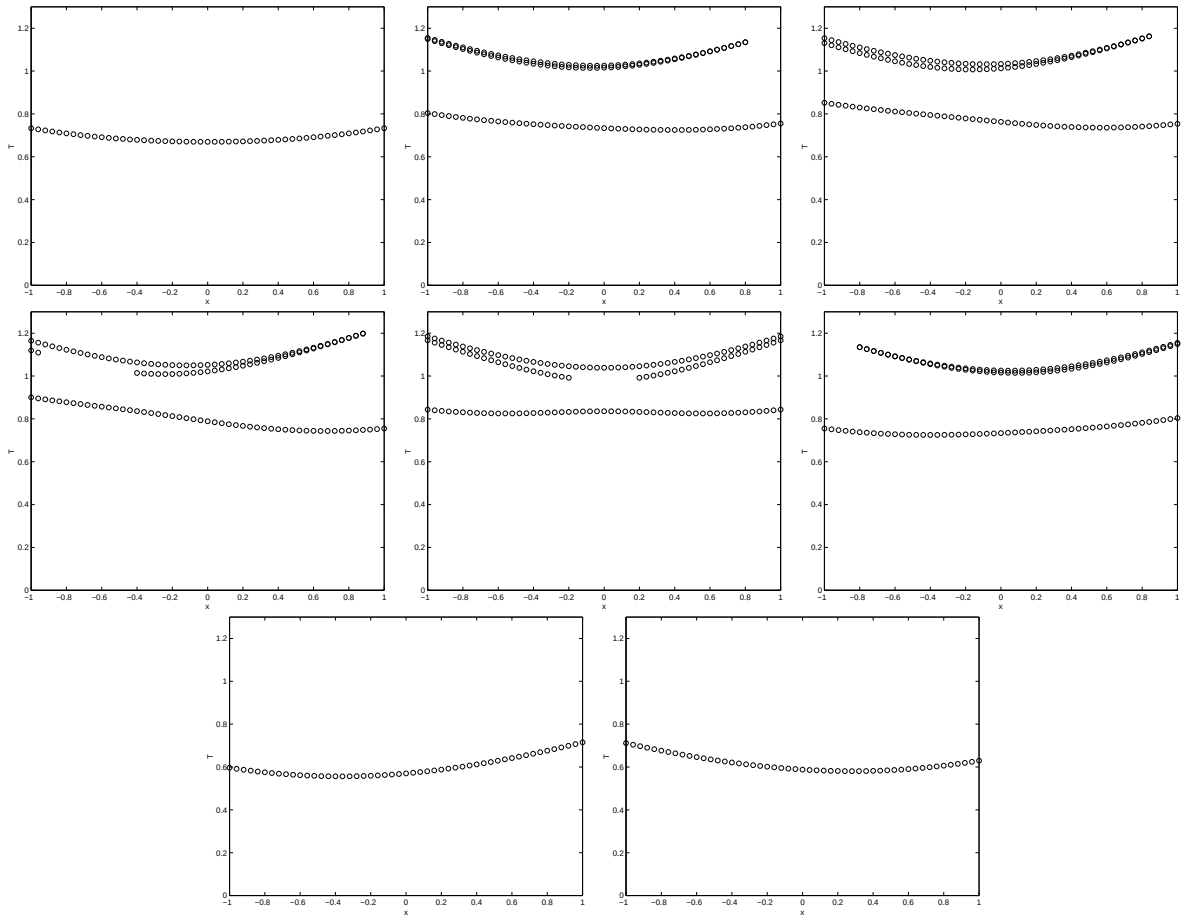

Figure 21: (Example 3) Arrivaltimes at $z = 1.5$ on the cross section $y = -1$ using $1 \leq i, j \leq 51$ and $1 \leq k, l \leq 401$ in the phase space with sources located at $(x, y)_s = (x, y)_{1,2,\cdots,8}$ respectively.

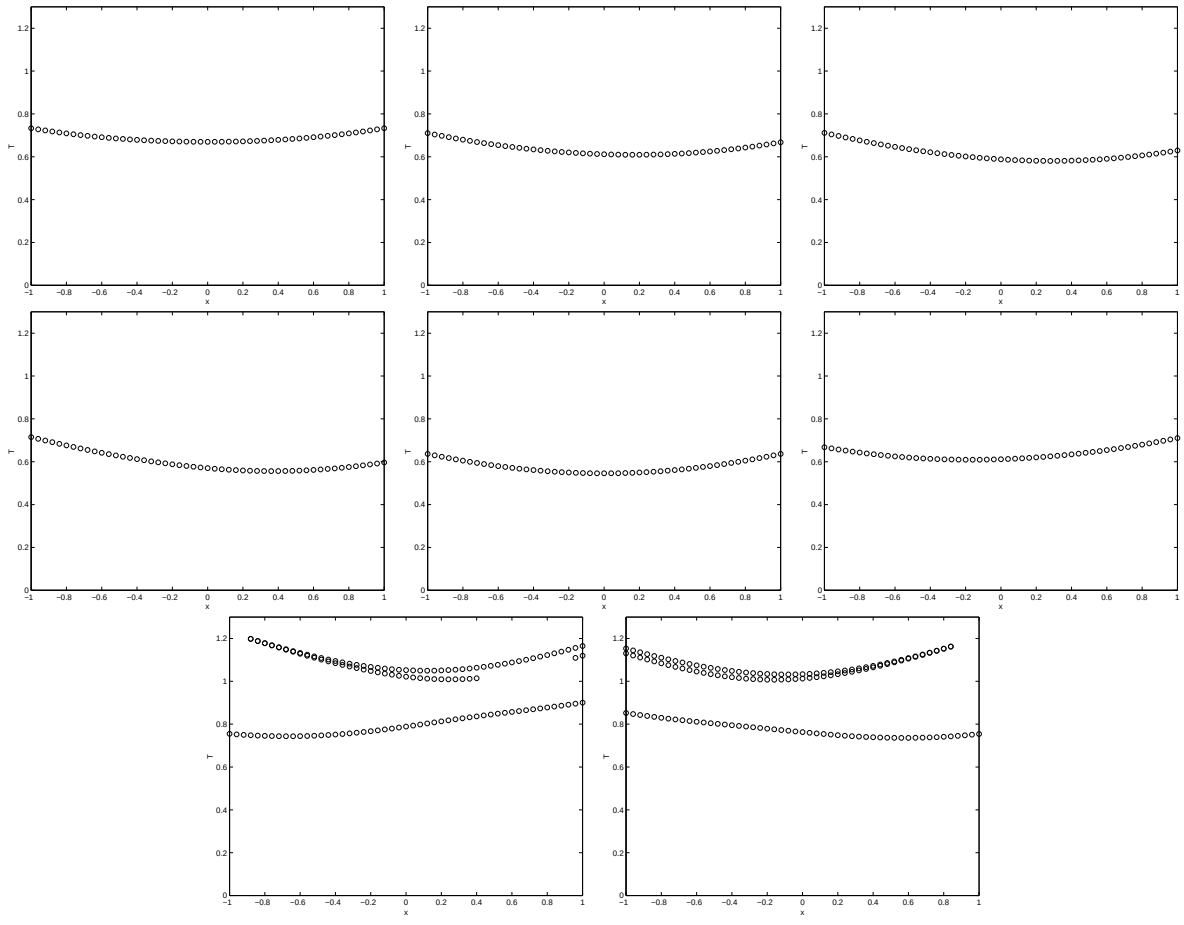

Figure 22: (Example 3) Arrivaltimes at $z = 1.5$ on the cross section $y = 1$ using $1 \leq i, j \leq 51$ and $1 \leq k, l \leq 401$ in the phase space with sources located at $(x, y)_s = (x, y)_{1,2,\cdots,8}$ respectively.

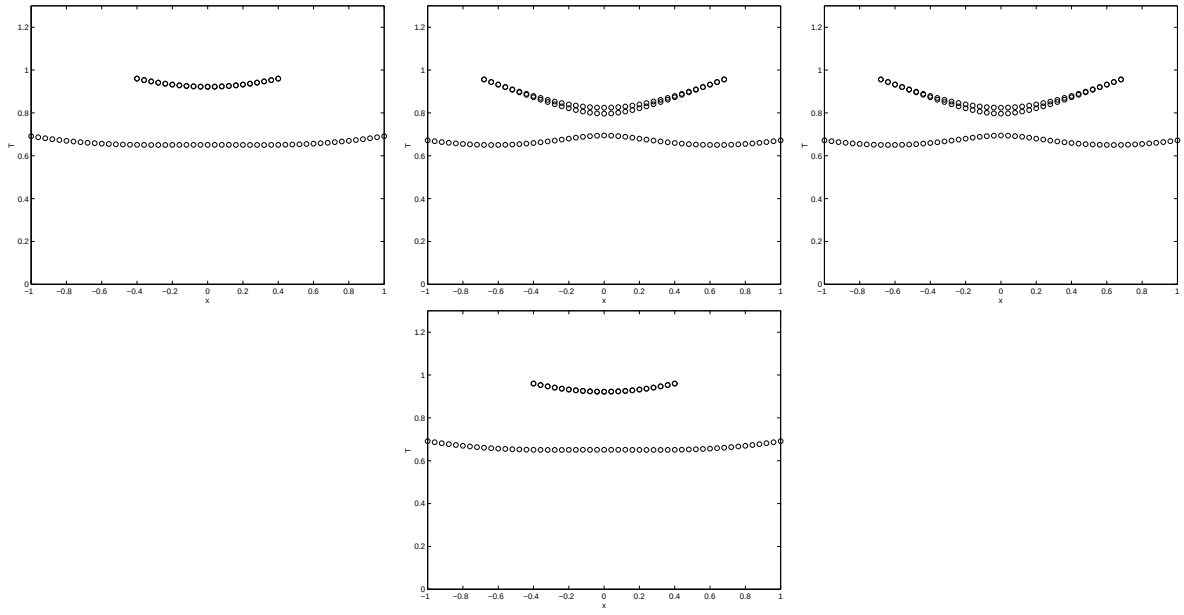

Figure 23: (Example 3) Arrivaltimes at $z = 1.5$ on different cross sections $y = y_j$ for $j = 11, 21, 31, 41$ using $1 \leq i, j \leq 51$ and $1 \leq k, l \leq 401$ in the phase space with sources located at $(x, y)_s = (0, 0)$.

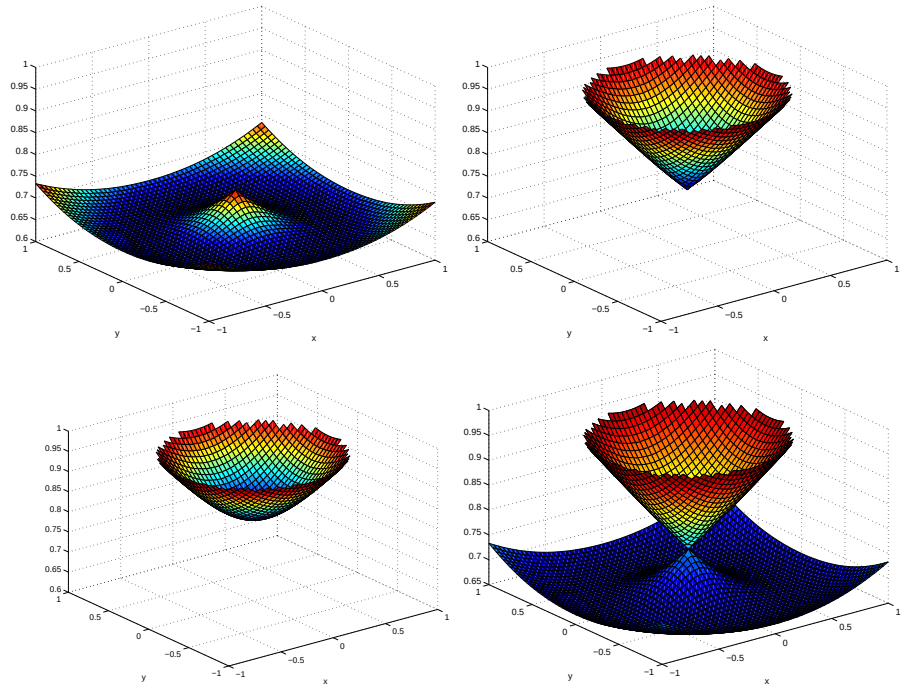

Figure 24: (Example 3) First, second and third arrivaltimes at $z = 1.5$ using $1 \leq i, j \leq 51$ and $1 \leq k, l \leq 401$ in the phase space with sources located at $(x, y)_s = (0, 0)$.

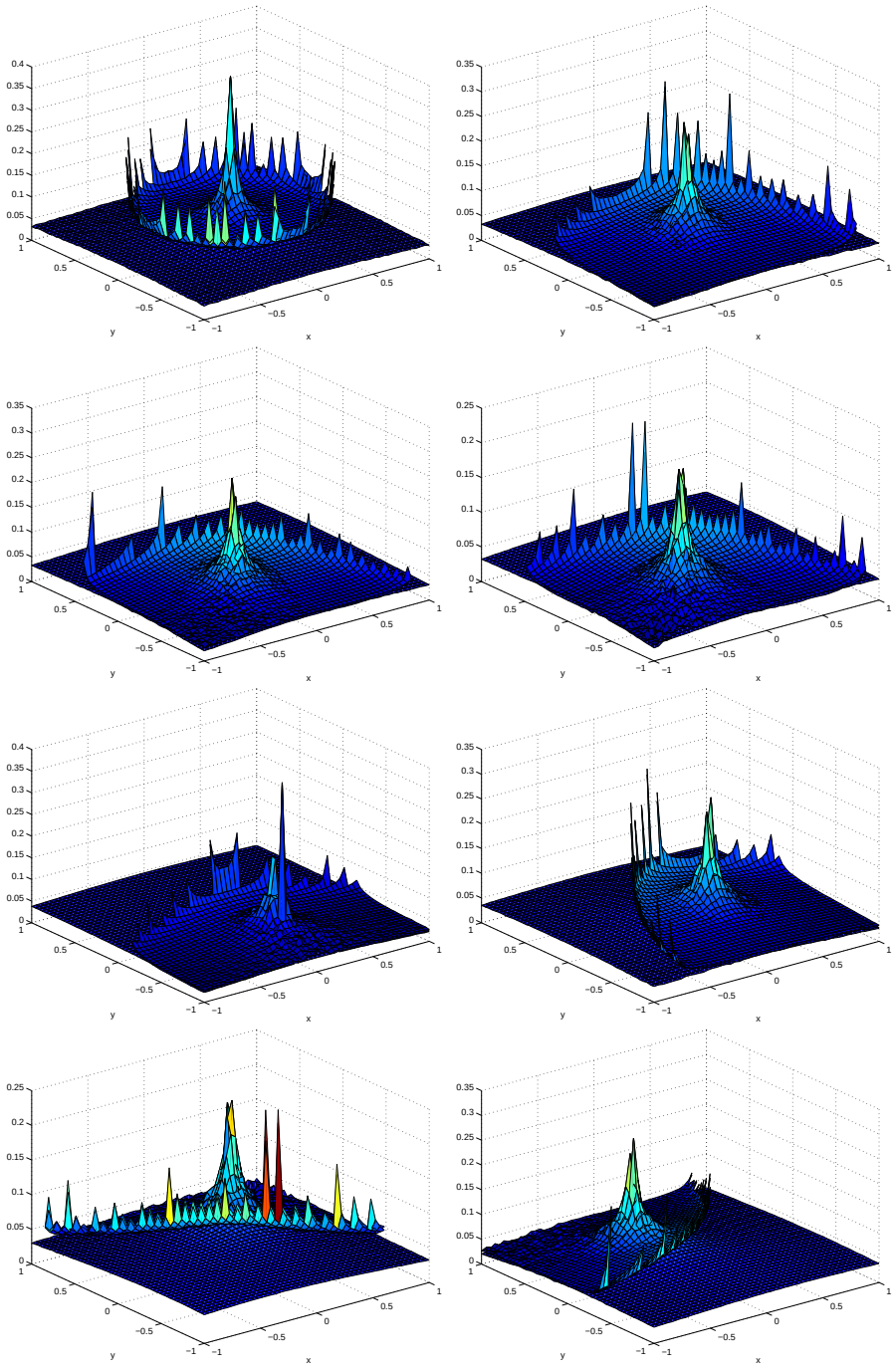

Figure 25: (Example 3) Amplitudes at $z = 1.5$ using $1 \leq i, j \leq 51$ and $1 \leq k, l \leq 401$ in the phase space with sources located at $(x, y)_s = (x, y)_{1,2,\cdots,8}$, respectively.

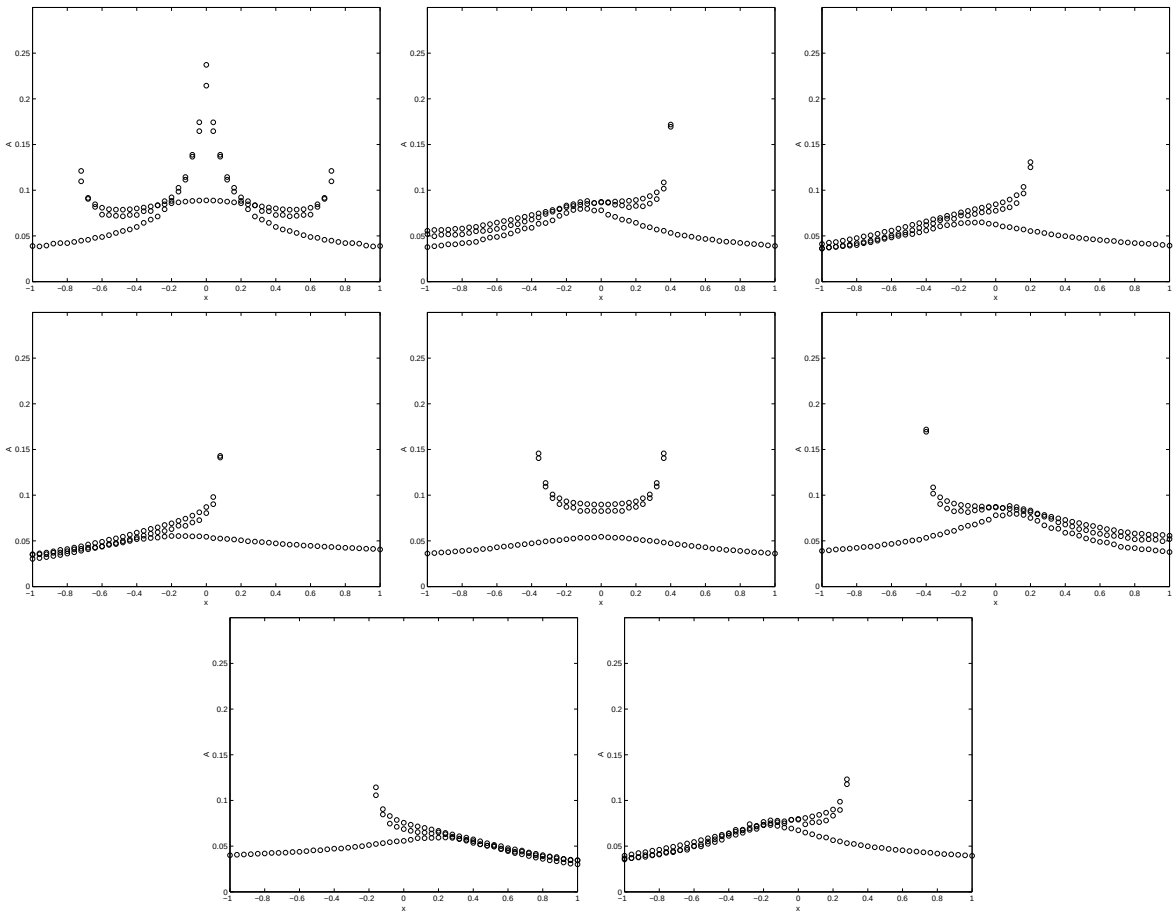

Figure 26: (Example 3) Amplitudes at $z = 1.5$ on the cross section $y = 0$ using $1 \leq i, j \leq 51$ and $1 \leq k, l \leq 401$ in the phase space with sources located at $(x, y)_s = (x, y)_{1,2,\cdots,8}$ respectively.

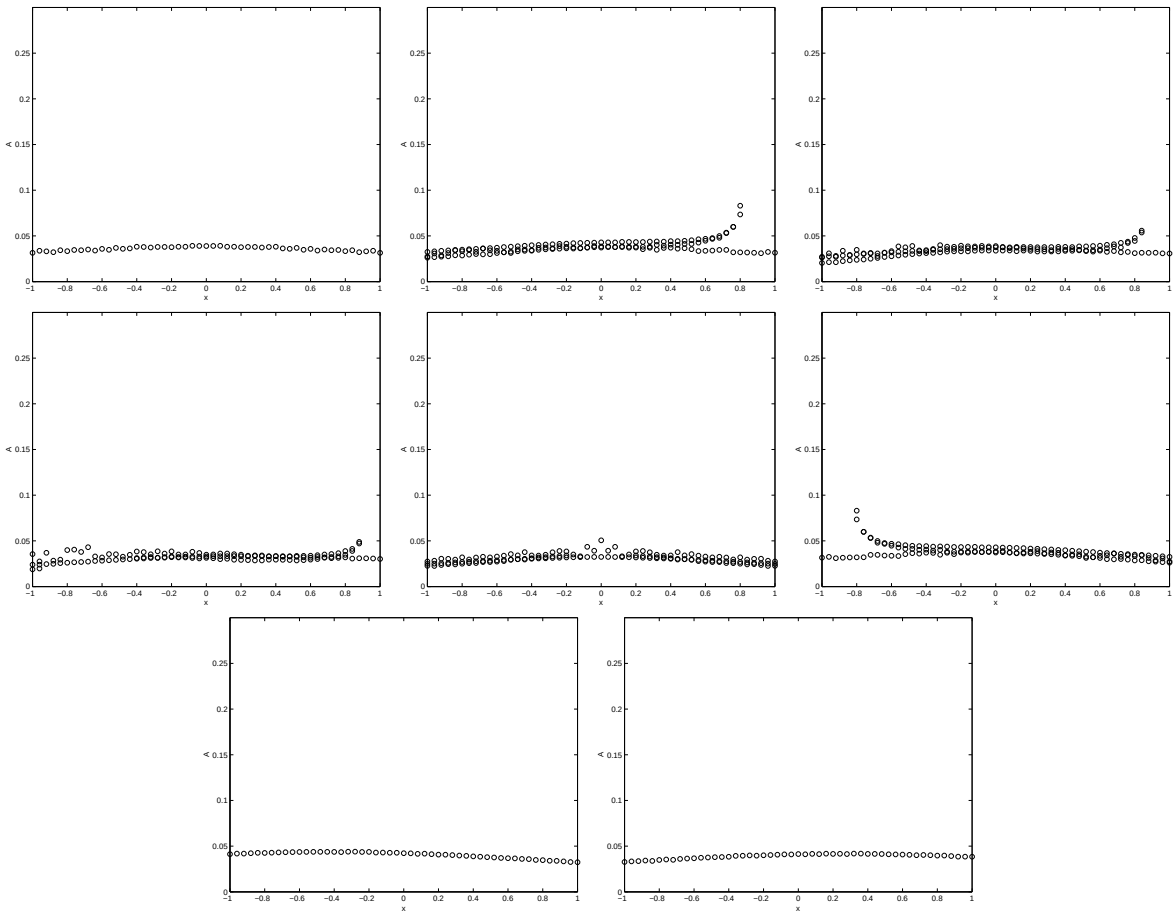

Figure 27: (Example 3) Amplitudes at $z = 1.5$ on the cross section $y = -1$ using $1 \leq i, j \leq 51$ and $1 \leq k, l \leq 401$ in the phase space with sources located at $(x, y)_s = (x, y)_{1,2,\cdots,8}$ respectively.

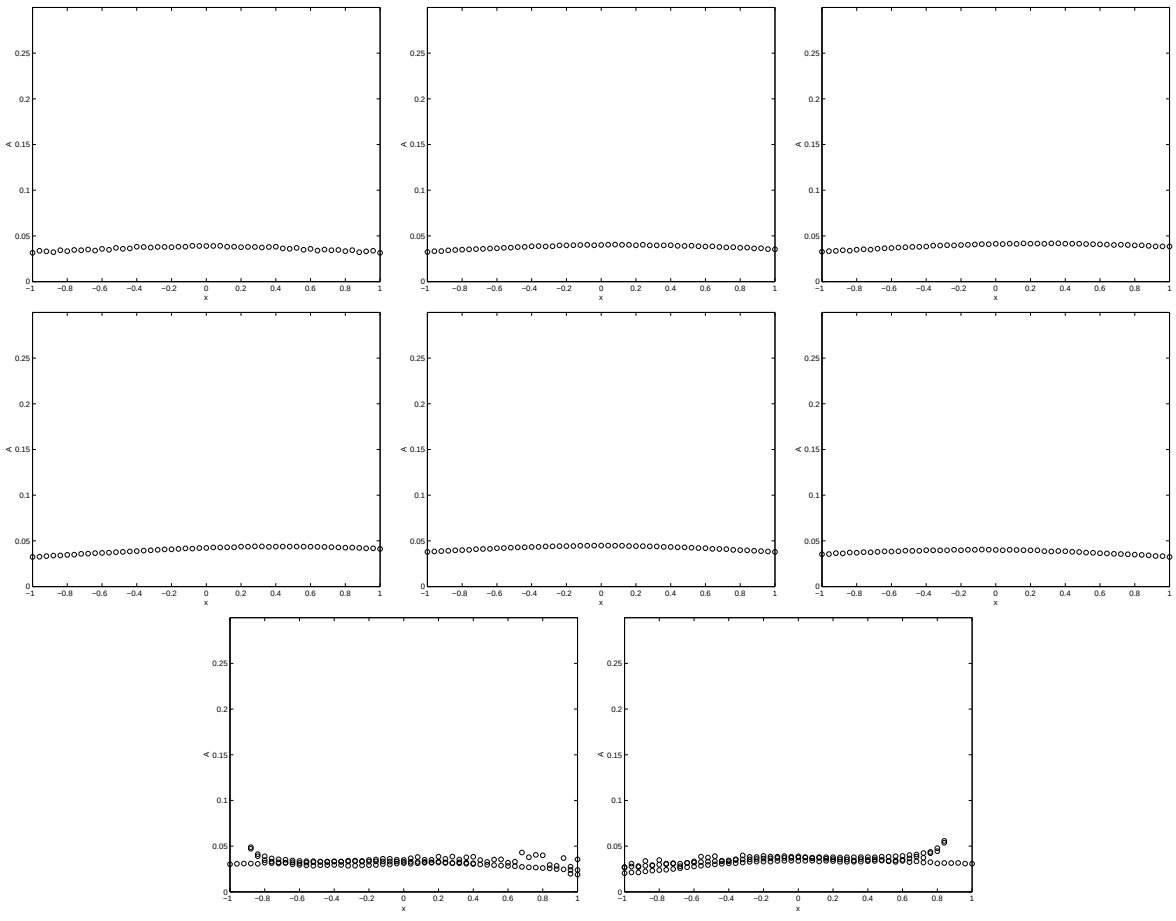

Figure 28: (Example 3) Amplitudes at $z = 1.5$ on the cross section $y = 1$ using $1 \leq i, j \leq 51$ and $1 \leq k, l \leq 401$ in the phase space with sources located at $(x, y)_s = (x, y)_{1,2,\cdots,8}$ respectively.

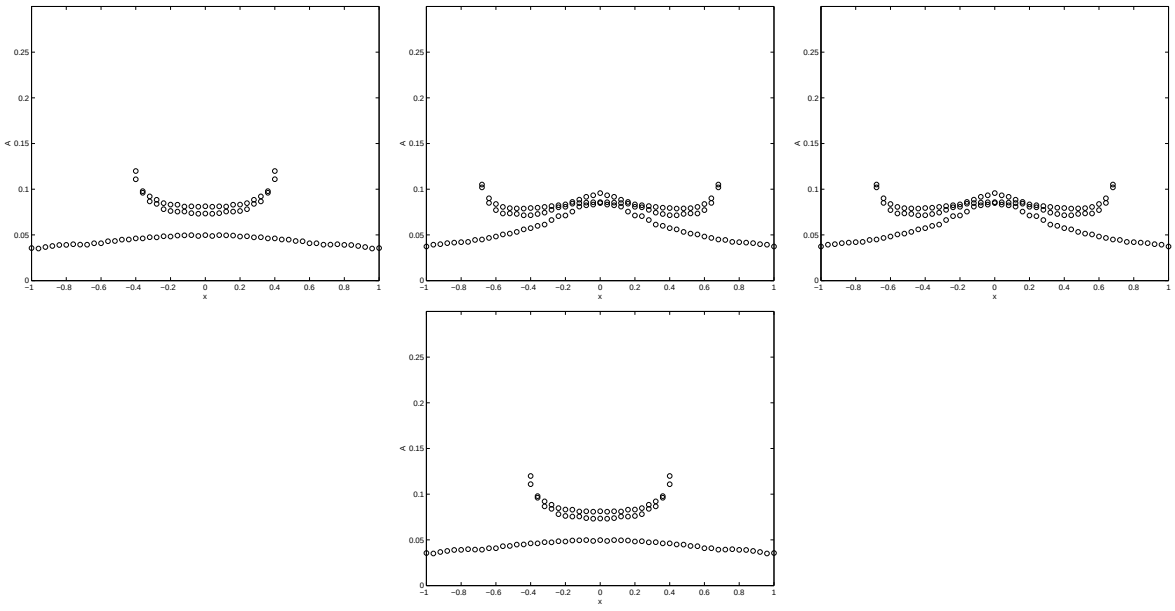

Figure 29: (Example 3) Amplitudes at $z = 1.5$ on different cross sections $y = y_j$ for $j = 11, 21, 31, 41$ using $1 \leq i, j \leq 51$ and $1 \leq k, l \leq 401$ in the phase space with sources located at $(x, y)_s = (0, 0)$.

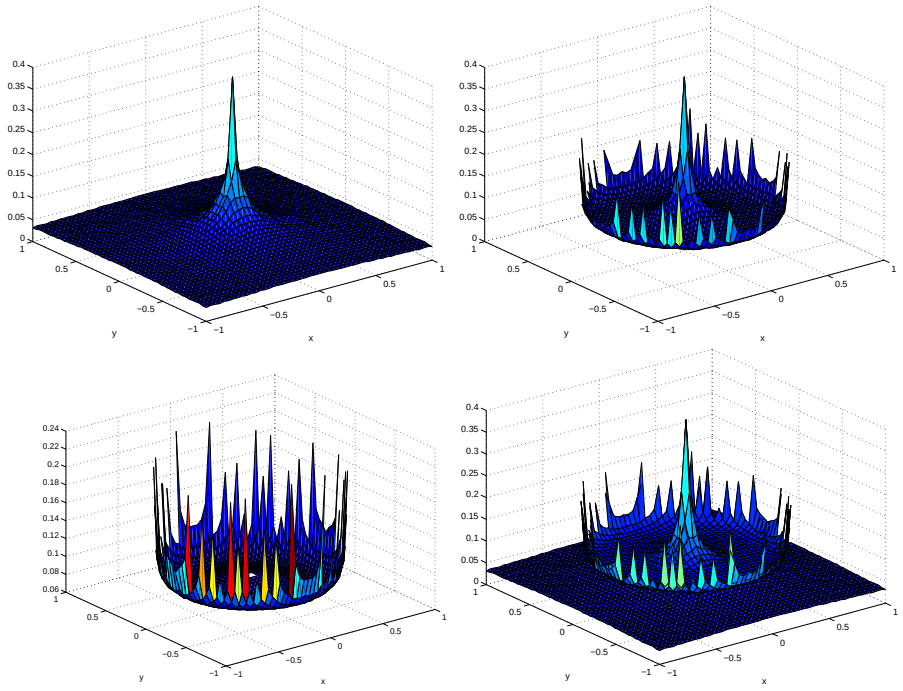

Figure 30: (Example 3) First, second and third arrival amplitudes at $z = 1.5$ using $1 \leq i, j \leq 51$ and $1 \leq k, l \leq 401$ in the phase space with sources located at $(x, y)_s = (0, 0)$.

6 Conclusion

We developed a level set method to compute the three dimensional multivalued geometrical optics term in the paraxial formulation. This method has two new features: it does not require reinitialization and it can handle multiple sources simultaneously. By using a semi-Lagrangian method in the paraxial formulation, the method has $O(N^2)$ rather than $O(N^4)$ memory storage requirement in the five dimensional phase space, where N is the number of mesh points along one direction. Although the computational complexity is still $O(MN^4)$, this disadvantage is largely overcome by the fact that $O(N^2)$ multiple sources can be treated simultaneously. Numerical examples demonstrated the efficiency and accuracy of the new method.

7 Acknowledgment

This research is supported by ONR Grant #N00014-02-1-0720.

8 Appendix

In this appendix, derivation of equations (28), (29) and (39) are given.

8.1 Equation (28)

The amplitude is transported by

$$\nabla \cdot (A^2 \nabla \tilde{T}) = 0\,. \tag{52}$$

Therefore, integrating along a ray tube and using the divergence theorem, we get

$$A\sqrt{\frac{d\sigma}{c}} = A_0\sqrt{\frac{d\sigma_0}{c_0}}\,, \tag{53}$$

where σ is the transverse section area of the ray tube. Furthermore,

$$A = A_0\sqrt{\frac{c\,d\sigma_0}{c_0\,d\sigma}}\,. \tag{54}$$

For a point source in an isotropic medium, we suppose that the source emits a directional spherical wave having a normalized radiation function $\frac{1}{4\pi}$ with unit strength,

$$A_0(\tilde{\Theta}, \tilde{\Psi}) = \frac{1}{4\pi R_0}, \tag{55}$$

where $\tilde{\Theta}$ and $\tilde{\Psi}$ are the spherical take-off angles, and R_0 is the radius of the small sphere centered at the source.

Then at small distances from the source,

$$d\sigma_0 = R_0^2 \sin\tilde{\Theta} d\tilde{\Theta} d\tilde{\Psi}\,. \tag{56}$$

Now consider an elementary ray tube,

$$dV = dxdydz = d\sigma ds \tag{57}$$

where ds is the differential element of arc length. Furthermore,

$$dV = \frac{\partial(x,y,z)}{\partial(\tilde{T},\tilde{\Theta},\tilde{\Psi})} d\tilde{T} d\tilde{\Theta} d\tilde{\Psi}, \tag{58}$$

$$ds = c\, d\tilde{T}, \tag{59}$$

so

$$d\sigma = \frac{1}{c} \frac{\partial(x,y,z)}{\partial(\tilde{T},\tilde{\Theta},\tilde{\Psi})} d\tilde{\Theta} d\tilde{\Psi}. \tag{60}$$

Thus,

$$\tilde{A}(z;x,y) = \frac{c}{4\pi\sqrt{c_0}} \sqrt{\sin\tilde{\Theta} \left| \frac{\partial(\tilde{T},\tilde{\Theta},\tilde{\Psi})}{\partial(x,y,z)} \right|}. \tag{61}$$

8.2 Equation (29)

Now, consider the term

$$\left| \frac{\partial(\tilde{T},\tilde{\Theta},\tilde{\Psi})}{\partial(x,y,z)} \right|. \tag{62}$$

First, we extend all $\tilde{T}$, $\tilde{\Theta}$ and $\tilde{\Psi}$ to the phase space. Then, the x,y,z-derivatives we be replaced by

$$\begin{aligned}
\frac{\partial\tilde{(.)}}{\partial x} &= (.)_x + (.)_\theta \frac{\partial\theta}{\partial x} + (.)_\psi \frac{\partial\psi}{\partial x} \\
\frac{\partial\tilde{(.)}}{\partial y} &= (.)_y + (.)_\theta \frac{\partial\theta}{\partial y} + (.)_\psi \frac{\partial\psi}{\partial y} \\
\frac{\partial\tilde{(.)}}{\partial z} &= (.)_x + (.)_\theta \frac{\partial\theta}{\partial z} + (.)_\psi \frac{\partial\psi}{\partial z}\,.
\end{aligned} \tag{63}$$

Then, one can expand the determinant and, after some algebra, get

$$\left| \frac{\partial(\tilde{T},\tilde{\Theta},\tilde{\Psi})}{\partial(x,y,z)} \right| = \frac{1}{c\sin\theta\cos\psi} \frac{\Delta_1}{\Delta_2}\,. \tag{64}$$

These computations were carries out by *Mathematica*.

```
alpha1=a1*b2-a2*b1; alpha2=a1*b3-a3*b1; alpha3=a1*b4-a4*b1;
alpha4=a3*b2-a2*b3; alpha5=a4*b2-a2*b4; alpha6=a4*b3-a3*b4;

beta1=c3*d4-c4*d3; beta2=c2*d4-c4*d2; beta3=c3*d2-c2*d3;
beta4=c1*d4-c4*d1; beta5=c3*d1-c1*d3;
```

```
dthetadx=-beta4/beta1; dthetady=-beta2/beta1;
dthetadz=(d4*(-u1*c1-u2*c2-u3*c3-u4*c4)-
c4*(-u1*d1-u2*d2-u3*d3-u4*d4))/-beta1;
dpsidx=-beta5/beta1; dpsidy=-beta3/beta1;
dpsidz=(-d3*(-u1*c1-u2*c2-u3*c3-u4*c4)+
c3*(-u1*d1-u2*d2-u3*d3-u4*d4))/-beta1;

tz=-u1*tx-u2*ty-u3*ttheta-u4*tpsi+f;
thetaz=-u1*a1-u2*a2-u3*a3-u4*a4;
psiz=-u1*b1-u2*b2-u3*b3-u4*b4;

delta={ { tx+ttheta*dthetadx+tpsi*dpsidx,
ty+ttheta*dthetady+tpsi*dpsidy, tz+ttheta*dthetadz+tpsi*dpsidz },
{ a1+a3*dthetadx+a4*dpsidx, a2+a3*dthetady+a4*dpsidy,
thetaz+a3*dthetadz+a4*dpsidz }, { b1+b3*dthetadx+b4*dpsidx,
b2+b3*dthetady+b4*dpsidy, psiz+b3*dthetadz+b4*dpsidz } };

delta1={{a1,a2,a3,a4},{b1,b2,b3,b4},{c1,c2,c3,c4},{d1,d2,d3,d4}};

jadelta=FullSimplify[Det[delta]];
jadelta1=Det[delta1];
jadelta2=c3*d4-c4*d3;

Simplify[jadelta*jadelta2/jadelta1]
```

where

$$a_{1,2,3,4} = \Theta_{x,y,\theta,\psi}\,,\; b_{1,2,3,4} = \Psi_{x,y,\theta,\psi}\,,\; c_{1,2,3,4} = \phi^1_{x,y,\theta,\psi}\,,\; d_{1,2,3,4} = \phi^2_{x,y,\theta,\psi}\,. \tag{65}$$

With everything, we have

$$A = \frac{\sqrt{\sin\Theta}}{4\pi}\frac{c}{\sqrt{c_0}}\sqrt{\frac{1}{c\sin\theta\cos\psi}\frac{\Delta_1}{\Delta_2}}\,, \tag{66}$$

where Θ and θ are the takeoff angle and the arrival angle of the ray from the source respectively.

And, therefore, equation (29).

8.3 Equation (39)

Δ_1 is given by

$$\Delta_1 = \begin{vmatrix} \phi^1_x & \phi^2_x & \Theta_x & \Psi_x \\ \phi^1_y & \phi^2_y & \Theta_y & \Psi_y \\ \phi^1_\theta & \phi^2_\theta & \Theta_\theta & \Psi_\theta \\ \phi^1_\psi & \phi^2_\psi & \Theta_\psi & \Psi_\psi \end{vmatrix}\,. \tag{67}$$

Now, we differentiate this with respect to z, we have

$$
\begin{aligned}
(\Delta_1)_z \;=\; & \begin{vmatrix} (\phi^1_z)_x & (\phi^2_z)_x & (\Theta_z)_x & (\Psi_z)_x \\ \phi^1_y & \phi^2_y & \Theta_y & \Psi_y \\ \phi^1_\theta & \phi^2_\theta & \Theta_\theta & \Psi_\theta \\ \phi^1_\psi & \phi^2_\psi & \Theta_\psi & \Psi_\psi \end{vmatrix} + \begin{vmatrix} \phi^1_x & \phi^2_x & \Theta_x & \Psi_x \\ (\phi^1_z)_y & (\phi^2_z)_y & (\Theta_z)_y & (\Psi_z)_y \\ \phi^1_\theta & \phi^2_\theta & \Theta_\theta & \Psi_\theta \\ \phi^1_\psi & \phi^2_\psi & \Theta_\psi & \Psi_\psi \end{vmatrix} + \\
& \begin{vmatrix} \phi^1_x & \phi^2_x & \Theta_x & \Psi_x \\ \phi^1_y & \phi^2_y & \Theta_y & \Psi_y \\ (\phi^1_z)_\theta & (\phi^2_z)_\theta & (\Theta_z)_\theta & (\Psi_z)_\theta \\ \phi^1_\psi & \phi^2_\psi & \Theta_\psi & \Psi_\psi \end{vmatrix} + \begin{vmatrix} \phi^1_x & \phi^2_x & \Theta_x & \Psi_x \\ \phi^1_y & \phi^2_y & \Theta_y & \Psi_y \\ \phi^1_\theta & \phi^2_\theta & \Theta_\theta & \Psi_\theta \\ (\phi^1_z)_\psi & (\phi^2_z)_\psi & (\Theta_z)_\psi & (\Psi_z)_\psi \end{vmatrix}
\end{aligned} \tag{68}
$$

Next, using (23) and the fact that the take-off angels Θ and Ψ are constant along the characteristics given by the system (33), i.e.

$$
\begin{aligned}
\Theta_z + \mathbf{u}\cdot\nabla\Theta &= 0 \\
\Psi_z + \mathbf{u}\cdot\nabla\Psi &= 0\,,
\end{aligned} \tag{69}
$$

we expand the right hand side of (68) and replace all z-derivative by x, y, θ, ψ-derivatives. We can get equation (39) by some algebras.

These computations can be done using the following commands in *Mathematica*.

```
delta1={ {a11,a12,a13,a14}, {a21,a22,a23,a24}, {a31,a32,a33,a34},
{a41,a42,a43,a44} };

a={ {
-u11*a11-u21*a21-u31*a31-u41*a41-u1*a111-u2*a112-u3*a113-u4*a114,
-u11*a12-u21*a22-u31*a32-u41*a42-u1*a121-u2*a122-u3*a123-u4*a124,
-u11*a13-u21*a23-u31*a33-u41*a43-u1*a131-u2*a132-u3*a133-u4*a134,
-u11*a14-u21*a24-u31*a34-u41*a44-u1*a141-u2*a142-u3*a143-u4*a144
}, {a21,a22,a23,a24}, {a31,a32,a33,a34}, {a41,a42,a43,a44} };

b={ {a11,a12,a13,a14}, {
-u12*a11-u22*a21-u32*a31-u42*a41-u1*a211-u2*a212-u3*a213-u4*a214,
-u12*a12-u22*a22-u32*a32-u42*a42-u1*a221-u2*a222-u3*a223-u4*a224,
-u12*a13-u22*a23-u32*a33-u42*a43-u1*a231-u2*a232-u3*a233-u4*a234,
-u12*a14-u22*a24-u32*a34-u42*a44-u1*a241-u2*a242-u3*a243-u4*a244
}, {a31,a32,a33,a34}, {a41,a42,a43,a44} };

c={ {a11,a12,a13,a14}, {a21,a22,a23,a24}, {
-u13*a11-u23*a21-u33*a31-u43*a41-u1*a311-u2*a312-u3*a313-u4*a314,
-u13*a12-u23*a22-u33*a32-u43*a42-u1*a321-u2*a322-u3*a323-u4*a324,
-u13*a13-u23*a23-u33*a33-u43*a43-u1*a331-u2*a332-u3*a333-u4*a334,
-u13*a14-u23*a24-u33*a34-u43*a44-u1*a341-u2*a342-u3*a343-u4*a344
}, {a41,a42,a43,a44} };

d={ {a11,a12,a13,a14}, {a21,a22,a23,a24}, {a31,a32,a33,a34}, {
-u14*a11-u24*a21-u34*a31-u44*a41-u1*a411-u2*a412-u3*a413-u4*a414,
```

```
-u14*a12-u24*a22-u34*a32-u44*a42-u1*a421-u2*a422-u3*a423-u4*a424,
-u14*a13-u24*a23-u34*a33-u44*a43-u1*a431-u2*a432-u3*a433-u4*a434,
-u14*a14-u24*a24-u34*a34-u44*a44-u1*a441-u2*a442-u3*a443-u4*a444 }
};

e=Det[a]+Det[b]+Det[c]+Det[d];

f=e+Det[delta1]*(u11+u22+u33+u44);

In[58]:= D[f,u11]

Out[58]= 0

In[59]:= D[f,u22]

Out[59]= 0

In[60]:= D[f,u33]

Out[60]= 0

In[61]:= D[f,u44]

Out[61]= 0

g1={ {a111,a121,a131,a141}, {a21,a22,a23,a24}, {a31,a32,a33,a34},
{a41,a42,a43,a44} };

g2={ {a11,a12,a13,a14}, {a211,a221,a231,a241}, {a31,a32,a33,a34},
{a41,a42,a43,a44} };

g3={ {a11,a12,a13,a14}, {a21,a22,a23,a24}, {a311,a321,a331,a341},
{a41,a42,a43,a44} };

g4={ {a11,a12,a13,a14}, {a21,a22,a23,a24}, {a31,a32,a33,a34},
{a411,a421,a431,a441} };

g=Det[g1]+Det[g2]+Det[g3]+Det[g4];

In[84]:= D[f,u1]+g

Out[84]= 0

h1={ {a112,a122,a132,a142}, {a21,a22,a23,a24}, {a31,a32,a33,a34},
{a41,a42,a43,a44} };
```

```
h2={ {a11,a12,a13,a14}, {a212,a222,a232,a242}, {a31,a32,a33,a34},
{a41,a42,a43,a44} };

h3={ {a11,a12,a13,a14}, {a21,a22,a23,a24}, {a312,a322,a332,a342},
{a41,a42,a43,a44} };

h4={ {a11,a12,a13,a14}, {a21,a22,a23,a24}, {a31,a32,a33,a34},
{a412,a422,a432,a442} };

h=Det[h1]+Det[h2]+Det[h3]+Det[h4];

i1={ {a113,a123,a133,a143}, {a21,a22,a23,a24}, {a31,a32,a33,a34},
{a41,a42,a43,a44}};

i2={ {a11,a12,a13,a14}, {a213,a223,a233,a243}, {a31,a32,a33,a34},
{a41,a42,a43,a44}};

i3={ {a11,a12,a13,a14}, {a21,a22,a23,a24}, {a313,a323,a333,a343},
{a41,a42,a43,a44} };

i4={ {a11,a12,a13,a14}, {a21,a22,a23,a24}, {a31,a32,a33,a34},
{a413,a423,a433,a443} };

i=Det[i1]+Det[i2]+Det[i3]+Det[i4];

j1={ {a114,a124,a134,a144}, {a21,a22,a23,a24}, {a31,a32,a33,a34},
{a41,a42,a43,a44} };

j2={ {a11,a12,a13,a14}, {a214,a224,a234,a244}, {a31,a32,a33,a34},
{a41,a42,a43,a44} };

j3={ {a11,a12,a13,a14}, {a21,a22,a23,a24}, {a314,a324,a334,a344},
{a41,a42,a43,a44} };

j4={ {a11,a12,a13,a14}, {a21,a22,a23,a24}, {a31,a32,a33,a34},
{a414,a424,a434,a444} };

j=Det[j1]+Det[j2]+Det[j3]+Det[j4];

In[103]:= Simplify[f+g*u1+h*u2+i*u3+j*u4]

Out[103]= 0

In[105]:= Simplify[e+Det[delta1]*(u11+u22+u33+u44)+
g*u1+h*u2+i*u3+j*u4]
```

```
Out[105]= 0
```

References

[1] D. Adalsteinsson and J.A. Sethian. A fast level set method for propagating interfaces. *J. Comput. Phys.*, 118:269–277, 1995.

[2] J.-D. Benamou. Direct solution of multi-valued phase-space solutions for Hamilton-Jacobi equations. *Comm. Pure Appl. Math.*, 52:1443–1475, 1999.

[3] J. D. Benamou. An introduction to Eulerian geometrical optics (1992 - 2002). *J. Sci. Comp.*, 19:63–93, 2003.

[4] R. Burridge, M. V. de Hoop, D. Miller, and C. Spencer. Multiparameter inversion in anisotropic media. *Geophys. J. Internat.*, 134:757–777, 1998.

[5] L.-T. Cheng, H. Liu, and S. J. Osher. High frequency wave propagation in Schrodinger equations using the level set method. *Comm. Math. Sci.*, 1:593–621, 2003.

[6] L.-T. Cheng, S. J. Osher, and J. Qian. Level set based Eulerian methods for multivalued traveltimes in both isotropic and anisotropic media. In *73st Ann. Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts*, pages 1801–1804. Soc. Expl. Geophys., Tulsa, OK, 2003.

[7] J. F. Claerbout. *Imaging the Earth's interior*. Blackwell Scientific Pub., Palo Alto, CA, 1984.

[8] B. Engquist and O. Runborg. Computational high frequency wave propagation. In *Acta Numerica, Pages 1-86*. Cambridge University Press, Cambridge, United Kingdom, 2003.

[9] B. Engquist, O. Runborg, and A-K Tornberg. High frequency wave propagation by the segment projection method. *J. Comp. Phys.*, 178:373–390, 2002.

[10] M. Falcone and R. Ferretti. Semi-lagrangian schemes for hamilton-jacobi equations, discrete representation formulae and godunov methods. *J. Comput. Phys.*, 175:559–575, 2002.

[11] S. Fomel and J. Sethian. Fast phase space computation of multiple traveltimes. *Proc. Nat. Aca. Sci.*, 99:7329–7334, 2002.

[12] S. Geoltrain and J. Brac. Can we image complex structures with first-arrival traveltime. *Geophysics*, 58:564–575, 1993.

[13] L. Gosse. Using K-branch entropy solutions for multivalued geometric optics computations. *J. Comput. Phys.*, 180:155–182, 2002.

[14] L. Gosse and P. A. Markowich. Multiphase semiclassical approximations of an electron in a one-dimensional crystalline lattice. Preprint, 2003.

[15] S. Gray and W. May. Kirchhoff migration using eikonal equation traveltimes. *Geophysics*, 59:810–817, 1994.

[16] S. Jin and X. Li. Multi-phase computations of the semi-classical limit of the Schrodinger equation and related problems. Physica D (in press), 2003.

[17] S. Jin, H. Liu, S. Osher, and R. Tsai. Computing multivalued physical observables for the semi-classical limit of the Schrödinger equation. Preprint, Level Set Systems, 2003.

[18] S. Jin and S. Osher. A level set method for the computation of multivalued solutions to quasi-linear hyperbolic PDEs and Hamilton-Jacobi equations. *Comm. Math. Sci.*, 1:575–591, 2003.

[19] B. R. Julian and D. Gubbins. Three-dimensional seismic ray tracing. *J. Geophys.*, 43:95–113, 1977.

[20] H. Liu, L. T. Cheng, and S. Osher. A level set formulation for tracking multivalued solutions of nonlinear first order equations. Preprint, Level Set Systems, 2003.

[21] X. Liu, S. Osher, and T. Chan. Weighted essentially non-oscillatory schemes. *J. Comput. Phys.*, 115:200–212, 1994.

[22] V. P. Maslov and M. V. Fedoriuk. *Semi-classical approximation in quantum mechanics*. D. Reidel Publishing Company, 1981.

[23] C. Min. Local level set methods in high dimension and codimension. Preprint, UCLA CAM report, 2003.

[24] S. Operto, S. Xu, and G. Lambare. Can we image quantitatively complex models with rays. *Geophysics*, 65:1223–1238, 2000.

[25] S. Osher, L.T. Cheng, M. Kang, H. Shim, and Y.H. Tsai. Geometric optics in a phase-space-based level set and eulerian framework. *J. Comput. Phys.*, 179:622–648, 2002.

[26] D. Peng, B. Merriman, S. Osher, H. K. Zhao, and M. Kang. A pde-based fast local level set method. *J. Comput. Phys.*, 155:410–438, 1999.

[27] J. Qian, L.-T. Cheng, and S.J. Osher. A level set based Eulerian approach for anisotropic wave propagations. *Wave Motion*, 37:365–379, 2003.

[28] J. Qian and S. Leung. A local level set method for paraxial multivalued geometric optics. Submitted to SIAM J. Sci. Comp.; UCLA CAM report 03-60, 2003.

[29] J. Qian and S. Leung. A level set method for paraxial multivalued traveltimes. *J. Comp. Phys.*, 197:711–736, 2004.

[30] J. Qian and W. W. Symes. Adaptive finite difference method for traveltime and amplitude. *Geophysics*, 67:167–176, 2002.

[31] S. J. Ruuth, B. Merriman, and S. J. Osher. A fixed grid method for capturing the motion of self-intersecting interfaces and related PDEs. *J. Comput. Phys.*, 151:836–861, 1999.

[32] C. W. Shu. Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. In B. Cockburn, C. Johnson, C.W. Shu, and E. Tadmor, editors, *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations*, volume 1697, pages 325–432. Springer, 1998. Lecture Notes in Mathematics.

[33] J. Steinhoff, M. Fan, and L. Wang. A new Eulerian method for the computation of propagating short acoustic and electromagnetic pulses. *J. Comput. Phys.*, 157:683–706, 2000.

[34] J. Strain. Tree methods for moving interfaces. *J. Comput. Phys.*, 151:616–648, 1998.

[35] J. Strain. Semi-Lagrangian methods for level set equations. *J. Comput. Phys.*, 151:498–533, 1999.

[36] W. W. Symes. Mathematics of reflection seismology. In *Annual Report, The Rice Inversion Project, (http://www.trip.caam.rice.edu/)*. Rice University, 1995.

[37] W. W. Symes. A slowness matching finite difference method for traveltimes beyond transmission caustics. In *68th Ann. Internat. Mtg., Soc. Expl. Geophys., Expanded Abstracts*, pages 1945–1948. Soc. Expl. Geophys., 1998.

[38] W. W. Symes and J. Qian. A slowness matching Eulerian method for multivalued solutions of eikonal equations. *J. Sci. Comp.*, 19:501–526, 2003.

[39] V. Vinje, E. Iversen, K. Åstebøl, and H. Gjøystdal. Estimation of multivalued arrivals in 3d models using wavefront construction - part 1. *Geophys. Prosp.*, 44:819–842, 1996.

[40] V. Vinje, E. Iversen, and H. Gjystdal. Traveltime and amplitude estimation using wavefront construction. *Geophysics*, 58:1157–1166, 1993.