Journal of Computational Physics **178,** 373–390 (2002)
doi:10.1006/jcph.2002.7033, available online at http://www.idealibrary.com on IDEAL®

High-Frequency Wave Propagation by the Segment Projection Method

Björn Engquist,*[,1] Olof Runborg,†[,2] and Anna-Karin Tornberg‡[,3]

*Department of Mathematics, Princeton University, Princeton, New Jersey; †Department of Numerical Analysis and Computer Science, KTH, Stockholm, Sweden; and ‡Courant Institute of Mathematical Sciences, New York University, New York, New York

E-mail: engquist@math.princeton.edu; olofr@nada.kth.se; and tornberg@cims.nyu.edu

Received May 1, 2001; revised February 13, 2002

Geometrical optics is a standard technique used for the approximation of high-frequency wave propagation. Computational methods based on partial differential equations instead of the traditional ray tracing have recently been applied to geometrical optics. These new methods have a number of advantages but typically exhibit difficulties with linear superposition of waves. In this paper we introduce a new partial differential technique based on the segment projection method in phase space. The superposition problem is perfectly resolved and so is the problem of computing amplitudes in the neighborhood of caustics. The computational complexity is of the same order as that of ray tracing. The new algorithm is described and a number of computational examples are given, including a simulation of waveguides. © 2002 Elsevier Science (USA)

Key Words: wave equation; eikonal equation; geometrical optics; segment projection method.

1. INTRODUCTION

High-frequency wave propagation is well approximated by asymptotic formulations such as geometrical optics and the geometrical theory of diffraction. These formulations can be the basis of computations or they can be used analytically for the understanding of high-frequency phenomena.

[1] The research was partially supported by NSF Grant DMS-0072112 and PSCI at KTH.

[2] The research was carried out at the Program for Applied and Computational Mathematics, Princeton University, supported by NSF KDI Grant DMS-9872890.

[3] The research was carried out at the Department of Numerical Analysis and Computer Science, KTH, Stockholm, supported by TFR Grant 222-2000-434.

0021-9991/02 $35.00

Let us consider the scalar wave equation

$$\begin{aligned} \frac{\partial^2 u(\mathbf{x},t)}{\partial t^2} &= c(\mathbf{x})^2 \Delta u(\mathbf{x},t), \quad \mathbf{x}\in\mathbb{R}^d, t>0, \\ u(\mathbf{x},0) &= u_0(\mathbf{x}), \qquad\qquad \mathbf{x}\in\mathbb{R}^d, \\ \frac{\partial u(\mathbf{x},0)}{\partial t} &= u_1(\mathbf{x}), \qquad\qquad \mathbf{x}\in\mathbb{R}^d. \end{aligned} \tag{1}$$

When u_0 or u_1 contains high-frequency components it is customary to use the ansatz

$$u(\mathbf{x},t) \sim e^{i\omega\varphi(\mathbf{x},t)} \sum_{j=0}^{\infty} a_j(\mathbf{x},t)(i\omega)^j, \tag{2}$$

or

$$u(\mathbf{x},t) \sim e^{i\omega(\varphi(\mathbf{x})-t)} \sum_{j=0}^{\infty} a_j(\mathbf{x})(i\omega)^j, \tag{3}$$

to derive the eikonal equations for the phase φ [9]. With $\omega\to\infty$ we get from (2)

$$\frac{\partial\varphi}{\partial t} + c(\mathbf{x})|\nabla\varphi| = 0, \tag{4}$$

and from (3)

$$|\nabla\varphi| = \frac{1}{c(\mathbf{x})}. \tag{5}$$

The transport equations for the leading amplitude functions a_0 are for (2) and (3), respectively, given by

$$\frac{\partial a_0}{\partial t} + c\frac{\nabla\varphi\cdot\nabla a_0}{|\nabla\varphi|} + \frac{c^2\nabla\varphi - \frac{\partial^2\varphi}{\partial t^2}}{2c|\nabla\varphi|}a_0 = 0, \tag{6}$$

$$2\nabla\varphi\cdot\nabla a_0 + \Delta\varphi a_0 = 0. \tag{7}$$

Ray tracing can be regarded as the method of characteristics applied to the eikonal equations. The ray $\mathbf{x}(t)$ is parameterized by t and the local direction of the ray is $\mathbf{p} = \nabla\varphi(\mathbf{x})$, $|\mathbf{p}| = c^{-1} = \eta$,

$$\frac{d\mathbf{x}}{dt} = c^2\,\mathbf{p}, \tag{8}$$

$$\frac{d\mathbf{p}}{dt} = c\nabla_x\eta. \tag{9}$$

The system (8), (9) can be augmented by an equation for the amplitude and approximated by standard numerical methods for ordinary differential equations.

Ray tracing techniques have difficulties with resolution in regions of diverging rays, so simulations based directly on the eikonal equation have recently been introduced [22, 23] (see also [7, 8]).

The eikonal equation is of Hamilton–Jacobi type and has a unique viscosity solution [4]. This means that the phase has a unique value and that crossing rays or linear superposition of waves are not allowed. The viscosity solution only gives the first arriving ray or wave front.

One way to resolve this problem is to geometrically decompose the region of the independent variables where there should be multiple rays. Extra-eikonal equation computations can be performed in these domains to approximate each separate wave field in the superposition. This was briefly discussed in [5] and more systematically developed by Benamou [1, 2] and Symes [18].

It is possible to derive a Vlasov-type partial differential equation from (8), (9) which gives the amplitude and allows for superposition [11],

$$\frac{\partial w}{\partial t} + c^2 \mathbf{p} \cdot \nabla_x w + c \nabla_x \eta \cdot \nabla_p w = 0. \tag{10}$$

The density $w = w(\mathbf{x}, \mathbf{p}, t)$ is a function of location, the ray vector, and time. It is related to the amplitude as $a_0^2 = \int w \, d\mathbf{p}$. If w has support on $|\mathbf{p}| = c^{-1}$ initially this will be valid for all time and w can be regarded as a function of $\mathbf{p}/|\mathbf{p}|$ and thus of six independent variables for $\mathbf{x} \in \mathbb{R}^3$.

When (10) is used for simulations the drawback is the computational complexity. Let h be of the order of the numerical step size for the independent variables. The computational complexity is $O(h^{-6})$ in $\mathbb{R}^3$ and $O(h^{-4})$ in $\mathbb{R}^2$ for an explicit difference method approximating (10). This should be compared to simulations based on ray tracing for the time harmonic problem. The corresponding complexities for the same resolutions are $O(h^{-3})$ in $\mathbb{R}^3$ and $O(h^{-2})$ in $\mathbb{R}^2$. The complexities for explicit approximations of the eikonal equation (4) are $O(h^{-4})$ in $\mathbb{R}^3$ and $O(h^{-3})$ in $\mathbb{R}^2$. The fast marching method [14] can be used for (4) and the complexity is reduced to $O(h^{-3} \log h)$ and $O(h^{-2} \log h)$, respectively.

A different approach was taken in [6, 12]. The eikonal equation is replaced by a system of nonlinear conservation laws that allows for a limited number of linear superpositions. The conservation laws are derived by integrating Eq. (10) with respect to $\mathbf{p}$ and using the finite number of superpositions as closure. The number of independent variables is then reduced to four in $\mathbb{R}^3$ and three in $\mathbb{R}^2$ but the numerical approximations exhibit problems when there are several crossing rays (see also [3] for the technique of reducing the number of independent variables).

Another method was introduced by Steinhoff and collaborators in [15–17] and further refined by Ruuth *et al.* in [13]. The dependent variables in this technique are essentially the coordinates of the closest point on a wave front from a given $\mathbf{x}$-coordinate. This clever choice of representation and a time-stepping scheme following the rules of geometrical optics allows for linear superposition. The present forms of the method have higher complexity than ray tracing and certain cases will not be correctly described with the initial version. One such example is a wave front given by a collapsing circle with two parallel tangent lines. When the circle and the tangent lines have been reduced to one line the information of the circle is lost and cannot be recovered. The method is quite straightforward and is being further developed.

The new technique we introduce here is also based on the tracking of fronts, but it is done in phase space. In two space dimensions a front is given as a curve in $xy\theta$ space where $\theta = \cos^{-1}(p_1/|\mathbf{p}|)$ is the angle between the normal of the front and the x-axis. The curve is represented by its projection onto the two-dimensional xy, $x\theta$, and $y\theta$ spaces. The representation of the curve in each of these spaces is evolved by the segment projection method [19]. The segment projection method allows for the superposition of fronts and the use of the phase plane makes it possible to describe caustics, including a correctly approximated amplitude. The motion of the curve is determined by the solution of partial

differential equations in one space variable; in spite of the extension to the phase plane the computational complexity is not more than that of standard ray tracing. This is a result of the projection onto the lower dimensional spaces. In the segment projection a curve in two dimensions is represented as a union of segments. These segments are chosen such that they can be given as a function of one of the variables. In three dimensions the segments are projected onto the two-dimensional coordinate planes. The spatial resolution should be of the same order as is required by ray tracing. The method for the evolution of the segments is explicit and the number of steps will in general be proportional to the number of steps in the ODE solver for the rays.

The general form of the segment projection method for one-dimensional curves in $\mathbb{R}^2$ and $\mathbb{R}^3$ is given in Section 2. This is followed in Section 3 by a presentation of the method adapted to geometrical optics. The final section contains numerical examples. In these sections we use the notation $\mathbf{x} = (x_1, x_2)$ or (x, y) whenever convenient.

2. THE SEGMENT PROJECTION METHOD

The segment projection method is a computational method for tracking the dynamic evolution of interfaces. It was recently developed in [19, 20]. The basic idea is to represent a curve or surface as a union of segments. Each segment is chosen such that it can be given as a function of the independent variables. The representation is thus analogous to a manifold being defined by an atlas of charts. The motions of the individual segments are given by partial differential equations based on the physics describing the evolution of the interfaces.

General software has been developed for curves γ in $\mathbb{R}^2$ and we give a brief overview of the method. The segments are here represented by functions $Y_j(x)$ and $X_k(y)$. The domains of the independent variables of these functions are projections of the segments onto the coordinate axis. The coordinates of the points on γ are given by $(x, y) = (x, Y_j(x))$ or $(x, y) = (X_k(y), y)$. For each point on a curve γ, there is a least one segment defining the curve. To make the description complete, information about the connectivity of segments must also be provided. For each segment in one variable there is information regarding which parts of the curve have overlap with segments in the other variable, as well as pointers to these segments.

The number of segments needed to describe a curve depends on the shape of the curve. An extremum of a function $Y_j(x)$ defines a separation point for the y-segments, as no segment given as a function of y can continue past this point. Similarly, an extremum of a function $X_k(y)$ defines a separation point for the x-segments. A sketch of a distribution of segments is shown in Fig. 1. For moving interfaces, $Y_j = Y_j(x, t)$ and $X_k = X_k(y, t)$ are also functions of time.

The segments are moved by equations of motion, and after each numerical advection step, the segment representation is reinitialized. Dynamic creation and elimination of segments are employed to follow the evolution of the curves. New segments are created if new extrema have appeared, and segments are removed when extrema disappear. The connectivity of segments must be kept updated such that the pointers relating the segments represents the current configuration. If we assume that the lower deformed circle in Fig. 1 has evolved from the circle above it, a new maximum and two new minima have appeared in the lower x-segment. The number of y-segments should then be increased, as is seen in the figure.

For each segment, the domain of the independent variable must be defined. These segments are numerically given by arrays for Y_j and X_k. The domains of the independent

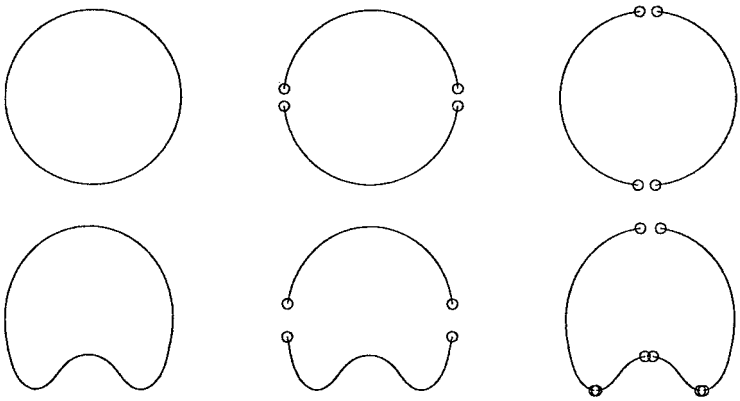

FIG. 1. Segment structure for circle and deformed circle. (Left) Curve γ; (middle) x-segments; (right) y-segments.

variables, the arrays, and information about connectivity between the segments define the structure that represents the curve.

From the definition of an x-segment, an ordered set of numbers is created that contains the start and endpoints of the segment, together with the extremum points of the segment. The intervals between these points correspond to different segments of the other variable. It is necessary to keep track of the connections between these segments.

Very short segments are not practical to work with for the evolution of γ. They are, however, needed in the structure of connectivity and are defined as empty segments. If the difference between two consecutive extrema in Y_j is less than $2\Delta y$ in absolute value, the corresponding y-segment is defined as empty.

Let a velocity field $\mathbf{u} = (u, v)^T$ be given by which the curve should move. The segments $y = Y(x, t)$ and $x = X(y, t)$ are updated according to the partial differential equations

$$\frac{\partial Y}{\partial t} + u\frac{\partial Y}{\partial x} = v, \tag{11}$$

$$\frac{\partial X}{\partial t} + v\frac{\partial X}{\partial y} = u. \tag{12}$$

Note that there is only one spatial variable present in each of these equations. Quantities that are transported by the velocity field can also be defined as functions on the segments. Let $\mathcal{F}^t(x, y)$ be the flow generated by $\mathbf{u}$. If $r(x, y, t)$ evolves according to the ODE

$$\frac{dr(\mathcal{F}^t, t)}{dt} = h(\mathcal{F}^t, r(\mathcal{F}^t, t), t),$$

for fixed (x, y), then

$$\frac{\partial R^x}{\partial t} + u\frac{\partial R^x}{\partial x} = h(x, Y, R^x, t), \tag{13}$$

$$\frac{\partial R^y}{\partial t} + v\frac{\partial R^y}{\partial y} = h(X, y, R^y, t), \tag{14}$$

where $R^x(x, t) = r(x, Y(x, t), t)$ and $R^y(y, t) = r(Y(y, t), y, t)$.

Boundary conditions must be defined for the segments. They are either given in the original problem formulation or interpolated from an overlapping segment in the other coordinate direction. Note that this interpolation is well defined as an interpolation on an irregular mesh from the discrete segment in the other coordinate direction.

After the numerical advection step based on (11), (12), we need to review the segment structure. If no new extrema have appeared, and no old have disappeared, no change needs to be made in the structure of the segments. We need, however, to update the positions of the extremum points, and to correct the corresponding segment information according to this. If new extrema have appeared or disappeared, the structure of the segments defined as functions of the other spatial variable needs to be modified.

Any moving curve γ is represented by overlapping segments. These segments evolve individually and may separate slightly in the overlapping regions due to numerical errors. A reinitialization is applied in every time step to realign the segments. This is done by a weighted interpolation. A segment will typically yield the most accurate description of the curve if its slope is small.

For a given point x_k, an x-segment has the value $f_k = Y(x_k, t)$. Denote by $\tilde{f}_k$ the value at x_k, interpolated from the discretized segment in the y-direction. Compute the slope of the two segments at this point and denote the average by s_k. The new value at x_k is set to

$$f_k^{\text{new}} = f_k + \theta(s_k)(\tilde{f}_k - f_k), \tag{15}$$

where the weight $\theta(s)$ is a function of the slope s, given by

$$\theta(s) = \begin{cases} 0 & \text{for } s < 1/\beta, \\ \frac{\beta}{\beta^2 - 1}(s - 1/\beta) & \text{for } 1/\beta \le s \le \beta, \\ 1 & \text{for } s > \beta. \end{cases} \tag{16}$$

The numerical examples in Section 4 are done with $\beta = 2$. For very coarse grids a smaller β could be used. The same interpolation is also applied to the functions R^x and R^y that are defined on the segments.

When different parts of γ cross each other, geometric rules for the segment interaction must be given. Examples are the merging of two bubbles in multiphase flow and the reflection of a wave front γ_1 meeting a curve γ_2, representing a perfect reflector.

The advection and reinitialization process for a structure of segments, representing a curve γ, can be summarized as follows:

1. Advect all x- and y-segments from a velocity field $\mathbf{u} = (u, v)$ and evolve associated quantities defined on the segments by numerical approximations of (11)–(14).

2. Update the segment structure. For each segment (a) check to see if any new local extrema have appeared or if any local extrema have disappeared by examining any change in monotonicity in the sequence of the points defining the segments, and split or merge corresponding segments of the other coordinate; (b) review the use of empty segments and add or remove short segments, as described above; and (c) use the position of the extrema to update the start index and the stop index of the independent variable for the segment of the other coordinate.

3. For each segment whose domain of definition has increased, new values need to be defined. These are interpolated from the corresponding segment in the other coordinate direction.

4. Interpolate the segment between overlapping parts of the x- and y-segments. The new values are assigned using a weight function based on the slopes of the segments.

5. Rearrange the segment structure from the rules of segment interactions.

These steps are generic and essentially the same for different applications. A common software will thus apply to different problems with only minor modifications in, for example, the advection and the interaction algorithms.

Curves of codimension two in $\mathbb{R}^3$ are approximated by their projections onto the two-dimensional coordinate planes. If a curve

$$\gamma(t) : \{X_1(s,t), X_2(s,t), X_3(s,t)\}$$

is parameterized by s and evolves by the velocity field

$$\mathbf{u}(\mathbf{x},t) = (u_1(\mathbf{x},t), u_2(\mathbf{x},t), u_3(\mathbf{x},t)),$$

we have

$$\frac{dX_j}{dt} = u_j(\mathbf{X},t), \quad j = 1,2,3.$$

Let the projection of γ onto the x_jx_k-plane be represented by a set of segment functions of the type $x_j = X^{jk}(x_k,t)$ and $x_k = X^{kj}(x_j,t)$. The evolution of the segment functions in all three projection planes is then given by the equations

$$\frac{\partial X^{jk}}{\partial t} + u_k \frac{\partial X^{jk}}{\partial x_k} = u_j, \quad j = 1,2,3; \quad k = 1,2,3; \quad j \neq k. \tag{17}$$

The projections in the x_jx_k-planes are updated in time following steps 1–5 above. A sixth step is then added with interpolation between the representations in the three coordinate planes. This step is similar to step 4 and it is in general needed in order to define u_k and u_j in (17). The simulations presented in this paper do not require such interpolations, however.

There is yet no general software for two-dimensional surfaces in $\mathbb{R}^3$. The principle is analogous to the lower dimensional case. The surface Σ is represented by functions defined on the three coordinate planes. The functions $x_\ell = X_\ell^{jk}(x_j,x_k,t)$ define the segments as in $\mathbb{R}^2$ and the union of segments defines Σ. Given the velocity field $\mathbf{u}(\mathbf{x},t)$, the motion of the segments is given by

$$\frac{\partial X_\ell^{jk}}{\partial t} + u_j \frac{\partial X_\ell^{jk}}{\partial x_j} + u_k \frac{\partial X_\ell^{jk}}{\partial x_k} = u_\ell,$$
$$j = 1,2,3, \quad k = 1,2,3, \quad \ell = 1,2,3, \quad j \neq k, \quad j \neq \ell, \quad k \neq \ell.$$

3. THE SEGMENT PROJECTION METHOD FOR GEOMETRICAL OPTICS

The segment projection method is applied to track the evolution in phase space of fronts that are given by geometrical optics. For two space dimensions a curve γ in $\mathbb{R}^3$ is tracked and for three space dimensions a surface Σ in $\mathbb{R}^5$ is evolved. We mainly discuss the two-dimensional case and only give one three-dimensional example, in Section 4.2. Section 3.1 discusses how to reduce the two-dimensional case to just one dimension. In the presentation of the method, we may assume that γ or Σ and their projections are functions. The segment

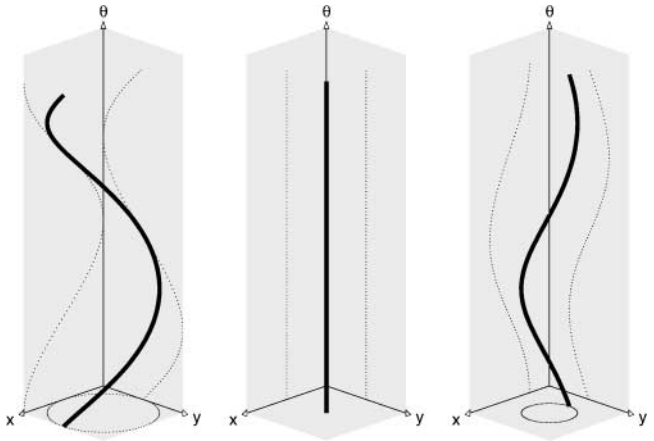

FIG. 2. Phase plane curve γ: thick line, with projections onto xy-, $x\theta$-, and $y\theta$-planes. (Left frame) Initial circular wave front, $t = 0$; (middle frame) focus at $t = 1$; (right frame) $t = 1.5$.

projection technique reduces the general case to a set of segments that are functions of some of the independent variables.

Let the independent variables be x, y, and θ. The orthogonal projections of γ onto the xy-, $x\theta$-, and $y\theta$-planes are denoted by γ_{xy}, $\gamma_{x\theta}$, and $\gamma_{y\theta}$, respectively. The evolution of $\gamma = \gamma(t)$ will be determined by the two-dimensional segment projection method, as presented in Section 2.

In Fig. 2, left frame, an initial circular wave front is given in the xy-plane. This frame also displays the phase plane curve γ in $\mathbb{R}^3$ together with its $x\theta$- and $y\theta$-projections. Let the circular wave front contract with time and be focused to a point $(x, y) = (1, 1)$ at time $t = 1$, $c(\mathbf{x}) \equiv 1$. The representations of γ at $t = 1$ in the $x\theta$- and $y\theta$-planes are smooth and the evolution can easily be continued to $t > 1$.

The evolution of a wave front in the xy-plane is given by the velocity $c(\mathbf{x})$ in its normal direction $\mathbf{n}$. The velocity field $\mathbf{u} = (u, v)$ in the xy-plane is thus

$$(u, v) = c(\mathbf{x})\mathbf{n} = c(\mathbf{x})\ (\cos\theta, \sin\theta), \tag{18}$$

where θ is the angle between the normal vector and the x-axis. The normal vector can be computed from the segment functions.

From the general equations (11) and (12) and the velocity field (18), we get the Eulerian form of the evolution equations for the x- and y-segments in the xy-plane, respectively,

$$\begin{cases} \frac{\partial Y^x}{\partial t} + c(x, Y^x(x,t))\cos\theta\,\frac{\partial Y^x}{\partial x} = c(x, Y^x(x,t))\sin\theta, \\ \frac{\partial X^y}{\partial t} + c(X^y(y,t), y)\sin\theta\,\frac{\partial X^y}{\partial y} = c(X^y(y,t), y)\cos\theta. \end{cases} \tag{19}$$

The approach of tracking the front only in the xy-plane, computing θ from the segments, breaks down at caustics. Therefore, the front must be tracked in phase space, and the other two projections are needed.

With $\mathbf{p} = (\cos\theta, \sin\theta)/c(\mathbf{x})$ we can derive from (8), (9)

$$\frac{d\theta}{dt} = \frac{\partial c}{\partial x}\sin\theta - \frac{\partial c}{\partial y}\cos\theta. \tag{20}$$

This equation, together with (18), gives the velocity field needed to apply (11) and (12) to the segment equations in the $x\theta$- and $y\theta$-planes.

Let the x- and θ-segments in the $x\theta$-plane be denoted by Θ^x and X^θ and the y and θ segments in the $y\theta$-plane be Θ^y and Y^θ, respectively. The segment equations are

$$\begin{cases} \frac{\partial \Theta^x}{\partial t} + c\cos\theta \frac{\partial \Theta^x}{\partial x} = \alpha, \\ \frac{\partial X^\theta}{\partial t} + \alpha \frac{\partial X^\theta}{\partial \theta} = c\ \cos\theta, \end{cases} \tag{21}$$

$$\begin{cases} \frac{\partial \Theta^y}{\partial t} + c\sin\theta \frac{\partial \Theta^y}{\partial y} = \alpha, \\ \frac{\partial Y^\theta}{\partial t} + \alpha \frac{\partial Y^\theta}{\partial \theta} = c\ \sin\theta, \end{cases} \tag{22}$$

$$\alpha = \frac{\partial c(\mathbf{x})}{\partial x}\sin\theta - \frac{\partial c(\mathbf{x})}{\partial y}\cos\theta. \tag{23}$$

The one-dimensional hyperbolic equations above are easily solved by standard numerical methods. Note that the representation of the phase plane curve γ may be degenerate for the projection onto one of the coordinate planes, but there will always be two projections for which γ is well represented.

When η is constant the amplitude on the curve can easily be calculated by post processing of the results from (21), (22). (In the next section we compute the amplitude for problems with variable η.) By (10), the intensity $A^2 = \int w\, d\mathbf{p}$ is a conserved quantity in time and changes (pointwise) only through geometrical spreading. Consider for instance an initial curve $(x_0(\theta),\ y_0(\theta),\ \theta)$ with amplitude $A_0(\theta)$ moving in the normal direction $(\cos\theta, \sin\theta)^T$. At time t it will have evolved to $(x(\theta, t),\ y(\theta, t),\ \theta)$, where

$$x(\theta, t) = x_0(\theta) + t\cos\theta, \qquad y(\theta, t) = y_0(\theta) + t\sin\theta.$$

Let $J(\theta, t)$ be the Jacobian of the mapping $(\theta, t) \mapsto (x(\theta, t),\ y(\theta, t))$ and set $q = |\det J|$. Then the amplitude at time t is given by

$$A^2(\theta, t) = \frac{A_0^2(\theta) q(\theta, 0)}{q(\theta, t)}, \tag{24}$$

and since $(x_\theta,\ y_\theta) \perp (\cos\theta, \sin\theta)$ by the construction

$$q(\theta, t) = |x_\theta y_t - y_\theta x_t| = |x_\theta \sin\theta - y_\theta \cos\theta| = ((x_\theta(\theta, t))^2 + (y_\theta(\theta, t))^2)^{1/2}.$$

We note finally that q can be computed from $X^\theta(\theta, t)$ and $Y^\theta(\theta, t)$ in (21), (22).

3.1. Reduction to One Dimension

In many problems the x-component of the ray vector is always positive and the x-axis can thus be used as the evolution direction. Time is not explicitly needed in the calculation and θ can be computed as a function of x and y, and y as a function of x and θ. From (8), (9), (20) we get the velocity field for this setting,

$$\mathbf{u} = (u, v)^T = \left(\tan\theta,\ \frac{1}{\eta}(\eta_y - \eta_x \tan\theta)\right)^T,$$

and the partial differential equation for the segments $\theta = \Theta(x, y)$ and $y = Y(x, \theta)$ are

$$\frac{\partial \Theta}{\partial x} + u\frac{\partial \Theta}{\partial y} = v,$$
$$\frac{\partial Y}{\partial x} + v\frac{\partial Y}{\partial \theta} = u.$$

The arrival time, T, is now a quantity defined on the phase plane curve, and it can be computed according to (13), (14). Let $\mathcal{F}^x(y, \theta) = (\mathcal{F}_1^x, \mathcal{F}_2^x)^T$ be the flow generated by $\mathbf{u}$. By inverting (8), we see that $dT(x, \mathcal{F}^x)/dx = \eta/\cos\theta$. This yields the following differential equations, defined for the y- and θ-segments, respectively,

$$\begin{aligned} \frac{\partial T^y}{\partial x} + u\frac{\partial T^y}{\partial y} &= \frac{\eta}{\cos\theta}, \\ \frac{\partial T^\theta}{\partial x} + v\frac{\partial T^\theta}{\partial \theta} &= \frac{\eta}{\cos\theta}. \end{aligned} \tag{25}$$

To derive the equations for the amplitude, we consider the reduced form of (10) for the phase space density $w = w(x, y, \theta)$,

$$\frac{\partial w}{\partial x} + u\frac{\partial w}{\partial y} + v\frac{\partial w}{\partial \theta} = 0, \tag{26}$$

where again the amplitude is given by $A^2 = \int w\, d\theta$. Suppose the initial wave at $x = 0$ is a curve in the phase plane defined by $G(y, \theta) = 0$ with amplitude $A_0(y, \theta)$. This corresponds to the initial data

$$w(0, y, \theta) = w_0(y, \theta) = A_0^2(y, \theta)\delta(G(y, \theta))|G_\theta(y, \theta)|$$

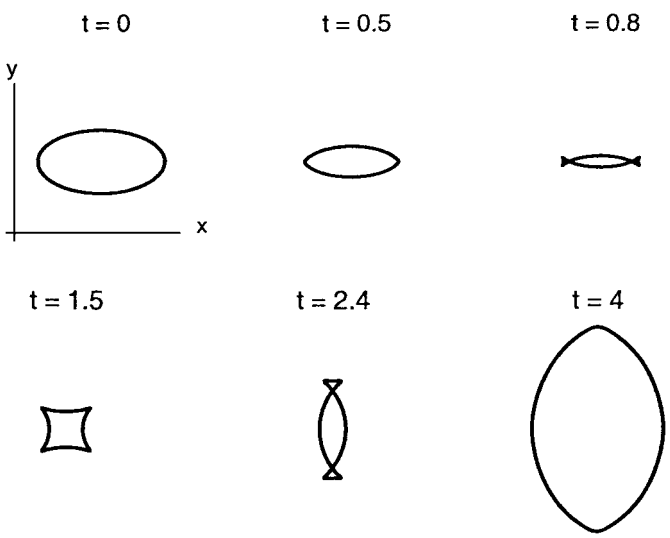

FIG. 3. Evolution of an initially elliptical wave front in the xy-plane.

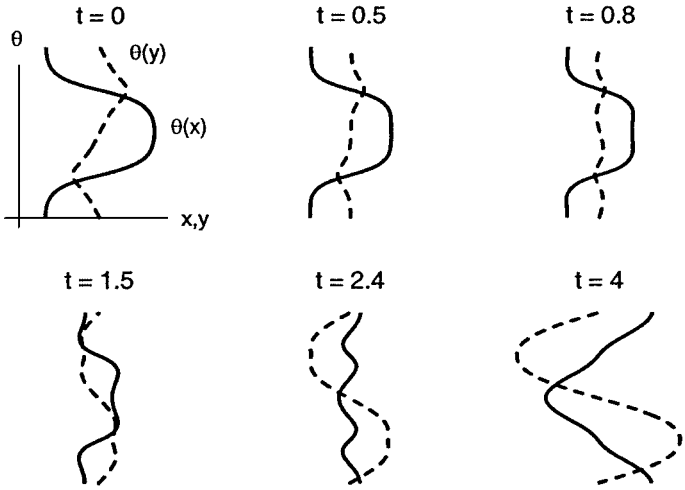

FIG. 4. Projection of phase plane curves γ. (Solid lines) $x\theta$-plane; (dashed lines) $y\theta$-plane.

for (26) whose solution is then $w(x, y, \theta) = w_0(\mathcal{F}^{-x}(y, \theta))$. When Θ is well defined, $G(\mathcal{F}^{-x}(y, \Theta(x, y))) = 0$, and the amplitude on the segment is given by

$$A^2(x, y) = \int w_0(\mathcal{F}^{-x}(y, \theta))\, d\theta = \frac{A_0^2(\mathcal{F}^{-x})|G_\theta(\mathcal{F}^{-x})|}{\left|\nabla_{y\theta} G(\mathcal{F}^{-x}) \cdot \frac{\partial \mathcal{F}^{-x}}{\partial \theta}\right|}(y, \Theta(x, y))$$

$$= \frac{A_0^2(\mathcal{F}^{-x})|G_\theta(\mathcal{F}^{-x})||\det \mathcal{J}^x|}{\left|G_y(\mathcal{F}^{-x})\frac{\partial \mathcal{F}_1^x}{\partial \theta} - G_\theta(\mathcal{F}^{-x})\frac{\partial \mathcal{F}_1^x}{\partial y}\right|}(y, \Theta(x, y)),$$

where $\nabla_{y\theta} = (\partial_y, \partial_\theta)$ and $\mathcal{J}^x$ is the Jacobian of $\mathcal{F}^x$. The same equality holds when

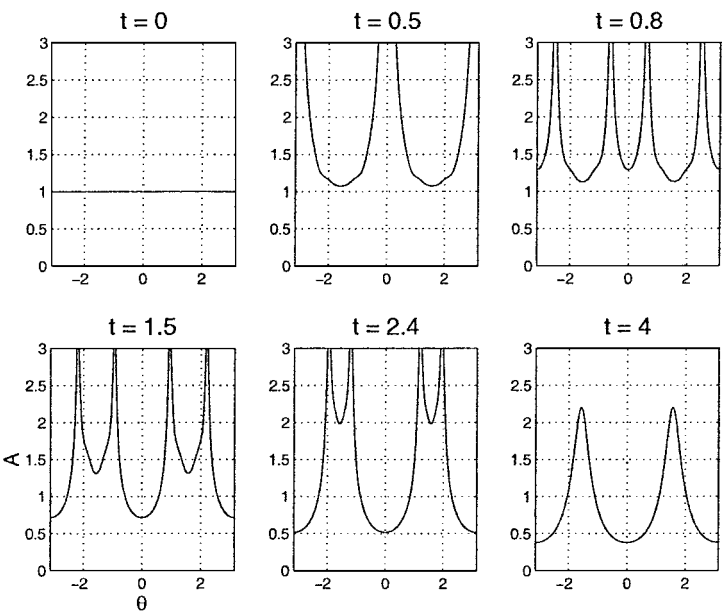

FIG. 5. Evolution of amplitude as a function of θ.

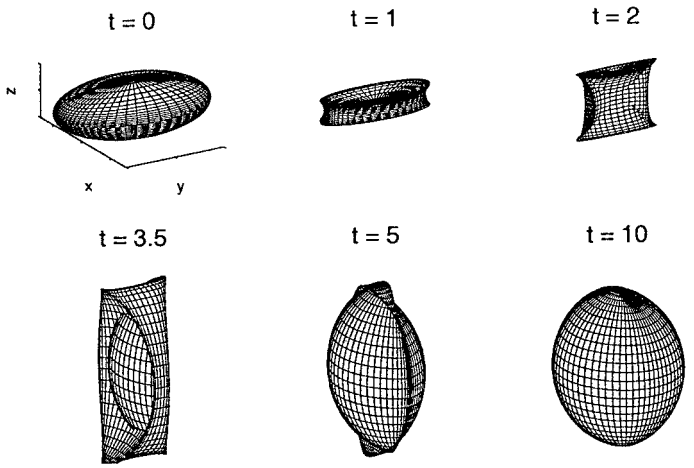

FIG. 6. Evolution of wave front in xyz space.

$(y, \Theta(x, y))$ is replaced by $(Y(x, \theta), \theta)$. In order to compute $A(x, y)$ we must hence also evolve $\mathcal{F}^{-x}$ and $\mathcal{J}^x$ as quantities on the curve. Let $D\mathbf{u}$ be the Jacobian of $\mathbf{u}(x, y, \theta)$ with respect to (y, θ). We have

$$\frac{d\mathcal{F}^{-x}(\mathcal{F}^x)}{dx} = 0, \qquad \frac{d\mathcal{J}^x(\mathcal{F}^x)}{dx} = D\mathbf{u}(\mathcal{F}^x)\mathcal{J}^x(\mathcal{F}^x). \tag{27}$$

Note, both $\mathcal{F}^{-x}$ and $\mathcal{J}^x$ remain bounded and smooth also at caustics, where A becomes infinite.

The quantities are then given by the PDEs

$$\frac{\partial F^y}{\partial x} + u\frac{\partial F^y}{\partial y} = 0, \qquad \frac{\partial J^y}{\partial x} + u\frac{\partial J^y}{\partial y} = D\mathbf{u}J^y,$$
$$\frac{\partial F^\theta}{\partial x} + v\frac{\partial F^\theta}{\partial \theta} = 0, \qquad \frac{\partial J^\theta}{\partial x} + v\frac{\partial J^\theta}{\partial \theta} = D\mathbf{u}J^\theta.$$

Here, F^y, F^θ give $\mathcal{F}^{-x}$ and J^y, J^θ give $\mathcal{J}^x$ on the segments $\Theta(x, y)$ and $Y(x, \theta)$, respectively.

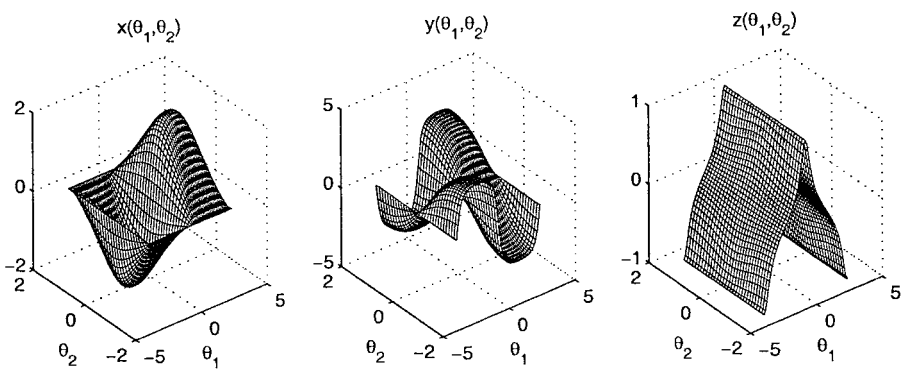

FIG. 7. Projections of phase space surface Σ onto different coordinate planes at $t = 0$.

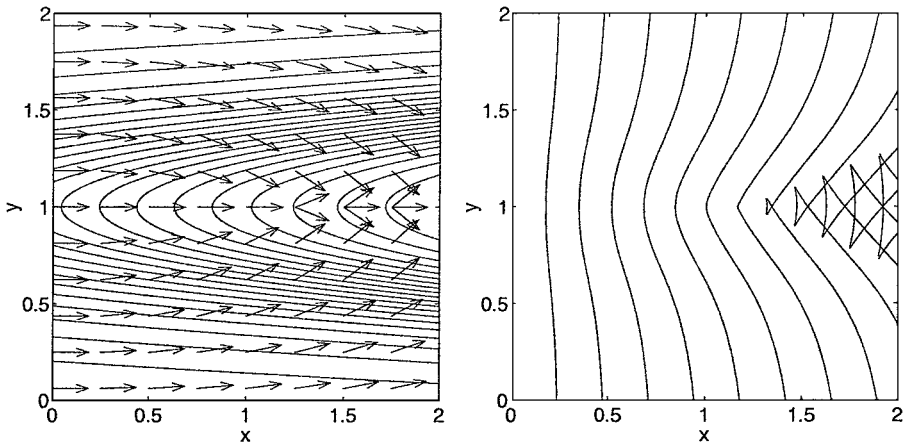

FIG. 8. Results for the lens simulation. (Left frame) Local ray directions with contour lines of index of refraction overlayed; (right frame) wave front in the xy-plane for a sequence of arrival times (T values).

Initial data for those equations are $F^y(0, y) = (y, \Theta(0, y))^T$, $F^\theta(0, \theta) = (Y(0, \theta), \theta)^T$, and $J^y(0, y) = J^\theta(0, \theta) = I$, the identity.

4. NUMERICAL EXAMPLES

We present four computational examples in order to describe different aspects of the method. All of the examples involve caustics and superposition. The main purpose is to give a proof of concept, and for simplicity we have used the first-order-accurate Lax–Friedrichs scheme as evolution method.

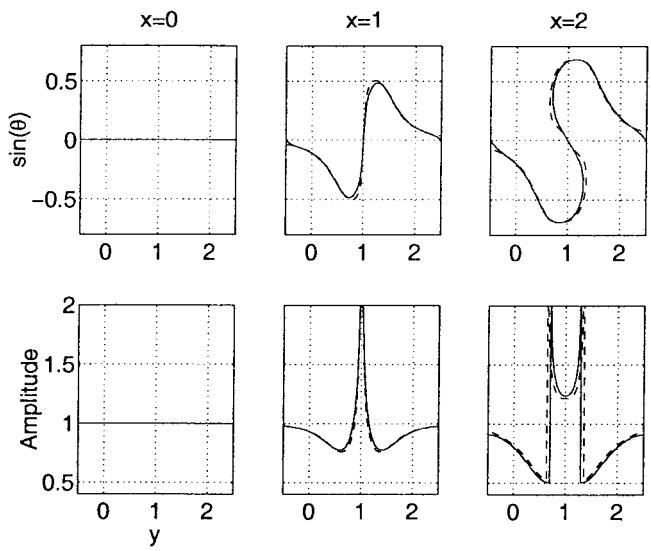

FIG. 9. Results at different x-positions for the lens simulation. (Upper row) Phase plane curve; (lower row) amplitude; (solid line) $\Delta y = \Delta\theta = 0.01$; (dashed line) high-resolution ray tracing.

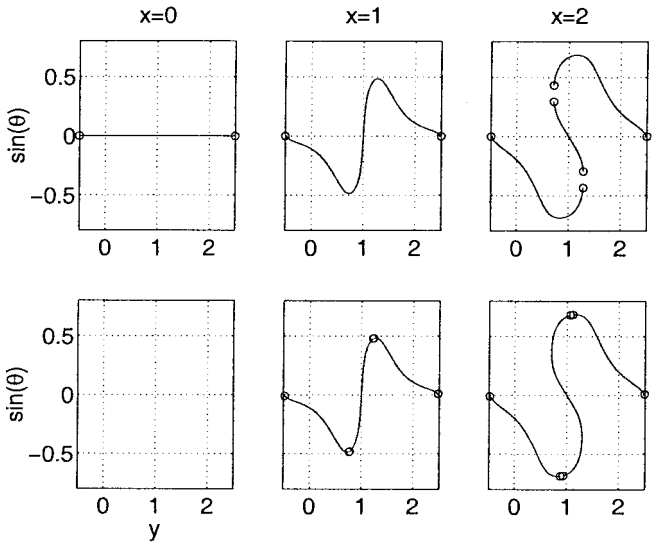

FIG. 10. Segment structure for lens simulation corresponding to phase plane curve in Fig. 9. (Upper row) y-Segments; (Lower row) θ-segments.

4.1. Contracting Elliptical Wave Front

The initial values in this example correspond to a one-dimensional elliptical wave front in $\mathbb{R}^2$ (see Fig. 3). The initial motion is contraction and the index of refraction is constant. The projections of the front in phase space onto the $x\theta$- and $y\theta$-planes are smooth even through caustics (Fig. 4). The function $\alpha \equiv 0$ (see (23)), which simplifies the calculations. No differential equation needs to be solved in the xy-plane. Equations (21), (22) are sufficient and the xy-location of the wave front is given by X^θ, Y^θ. With constant index of refraction in the examples of Sections 4.1 and 4.2 there is no need to perform the interpolation that was discussed in Section 2 as a sixth step in the segment projection process. There is no problem in calculating the amplitude through the formation of caustics using (24) (Fig. 5), even on a coarse grid; the quantities that we solve for are smooth. However, in order to resolve the spikes in the postprocessed amplitude for the presentation, we used a fairly dense grid, $\Delta\theta = 2\pi/512 \approx 0.01$.

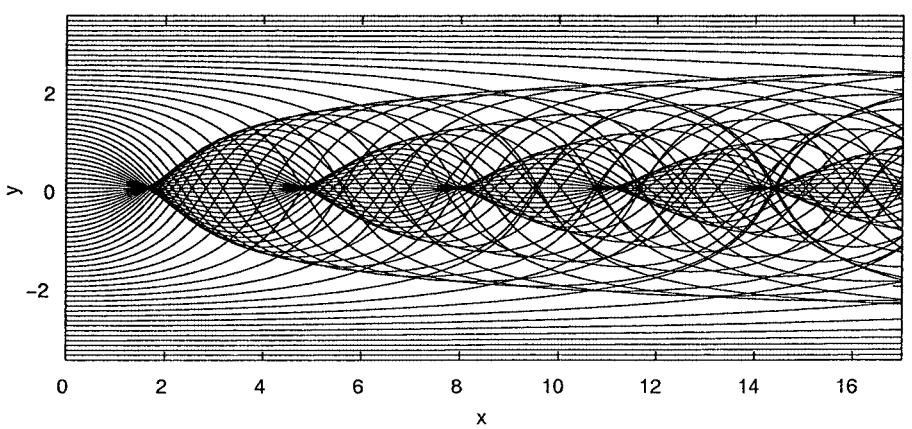

FIG. 11. Rays from initial plane wave for waveguide.

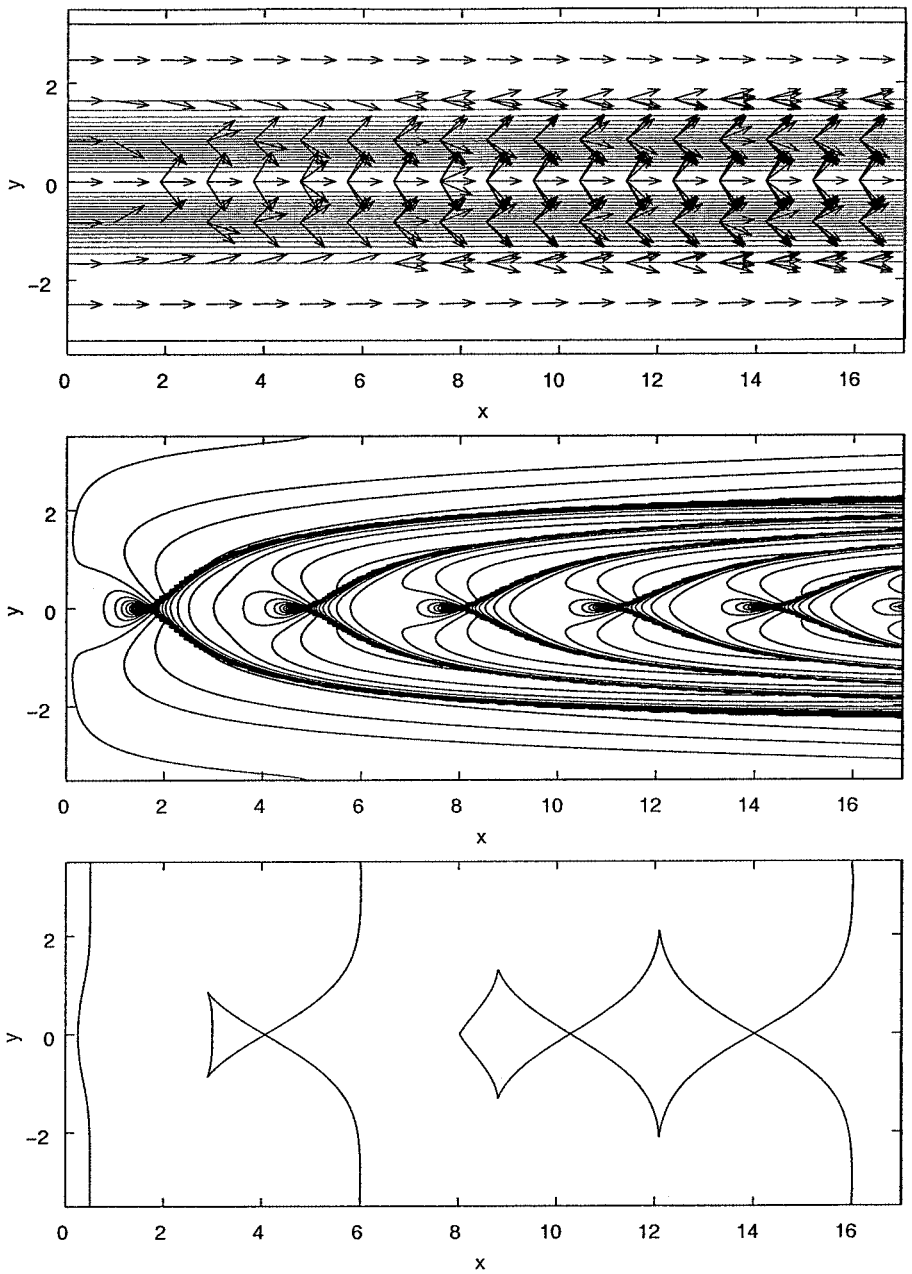

FIG. 12. Results for the waveguide simulation. (Top frame) Local ray directions with contour lines of index of refraction overlayed; (middle frame) amplitude, contour lines of $\min(A, 4)$; (bottom frame) wave fronts in the xy-plane at $T = 0.5, 6, 16$.

4.2. Contracting Ellipsoidal Wave Front

The example in Section 4.1 is here extended to a surface in $\mathbb{R}^3$. Even though general software for the three-dimensional segment projection method has not yet been developed this simulation can be done. The reason is that the index of refraction is constant and then only one segment is needed in each of the coordinate planes given in Fig. 7, which displays the projections of the initial surface in phase space onto the $x\theta_1\theta_2$, $y\theta_1\theta_2$, and $z\theta_1\theta_2$ spaces. In

Fig. 6 we see the evolution of the wave front in xyz space at different times. For the general case of variable index of refraction a larger number of segments could be required. It is in principle also possible in the segment projection method. This will not increase the overall computational complexity of the algorithm, but the software becomes more complicated (see [21] for a simple three-dimensional example with several overlapping segments). The grid resolution used in the computations was $\Delta\theta_1 = \Delta\theta_2 = 2\pi/60 \approx 0.1$.

4.3. Focusing from a Lens

In this example with variable index of refraction a wave front passes through a lens, modeled by the index of refraction

$$\eta(x, y) = 1.5 - \frac{1}{\pi}\arctan(5[(y-1)^2 - 0.1(x-0.5)]).$$

The left frame in Fig. 8 shows the direction of the rays and isocurves for the index of refraction. In the right frame, the initial planar wave front is given together with the wave fronts at later times. This includes the development of caustics.

Since all rays go in the positive x-direction in this simulation, we can use the reduced model of Section 3.1. The arrival time T is a well-defined quantity on the phase plane curve. It is computed via (26) and the y-segment functions $T^y(x, y)$ are used to plot the wave fronts in Fig. 8. Note, since we use a constant Δx and a uniform y grid, T is in fact obtained on a uniform $x \times y$ grid. At each point the number of y-segments corresponds to the number of crossing wave fronts. Figure 9 compares the results obtained with the segment projection method on a grid with $\Delta y = \Delta\theta = 0.01$ and a ray-traced reference solution, in order to give an indication of the numerical error. The segments Θ and Y are given in Fig. 10.

4.4. Waveguide

In this simulation an incoming plane wave at $x = 0$ with constant amplitude $A = 1$ enters a waveguide. The variable index of refraction in the waveguide, $\eta(x, y) = 1 + \exp(-y^2)$, causes the rays to bend. The ray pattern is given in Fig. 11. The corresponding local ray

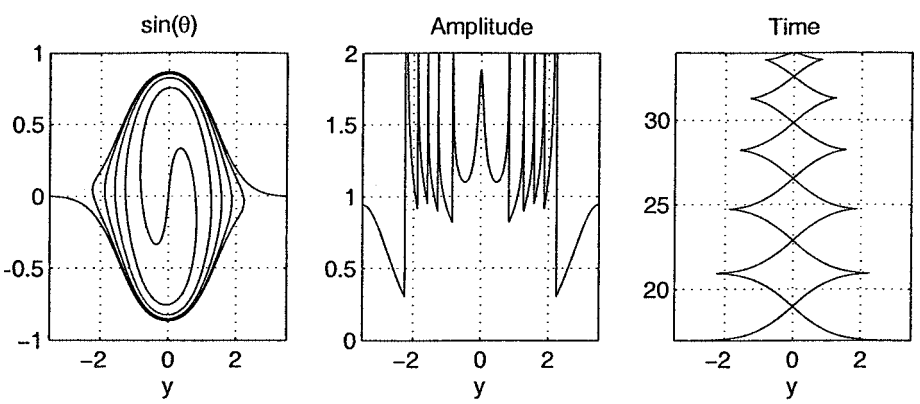

FIG. 13. Results for waveguide at $x = 17$ as a function of y. (Left frame) $\sin(\theta)$; (middle frame) amplitude A; (right frame) time T.

directions, amplitude, and wave fronts as computed by the segment projection method are shown in Fig. 12. The technique described in Section 3.1 of having the x-axis as evolution direction is also used in this application. The phase space curve in the $y\theta$-plane becomes complicated at larger x-values but it is still handled well by the segment projection method (see Fig. 13). The grid resolution was high, $\Delta y = 7/4096 \approx 0.002$, to get an accurate rendering of the amplitude at the far end of the waveguide.

5. CONCLUSION

We have presented a new PDE-based method for computing the geometrical optics approximation of the wave equation. The wavefront is tracked directly in phase space through an adapted version of the segment projection method. In this way, there is no extra difficulty when computing solutions with multiple crossing waves. Nor do caustics present a problem when computing amplitudes. The method is defined on an Eulerian grid and does not require regridding, due to diverging rays. The computational complexity is of the same low order as that of ray tracing and the algorithm explicitly defines the wave front.

We have applied the method to a number of examples and have shown how to use it for efficiently computing multivalued travel times as well as amplitudes. The examples cover problems with constant and variable index of refraction in two dimensions and a constant coefficient problem in three dimensions. The computations in the two-dimensional examples were done by codes based on the general segment projection method software. The three-dimensional example was done by a special purpose code.

Finally, we should remark that our method can also be used for more general Hamilton–Jacobi equations. Given a Hamiltonian $H(\mathbf{x}, \nabla\varphi)$, we can just replace the system (8), (9) by

$$\frac{d\mathbf{x}}{dt} = \nabla_p H(\mathbf{x}, \mathbf{p}),$$
$$\frac{d\mathbf{p}}{dt} = -\nabla_x H(\mathbf{x}, \mathbf{p}),$$

and the transport equation (10) by

$$\frac{\partial w}{\partial t} + \nabla_p H \cdot \nabla_x w - \nabla_x H \cdot \nabla_p w = 0.$$

Geometrical optics is the special case when $H(\mathbf{x}, \nabla\varphi) = c(\mathbf{x})|\nabla\varphi|$.

REFERENCES

1. J.-D. Benamou, Big ray tracing: Multivalued travel time field computation using viscosity solutions of the eikonal equation, *J. Comput. Phys.* **128**, 463 (1996).
2. J.-D. Benamou, Direct solution of multivalued phase space solutions for Hamilton–Jacobi equations, *Commun. Pure Appl. Math.* **52**(11), 1443 (1999).
3. Y. Brenier and L. Corrias, A kinetic formulation for multibranch entropy solutions of scalar conservation laws, *Ann. Inst. H. Poincaré Ana. Non Linéaire* **15**(2), 169 (1998).
4. M. Crandall and P. Lions, Viscosity solution of Hamilton-Jacobi equations, *Trans. Am. Math. Soc.* **227**, 1 (1983).

5. B. Engquist, E. Fatemi, and S. Osher, Numerical solution of the high frequency asymptotic expansion for the scalar wave equation, *J. Comput. Phys.* **120**, 145 (1995).
6. B. Engquist and O. Runborg, Multi-phase computations in geometrical optics, *J. Comput. Appl. Math.* **74**, 175 (1996).
7. L. Gosse and F. James, Convergence results for an inhomogeneous system arising in various high frequency approximations, *Numer. Math.* (2001), doi:10.1007/S002110100309.
8. L. Gosse and F. James, Numerical approximations of one-dimensional linear conservation equations with discontinuous coefficients, *Math. Comp.* **69**, 987 (2000).
9. J. Keller, Geometrical theory of diffraction, *J. Opt. Soc. Am.* **52** (1962).
10. S. Osher and J. A. Sethian, Fronts propagating with curvature dependent speed: Algorithms based on Hamilton–Jacobi formulations, *J. Comput. Phys.* **79**, 12 (1988).
11. O. Runborg, *Multiscale and Multiphase Methods for Wave Propagation*, Ph.D. thesis (Department of Numerical Analysis and Computer Science, Royal Institute of Technology (KTH), Stockholm, Sweden, 1998); ISBN 91-7170-318-7, TRITA-NA-9818.
12. O. Runborg, Some new results in multiphase geometrical optics, *M2AN Math. Model. Numer. Anal.* **34**, 1203 (2000).
13. S. J. Ruuth, B. Merriman, and S. Osher, A fixed grid method for capturing motion of self-intersecting wavefronts and related PDEs, *J. Comput. Phys.* **163**, 1 (2000).
14. J. A. Sethian, *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision and Materials Science.* (Cambridge University Press, Cambridge, UK, 1999).
15. J. Steinhoff, Y. Wenren, D. Underhill, and E. Puskas, Computation of short acoustic pulses, in *Proceedings, 6th International Symposium on CFD, Lake Tahoe NV, Sept. 1995.*
16. J. Steinhoff and M. Fan, *Eulerian Computation of Evolving Surfaces, Curves and Discontinuous Fields*, University of Tennessee Space Institute Report (University of Tennessee, Tullahoma, 1998).
17. 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 (2000).
18. W. Symes, A slowness matching finite difference method for traveltimes beyond transmission caustics, Preprint, Dept. of Computational and Applied Mathematics (Rice University, 1996).
19. A.-K. Tornberg, *Interface Tracking Methods with Applications to Multiphase Flows*, Ph.D. thesis (Department of Numerical Analysis and Computer Science, Royal Institute of Technology (KTH), Stockholm, Sweden, 2000); ISBN 91-7170-558-9, TRITA-NA 0010.
20. A.-K. Tornberg and B. Engquist, Interface tracking in two-phase flows, in *Multifield Problems, State of the Art* (Springer-Verlag, Berlin, 2000), p. 55.
21. A.-K. Tornberg and B. Engquist, The segment projection method for interface tracking, submitted for publication.
22. J. van Trier and W. W. Symes, Upwind finite-difference calculation of traveltimes, *Geophysics* **56**(6), 812 (1991).
23. J. Vidale, Finite-difference calculation of traveltimes, *B. Seismol. Soc. Am.* **78**(6), 2062 (1988).