A method for simulation of underwater acoustic wavefront propagation

Sheri L. Martinelli [a)b)]

Torpedo Systems Department,

Naval Undersea Warfare Center,

1176 Howell Street,

Newport,

Rhode Island 02841

(Dated: November 30, 2010)

[a)] Also at Division of Applied Mathematics, Brown University, Providence, Rhode Island 02912

Abstract

An algorithm for computing wavefronts, based on the high frequency approximation to the wave equation, is presented. This technique applies the level set method to acoustic wavefront propagation in the time domain. The level set method allows for computation of the acoustic phase function using established numerical techniques to solve a first order transport equation to a desired order of accuracy. Traditional methods for solving the eikonal equation directly on a fixed grid limit one to only the first arrivals, so these approaches are not useful when multi-path propagation is present. Applying the level set model to the problem allows for the time domain computation of the phase function on a fixed grid, without having to restrict to first arrival times. The implementation presented has no restrictions on range dependence or direction of travel, and offers improved efficiency over solving the full wave equation which under the high frequency assumption requires a large number of grid points to resolve the highly oscillatory solutions. Comparisons to analytical solutions are presented where available, and numerical results are validated by comparing results with exact solutions where available, a full wave equation solver, and with wavefronts extracted from ray tracing software.

PACS numbers: 43.30.Gv, 43.58.Ta, 43.20.Bi, 43.20.El

I. INTRODUCTION

In this work, a fixed-grid model is applied to computational high frequency acoustic propagation. The proposed method builds upon the general foundation established by Osher, Cheng, Kang, Shim, and Tsai[1] in which a basic level set method for geometric optics was introduced. High frequency propagation modelling is traditionally accomplished via ray tracing. Rather than solve for the acoustic pressure directly, the geometric optics approximation to the wave equation is employed to solve for a more slowly varying phase function and a separate amplitude function. Ray tracing solves the eikonal equation for the phase using the method of characteristics. When rays (characteristics) diverge, eventually they do not cover enough physical space, and well-resolved solutions are not available on any uniform grid.

Several computational approaches in addition to ray tracing already exist which can accurately solve the equations of acoustic propagation. However, these are not appropriate methods at high frequencies where required grid sizes become large enough to overwhelm computational resources. Ray tracing is therefore the current standard for high frequency or long range propagation modeling. The level set method may provide an alternative to ray tracing for solving the high frequency approximation to the wave equation that allows the simulation user greater control over the accuracy of the solutions in certain applications.

The difficulties with the Lagrangian approach are familiar from studies of long range propagation. The ray chaos problem was discussed in [2]. The term "ray chaos" generally refers to the phenomenon whereby small perturbations in the ray shooting angle result in large variations in the resulting trajectories. When chaotic rays are present, a high degree of precision is required to specify shooting angles in order to be able to locate eigenrays. In [3], Collins and Kuperman suggested an alternative method to compute eigenrays in the presence of ray chaos, i.e., the boundary value problem perspective vice an initial value problem (e.g., shooting method) for locating eigenrays, but their method relied on direct

[b)] sheri_martinelli@brown.edu

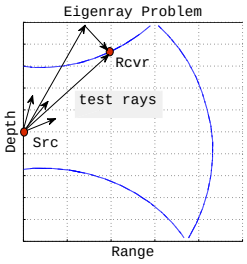

FIG. 1. Arrivals at a given point in space are determined by wavefront crossing rather than shooting test rays

path optimization and did not allow for bottom or surface reflections. Godin[4] examined the behavior of rays versus that of wavefronts under weak sound speed fluctuations (but not in the presence of ray chaos) and showed that wavefronts are much more stable than rays, due to the fact that the significant ray perturbations tend to occur along the wavefronts rather than across them. Another related idea is exact ray theory[5], which requires a previously computed solution to the Helmholtz equation so is limited to use in aiding the understanding of solutions that have already been computed.

The level sets approach is a wavefront-based model that effectively traces all rays simultaneously, though in a higher dimensional space. By solving on a fixed grid in the full space and evolving entire wavefronts in time, the eigenray problem is eliminated. Thus in complex domains, e.g., shallow water, even though the level set problem is posed in high dimensional phase space, improvement in computational speed may be observed because it would no longer be necessary to use a large number of test rays to determine a solution at a given point in space; this idea is illustrated in Figure 1.

Level set methods are generic, computational techniques introduced by Osher and Sethian[6] for tracking the evolution of moving curves and surfaces. The advantage of this approach is that standard partial differential equation (PDE) solvers (e.g., finite difference or finite volume approximations) can be employed to solve the problem on uniform grids which may then be refined to reduce the global error. Level set methods achieve this by representing the propagating surface implicitly as the zero level set of a function in a higher

dimensional space. This zero level set is then transported via the underlying velocity field. In the case of acoustic propagation in isotropic media, the propagation direction is normal to the propagating surface (wavefront). The extension to phase space allows for the computation of multi-valued solutions in the physical space. Multiple arrivals are handled naturally by ray tracing, but become an issue when solving on a fixed grid. In physical space, the multiply-valued solutions violate well-posedness of the problem. The bicharacteristic curves, an extension of the characteristics to the phase space, are single-valued however and hence working in the phase space, one can capture multipath arrivals.

Eulerian (fixed grid) geometric optics has been an active research area in the scientific computing community for quite some time. Benamou[7] provides an overview of approaches to this problem. The most similar to the level set method is the segment projection method[8] in which wavefronts are tracked in phase space as projections onto each two dimensional subspace of the three dimensional phase space. This method is effective, but requires complicated bookkeeping in order to reconstruct wavefronts. The approach of Osher, et al. propagates the entire wavefront in the phase space where the bicharacteristics of the eikonal equation are well-behaved. In [9], Qian, et al. build upon [1] by extending the method to propagation in anisotropic materials. In [10], a method for incorporating reflecting boundaries is introduced. Qian and Leung[11,12] developed a level set method for the paraxial approximation. This approximation is commonly applied to ray tracing implementations as it reduces the number of independent equations to solve by propagating in a direction that increases monotonically with time (e.g., range). The approach of Qian and Leung reduces the dimensionality of the problem, but introduces an additional equation to be solved in the phase space to compute arrival times. This method addresses long range propagation problems, but the present work is primarily concerned with reflections and scattering back toward the source, which are precluded by the paraxial assumption.

The purpose of this work is to demonstrate an application of these foundations to the specific problem of high frequency acoustics. Section II reviews background material and provides an overview of the method. Section III offers a description of the implementation,

presenting the necessary components to implement a level set method for underwater acoustics, and in Section IV, some preliminary results demonstrating the algorithm's performance in a few sample cases including varying sound speed profiles and reflecting boundaries are presented. Results are summarized and section V concludes.

II. BACKGROUND

A. High frequency acoustics and the eikonal equation for the phase

The high frequency wave equation results from application of a classical asymptotic approximation to the standard wave equation. Following [13], start with the d-dimensional linear wave equation for the acoustic pressure in a medium with constant density and variable sound speed function given by $c(\mathbf{x})$:

$$p_{tt} - c(\mathbf{x})^2 \triangle p = 0, \tag{1}$$

where $(t, \mathbf{x}) \in \mathbb{R}^+ \times \Omega, \Omega \subset \mathbb{R}^d$, and it is assumed that appropriate initial and boundary conditions are available. At very high frequencies, the solutions to this equation exhibit rapid oscillations. In order to be able to reasonably compute these solutions, apply the geometric optics approximation by assuming a solution of the form

$$p(t, \mathbf{x}) = e^{i\omega S(t,\mathbf{x})} \sum_{k=0}^{\infty} A_k(t, \mathbf{x})(i\omega)^{-k}. \tag{2}$$

Upon substitution into the wave equation (1), this expression yields the eikonal equation for the phase function S from the highest order terms in ω

$$S(t, \mathbf{x}) \pm c|\nabla S(t, \mathbf{x})| = 0 \tag{3}$$

and a transport type equation for the first amplitude term

$$(A_0)_t + c(\mathbf{x}) \frac{\nabla S \cdot \nabla A_0}{|\nabla S|} + \frac{c(\mathbf{x})^2 \triangle S - S_{tt}}{2c(\mathbf{x})|\nabla S|} A_0 = 0. \tag{4}$$

Similar transport equations can also be derived for the remaining amplitude terms, however for $\omega >> 1$, only the two leading terms in the expansion are significant (this is the geometric optics approximation).

The weakly coupled system consisting of (3) along with (4) form the equations of high frequency acoustics.

B. Level set method for the eikonal equation

The eikonal equation is a Hamilton-Jacobi type equation with Hamiltonian given by $H(\mathbf{x}, \mathbf{k}) = c(\mathbf{x})\,|\mathbf{k}|$. The vector-valued variable $\mathbf{k}$ is associated with ∇S. First order PDE of this type can be solved locally via the method of characteristics by expressing (3), where, without loss of generality, only the equation with the plus sign (propagation outward from the initial wavefront) is considered, in terms of the phase space variables $\mathbf{x}$ and $\mathbf{k}$. The Hamilton-Jacobi formulation is in the full phase space, where the variable $\mathbf{k}$ is the generalized momentum. The characteristic equations are given by

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \nabla_k H(\mathbf{x}, \mathbf{k}) = c(\mathbf{x}(t)) \frac{\mathbf{k}(t)}{|\mathbf{k}(t)|} \\ \dot{\mathbf{k}}(t) &= -\nabla_x H(\mathbf{x}, \mathbf{k}) = -|\mathbf{k}(t)| \nabla_x c(\mathbf{x}(t)) \end{aligned} \tag{5}$$

with given consistent initial conditions $\mathbf{x}(0) = \mathbf{x}_0, \mathbf{k}(0) = \mathbf{k}_0 = \nabla_x S(0, \mathbf{x}_0)$. Considered as a pair, the bicharacteristic curves $(\mathbf{x}(t), \mathbf{k}(t))$ occupy the two-dimensional phase space, and differentiating with respect to time shows that $H(\mathbf{x}(t), \mathbf{k}(t))$ is constant along them. Taking $H \equiv 1$, the condition

$$|\mathbf{k}| = |\nabla S| = \frac{1}{c(\mathbf{x})} \tag{6}$$

is recovered.

The ray tracing approach involves computing $\mathbf{x}(t)$ from (5), typically using arclength parameterization rather than unscaled time. The difficulty with solving (3) on a fixed grid is that, in general, rays may cross, generating multi-valued solutions for the phase function S at some points. Thus standard PDE solvers fail in this case as convergence requires uniqueness of the solution. One of the earlier methods for dealing with this situation was to compute the viscosity solution[14] which forces uniqueness by computing only the first arrival time. However, in applications, multi-valued arrival times are often desired. The level set

method is able to handle multi-valued arrival times by working in the phase space, where the bicharacteristics do not suffer from this problem[1].

The difference between ray tracing and the level set method is in the representation of the solution. With ray tracing, the ordinary differential equations (5) are solved along the ray from a starting point, x_0, on the initial wavefront. Often these equations are parameterized with respect to arclength in cylindrical coordinates in the presence of azimuthal symmetry, (r, z), and the initial conditions are specified by a take-off angle. The level set method is based rather on an implicit representation of the wavefront, that is, the wavefront is not expressed explicitly as a function in the physical space, but is instead embedded as the zero level set of a function that is defined in a higher dimensional space (the phase space). Thus, a function that may be multi-valued or otherwise poorly behaved in the physical space is represented by a smoothly varying and well-defined quantity in the phase space. In fact, given the restriction (6), it is not necessary to utilize the full phase space in order to find S, since the magnitude $|\mathbf{k}|$ is fixed, the dimension can be reduced by one. For an example with two dimensional physical space, let

$$\mathbf{k} = \begin{pmatrix} |\mathbf{k}| \cos\theta \\ |\mathbf{k}| \sin\theta \end{pmatrix}, \tag{7}$$

then only θ need be considered an independent variable. In this representation, θ represents the propagation direction of the wavefront, and $\mathbf{k} = \left(\frac{\cos(\theta)}{c}, \frac{\sin(\theta)}{c}\right)$. So for two-dimensional physical space, the reduced phase space has three dimensions, and for fully three-dimensional acoustics, the reduced phase space would be five-dimensional. When computational efficiency is a priority, it is desirable to work in the reduced phase space to limit the number of grid points needed to meet a specified error tolerance.

In two-dimensional propagation, since the wavefront is a curve in three-dimensional reduced phase space, define a vector-valued function

$$\mathbf{\Phi} = \begin{pmatrix} \phi_1 \\ \phi_2 \end{pmatrix}, \tag{8}$$

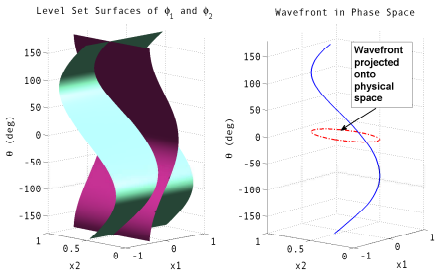

FIG. 2. Implicit representation of 2D wavefront in the phase space (Color online)

referred to as the "level set function"[1] in the phase space, with two components, $\phi_1(t, \mathbf{x}, \mathbf{k})$ and $\phi_2(t, \mathbf{x}, \mathbf{k})$, such that the initial wavefront is embedded in the intersection of the zero level sets of ϕ_1 and ϕ_2, i.e., the projection onto the physical space of the set $\{(\mathbf{x}, \mathbf{k}) | \phi_1(0, \mathbf{x}, \mathbf{k}) = \phi_2(0, \mathbf{x}, \mathbf{k}) = 0\}$. For full three dimensional propagation, the reduced phase space has five dimensions and the wavefront is a surface, so the level set function would have three components with the wavefront corresponding to the intersection of the zero level sets of all three component functions[1]. Figure 2 offers an example in two dimensions of the implicit wavefront representation as the intersection of level set surfaces for two level set functions.

To evolve the level set functions, note that the wavefront propagates in the direction of its normal, which is the local ray direction, given by $\frac{\mathbf{k}}{|\mathbf{k}|}$. This leads to the condition

$$\tilde{\mathbf{V}} \cdot \nabla \mathbf{\Phi} = 0, \tag{9}$$

where $\tilde{\mathbf{V}}$ is the direction of the characteristics, given in the most general formulation as

$$\tilde{\mathbf{V}} = \begin{pmatrix} \nabla_k H(\mathbf{x}, \mathbf{k}) \\ -\nabla_x H(\mathbf{x}, \mathbf{k}) \end{pmatrix}. \tag{10}$$

In the reduced phase space for the two dimensional problem, the level set functions thus satisfy the following transport equation

$$\frac{\partial \phi_i}{\partial t} + \mathbf{V} \cdot \nabla \phi_i = 0, i = 1, 2 \tag{11}$$

with the velocity field $\mathbf{V}$ given as

$$\mathbf{V} = \begin{pmatrix} c(\mathbf{x})\cos\theta \\ c(\mathbf{x})\sin\theta \\ \frac{\partial c}{\partial x_1}\sin\theta - \frac{\partial c}{\partial x_2}\cos\theta \end{pmatrix} \tag{12}$$

where $\mathbf{x} = (x_1, x_2)$ and θ is the direction of propagation in the $x_1 - x_2$ coordinate plane. This velocity field is derived directly from the ray equations (5). The functions ϕ_1 and ϕ_2 are defined at $t = 0$ so that the known initial wavefront is embedded as the intersection of the zero level surfaces of the two functions. It is convenient if the source can be parameterized in θ so that it can be described as

$$x_1 = f_1(\theta)$$
$$x_2 = f_2(\theta).$$

Then ϕ_1 and ϕ_2 can be initialized as

$$\phi_1(0, x_1, x_2, \theta) = x_1 - f_1(\theta)$$
$$\phi_2(0, x_1, x_2, \theta) = x_2 - f_2(\theta). \tag{13}$$

For instance, if the wavefront is a circle in the x_1x_2 plane with radius α centered at some point (x_1^0, x_2^0),

$$\phi_1(0, x_1, x_2, \theta) = x_1 - x_1^0 - \alpha\cos\theta$$
$$\phi_2(0, x_1, x_2, \theta) = x_2 - x_2^0 - \alpha\sin\theta$$

defines such a choice. Letting $\alpha \to 0$ gives an appropriate initial condition for a point source at (x_1^0, x_2^0). In this case, the two level set surfaces are orthogonal, an important property since recovering the wavefront involves seeking the intersection of the level set surfaces. Equation (11) can be evolved in time using appropriate numerical techniques and the wavefront, $W(x_1, x_2; t)$ may be recovered at any time t as

$$W(x_1, x_2; t)$$
$$= \{(x_1, x_2) | \phi_1(t, x_1, x_2, \theta) = \phi_2(t, x_1, x_2, \theta) = 0\}. \tag{14}$$

A good introduction to the level set method for geometric optics can be found in [1].

III. IMPLEMENTATION

A. Solving the level set equations for underwater acoustics

The implementation discussed in this work is for computing the phase function in two-dimensional propagation. Equation (11) is solved in Cartesian coordinates with a uniform line source, parallel to the y-axis as in Figure 3. Thus, the wave equation is reduced by symmetry to two dimensions given by (x, z), where $z = 0$ at the surface, and increases with increasing depth. If azimuthal symmetry is present in the domain and sound speed profile, a range-depth solution can be obtained by restricting to the right half of the $x - z$ plane. Take the source to be located at $(0, z_s)$, then the level set functions can be initialized by setting

$$\begin{aligned} \phi_1(0, x, z, \theta) &= x \\ \phi_2(0, x, z, \theta) &= z - z_s. \end{aligned} \tag{15}$$

Solving the level set equations equates to solving a first order transport equation in the reduced phase space, (x, z, θ):

$$\begin{aligned} &f_t + c(x, z)\cos(\theta) f_x + c(x, z)\sin(\theta) f_z \\ &\quad + (c_x \sin(\theta) - c_z \cos(\theta)) f_\theta = 0 \end{aligned} \tag{16}$$

The transport equation is a conservation law with variable coefficients which can be solved using upwind finite differences as described in [15]. The complications arise from the boundary conditions. In this work, two types of boundary conditions have been applied: pure reflection and absorbing. The absorbing condition is handled naturally; the wavefront simply flows out of the domain. However, recall that the level set equations are solved in

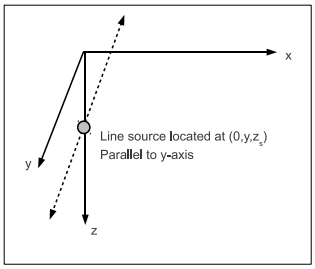

FIG. 3. Sample problem geometry: a uniform line source parallel to the y-axis

the full domain and for all $-\pi \leq \theta < \pi$, hence an inflow condition must be imposed at the boundary. One way to do this is to modify the differencing at inflow to use only data within the truncated domain. Another option is to impose a zero flow Neumann condition,

$$\frac{df}{dx}\Big|_{x_0} = 0. \tag{17}$$

Any such domain truncation will distort the solution near these boundaries, and the effect will worsen over time. In the level set method, one is only interested in what happens at the wavefront so distortion of the level set functions at inflow is not an issue since the wavefront is sufficiently removed from the inflow grid points. It is important to keep in mind though that for very long time integration or for very sparse grids, the level set functions may need to be re-initialized periodically to prevent distortion of data close to the wavefront. For a discussion of reinitialization, see [1]. Ultimately, for computational efficiency, a local level set method[16] should be used. In this method, the fact that the values of the level set functions far from the wavefront are not of interest is exploited in order to reduce the required amount of computational work by solving the level set equations only within a band of grid points containing the wavefront.

The reflection boundary condition poses a slight complication. To impose a reflection

boundary condition at, say, the surface $\{z = 0\}$, set[10]

$$\phi_1(t, x, 0, \theta_{refl}) = \phi_1(t, x, 0, \theta_{inc})$$
$$\phi_2(t, x, 0, \theta_{refl}) = \phi_2(t, x, 0, \theta_{inc})$$
$$\theta_{refl} = \pi + \theta_{inc} \tag{18}$$

in accordance with Snell's Law. Here, θ_{inc} and θ_{refl} are, respectively, the angles incident upon and reflected from the surface. For a general boundary, the condition would be

$$\theta_{refl} = 2\theta_B - \theta_{inc} - \pi, \tag{19}$$

where θ_B is the angle of the outward normal to the reflecting surface. This construction, under the velocity field given by (12) results in a discontinuity at the boundary of size $2c_0t$ in the case of a constant sound speed $c(x, z) = c_0$ at vertical incidence. This can be seen by recognizing that under a constant wave speed, (11) has the solutions

$$\phi_1(t, x, z, \theta) = x - c_0 t \cos\theta$$
$$\phi_2(t, x, z, \theta) = z - z_s - c_0 t \sin\theta.$$

Of course, vertical incidence is a worst-case scenario; at less severe angles, the vertical component of the velocity field, $c_0 \sin\theta$, is reduced in magnitude. However, the discontinuity will worsen in severity for higher sound speeds and for longer periods of time integration. Such discontinuities are known to produce spurious oscillations when standard finite differencing methods are used. Actually, the simplest upwind method - forward Euler time integration with first order upwind spatial differencing, is highly dissipative - so if the source is placed sufficiently far from the reflecting boundary, the oscillations might not be observed, but when a source is located near the boundary, the oscillations are not sufficiently suppressed so a different approach is needed. To mitigate this issue, a Weighted Essentially Non-Oscillatory (WENO)[17] interpolation based method is employed to obtain the spatial derivatives, and this is coupled with Total Variation Diminishing Runge-Kutta (TVDRK)[18] time integration. The first order TVDRK is equivalent to Forward Euler. These are Runge-Kutta methods

designed to ensure that oscillations in the numerical approximation diminish over time. It is also necessary to be careful about interpolating in θ near the boundary for this reason. For this purpose, coefficients for WENO interpolation at an arbitrary point were derived and are presented in the Appendix.

WENO methods are high order methods; that is, for a method with order of accuracy ν, the numerical solution on a grid with size Δx spacing converges to the true solution like Δx^ν as the grid is refined. While higher order methods generally involve more computational work per gridpoint, there is a trade-off in that high order solvers require fewer grid points to meet a given error tolerance. To compute the time derivatives, first, second, and third order TVDRK methods are used. The WENO solvers that have been implemented for this work have orders $\nu = 1, 3$, or 5. The subsequent examples were produced using a combination of 5^{th} order WENO and 3^{rd} order TVDRK. WENO methods are discussed further in the Appendix; for further details on TVDRK methods, the reader should refer to [18].

The other matter with respect to boundary handling is the fact that the boundary location might not be in the grid. One could use a non-uniform grid, but this affects the accuracy of the underlying WENO method. Instead, experimentation has shown that approximating the location of the boundary by the nearest grid points yields convergence. Two examples for which the boundary does not conform to the grid are presented in Section IV.

B. Additional processing

Solving the level set equations in the phase space is the first step. Additional processing is required to extract the arrival data. In order to obtain the wavefronts, it is necessary to first locate the zero level sets of ϕ_1 and ϕ_2, and then locate the intersection. Standard graphics routines for surface reconstruction are typically based on the marching cubes algorithm[19]. Existing generic software can be used to compute the zero level set surfaces, but more specialized routines are required to locate the intersection. For example, a typical isosurface

extraction routine returns a triangulated surface consisting of arrays of triangle faces and vertices representing the zero isosurfaces for each function. To extract the wavefronts depicted in Section IV, a basic bounding box test was applied to identify pairs of intersecting triangles, then the line segment along which the intersection lies was reconstructed. In this way, plots are produced from the individual line segments.

IV. RESULTS

In this section, some computational results are presented to show convergence of the wavefronts computed using the level set method, and compare the solutions to those of the full wave equation.

A. Comparison to exact solutions

The examples in this section are test cases for which analytical solutions are available to test for convergence. Results for two sound speed profiles are presented in the absence of reflections from surface or bottom. The results are based on implementation of a fifth order WENO scheme coupled with third order TVD Runge-Kutta time integration on an $N \times N \times N$ grid. Comparison is made with the exact solutions after 0.1 seconds and errors are reported in the max norm, evaluated in a fixed neighborhood of the wavefront (and for the linear profile example, away from vertical angles where the exact solution is poorly behaved). The theoretical convergence rate is not observed immediately, but even for comparatively sparse grids, observed convergence is faster than first order.

1. Isovelocity profile

This example uses $c = 1.0$ km/s, independent of location in space, and a point source at $z = z_s = 0.5$ km with $x \in [-1, 1]$ and $z \in [0, 1]$. In this case the characteristics are straight lines so $\theta(t) = \theta_0$ and it is easy to see that the solutions to the level set equations are given as $\phi_1(t, x, z, \theta) = x - t\cos(\theta)$, and $\phi_2(t, x, z, \theta) = z - z_s - t\sin(\theta)$. The results, evaluated at

$t = 0.1$ s and without reflection, are presented in Table I.

Initially, the solution appears to be converging like $\frac{1}{N^2}$. The expected fifth order convergence is observed at $N = 160$, then the error decreases even more rapidly as N is increased to 320.

2. Linear profile

An exact solution to the level set equations is available for the case of a profile linear in depth: $c(z) = \alpha z + \beta$, where α and β are constants. For this example, $\alpha = 0.5$ and $\beta = 1.0$ are chosen, where units are again in km/s. The initial condition is a point source at $z = z_s = 0.5$ km with $x \in [-1, 1]$ and $z \in [0, 1]$. The exact solution for this case is given (away from $\theta = \frac{\pi}{2}$) by

$$\phi_1(t, x, z, \theta) = x + \hat{\gamma}\left(z + \frac{\beta}{\alpha}\right)\frac{1 - e^{2\alpha t}}{1 + \hat{\gamma}^2 e^{2\alpha t}} \tag{20}$$

$$\phi_2(t, x, z, \theta) = \left(z + \frac{\beta}{\alpha}\right)\frac{(1 + \hat{\gamma}^2)\, e^{\alpha t}}{1 + \hat{\gamma}^2 e^{2\alpha t}} - \frac{\beta}{\alpha} - z_s \tag{21}$$

where

$$\hat{\gamma} = \tan\left(\frac{\theta}{2} + \frac{\pi}{4}\right). \tag{22}$$

This solution can be derived by applying the method of characteristics to Equation 16 with $c(x, z) = \alpha z + \beta$ for prescribed constants α and β. The results of this investigation are presented in Table II.

The observed convergence rate is somewhat slower in this case than for the isovelocity profile, but still fifth order convergence is observed at $N = 160$ and rapidly begins to approach the true solution for larger N. The slower convergence in this case is likely due to the fact that the isovelocity problem is actually a two-dimensional problem in the phase space, since the direction of travel, θ, is constant and hence the θ component of the velocity field is zero. When a linearly varying sound speed is introduced, the problem is fully three-dimensional.

B. Examples involving reflection

This section presents computational results for profiles with reflecting boundaries, all computed with a 50 x 50 x 50 grid. The format for presenting the results in Figures 4 to 6 is time snapshots of the wavefront superimposed on contours from a full wave equation result for the same profile and ocean geometry. Including the amplitude contours from a finite frequency full wave solver serves as a check on the accuracy of the solutions. The profiles used were selected for theoretical illustration and are not realistic ocean profiles, which would require smooth interpolation from data.

1. Linear in depth and range profile with flat, reflecting bottom

Figure 4 applies a profile linear in both x and z: $c(x, z) = 0.5x + 0.5z + c_0$ to the same scenario. This example serves to illustrate that sound speed range dependence is a natural feature of this algorithm. The full wave equation solver is a finite frequency solution so it does not produce wavefronts exactly, but the wavefront is located where the contours appear to be converging together, corresponding to a rapid change in amplitude. The thick, black line representing the wavefront computed at each specified time step is seen to lie right on the leading edge of the converging contours, as expected, even after multiple reflections.

2. Isovelocity profile with sloping bottom

The next two examples are included to illustrate the handling of a bottom geometry that is dependent on the x variable. The simplest such type of domain incorporates a sloping bottom. Figure 5 shows the results with an upslope of 9° from horizontal, while Figure 6 was produced with a downslope of 9° from horizontal. The computational challenge presented by this case is the representation of a boundary that does not conform to the Cartesian grid employed to extract the desired convergence rate from the WENO method which is heavily dependent on uniform grid spacing, or equivalently, a smooth mapping to a uniform grid. Through experimentation, it was discovered that greater stability was achieved by

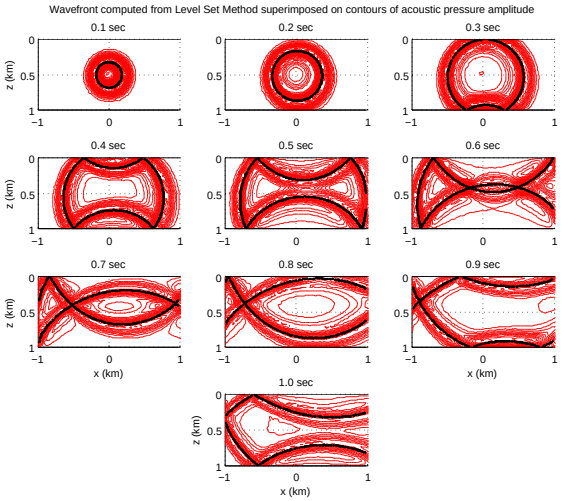

FIG. 4. Time snapshots of wavefronts (thick, black curve) computed using level sets superimposed on full field amplitude contours: $c(x, z) = 0.5x + 0.5z + c_0$ km/s (Color online)

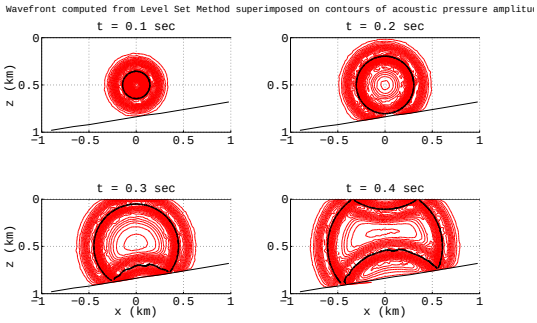

FIG. 5. Time snapshots of wavefronts (thick, black curve) computed using level sets: 9° upslope (Color online)

implementing the boundary condition at the nearest grid point to the physical boundary location, rather than modifying the grid to include the boundary. Although this results in some irregularity in the extracted wavefronts post-reflection, the figures indicate a very nice match with the apparent true wavefront location. The boundary effects have been shown to be reduced as the grid is resolved. Both examples use the sound speed $c = 1.5$ km/s.

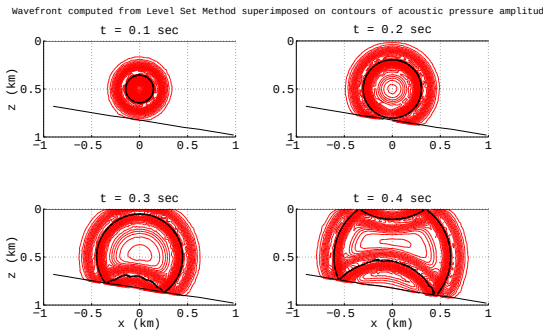

FIG. 6. Time snapshots of wavefronts (thick, black curve) computed using level sets: 9° downslope (Color online)

C. Examples with variable sound speed profiles

1. Propagation in a channel

In this example, a profile that simulates a waveguide presents an interesting comparison between the ray trace (Figure 8) and wavefront models. The model is an acoustic analog of the quantum harmonic oscillator, discussed in more detail by Foreman[5]; this profile yields a normal mode solution that is tractable at low frequencies, but the behavior becomes more complicated in the high frequency limit. A source is placed at $z_s = 500$ m in a field with $c(z) = \frac{c_0}{\sqrt{1-\left(1-\frac{z}{z_s}\right)^2}}$, with a sound speed of $c_0 = 1500$ m/s (Figure 7) at the source depth. This scenario represents a waveguide, symmetric about the sound channel axis at $z = z_s$, as $c(z)$ approaches infinity at the channel boundaries $z = 0$ km and $z = 1$ km. The ray trace in Figure 8 shows the rays are oscillating functions about the sound channel axis. In the wavefront model, this corresponds to a periodically self-crossing wavefront. Figure 9 shows snapshots of the wavefront computed using the level set technique with wavefronts extracted from a ray trace superimposed for validation. The match is very good away from the source. The errors are greater closer to the source, but this can be resolved by using a finer grid in the level set implementation.

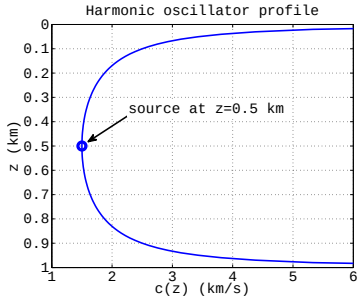

FIG. 7. Sound speed as a function of depth for harmonic oscillator profile

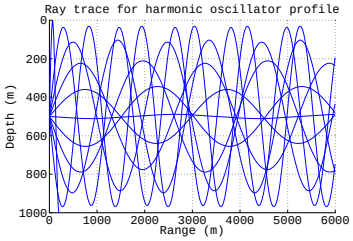

FIG. 8. Ray trace for the harmonic oscillator profile

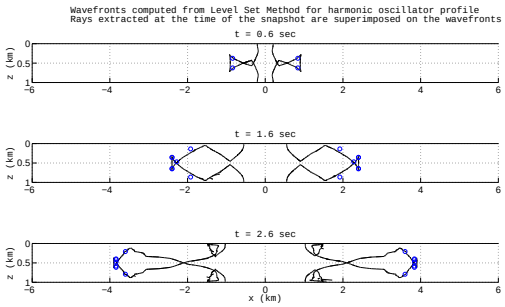

FIG. 9. A subset of the time snapshots of wavefronts computed using level sets: $c(z) = \frac{c_0}{\sqrt{1-\left(1-\frac{z}{z_s}\right)^2}}$, with wavefronts extracted from an analogous ray trace appearing as circles superimposed on the level sets wavefronts (Color online)

D. Examples with scattering

1. Scattering off bottom features

The following examples explore range dependence a little further by introducing rectangular (grid-conforming) features in the domain. In order to avoid overcomplicating the scenarios, the sound speed profile was held constant at $c = 1.5$ km/s. In Figure 10, a rectangular scatterer is placed on the "ocean" bottom. As would be expected from a pure geometric optics approximation, the diffraction at the corner is not represented by the level set method. Away from the corner, the results agree nicely with the full wave equation model, however a spurious zero level set appears in the phase space when the wavefront reflects off of the object. This can be understood more easily when a line source is considered rather than the point source. Recall that the wavefront is represented in the model as the intersection of zero level sets of two functions. Thus the function is positive on one side of the front, say the leading part and negative on the other (lagging). When part of the wavefront reflects off the object, the positive values reflect back in the opposite direction in the phase space, but the part that does not encounter the object retains the same sign in that direction, creating a discontinuity. Because there is a change of sign over this discontinuity, a zero level set exists as an artifact. One could apply edge detection to remove this artifact, but that is not terribly reliable as an approach because it is difficult to distinguish numerically between a discontinuity and a steep gradient in a smooth function. Instead, since this is a non-physical effect, a proper computation of the amplitude, when incorporated, should assign negligible weight to those locations along the computed wavefront.

2. Scattering off an obstacle

This final example, in Figure 11, is included to demonstrate a potentially useful application for an acoustic level set method. The rectangular geometry is kept for the sake of conforming to the grid, but this time the scatterer is an object in a field, with absorbing

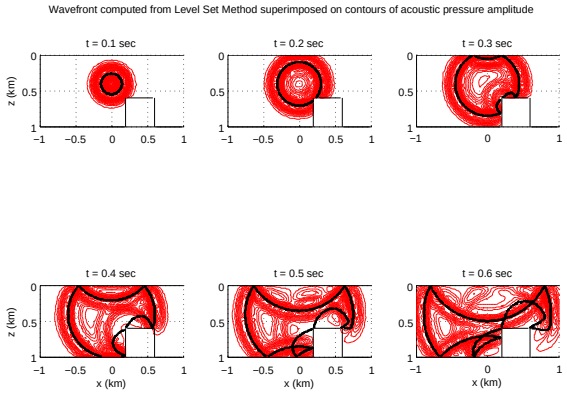

FIG. 10. Time snapshots of wavefronts (thick, black curve) computed using level sets in the presence of a rectangular bottom scatterer (Color online)

boundaries, which could represent an object in the deep ocean. The point source is located at $z_s = 500$m and the sound speed is again $c = 1.5$ km/s. This example also suffers from the same effects as the above examples (i.e., the spurious zero level sets upon reflection). However this example really demonstrates how this method changes the point of view for a scattering problem from the ray-based situation to having the ability to fully capture shape information from acoustic reflection off an object. A ray-based model involves the specification of locations on the object, dependent upon its orientation if it's not spherically symmetric, to which to trace eigenrays in order to simulate a response. Using a wavefront model, one can fully utilize the object's shape information and avoid having to compute eigenrays.

V. CONCLUSION

This work has introduced a method that applies a level set formulation to the high frequency acoustics problem; the model was discussed in section II, its implementation described in section III, and some basic examples provided in section IV to validate the model and illustrate its capabilities. Level set methods provide an alternative framework for numerical solutions to the high frequency approximation to the wave equation. By

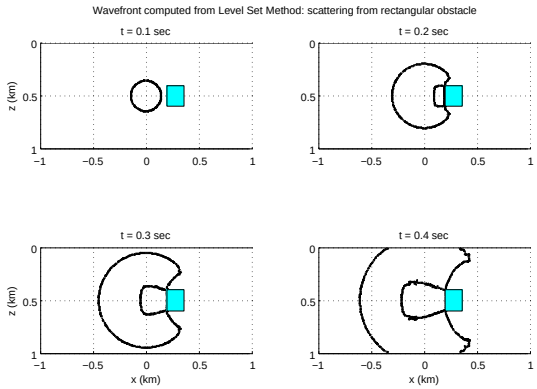

FIG. 11. Time snapshots of wavefronts computed using level sets in the presence of a scattering obstacle in the field

transforming to fixed-grid point of view, it may be more efficient to utilize this approach in certain applications where computational efficiency is a priority and ray traces fail to cover the physical domain adequately (e.g., range-dependent shallow water environments). There is a trade-off in that solving equations in the higher dimensional phase space for general problems requires much more in terms of computational resources than ray tracing. Whether the level set method can offer a viable alternative to ray tracing will be dependent on the complexity of the domain geometry and whether there are multiple sources and receivers present. There are several open issues left to future work. Certainly the computation of amplitude needs to be addressed. A fully three-dimensional propagation implementation would be useful and interesting. It would also be interesting to consider incorporating higher order effects (e.g., diffraction) into the model. Since the level set method and ray tracing are both based on the geometric optics approximation, the level set method also leads to unbounded amplitudes at caustics, so it would be worth investigating approaches to computing physically realistic amplitude values in such situations.

Acknowledgments

The author would like to thank Dr. Drumheller (ONR 333) and Dr. Soukup (NRL and ONR 333) for continued support of this program. This work is also a result of initial funding by the In-house Laboratory Independent Research (ILIR) program at Naval Undersea Warfare Center (NUWC), Newport, RI, and is partially funded by the Science, Mathematics, And Research for Transformation (SMART) program. I thank Dr. Jan Hesthaven of Brown University for his technical oversight and advice, Dr. Kevin Bongiovanni of Raytheon/BBN for suggesting this topic, and Dr. Andrew Fredricks also of NUWC for helpful guidance. Zhu Xueyu of Brown Univerity contributed the Discontinuous Galerkin - Perfectly Matched Layers (DG-PML) full wave equation code to verify solutions. Ray trace results were obtained from the code RAY produced at Woods Hole Oceanographic Institute[20]. Finally, I would like to acknowledge the helpful comments and support of the acoustics community in general.

APPENDIX: ESSENTIALLY NON-OSCILLATORY (ENO) AND WEIGHTED ENO (WENO) INTERPOLATION

Finite difference approximation to a function's derivative is based on the use of an interpolating polynomial over a subset of the grid (stencil) containing the point at which the derivative is to be evaluated. The derivative of the polynomial is used to approximate the derivative of the underlying function. For smooth functions, the number of points in the stencil determines the order of accuracy of the approximation, e.g., a stencil with $k+1$ points can produce an approximation that converges as $O(\Delta x^k)$. However when a function develops discontinuities, one has to be careful in choosing the appropriate interpolation. In ENO interpolation, introduced by Harten et al.[21] and later refined by Shu and Osher[22,23], one seeks to choose an a appropriate stencil on which to approximate the function by Lagrange polynomials. The idea is to isolate points of discontinuity, using a measure of the function's variation. That is, the stencil is adaptive, versus using a fixed stencil as in a standard

interpolation routine. WENO interpolation, extends ENO by combining the results from all candidate stencils to achieve $O(\Delta x^{2k-1})$ accuracy, where k is the accuracy order of the underlying ENO method. Only the one-dimensional problem is discussed here, but the extension to multiple dimensions for implementation of a level set method is straightforward. In particular, on the Cartesian grid, the same operations are performed in each dimension.

1. ENO finite difference reconstruction

As in Shu's comprehensive report[17], for polynomial reconstruction in one dimension, establish a uniform grid $\{x_{i-\frac{1}{2}}\}_{i=0}^{N}$, with $\Delta x = x_{\frac{1}{2}} - x_{-\frac{1}{2}}$. That the grid is uniform is critical for the finite difference formulation, but this requirement can be relaxed if a finite volume scheme is employed instead, or if a smooth transformation to a uniform grid is available. The present implementation is built on finite differences, so a uniform grid is assumed. Define the cells $I_i = \left[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}\right]$ for $i = 0, \ldots, N-1$. Assume the values of the function $f(x)$ are available at the cell centers, $\{f(x_i) \equiv f_i\}_{i=0}^{N-1}$. The goal of ENO is to use these values to construct an approximation, $\hat{f}_{i+\frac{1}{2}}$, such that

$$\left.\frac{\partial f}{\partial x}\right|_{x_i} = \frac{\hat{f}_{i+\frac{1}{2}} - \hat{f}_{i-\frac{1}{2}}}{\Delta x} + O(\Delta x^k). \tag{A.1}$$

In the context of conservation laws, $\hat{f}_{i+\frac{1}{2}}$ is referred to as a numerical flux function.

For each cell I_i, define the k-point stencil $S_r(i)$:

$$S_r(i) = \{x_{i-r}, \ldots, x_{i+s}\}, \tag{A.2}$$

where $r + s + 1 = k$. In ENO, one begins with a single cell stencil then adaptively adds points depending upon the value of some smoothness measure, e.g., the divided difference, so that r depends on the cell i. The function values f_i are viewed as the cell averages of some unknown function, $h(x)$, so that

$$f(x) = \frac{1}{\Delta x}\int_{x-\frac{\Delta x}{2}}^{x+\frac{\Delta x}{2}} h(\xi)d\xi. \tag{A.3}$$

Take $H(x)$ to be the primitive of $h(x)$, i.e., $H(x) = \int_{-\infty}^{x} h(\xi)d\xi$. Under this point of view, rather than directly approximating $f(x)$, the relationship of the implicitly defined

function $h(x)$ to $f(x)$ is $f'(x) = \frac{1}{\Delta x}\left(h(x+\frac{\Delta x}{2}) - h(x-\frac{\Delta x}{2})\right)$, which allows one to define a flux function in terms of the reconstructed values of h at the cell edges, $\left\{h^-_{i+\frac{1}{2}}\right\}_{i=0}^{N-1}$ and $\left\{h^+_{i-\frac{1}{2}}\right\}_{i=0}^{N-1}$. The $-$ and $+$ superscripts indicate the limits from the left and right, respectively. To find these approximations, seek a polynomial $p_i(x)$ defined on $S_r(i)$ such that p_i approximates h in the following sense:

$$\frac{1}{\Delta x}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} p_i(\xi)d\xi = f_i. \tag{A.4}$$

To achieve an estimate satisfying (A.1), require that the polynomial p_i has degree at most $k-1$. Now define the primitive of p_i as $P_i(x) = \int_{-\infty}^{x} p_i(\xi)d\xi$. Then, $P_i(x)$ is related to f as

$$\frac{1}{\Delta x}\left(P_i(x_{i+\frac{1}{2}}) - P_i(x_{i-\frac{1}{2}})\right) = f_i.$$

There is also the relationship

$$h^-_{i+\frac{1}{2}} = p_i(x_{i+\frac{1}{2}}) = P_i'(x_{i+\frac{1}{2}})$$
$$h^+_{i-\frac{1}{2}} = p_i(x_{i-\frac{1}{2}}) = P_i'(x_{i-\frac{1}{2}}).$$

Take $P_i(x)$ to be the Lagrange polynomial interpolating the function $H(x)$ over the $k+1$ points $x_{i-r-1/2}, \ldots, x_{i+s+1/2}$, so P_i satisfies

$$P_i(x) = \sum_{m=0}^{k} H(x_{i-r+m-1/2})\ell^k_{i,r,m}(x).$$

The form of the Lagrange polynomial can be found in any basic numerical analysis textbook, and is represented here as

$$\ell^k_{i,r,m}(x) = \prod_{\substack{l=0\\ l\neq m}}^{k} \frac{x - x_{i-r+l-1/2}}{x_{i-r+m-1/2} - x_{i-r+l-1/2}}.$$

Now the function $H(x)$ is only implicitly defined, so to eliminate the dependence on the unknown function, write

$$\begin{aligned} P_i(x) - H(x_{i-r-1/2}) = & \\ & \sum_{m=0}^{k}\left(H(x_{i-r+m-1/2}) - H(x_{i-r-1/2})\right)\ell^k_{i,r,m}(x) \\ & = \sum_{m=1}^{k}\left(\sum_{j=0}^{m-1} f_{i-r+j}\Delta x\right)\ell^k_{i,r,m}(x), \end{aligned} \tag{A.5}$$

so that $P_i(x)$ is expressed in terms of the data, $\{f_j\}$. The extra constant is irrelevant since the quantity of interest is the derivative, $p_i(x)$. Upon differentiating (A.5), rearranging terms, and substituting $x = x_{i+1/2}$ and $x = x_{i-1/2}$, one can see that the reconstructed values on the stencil $S_r(i)$ have the form

$$h^-_{i+\frac{1}{2}} = \sum_{j=0}^{k-1} c_{rj} f_{i-r+j}, \quad h^+_{i-\frac{1}{2}} = \sum_{j=0}^{k-1} \tilde{c}_{rj} f_{i-r+j}, \tag{A.6}$$

where the constants c_{rj} and $\tilde{c}_{rj}$ are grid-dependent as derived from the derivatives of the Lagrange polynomials. Furthermore, given the definitions of $h^-_{i+\frac{1}{2}}$ and $h^+_{i+\frac{1}{2}}$ and the stencil $S_r(i)$, it is apparent that $\tilde{c}_{rj} = c_{r-1,j}$. The values of c_{rj} as presented in [17] are provided here for completeness in Table III.

2. WENO finite difference reconstruction

WENO is an extension of ENO, based on the observation that ENO adaptively selects a single stencil for each cell and so effectively uses $2k-1$ cells to obtain order k accuracy. Instead, Liu, Osher, and Chan[24] proposed that all candidate stencils be combined into an order $2k-1$ accurate reconstruction. This is accomplished by defining weights on each stencil so that approximations on stencils where the function appears to be smooth are given a significant weight and those in stencils containing a discontinuity are given very small weight. Let $h^{(r)}_{i+1/2}$ represent the reconstruction on the r^{th} stencil $S_r(i) = \{x_{i-r}, \ldots, x_{i-r+k-1}\}$, that is, using (A.6),

$$h^{(r)}_{i+1/2} = \sum_{j=0}^{k-1} c_{rj} f_{i-r+j}, \quad r = 0, \ldots, k-1. \tag{A.7}$$

Then there are weights ω_r satisfying $\omega_r \geq 0$ and $\sum_{r=0}^{k-1} \omega_r = 1$ such that

$$h_{i+1/2} = \sum_{r=0}^{k-1} \omega_r h^{(r)}_{i+1/2} = h(x_{i+1/2}) + O(\Delta x^{2k-1}).$$

To construct these weights, first observe that for a smooth functions, a Taylor expansion reveals that there are constants d_r such that

$$h_{i+1/2} = \sum_{r=0}^{k-1} d_r h^{(r)}_{i+1/2}, \quad h_{i-1/2} = \sum_{r=0}^{k-1} d_{k-1-r} h^{(r)}_{i-1/2} \tag{A.8}$$

approximates $h(x_{i+1/2})$ and $h(x_{i-1/2})$ (by symmetry) with error $O(\Delta x^{2k-1})$. These constants are repeated from [17] in Table IV for completeness.

When the stencil contains a discontinuity, it is desired that the stencil be assigned a weight $\omega_r << 1$. Thus the weight should be close to d_r in smooth regions, and be moderated by some smoothness factor in non-smooth regions. The weights take the form

$$\omega_r = \frac{\alpha_r}{\sum_{s=0}^{k-1} \alpha_s} \tag{A.9}$$

for $r = 0, \ldots, k-1$, where

$$\alpha_r = \frac{d_r}{(\epsilon + \beta_r)}, \quad \tilde{\alpha}_r = \frac{d_{k-1-r}}{(\epsilon + \beta_r)^2}, \tag{A.10}$$

where β_r is the smoothness factor, and ϵ is just a small constant to prevent division by zero. A choice for β_r is given in [25] that is based on the variation of the reconstruction polynomial $p_r(x)$ defined by stencil $S_r(i)$ in cell I_i:

$$\beta_r = \sum_{l=1}^{k-1} \int_{x_{i-1/2}}^{x_{i+1/2}} \Delta x^{2l-1} \left(\frac{\partial p_r(x)}{\partial x} \right)^2 dx. \tag{A.11}$$

The expressions for computing β_r from the function values $\{f_i\}_{i=0}^{N-1}$ can be derived from the form of the reconstruction polynomial, but are repeated here from [17] again for reference for $k = 2, 3$. When $k = 1$, the stencil is fixed, so $\omega_0 = 1$.

For $k = 2$,

$$\beta_0 = (f_{i+1} - f_i)^2$$
$$\beta_1 = (f_i - f_{i-1})^2$$

yields a third order WENO scheme.

For $k = 3$,

$$\beta_0 = \frac{13}{12}(f_i - 2f_{i+1} + f_{i+2})^2 + \frac{1}{4}(3f_i - 4f_{i+1} + f_{i+2})^2$$
$$\beta_1 = \frac{13}{12}(f_{i-1} - 2f_i + f_{i+1})^2 + \frac{1}{4}(f_{i-1} - f_{i+1})^2$$
$$\beta_2 = \frac{13}{12}(f_{i-2} - 2f_{i-1} + f_i)^2 + \frac{1}{4}(f_{i-2} - 4f_{i-1} + 3f_i)^2.$$

yields a fifth order WENO scheme. It is possible to derive higher order WENO schemes as well, however only third and fifth order WENO are employed in the implementation discussed in this work. The WENO procedure is summarized in Algorithm 1.

Algorithm 1 WENO Reconstruction of $h^+_{i-1/2}$, $h^-_{i+1/2}$

Require: $i \geq 0$, data $\{f_{i+r}\}_{r=-k}^{k}$, $\epsilon > 0$ $\{\epsilon = 10^{-6}$ as implemented$\}$

for $r = 0, \ldots, k-1$ **do**

Compute $h^{(r)}_{i+1/2}$ and $h^{(r)}_{i-1/2}$ using (A.6), and Table III

Compute β_r using the formulae given, or (A.11)

Compute α_r and $\tilde{\alpha}_r$ using Table IV and (A.10)

end for

Compute $\sum_{s=0}^{k-1} \alpha_s$ and $\sum_{s=0}^{k-1} \tilde{\alpha}_s$

for $r = 0, \ldots, k-1$ **do**

Set $\omega_r = \frac{\alpha_r}{\sum_{s=0}^{k-1} \alpha_s}$ and $\tilde{\omega}_r = \frac{\tilde{\alpha}_r}{\sum_{s=0}^{k-1} \tilde{\alpha}_s}$

end for

Construct $h^-_{i+1/2} = \sum_{r=0}^{k-1} \omega_r h^{(r)}_{i+1/2}$ and $h^+_{i-1/2} = \sum_{r=0}^{k-1} \tilde{\omega}_r h^{(r)}_{i+1/2}$

3. Computing the spatial derivatives from the reconstruction

Ultimately the goal is to approximate the spatial operator for the one-dimensional transport equation having the form

$$\frac{\partial f}{\partial t} = -v(x)\frac{\partial f}{\partial x}. \tag{A.12}$$

Algorithm 1 reconstructs the implicitly defined function $h(x)$ at the cell endpoints, but it is still necessary to define an upwind flux function in order to approximate the spatial operator, and in particular, $\frac{\partial f}{\partial x}$ at the cell center, x_i. For this implementation, a Roe flux is selected because it can be computed very quickly. Again following [17], define the Roe speed for (A.12) as

$$\bar{a}_{i+1/2} = -v(x)\frac{f_{i+1} - f_i}{\Delta x}. \tag{A.13}$$

If $\bar{a}_{i+1/2} \geq 0$, then use $\hat{f}_{i+1/2} = h^-_{i+1/2}$ in (A.1), otherwise use $\hat{f}_{i+1/2} = h^+_{i+1/2}$. Thus the semi-discrete form is

$$\left.\frac{\partial f}{\partial t}\right|_{x_i} = -v(x_i)\frac{\hat{f}_{i+1/2} - \hat{f}_{i-1/2}}{\Delta x}. \tag{A.14}$$

4. WENO reconstruction at an arbitrary point

Suppose a function $f(x)$ is given, with values known at the points on the grid $\{x_i\}_{i=0}^{N-1}$, with the grid cells and spacing defined as above. The function $f(x)$ may have discontinuities and it is necessary to find the value at an arbitrary point x^*, where $x^* \in I_i$ for some $i \in \{0, 1, \ldots, N-1\}$. Because of the presence of discontinuities, it is desirable to use a specialized interpolation scheme that captures the behavior of the function accurately in the smooth regions and limits the degradation of the approximation in non-smooth regions of the grid. This section describes how the WENO procedure is adapted to handle this problem. The WENO procedure results in a degree k polynomial representation of the function $h(x)$, denoted $p(x)$, which relates to $f(x)$ as in (A.3),

$$p(x) = \sum_{j=0}^{k-1} \gamma_j x^j,$$

with the constants γ_j, $j = 0, \ldots, k-1$ determined by the WENO procedure.

The goal is to approximate the value of $f(x^*)$ using the values of $f(x)$ at the cell centers, f_i. In fact, for the approximation over the the stencil $S_r(i) = \{x_{i-r}, x_{i-r+1}, \ldots, x_{i-r+k-1}\}$, if $p_f(x)$ is the approximating polynomial to $f(x)$ on cell I_i, then

$$p_f^{(r)}(x^*) = \sum_{j=0}^{k-1} \hat{c}_{rj} f_{i-r+j}, \tag{A.15}$$

for $r = 0, \ldots, k-1$. The constants $\hat{c}_{rj}$ are computed to achieve an order k approximation to the function f. To determine the constants, let I_i be the cell in which the point x^* is located, and define (on a uniform grid)

$$\alpha = \frac{x^* - x_{i-1/2}}{\Delta x} \tag{A.16}$$

so that $0 \le \alpha < 1$ and expressed alternately:

$$x^* = x_{i-1/2} + \alpha \Delta x. \tag{A.17}$$

Given this expression, assume f is smooth in the cell I_i, expand f in a Taylor series about the point $x_{i-1/2}$ (for convenience, this way α is non-negative) and substitute into the sum, matching terms to achieve the desired order of accuracy. The constants are listed in Table V for $k = 1, 2, 3$.

Following the above procedure results in k approximations to the value $f(x^*)$, denoted $f^{*,(r)} \equiv p_f^{(r)}(x^*)$, $r = 0, \ldots, k-1$. As in the above development of WENO, if the function $f(x)$ is smooth, there are constants $\hat{d}_r$ such that

$$f^* = \sum_{r=0}^{k-1} \hat{d}_r f^{*,(r)} = f(x^*) + O(\Delta x^{2k-1}). \tag{A.18}$$

To find these coefficients, expand $f^{*,(r)}$ about x^* and match terms in the expansion. Consistency always results in the condition $\sum_{r=0}^{k-1} d_r = 1$. For $k = 1, 2, 3$, the constants are listed in Table VI.

The WENO procedure for approximating the value of $f(x^*)$ is obtained by applying Algorithm 1 directly to the values $f^{*,(r)}$ in place of $h_{i+1/2}^{(r)}$ and $h_{i-1/2}^{(r)}$.

REFERENCES

[1] 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", Jour. Comp. Phys. **179**, 622-648 (2002).

[2] K. Smith, M. Brown, and F. Tappert, "Ray chaos in underwater acoustics", J. Acoust. Soc. Amer. **91**, 1939-49 (1992).

[3] M. Collins and W. Kuperman, "Overcoming ray chaos", J. Acoust. Soc. Amer. **95**, 3167-70 (1994).

[4] O. Godin, "Restless rays, steady wave fronts", J. Acoust. Soc. Amer. **122**, 3353-63 (2007).

[5] T. Foreman, "An exact ray theoretical formulation of the Helmholtz equation", J. Acoust. Soc. Amer. **86**, 234-246 (1989).

[6] S. Osher and J. Sethian, "Fronts propagating with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulations", Jour. Comp. Phys. **79**, 12-49 (1988).

[7] J. Benamou, "An Introduction to Eulerian geometric optics (1992-2002)", Jour. Sci. Comp. **19**, 63-93 (2003).

[8] B. Engquist, O. Runborg, and A. Tornberg, "High frequency wave propagation by the segment projection method", Jour. Comp. Phys. **178**, 373-390 (2002).

[9] J. Qian, L.-T. Cheng, and S. Osher, "A Level set-based Eulerian approach for anisotropic wave propagation", Wave Motion **37**, 365-379 (2003).

[10] L.-T. Cheng, S. Osher, M. Kang, H. Shim, and Y.-H. Tsai, "Reflection in a level set framework for geometric optics", Computer Modeling in Engineering and Sciences **5**, 347-360 (2004).

[11] J. Qian and S. Leung, "A level set based Eulerian method for paraxial multivalued traveltimes", Jour. Comp. Phys. **197**, 711-736 (2004).

[12] J. Qian and S. Leung, "A local level set method for paraxial geometrical optics", SIAM J. Sci. Comput. **28**, 206-223 (2006).

[13] B. Engquist and O. Runborg, "Computational high frequency wave propagation", Acta Numerica 181-266 (2003).

[14] M. Crandall and P.-L. Lions, "Viscosity Solutions of Hamilton-Jacobi equations", Trans. Amer. Math. Soc. **277**, 1-42 (1983).

[15] S. Osher and R. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces* (Springer-Verlag, New York, 2003), Vol. **153** of *Applied Mathematical Sciences*, Chap. 3, pp. 29-37.

[16] D. Peng, B. Merriman, S. Osher, H. Zhao, and M. Kang, "A PDE-based fast local level set method", Jour. Comp. Phys. **155**, 410-438 (1999).

[17] C.-W. Shu, "Essentially Non-Oscillatory and Weighted Essentially Non-Oscillatory Schemes for Hyperbolic Conservation Laws", ICASE Report 97-65, NASA/CR-97-206253, National Aeronautics and Space Administration, Hampton, VA (1997).

[18] S. Gottlieb and C.-W. Shu, "Total Variation Diminishing Runge-Kutta Schemes", Mathematics of Computation **67**, 73-85 (1998).

[19] W. Lorenson and H. Cline, "Marching Cubes: a high resolution 3D surface reconstruction algorithm", Comp. Graphics **21**, 163-169 (1987).

[20] J. Bowlin, J. Spiesberger, T. Duda, and L. Freitag, "Ocean Acoustical Ray-Tracing Software RAY", Tech. Rept. WHOI-93-10 Woods Hole Oceanographic Institution, Woods Hole, MA (1992).

[21] A. Harten, B. Engquist, S. Osher, and S. Chakravarthy, "Uniformly high-order accurate essentially non-oscillatory schemes III", J. Comput. Phys. **71**, 231-303 (1987).

[22] C.-W. Shu and S. Osher, "Efficient implementation of essentially non-oscillatory shock capturing schemes", J. Comput. Phys. **77**, 439-471 (1988).

[23] C.-W. Shu and S. Osher, "Efficient implementation of essentially non-oscillatory shock capturing schemes II", J. Comput. Phys. **83**, 32-78 (1989).

[24] X.-D. Liu, S. Osher, and T. Chan, "Weighted essentially nonoscillatory schemes", J. Comput. Phys. **115**, 200-212 (1994).

[25] G. Jiang and C.-W. Shu, "Efficient implementation of weighted ENO schemes", J. Comput. Phys. **126**, 202-228 (1996).

TABLE I. Accuracy analysis 1: c = constant

N	Error (max norm)	Effective Order
20	3.306945E-03	–
40	8.509789E-04	1.96
80	1.164058E-04	2.87
160	2.695106E-06	5.43
320	2.312644E-09	10.19

TABLE II. Accuracy analysis 2: $c(z)$ linear

N	Error (max norm)	Effective Order
20	5.253662E-03	–
40	3.423236E-03	0.62
80	4.712438E-04	2.86
160	1.382231E-05	5.09
320	5.375073E-08	8.01

TABLE III. Constants for reconstruction in (A.6), for $k = 1, 2, 3$

k	r	j=0	j=1	j=2
1	-1	1		
	0	1		
2	-1	$\frac{3}{2}$	$-\frac{1}{2}$	
	0	$\frac{1}{2}$	$\frac{1}{2}$	
	1	$-\frac{1}{2}$	$\frac{3}{2}$	
3	-1	$\frac{11}{6}$	$-\frac{7}{6}$	$\frac{1}{3}$
	0	$\frac{1}{3}$	$\frac{5}{6}$	$-\frac{1}{6}$
	1	$-\frac{1}{6}$	$\frac{5}{6}$	$\frac{1}{3}$

	2	$\frac{1}{3}$	$-\frac{7}{6}$	$\frac{11}{6}$

TABLE IV. Constants d_r from (A.8), for $k = 1, 2, 3$

k	d_0	d_1	d_2
1	1		
2	$\frac{2}{3}$	$\frac{1}{3}$	
3	$\frac{3}{10}$	$\frac{3}{5}$	$\frac{1}{10}$

TABLE V. Constants for interpolation in (A.15), for $k = 1, 2, 3$

k	r	j=0	j=1	j=2
1	0	1		
2	0	$\frac{3}{2} - \alpha$	$\alpha - \frac{1}{2}$	
	1	$\frac{1}{2} - \alpha$	$\alpha + \frac{1}{2}$	
3	0	$1 - \frac{(\alpha-\frac{1}{2})(\frac{7}{2}-\alpha)}{2}$	$(\alpha - \frac{1}{2})(\frac{5}{2} - \alpha)$	$\frac{(\alpha-\frac{3}{2})(\alpha-\frac{1}{2})}{2}$
	1	$1 - \frac{(\alpha+\frac{1}{2})(\frac{5}{2}-\alpha)}{2}$	$(\alpha + \frac{1}{2})(\frac{3}{2} - \alpha)$	$\frac{(\alpha+\frac{1}{2})(\alpha-\frac{1}{2})}{2}$
	2	$1 - \frac{(\alpha+\frac{3}{2})(\frac{3}{2}-\alpha)}{2}$	$(\alpha + \frac{3}{2})(\frac{1}{2} - \alpha)$	$\frac{(\alpha+\frac{3}{2})(\alpha+\frac{1}{2})}{2}$

TABLE VI. Constants $\hat{d}_r$ from (A.18), for $k = 1, 2, 3$

k	d_0	d_1	d_2
1	1		
2	$\frac{\alpha+\frac{1}{2}}{2}$	$\frac{\frac{3}{2}-\alpha}{2}$	
3	$\frac{(2\alpha+3)(2\alpha+1)}{48}$	$\frac{(2\alpha+3)(5-2\alpha)}{24}$	$\frac{(2\alpha-3)(-5+2\alpha)}{48}$

LIST OF FIGURES