WaveQ3D: Fast and accurate acoustic transmission loss (TL) eigenrays, in littoral environments, for active sonar simulation/stimulation systems

Sean M. Reilly
Department of Ocean Engineering, University of Rhode Island,
Narragansett RI, USA
campreilly@my.uri.edu

Gopu Potty
Department of Ocean Engineering, University of Rhode Island,
Narragansett RI, USA
potty@egr.uri.edu

David Thibaudeau
AEgis Technologies Group, Inc.,
North Kingstown RI, USA
dthibaudeau@aegistg.com

Michael Goodrich
Alion Science and Technology Corporation,
Norfolk VA, USA
mgoodrich@alionscience.com

Revised (26 June 2014)

The Wavefront Queue 3–D (WaveQ3D) model is a research effort to create fast and accurate acoustic transmission loss (TL) eigenrays, in littoral environments, for active sonar simulation/stimulation systems. WaveQ3D is based on ray theory because Parabolic Equation and Normal Mode models run prohibitively slow at active sonar frequencies above 1000 Hz, where the number of propagating modes is large. To extend applicability to lower frequencies, WaveQ3D augments ray theory with Gaussian beam techniques based on the Gaussian Ray Bundling (GRAB), which is certified for use down to 150 Hz. WaveQ3D is unique among Gaussian beam models in that it solves the eikonal equation in spherical Earth coordinates. The premise of this approach is that, when scenario geometries are constantly evolving, it is more computationally efficient to perform acoustic transmission loss (TL) in the latitude, longitude, altitude coordinates of the underlying environmental databases, than it is to convert the 3–D environments into a series of Nx2–D Cartesian slices. This approach also has the benefit of supporting out-of-plane, 3–D effects.

Keywords: Gaussian beams; 3–D modeling; range-dependent; time-domain.

1. Introduction

Sonar simulation/stimulation systems convert the real-time evolution of complex scenarios into acoustic signals, which are then injected into the sonar, and presented to the operator. They are often used to teach sonar operators to recognize and react to acoustic phenomena, in a controlled environment, at a much lower cost than at-sea exercises. Unfortunately, there

has been very little success in creating acoustic models that can compute results in real-time, for dynamic scenarios, in littoral environments, without significant losses in accuracy.* Most of the successes to date have involved the use of massively parallel computing systems with high acquisition costs.

The Wavefront Queue 3–D (WaveQ3D) model is a research effort to create fast and accurate acoustic transmission loss (TL) eigenrays, in littoral environments, for active sonar simulation/stimulation systems. In addition to TL, these eigenrays include the multipath travel time, phase, and propagation direction information needed by sonar simulation/stimulation systems. Our computation speed goal is to model one-way transmission loss for 100 targets on a single core of an average laptop.

WaveQ3D is based on ray theory because Parabolic Equation and Normal Mode models run prohibitively slow at active sonar frequencies above 1000 Hz, where the number of propagating modes is large. To extend applicability to lower frequencies, WaveQ3D augments ray theory with Gaussian beam techniques based on the Gaussian Ray Bundling (GRAB),[2] which is certified for use down to 150 Hz.[3] WaveQ3D is unique among Gaussian beam models[4,?,6] in that it solves the eikonal equation in spherical Earth coordinates.† The premise of this approach is that, when scenario geometries are constantly evolving, it is more computationally efficient to perform acoustic transmission loss (TL) in the latitude, longitude, altitude coordinates of the underlying environmental databases, than it is to convert the 3–D environments into a series of 2-DxN Cartesian slices. This approach also has the side benefit of supporting out-of-plane, 3–D effects.

The creation of the WaveQ3D model required the derivation of new equations for ray tracing, reflection, eigenrays finding, and Gaussian beam propagation loss. This paper introduces WaveQ3D to the research community by outlining that derivation. Future papers will focus on the accuracy of the WaveQ3D model relative to analytic solutions and at-sea results.

2. Ray Tracing in Spherical/Time Coordinates

Ray theory decomposes acoustic waves into surfaces of constant travel time (t) from the source (Fig. 1). The rays are a vector field that is normal to these surfaces at each point in space, and the route of these rays through the medium defines the direction of propagation. In the high frequency limit, spreading loss occurs as the energy of the wavefront is stretched over increasingly larger areas during propagation. Conventional ray theory uses the change in distance between rays to model the spreading effects of wavefront propagation. The fundamental equations of ray theory are derived[18] by seeking solutions to the Helmholtz

*Discussion with Michael Vaccaro, project manager for the High Fidelity Active Sonar Training (HiFAST) Project at the U.S. Office of Naval Research.

†Although the Ray-Tracing Program for the Ocean (HARPO) model [7] also computes ray paths on a curved Earth, its use of the Hamilton on an elliptical Earth introduces complexity that we do not believe will impact training scenarios. HARPO also does not compute transmission loss and other eigenrays products at this time.

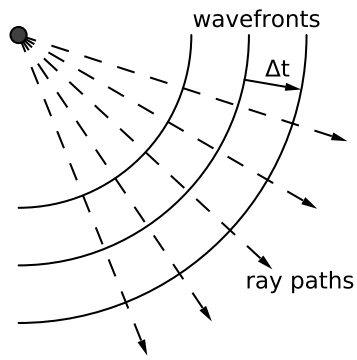

Fig. 1. Acoustic ray geometry.

equation (1) in the form given by Eq. (2)

$$\nabla^2 p + \frac{\omega^2}{c^2(\vec{r})} p(\vec{r}) = -\delta(\vec{r} - \vec{r}_0) \tag{1}$$

$$p(\vec{r}) = e^{i\omega t(\vec{r})} \sum_{j=0}^{\infty} \frac{A_j(\vec{r})}{(i\omega)^j} \,. \tag{2}$$

Equating terms of like order in ω, yields the infinite sequence of equations given by Eqs. (3), (4), and (5)

$$O(\omega^2) : \left|\vec{\nabla} t\right|^2 = \frac{1}{c^2(\vec{r})} \,, \tag{3}$$

$$O(\omega) : 2\vec{\nabla} A_0 \cdot \vec{\nabla} t + (\nabla^2 t) A_0 = 0 \,, \tag{4}$$

$$O(\omega^{1-j}) : 2\vec{\nabla} A_j \cdot \vec{\nabla} t + (\nabla^2 t) A_j = -\nabla^2 A_{j-1} \qquad \text{for j=1,2,...} \tag{5}$$

where $\vec{r}$ is the position coordinate along a ray path; c is the speed of sound in water; A is the wavefront amplitude; and t is the travel time along the ray path. The eikonal equation (3) defines the relationship between the direction of propagation and the speed of sound in water. The first transport equation (4) relates the spreading loss of the acoustic field to the divergence in the propagation direction. The remaining transport equations (5) relate the spreading loss of the acoustic field to diffraction effects. Eqs. (3) and (4) are an exact solution of the wave equation in the geometric limit, that is, when the sound speed gradient along the direction of motion changes slowly compared to the acoustic wavelength. Conventional ray theory accuracy starts to break down at lower frequencies where diffraction becomes a significant feature of acoustic propagation.

The analytic solution to the eikonal equation (3) is found[18] by relating $\vec{\nabla} t$ to $\hat{n}$, the direction of energy propagation along the ray paths

$$\hat{n} = \frac{d\vec{r}}{ds} = c\vec{\nabla} t \,. \tag{6}$$

This transforms the eikonal equation into a second order ordinary differential equation in terms of $\vec{r}$, c, and s

$$\frac{d}{ds}\left(\frac{1}{c}\frac{d\vec{r}}{ds}\right) = -\frac{1}{c^2}\vec{\nabla} c \,. \tag{7}$$

This can be reduced to a pair of simultaneous first order equations by introducing the temporary variable $\vec{\xi}$

$$\frac{d\vec{\xi}}{ds} = -\frac{1}{c^2}\vec{\nabla} c \,, \tag{8}$$

$$\frac{d\vec{r}}{ds} = c\vec{\xi} \,. \tag{9}$$

This set of equations can be solved using a series of arc length steps given initial conditions.[8,9] Note that although the ray paths represented by Eqs. (12) and (13) are independent of frequency, the loss along those paths will include the frequency dependent effects of seawater absorption and interface reflection.

Equation (9) illustrates that the temporary variable $\vec{\xi}$ is actually the direction of propagation scaled by the speed of sound, or equivalently, the wave number vector divided by the angular frequency

$$\vec{\xi} = \frac{\hat{n}}{c} = \frac{\vec{k}}{\omega} \,, \tag{10}$$

where $\vec{k}$ is the acoustic wave number vector; and ω is the angular frequency of the acoustic source. This definition of $\vec{\xi}$ allows the system of equations represented by Eqs. (8) and (9) to be initialized with the position and steering angle for each ray path at the acoustic source. Note that the steering angle is defined using the depression/elevation (μ) and the azimuthal steering (φ) launch angles of each ray relative to the source. Marching the simultaneous first order equations (8) and (9) through steps in arc length then generates ray paths throughout the water column. The spreading loss of the wavefront at any point is calculated by measuring the spreading between adjacent rays. As adjacent rays have different travel times in this treatment, there is an implicit assumption that propagation loss is being calculated for a continuous ensonification at a single frequency.

Instead of solving the ray equations in units of arc-length, WaveQ3D uses a change of variables (Eq. (11)) to transform equations (8) and (9) into functions of time.

$$\frac{d}{ds} = \frac{1}{c}\frac{d}{dt} \,, \tag{11}$$

$$\frac{d\vec{\xi}}{dt} = -\frac{1}{c}\vec{\nabla}c\,, \tag{12}$$

$$\frac{d\vec{r}}{dt} = c^2\vec{\xi}\,. \tag{13}$$

Computing acoustic propagation as a function of time has several advantages for training system implementations. First, because phase coherence between rays is preserved at each step, discontinuities between ray paths are reduced. Second, it allows the implementation to return propagation results to the calling program, in travel time order, before later paths have been computed. Third, it casts the solutions into a form that is very useful for calculating reverberation.

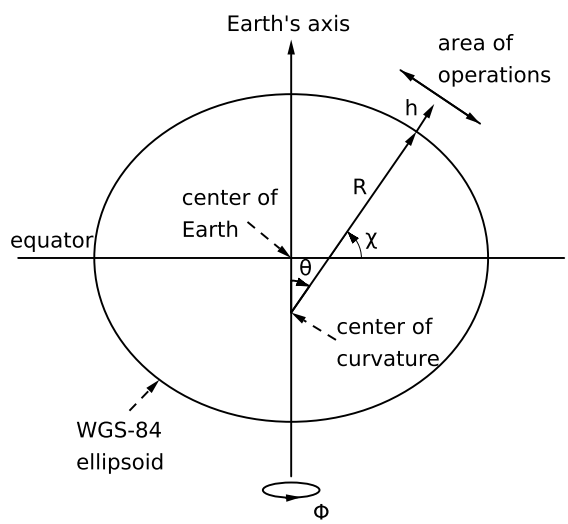

Fig. 2. Spherical earth coordinates.

The WaveQ3D model uses the World Geodetic System 1984 (WGS–84)[19] reference ellipse to represent zero altitude worldwide (mean sea level). A spherical polar coordinate system is constructed within each area of operations, and its radius is set equal to the WGS–84 radius of curvature at the center of the area. This coordinate system is illustrated in Fig. 2 where χ is latitude; ϕ is longitude; h is altitude above mean sea level; R is radius of earth's curvature for the area of operations; r is distance from the center of curvature (R+h); and θ is co-latitude ($90° - \chi$).

The WaveQ3D model defines the average radius of curvature in the area of operations using a combination of WGS–84 radii in the latitude and longitude directions[7]

$$w^2 \equiv 1 - e^2 cos\chi\,, \tag{14}$$

$$R_\chi = \frac{a(1-e^2)}{w^3} \,, \tag{15}$$

$$R_\phi = \frac{a}{w} \,, \tag{16}$$

$$R = \sqrt{R_\chi R_\phi} \tag{17}$$

where e is the WGS–84 elliptic eccentricity of earth (8.1819190842622x10^{-2}); a is the WGS–84 equatorial radius of the earth (6378137.0 m); (R_χ,R_ϕ) are the radii of curvature in the latitude and longitude directions; and R is the average radius of curvature in area of operations.

From these definitions, all locations in the area of operations can be represented in terms of the spherical polar properties (r,θ,ϕ). The transformation of Eqs. (12) and (13) into this coordinate system uses the fact that the ray position only has radial components, while the ray direction has components in all three dimensions.[‡]

$$\vec{r}(t) = r(t)\hat{r}(t) \,, \tag{18}$$

$$\vec{\xi}(t) = \alpha(t)\hat{r}(t) + \beta(t)\hat{\theta}(t) + \gamma(t)\hat{\phi}(t) \,, \tag{19}$$

$$\frac{d\vec{r}}{dt} = \frac{dr}{dt}\hat{r} + r\frac{d\hat{r}}{dt} \,, \tag{20}$$

$$\frac{d\vec{\xi}}{dt} = \frac{d\alpha}{dt}\hat{r} + \alpha\frac{d\hat{r}}{dt} + \frac{d\beta}{dt}\hat{\theta} + \beta\frac{d\hat{\theta}}{dt} + \frac{d\gamma}{dt}\hat{\phi} + \gamma\frac{d\hat{\phi}}{dt} \tag{21}$$

where $\alpha(t)$, $\beta(t)$, and $\gamma(t)$ are the radial, co-latitude, and longitude components of the normalized ray direction. Note that the unit vectors $\hat{r}$, $\hat{\theta}$, and $\hat{\phi}$ in Eqs. (20) and (21) change as a function of r, θ, and ϕ. The impact of these derivatives can be understood by casting them into their Cartesian coordinate equivalents before applying the time derivative

$$\hat{r}(t) = sin\theta(t)\, cos\phi(t)\, \hat{i} + sin\theta(t)\, sin\phi(t)\, \hat{j} + cos\theta(t)\, \hat{k} \,, \tag{22}$$

$$\hat{\theta}(t) = cos\theta(t)\, cos\phi(t)\, \hat{i} + cos\theta(t)\, sin\phi(t)\, \hat{j} - sin\theta(t)\, \hat{k} \,, \tag{23}$$

$$\hat{\phi}(t) = -sin\phi(t)\, \hat{i} + cos\phi(t)\, \hat{j} \,. \tag{24}$$

The chain rule, when applied to Eqs. (22), (23), and (24) yields the following time derivatives in spherical coordinates

$$\frac{d\hat{r}}{dt} = \frac{d\theta}{dt}\hat{\theta} + sin\theta\frac{d\phi}{dt}\hat{\phi} \,, \tag{25}$$

[‡]This derivation uses arrows for vectors with magnitude and direction (such as $\vec{r}$), carets for unit length vectors (such as $\hat{r}$), and plain text for magnitude parameters (such as r).

$$\frac{d\hat{\theta}}{dt} = -\frac{d\theta}{dt}\hat{r} + cos\theta\frac{d\phi}{dt}\hat{\phi}\,, \tag{26}$$

$$\frac{d\hat{\phi}}{dt} = -sin\theta\frac{d\phi}{dt}\hat{r}\,. \tag{27}$$

Applying Eqs. (25) through (27) to Eqs. (20) and (21) transforms the ray tracing equations into spherical earth coordinates

$$\frac{d\vec{r}}{dt} = \frac{dr}{dt}\hat{r} + r\frac{d\theta}{dt}\hat{\theta} + rsin\theta\frac{d\phi}{dt}\hat{\phi}\,, \tag{28}$$

$$\begin{aligned}\frac{d\vec{\xi}}{dt} = \left[\frac{d\alpha}{dt} - \beta\frac{d\theta}{dt} + \gamma sin\theta\frac{d\phi}{dt}\right]\hat{r} + \left[\frac{d\beta}{dt} + \alpha\frac{d\theta}{dt} - \gamma cos\theta\frac{d\phi}{dt}\right]\hat{\theta} \\ + \left[\frac{d\gamma}{dt} + (\alpha sin\theta + \beta cos\theta)\,\frac{d\phi}{dt}\right]\hat{\phi}\,.\end{aligned} \tag{29}$$

Matching terms for $\hat{r}$, $\hat{\theta}$, and $\hat{\phi}$ yields a system of six first-order, scalar differential equations

$$\frac{dr}{dt} = c^2\alpha\,, \tag{30}$$

$$r\frac{d\theta}{dt} = c^2\beta\,, \tag{31}$$

$$rsin\theta\frac{d\phi}{dt} = c^2\gamma\,, \tag{32}$$

$$\frac{d\alpha}{dt} - \beta\frac{d\theta}{dt} - \gamma sin\theta\frac{d\phi}{dt} = -\frac{1}{c}\frac{dc}{dr}\,, \tag{33}$$

$$\frac{d\beta}{dt} + \alpha\frac{d\theta}{dt} - \gamma cos\theta\frac{d\phi}{dt} = -\frac{1}{cr}\frac{dc}{d\theta}\,, \tag{34}$$

$$\frac{d\gamma}{dt} + (\alpha sin\theta + \beta cos\theta)\,\frac{d\phi}{dt} = -\frac{1}{crsin\theta}\frac{dc}{d\phi}\,. \tag{35}$$

When Eqs. (30) though (32) are plugged into Eqs. (33) though (35), the system is reduced to a state where all of the coordinate derivatives appear only once

$$\frac{dr}{dt} = c^2\alpha\,, \tag{36}$$

$$\frac{d\theta}{dt} = \frac{c^2\beta}{r}\,, \tag{37}$$

$$\frac{d\phi}{dt} = \frac{c^2\gamma}{rsin\theta} \ , \tag{38}$$

$$\frac{d\alpha}{dt} = -\frac{1}{c}\frac{dc}{dr} + \frac{c^2}{r}\left(\beta^2 + \gamma^2\right) \ , \tag{39}$$

$$\frac{d\beta}{dt} = -\frac{1}{cr}\frac{dc}{d\theta} - \frac{c^2}{r}\left(\alpha\beta + \gamma^2 cot\theta\right) \ . \tag{40}$$

$$\frac{d\gamma}{dt} = -\frac{1}{crsin\theta}\frac{dc}{d\phi} - \frac{c^2\gamma}{r}\left(\alpha + \beta cot\theta\right) \ , \tag{41}$$

WaveQ3D uses Equations (36) through (41) to compute the time evolution for the location and direction for any acoustic ray based on its initial conditions. These equations include the horizontal refraction effects and follow great circle routes between as they traverse latitudes and longitude. Environmental parameters and their derivatives are computed directly in geographic latitude, longitude, and altitude without reducing the problem to a series of Nx2–D Cartesian projections.

The WaveQ3D model uses an explicit, third order, Adams-Bashforth (AB3) algorithm[8] to propagate Eqs. (36) through (41) numerically. The AB3 algorithm, summarized in Eqs. (42), approximates each step in the solution using cached information from the three time steps that came before it

$$\vec{f}(t_{n+1}) = \vec{f}(t_n) + \delta t\left[\frac{23}{12}\frac{d\vec{f}}{dt}(t_n) - \frac{16}{12}\frac{d\vec{f}}{dt}(t_{n-1}) + \frac{5}{12}\frac{d\vec{f}}{dt}(t_{n-2})\right] \ . \tag{42}$$

where $\vec{f}$ is a vector of the positions, directions, and their derivatives such that

$$\vec{f} = \left[r, \theta, \phi, \alpha, \beta, \gamma, \frac{dr}{dt}, \frac{d\theta}{dt}, \frac{dr\phi}{dt}, \frac{d\alpha}{dt}, \frac{d\beta}{dt}, \frac{d\gamma}{dt}\right] . \tag{43}$$

When the past values are cached instead of re-calculated, AB3 is much faster than other integrators with similar accuracy. However, because AB3 is not self-starting, the WaveQ3D model uses a third order Runge-Kutta (RK3) algorithm[9] whenever the ray parameters must be initialized, or re-initialized

During the development of WaveQ3D, we discovered that the inclusion of spherical coordinate terms in Eqs. (36) through (41) had very little impact on the speed of ray trace calculations. The types of training applications that we are targeting rely on gridded environmental data that has been derived from at-sea measurements. We found that the interpolations needed to compute the sound speed and its derivatives from gridded data were much more computationally expensive than algebraic or trigonometric functions because of the search operations inherent in interpolation. This difference becomes even more pronounced if terms like $sin\theta$ and c^2/r are cached at the time that they are first used

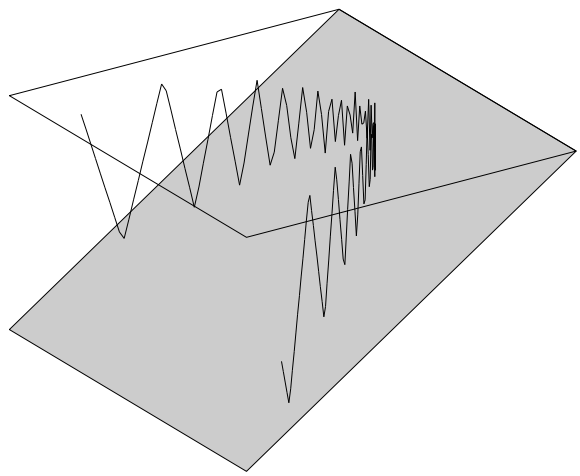

Fig. 3. Horizontal refraction from 3-D slope.

3. Boundary Reflections on a Spherical Earth

Fig. 3 illustrates WaveQ3D out-of-plane reflection from a 3-D slope. In this example, a single ray is launched obliquely up a 2.86 degree slope. Each time the ray reflects from the bottom, the tilt of the normal turns the ray slightly down slope. Eventually the ray stops traveling up toward the apex and turns down slope. Rays with steep launch angle have more bottom reflections and turn down slope faster than shalllow-angle rays.

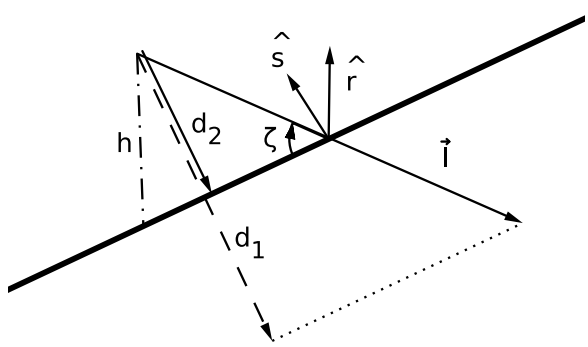

Fig. 4. Estimating the point of reflection.

Interface reflection in WaveQ3D starts with an estimate of the point in time when the

incident ray strikes the bottom. The derivation of the equations for estimating the time of reflection in spherical coordinates uses the symbols defined in Fig. 4 where $\vec{I}$ is the incident ray path; ζ is the incident grazing angle; $\hat{s}$ is the surface normal; $\hat{r}$ is the unit vector in the radial direction; h is the incident ray height above bottom; Δt is the normal time step; and δt is the time step needed to reach the interface. Note that, to simplify the geometry, Fig. 4 is drawn in the plane of reflection. Fig. 5 illustrates this same scenario in a 3–D perspective.

If the bottom slope is nearly constant across the length of the incident ray, then the ratio of the time steps is equivalent to the ratio of the distances normal to the surface

$$d_1 \equiv -\vec{I} \cdot \hat{s} = -\left(\frac{d\vec{r}}{dt} \cdot \hat{s}\right) \Delta t \,, \tag{44}$$

$$d_2 \equiv -h\hat{r} \cdot \hat{s} \,, \tag{45}$$

$$\frac{\delta t}{\Delta t} = \frac{d_2}{d_1} = \frac{h\,\hat{r} \cdot \hat{s}}{\left(\frac{d\vec{r}}{dt} \cdot \hat{s}\right) \Delta t} \,, \tag{46}$$

$$\delta t = \frac{h\,\hat{r} \cdot \hat{s}}{\frac{d\vec{r}}{dt} \cdot \hat{s}} \,. \tag{47}$$

At the ocean surface, this simplifies to

$$\delta t_{surface} = \frac{h}{\frac{dr}{dt}} \,, \tag{48}$$

where $\frac{dr}{dt}$ is the radial ray tracing component defined in Eq. (36).

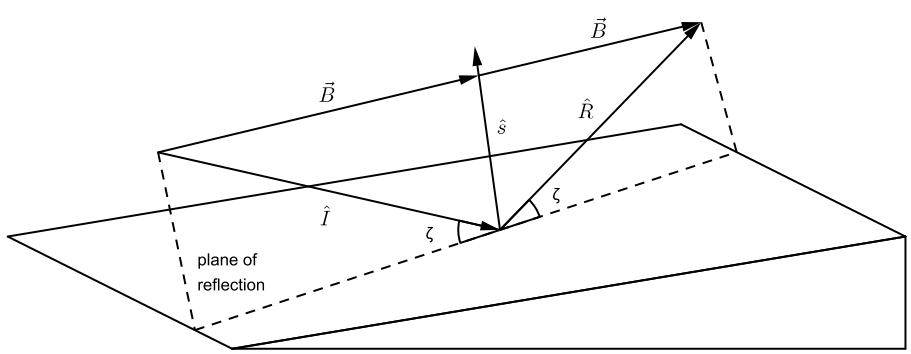

Fig. 5. Reflection from a 3–D slope.

The next step in WaveQ3D reflection modeling is estimating the direction of reflection from a 3–D slope. The derivation of the equations for estimating this direction in spherical coordinates uses the symbols defined in Fig. 5 where $\hat{I}$ is the incident ray path direction; $\hat{R}$ is the reflected ray path direction; and $\vec{B}$ is the component of the incident ray that is perpendicular to surface normal. Since the reflected ray has the same angle to the normal as the incident ray

$$\hat{R} = 2\vec{B} - \hat{I}\,, \tag{49}$$

$$\vec{B} = \hat{I} - (\hat{I} \cdot \hat{s})\hat{s}\,, \tag{50}$$

$$\hat{R} = \hat{I} - 2(\hat{I} \cdot \hat{s})\hat{s}\,. \tag{51}$$

For ocean surface reflections, these relationships negate the sign of the radial component while leaving the θ and ϕ direction components unchanged.

During development, we found that the Eq. (51) was very sensitive to the curvature of the incident ray. Using a second second order Taylor expansion for each component of the position, normalized direction, and sound speed, to find their precise values at the point defined by Eq. (47), improved the overall accuracy of the reflection model.

Most bottom depth databases grid the relief of the earth's surface into a series of geographic latitude and longitude points. WaveQ3D compute the spherical components of the surface normal (s_r, s_θ, s_ϕ) by equating the slope $(\sigma_\theta, \sigma_\phi)$ to the first derivative of b, the bottom's radial position at the point of reflection.

$$\Omega_\theta \equiv tan(\sigma_\theta) = \frac{1}{b}\frac{db}{d\theta}\,, \tag{52}$$

$$\Omega_\phi \equiv tan(\sigma_\phi) = \frac{1}{b\,sin\theta}\frac{db}{d\phi}\,, \tag{53}$$

$$s_\theta = -sin(\sigma_\theta) = -\frac{\Omega_\theta}{\sqrt{1+\Omega_\theta^2}}\,, \tag{54}$$

$$s_\phi = -sin(\sigma_\phi) = -\frac{\Omega_\phi}{\sqrt{1+\Omega_\phi^2}}\,, \tag{55}$$

$$s_r = \sqrt{1-(s_\theta^2+s_\phi^2)}\,. \tag{56}$$

After reflection, WaveQ3D must re-initialize the ray in a way that maintains the phase continuity of the wavefront. The WaveQ3D model uses a third order Runge-Kutta (RK3) algorithm to reverse propagation from the point of reflection back to the t_n, t_{n-1}, t_{n-2} points in time, during reinitialization.

4. Finding Eigenrays using Coherent Wavefronts

In the WaveQ3D model, eigenrays are derived from each target's geometry relative to a Closest Point of Approach (CPA) on the wavefront. A point on the wavefront is the CPA for a specific target if it has the smalled distance to that target relative to the wavefront points immediately surround it. The coordinates for this distance calculation are illustrated in Fig. 6 where $\vec{r}_p$ is the position of the eigenray target; $\vec{r}_{njk}$ is the position of a candidate point on the wavefront; d_{njk} is the distance from target to each point on wavefront; δt is the target offset along the direction of propagation; $\delta\mu$ is the target offset in the depression/elevation direction; and $\delta\varphi$ is the target offset in the azimuthal direction.

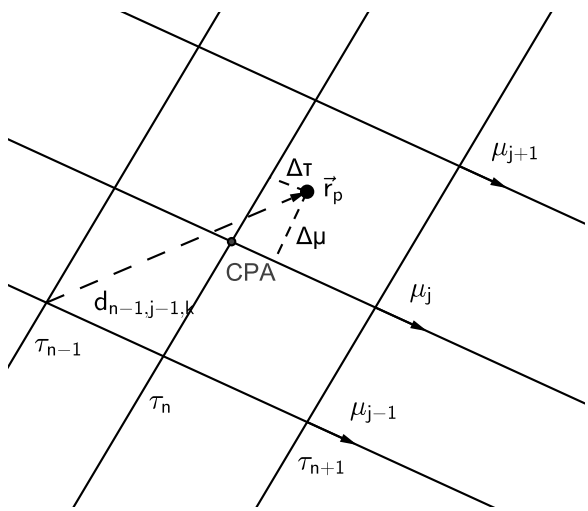

Fig. 6. Eigenray estimation geometry (side view: φ_k direction not shown).

WaveQ3D uses an efficient calculation for the square of the distance in spherical coordinates that is derived from the Haversine formula[12]

$$d_{njk}^2 = r_p^2 + r_{njk}^2 - 2\vec{r}_p \cdot \vec{r}_{njk} \,, \tag{57}$$

$$d_{njk}^2 = r_p^2 + r_{njk}^2 - 2r_p r_{njk} \left[sin(\chi_p) sin(\chi_{ijk}) + cos(\chi_p) cos(\chi_{ijk}) cos(\phi_p - \phi_{ijk}) \right] \,, \tag{58}$$

$$\begin{aligned} d_{njk}^2 = r_p^2 + r_{njk}^2 - 2r_p r_{njk} \Bigg[1 - 2 \Bigg\{ & sin^2 \left(\frac{\chi_p - \chi_{ijk}}{2} \right) \\ & + cos(\chi_p) cos(\chi_{ijk}) sin^2 \left(\frac{\phi_p - \phi_{ijk}}{2} \right) \Bigg\} \Bigg] \,, \end{aligned} \tag{59}$$

$$d_{njk}^2 \approx r_p^2 + r_{njk}^2 - 2r_p r_{njk} \left[1 - 2 \left\{ \left(\frac{\chi_p - \chi_{ijk}}{2} \right)^2 + cos(\chi_p) cos(\chi_{ijk}) \left(\frac{\phi_p - \phi_{ijk}}{2} \right)^2 \right\} \right] , \tag{60}$$

$$d_{njk}^2 \approx r_p^2 + r_{njk}^2 - 2r_p r_{njk} \left[1 - 2 \left\{ \left(\frac{\theta_p - \theta_{ijk}}{2} \right)^2 + sin(\theta_p) sin(\theta_{ijk}) \left(\frac{\phi_p - \phi_{ijk}}{2} \right)^2 \right\} \right] . \tag{61}$$

The accuracy of this approximation improves as the distance between the target and the wavefront decreases. Because the implementation caches values for $sin\theta$ at both the wavefront and target locations, Eq. (61) allows d_{njk}^2 to be computed without the use of any additional transcendental functions.

WaveQ3D uses the offset of each target relative to the travel time (t), depression/elevation launch angle (μ), and the azimuthal launch angle (φ) coordinates of the CPA to compute eigenrays. The WaveQ3D model calculates these offsets by expressing d_p^2, the square of the distance at the target point, as a second order Taylor series, relative to the CPA, in vector form

$$\vec{\rho} \equiv (\rho_1, \rho_2, \rho_3) \equiv (\delta t, \delta\mu, \delta\varphi) , \tag{62}$$

$$d_p^2 \approx \epsilon + \vec{g} \cdot \vec{\rho} + \frac{1}{2} \vec{\rho} \cdot \mathbf{H} \, \vec{\rho} , \tag{63}$$

$$\epsilon \equiv d^2 \big|_{CPA} , \tag{64}$$

$$\vec{g} \equiv \frac{\partial d^2}{\partial \vec{\rho}} \big|_{CPA} , \tag{65}$$

$$\mathbf{H} \equiv \frac{\partial^2 d^2}{\partial \vec{\rho}^2} \big|_{CPA} \tag{66}$$

where $\vec{\rho}$ is the target offset from CPA in vector form; $\vec{g}$ is the gradient of squared distance at CPA (3 elements), and $\mathbf{H}$ is the Hessian matrix of squared distance at CPA (3x3).

One way to solve this equation would be to search for a value of $\vec{\rho}$ for which Eq. (63) was zero. However, since d_p^2 is positive definite, the problem can be simplified by searching for the minimum value, indicated by a zero in the first derivative

$$\frac{\partial d_p^2}{\partial \vec{\rho}} = \vec{g} + \mathbf{H} \, \vec{\rho} = 0 , \tag{67}$$

$$\mathbf{H} \, \vec{\rho} = -\vec{g} , \tag{68}$$

$$\vec{\rho} = -\mathbf{H}^{-1} \, \vec{g} . \tag{69}$$

This treatment reduces the offset estimation problem to the calculation of the gradient of distance, the calculation of the Hessian matrix, and a matrix inversion. Note that the inverse of a 3x3 matrix has a simple analytic solution that allows it to be solved efficiently and without approximation.

Some eigenray products can be computed directly from this offset vector solution

$$t_p = t_n + \delta t \,, \tag{70}$$

$$\mu_p = \mu_j + \delta\mu \,, \tag{71}$$

$$\varphi_p = \varphi_k + \delta\varphi \tag{72}$$

where t_p is the travel time from the source; μ_p is the depression/elevation launch angle at source; and φ_p is the azimuthal launch angle at source. WaveQ3D computes the direction at the target by forward solving a 2nd order Taylor series for $\vec{\xi}$ in the neighborhood of the CPA. The calculation of the eigenray is completed by the calculation of propagation loss, which is discussed in the next section.

Note that this eigenray detection process is less much efficient than an equivalent calculation in Cartesian coordinates. The impact of this difference is minimized when the number of targets is small compared to the number ray tracing points; a good assumption for real-time, sonar simulation/stimulation systems. Unfortunately, this assumption may make the WaveQ3D model inefficient for tactical decision aids, because the number of target points is large in "what-if" applications.

5. Computing Propagation Loss using 3–D Gaussian beams

In conventional ray theory, the spreading of acoustic propagation loss is estimated by measuring the changes in ensonified area between ray paths. The intensity across the wavefront is inversely proportional to the change in a surface area segment compared to its area at the source. The Gaussian beam approach uses dynamic ray equations to compute the divergence of the acoustic field normal to the path of propagation. The GRAB model assumes that this divergence can be estimated from the wavefront shape directly, and this reduces the computational cost of transmission loss calculations.

In 2–D Gaussian beam models, the intensity at the target location is a summation of contributions from rays above and below the eigenray target. To extend this to three dimensions, WaveQ3D assumes that the wavefront spreading is independent in the μ and φ directions. This assumption decomposes the Gaussian ray contributions to a series of nearest neighbor lines in the μ and φ directions, as illustrated in Fig. 7.

$$G(\vec{r}_p) = \left(\sum_{j'=j-J}^{j+J} g_{j'}(\vec{r}_p) \right) \left(\sum_{k'=k-K}^{k+K} g_{k'}(\vec{r}_p) \right) \,, \tag{73}$$

where $G(\vec{r}_p)$ is the total Gaussian beam intensity at the eigenray target; (j, k) are the index numbers of the cell containing the eigenray target; $g_{j'}$ are the Gaussian beam contributions

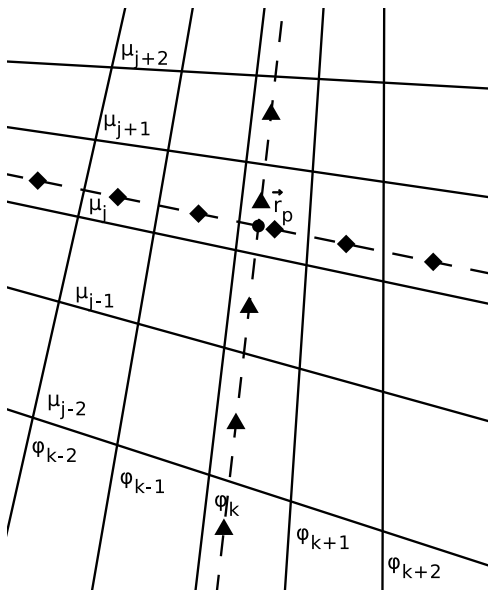

Fig. 7. Gaussian ray nearest neighbors (front view: t_n direction not shown).

along depression/elevation direction; $g_{k'}$ are the Gaussian beam contributions along the azimuthal direction; $2J+1$ are the number of significant beams in the depression/elevation direction; and $2K+1$ are the number of significant beams in the azimuthal direction.

The intensity of each Gaussian beam contribution is a function of the width of the beam and the distance along the wavefront to the eigenray target, normalized to the average distance across the beam at the source

$$N_{j'} = \frac{\int_{\varphi_{k'}}^{\varphi_{k'+1}} \left(\mu_{j'+1} - \mu_{j'}\right) d\varphi}{\int_{\varphi_{k'}}^{\varphi_{k'+1}} d\varphi} = \mu_{j'+1} - \mu_{j'} \ , \tag{74}$$

$$N_{k'} = \frac{\int_{\mu_{j'}}^{\mu_{j'+1}} \left(\varphi_{k'+1} - \varphi_{k'}\right) cos\left(\mu\right) d\mu}{\int_{\mu_{j'}}^{\mu_{j'+1}} d\mu} = \frac{sin(\mu_{j'+1}) - sin(\mu_{j'})}{\mu_{j'+1} - \mu_{j'}} \left(\varphi_{k'+1} - \varphi_{k'}\right) \ , \tag{75}$$

$$g_{j'}(\vec{r}_p) = \frac{N_{j'}}{\sqrt{2\pi w_{j'}^2}} \, exp\left(-\frac{d_{j'}^2}{2w_{j'}^2}\right) \ , \tag{76}$$

$$g_{k'}(\vec{r}_p) = \frac{N_{k'}}{\sqrt{2\pi w_{k'}^2}} \, exp\left(-\frac{d_{k'}^2}{2w_{k'}^2}\right) \ , \tag{77}$$

where $w_{j'}$ and $w_{k'}$ are the half-widths of the Gaussian beam (from the cell center to one edge) in the μ and φ directions; and $d_{j'}^2$ and $d_{k'}^2$ are the distance in the μ and φ directions from the Gaussian beam center to the eigenray target.

The WaveQ3D model aligns the ray paths with the edges of the Gaussian beam (full-width at half maximum). Distance terms in the μ direction are calculated using the following pattern

$$L = 2\, w_j \, \frac{\delta\mu}{\Delta\mu_j} \ , \tag{78}$$

$$d_j = L - w_j \ , \tag{79}$$

$$d_{j-1} = L - w_{j-1} \ , \tag{80}$$

$$d_{j-2} = L + 2w_{j-1} + w_{j-2} \ , \tag{81}$$

$$d_{j+1} = L - 2w_j - w_{j+1} \ , \tag{82}$$

$$d_{j+2} = L - 2w_j - w_{j+1} - w_{j+2} \tag{83}$$

where $\delta\mu$ is the eigenray offset in depression/elevation angle; $\Delta\mu$ is the depression/elevation width of this beam at the source; w_j is the width of beam j; and all widths have been interpolated to the time $t_n + \delta t$. More distant cells can be supported by adding additional $2w$ terms to Eqs. (81) and (83). The pattern in the azimuthal direction is similar.

GRAB[2] models the frequency dependent component of the beamwidth by giving each beam a minimum width

$$w'_j(f) = max\left(w_j, 2\pi\lambda\right) \ , \tag{84}$$

where λ is the wavelength of the signal being modeled; w_j is the cell width of beam j, and $w'_j(f)$ is the adjusted width of beam j. The λ term acts as the minimum amount of spreading that GRAB expects beams to project into neighboring areas. In GRAB, Gaussian beams are overlapped by 50% to reduce fluctuations in the vicinity of the mid-point between rays.[13] The number of Gaussian profiles is effectively twice the number of ray paths.

The WaveQ3D model treats beam width broadening as a convolution between a frequency dependent Gaussian propagation spread and a second Gaussian that represents the spatial spreading created by the sampling of the wavefront.

$$(w'_j(f))^2 = (2w_j)^2 + (2\pi\lambda)^2 \ . \tag{85}$$

which avoids the sudden transitions inherent in taking a maximum value. WaveQ3D manages the 50% overlap by multiplying the cell width term by two. Treating the two spreading terms in this way reduces the number of Gaussian beam calculations by two, Normalizing Eq. (76) by the combined effect of both spreading sources conserves energy across the wavefront.

During development, we evaluated the impact of these changes on propagation into a shadow zone using the downward refracting n^2 linear environment used in the development of GRAB's minimum beamwidth.

$$c(z) = \frac{c_0}{\sqrt{1 + \frac{2g_0 z}{c_0}}}, \tag{86}$$

where $c(z)$ is speed of sound as a function of depth; z is depth (positive is down); c_0 is the speed of sound at the ocean surface (=1550.0 m/s); and g_0 is the sound speed gradient at the ocean surface (=1.2 s^{-1}).

Because analytic results for this scenario did not include the effects of a round Earth, an Earth flattening correction needed to be applied to the sound speed profile before WaveQ3D could generate equivalent results. The Earth flattening term is the inversion of the term used by GRAB to incorporate Earth curvature into its calculations.[15] This correction is given by

$$c(r) = \frac{r}{R}\, c(z)\,, \tag{87}$$

where r is the radial distance from the center of curvature; R is the radius of earth's curvature in this area of operations; $c(z)$ is the original sound speed; and $c(r)$ is the modified sound speed.

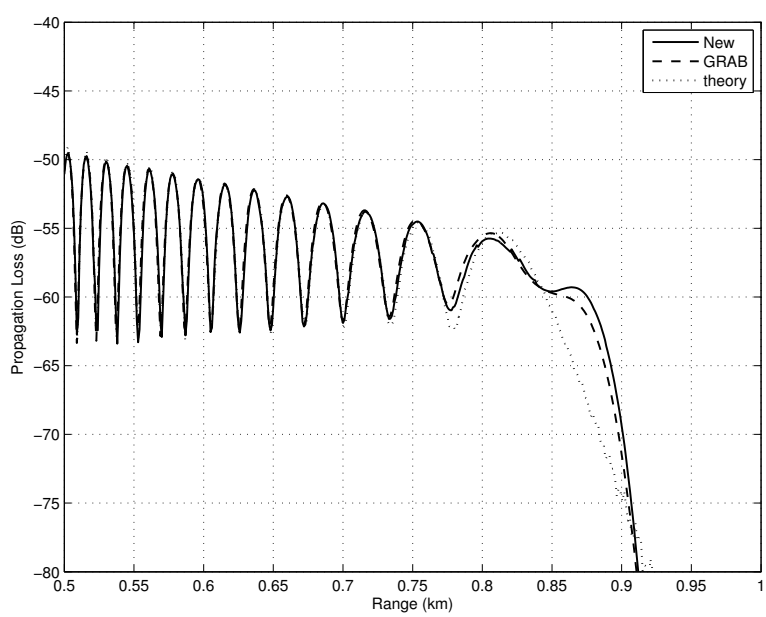

Fig. 8. Model comparisons at edge of shadow zone.

Fig. 8 compares the results of WaveQ3D, GRAB, and a full field calculation using the Fast Field Program (FFP) wavenumber integration model.[16,17] It plots the coherent propagation loss, as a function of range, for a continuous ensonification at 2000 Hz, with the source and receiver both at a depth of 75 meters. Note that, at all ranges, the FFP result is consistent with an ideal wave equation solution except for the presence of some minor implementation jitter in the ranges above 880 m. Prior to the shadow zone, all three models produce similar results. In the region beyond 840 m, the WaveQ3D and GRAB transmission loss values are similar to each other, but slightly higher than FFP. This result, in combination with other tests that we have performend, suggests that our 3–D implementation of GRAB's 2–D Gaussian beam appears to produce equivalent results.

6. Execution Speed Comparisons

The premise of the WaveQ3D approach is that, when target/sensor geometries are constantly evolving, it is more computationally efficient to perform acoustic transmission loss (TL) in the latitude, longitude, altitude coordinates of the underlying environmental databases, than it is to convert the 3–D environments into a series of 2-DxN Cartesian slices. Testing this claim requires a scenario that combines the extraction of data from a 3–D data set and the calculation of transmission loss for a large number of targets. Fortunately, one of the regressions tests that are distributed with GRAB includes just such a combination of features. Although the speeds of individual training systems will differ from this example (because they each extract 2–D radials in a different way) this scenario provides a useful baseline for evaluating the potential of the WaveQ3D approach.

Standard Test 14 was modified to compute transmission loss for a variable number of targets, at a depth of 100 m, and a range 100 km from the source, evenly spaced in azimuth. This test includes a grid of sound speeds and bottom depths in latitude and longitude coordinates for an area near Hawaii, between 16.2N-24.6N and 164.7W-155.4W. The source location was SW of Hawaii at 19.52N 160.5W, at a depth of 200 m. For CASS a radial was constructed for each target using a depression/elevation ray fan of +89 to -89 by 1 degree increments and the range of each target in each radial of 100 km. For WaveQ3D, a wavefront was constructed using depression/elevation angles from -90 to +90 of 181 tangentially spaced rays and an azimuthal spread from 0 to 360 using 25 evenly spaced rays. WaveQ3D propagates 4525 rays through the environment for 80 seconds by 100 millisecond time steps.

Fig. 8 illustrates timing tests of the relative speeds of the models, run on a Dell Latitude Laptop E6520 Intel i5-2540M CPU @2.60GHz, with a variable number of targets from 0 to 100 targets, in increments of 10. The ordinal axis illustrates the time required to compute transmission loss for all targets on this platform. The GRAB model requires approximately 476 ms per target. The measured speed of WaveQ3D is approximately 3.07 seconds plus 42 ms per target. For small numbers of targets, GRAB is faster, because WaveQ3D must propagate the wavefront in all directions for 80 seconds, regardless of the number of targets. GRAB only needs to construct 2–D radials if a target is actually present. However, as the

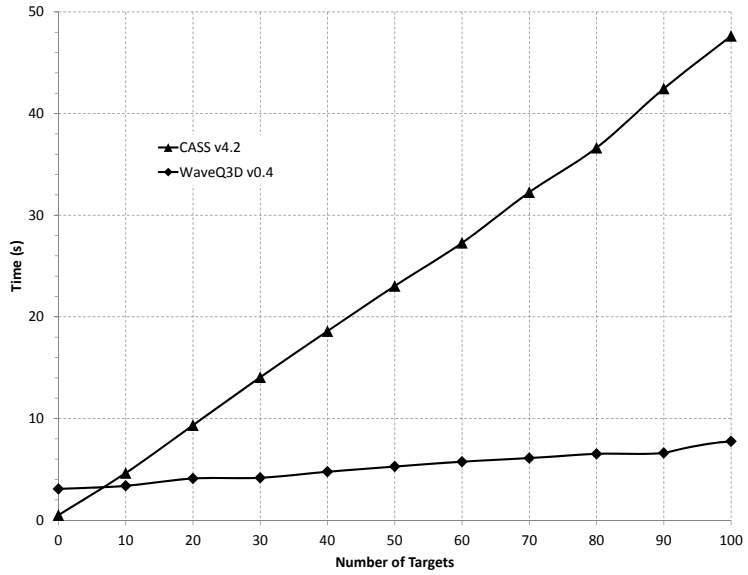

Fig. 9. Speed of transmission loss calculation.

number of targets grows large, WaveQ3D is faster because it has a much less overhead on a per target basis. The crossover point for this scenario appears to be approximately 7 targets. Although WaveQ3D may not be more computationally efficient than other models in all applications, these results illustrate its potential to make a significant speed improvement in active sonar simulation/stimulation systems where the number of targets is large.

7. Availability

WaveQ3D is freely available to the research community as an open-source product. It is distributed as part of the Under Sea Modeling Library (USML), a C++ package that provides the computational framewwork in which is WaveQ3D implemented. Formal releases and test results are distributed through the Ocean Modeling Library, a web site (http://oalib.hlsresearch.com) that is supported by the U.S. Office of Naval Research as a means of publishing software of general use to the international ocean acoustics community. Software developers can also participate directly in the WaveQ3D development process through the UnderSeaModelingLibrary project on GitHub (https://github.com/campreilly/UnderSeaModelingLibrary). Documentation on the application programmer's interface (API) for this software and additional test results are also available from both sources.

Acknowledgments

This paper was developed as part of Sean Reilly's PhD studies at the Ocean Engineering Department of the University of Rhode Island, under the direction of Dr. Gopu Potty and Dr. James Miller. Testing and productization of this model were funded by Michael Vaccaro as part of the High Fidelity Active Sonar Training (HiFAST) Project at the U.S. Office of Naval Research. The authors would also like to thank Dr. Charles Holland (ARL/PS) for his editorial help in preparing this manuscript.

References

1. P. A. Baxley, H. Bucker, M. B. Porter, Comparison of Beam Tracing Algorithms, *Proceedings of the Fifth European Conference on Underwater Acoustics (ECUA)* 2000.
2. H. Weinberg, R. E. Keenan, Gaussian ray bundles for modeling high-frequency propagation loss under shallow-water conditions, *J. Acoust. Soc. Amer.* **100** (1996) 1421.
3. Software Requirements Specification/Software Design Description and Software Test Description for the Oceanographic and Atmospheric Master Library Navy Standard Comprehensive Acoustic System Simulation Model Version 4.2A, *Naval Meteorology and Oceanography Command Report OAML-SRS-SDD-STD-83D* (2008).
4. V. Červený , M. M. Popov, and I. Pšencík, Computation of wave fields in inhomogeneous media - Gaussian beam approach, *Geophys. J. R. Astron. Soc.* **70** (1982) 109.
5. M. B. Porter, Y. C. Liu, Finite Element Ray Tracing, *International Conference on Theoretical and Computational Acoustics (ICTCA)* Volume 2 (1994) 947.
6. M. B. Porter, H. P. Bucker, Gaussian beam tracing for computing ocean acoustic fields, *J. Acoust. Soc. Amer.* **93** (1987) 1349.
7. R. M. Jones, J.P. Riley, T.M. Georges, HARPO: A Versatile Three-Dimensional Hamiltonian Ray-Tracing Program for Acoustics Waves in an Ocean with Irregular Bottom, *National Oceanic and Atmospheric Administration (NOAA) Report*, (1986).
8. Yakowitz and Szidarovszky, An Introduction to Numerical Computation (Macmillan Publishing, New York, 1986) pp. 306-311.
9. W. Press, S. Teukolsky, W. Vetterling, B. Flannery, Numerical Recipes in C (Cambridge University Press, New York, 1992), pp 747-752.
10. ETOPO1v2 Global Gridded 2-minute Database, National Geophysical Data Center, National Oceanic and Atmospheric Administration, U.S. Dept. of Commerce, http://www.ngdc.noaa.gov/mgg/global/ETOPO1.html.
11. F. Sturm, S. Ivansson, Y. M. Jiang, N. R. Chapman, Numerical investigation of out-of-plane sound propagation in a shallow water experiment, *J. Acoust. Soc. Amer.* **124** (2008) pp. 341-346.
12. R. W. Sinnott, Virtues of the Haversine, *Sky and Telescope* **68** (1984) 159.
13. H. Weinberg, R. E. Keenan, Gaussian ray bundles for modeling high-frequency propagation loss under shallow-water conditions, NUWC-NPT Technical Report 10,568 (1996).
14. M. A. Pedersen, D. F. Gordon, Normal-Mode and Ray Theory Applied to Underwater Acoustic conditions of Extreme Downward Refraction, *J. Acoust. Soc. Amer.* **51** (1972) 323.
15. C. L. Pekeris, Accuracy of the Earth-Flattening Approximation in the Theory of Microwave Propagation, *Phys. Rev.* **70** (1943).
16. F. R. DiNapoli, R. L. Deavenport, Theoretical and numerical Green's function field solution in a plane multilayered medium, *J. Acoust. Soc. Amer.* **67** (1980).
17. L. M. Brekhovskikh, Waves in Layered Media (Academic Press Inc., New York, 1980), 2nd Edition, Section 54.

18. F. B. Jensen, W. A. Kuperman, M. B. Porter, and H. Schmidt, Computational Ocean Acoustics (American Institute of Physics Press, New York, 1994) pp. 150-153.
19. WGS 84 Implementation Manual, Version 2.4, *European Organization for the Safety of Air Navigation, and the Institute of Geodesy and Navigation*, (1998).