UMPE:

The University of Miami Parabolic Equation Model

Version 1.1

Kevin B. Smith
Marine Physical Laboratory
Scripps Institution of Oceanography
University of California, San Diego

Frederick D. Tappert
Applied Marine Physics
Rosenstiel School of Marine and Atmospheric Science
University of Miami

MPL Technical Memorandum 432
May 1993
(revised September 1994)

Contents

Solving the acoustic wave equation using the parabolic approximation is a popular approach of many available underwater acoustic models. Here we develop and present a version of the PE model developed at the University of Miami from 1989 to the present under the guidance of Professor Fred Tappert. The model has aptly been named the UMPE model. Fundamentally a research model, the numerical approaches used here may be compared to other PE models and, subsequently, may begin a discussion on the diversity and validity of current PE methods.

This report is based on a collection of published articles by various authors and a compilation of unpublished lecture notes given by Fred Tappert. The motivation here is an attempt to address the fundamental properties of PE modeling and the implementation within the UMPE model. The framework for the algorithms and a description of the source code implementation will also be given. As upgrades to both the code and this text will occur inevitably, we are defining this model as version 1.0. Users are encouraged to contact either author with questions or numerical problems (i.e., bugs).

1. General theory of PE approximations

We begin by representing the time-harmonic acoustic field in a cylindrical coordinate system by

$$P(r, z, \varphi, \omega t) = p(r, z, \varphi) e^{-i\omega t} \quad . \tag{1.1}$$

Substituting this into the wave equation in cylindrical coordinates leads to the Helmholtz equation,

$$\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial p}{\partial r}\right) + \frac{1}{r^2}\frac{\partial^2 p}{\partial \varphi^2} + \frac{\partial^2 p}{\partial z^2} + k_0^2 n^2(r, z, \varphi) p = -4\pi P_0 \delta(\vec{x} - \vec{x}_S) \tag{1.2}$$

where $k_0 = \dfrac{\omega}{c_0}$ is the reference wavenumber, $n(r, z, \varphi) = \dfrac{c_0}{c(r, z, \varphi)}$ is the acoustic index of refraction, c_0 is the reference sound speed, and $c(r, z, \varphi)$ is the acoustic sound speed. It is within $c(r, z, \varphi)$ that all features of the environment are represented. The source function is that of a point source at coordinates $(r = 0, z = z_S)$ with reference source level P_0 defined as the pressure amplitude at a reference distance of $R_0 = 1$ m, and

$$\delta(\hat{x}) = \frac{1}{2\pi r}\delta(z - z_S)\delta(r) \quad . \tag{1.3}$$

Primarily, acoustic energy propagates outward from acoustic sources in the ocean in the horizontal direction. We therefore represent the pressure field by an outgoing Hankel function slowly modulated by an envelope function,

$$p(r, z, \varphi) = \psi(r, z, \varphi)H_0^{(1)}(k_0 r) \quad . \tag{1.4}$$

In the far-field, the Hankel function can be approximated by

$$H_0^{(1)}(k_0 r) \approx \left(\frac{2}{i\pi k_0 r}\right)^{1/2} e^{ik_0 r} \quad , \tag{1.5}$$

thus an alternative relationship between the acoustic pressure, $p(r, z, \varphi)$, and the slowly modulating envelope function, $\psi(r, z, \varphi)$, is

$$p(r, z, \varphi) = P_0\sqrt{\frac{R_0}{r}}\psi(r, z, \varphi)e^{ik_0 r} \quad . \tag{1.6}$$

This is the standard definition of the so-called "PE field function" ψ scaled such that at $r = R_0$, $|\psi| = 1$ and $|p| = P_0$. Substituting (1.6) into (1.2) and dropping the source contribution to the far-field solution yields the defining equation for the evolution of the PE field function,

$$\frac{\partial^2\psi}{\partial r^2} + 2ik_0\frac{\partial\psi}{\partial r} + \frac{1}{r^2}\frac{\partial^2\psi}{\partial\varphi^2} + \frac{\partial^2\psi}{\partial z^2} + k_0^2\left(n^2 - 1 + \frac{1}{4k_0^2 r^2}\right)\psi = 0 \quad . \tag{1.7}$$

Because this is a far-field approximation, we choose to neglect the two terms containing the $\frac{1}{r^2}$ factors. This will automatically invoke the uncoupled azimuth (UNCA) approximation by dropping the $\frac{\partial^2\psi}{\partial\varphi^2}$ term. Note that this is strictly valid only for $\frac{\partial^2\psi}{\partial\varphi^2} \ll \psi$. Should the environmental conditions become such that $\frac{\partial^2\psi}{\partial\varphi^2} \gg \psi$ and is non-negligible with respect to $r^2\frac{\partial\psi}{\partial r}$ then azimuthal coupling should not be ignored. We will return to this topic near the end of this report.

At this point, the only approximations we have applied to the wave equation are a far-field assumption and little or no azimuthal variations. Implicit in the far-field analysis is the small angle approximation, evident from the expression for the uniform ocean Green's function

$$p = \frac{1}{[r^2 + (z - z_S)^2]^{1/2}} e^{ik_0[r^2 + (z - z_S)^2]^{1/2}} \tag{1.8}$$

where z_S is the source depth. For small angles of propagation, $|\theta| \approx \frac{|z - z_S|}{r} \ll 1$, so

$$p \approx \frac{1}{r} e^{ik_0\left[r + \frac{(z - z_S)^2}{2r}\right]} = \frac{1}{\sqrt{r}}\psi e^{ik_0 r} \quad . \tag{1.9}$$

The corresponding form for ψ is found to satisfy a parabolic type equation. The general form of the parabolic approximation to the wave equation simply follows then from the acknowledgment that ψ is a slowly varying function in range and we may neglect the term $\frac{\partial^2\psi}{\partial r^2}$. Thus, Eq. (1.7) takes the form

$$\frac{\partial\psi}{\partial r} = \frac{i}{2k_0}\frac{\partial^2\psi}{\partial z^2} + \frac{ik_0}{2}(n^2 - 1)\psi \quad . \tag{1.10}$$

Eq. (1.10) was first introduced to the underwater acoustics community by Tappert (1974).

We have now reduced a second order differential equation to a first order one, thereby allowing solutions via a non-iterative marching algorithm. Note that we may rewrite Eq. (1.10) as

$$ik_0^{-1}\frac{\partial\psi}{\partial r} = H_{op}\psi = (T_{op} + U_{op})\psi \tag{1.11}$$

where the operators

$$T_{op} = -\frac{1}{2}\left(-ik_0^{-1}\frac{\partial}{\partial z}\right)^2 = -\frac{1}{2k_0^2}\frac{\partial^2}{\partial z^2} \tag{1.12}$$

and

$$U_{op} = U(r, z, \varphi) = -\frac{1}{2}(n^2 - 1) \quad . \tag{1.13}$$

This representation of the operators as kinetic and potential energy operators is especially insightful when one wishes to form the ray equations which have Hamiltonian form (e.g., Smith et al., 1992). In Eq. (1.11), the function ψ is a vector (in z) in Hilbert space. The relationship between the values of ψ at different ranges can now be expressed as

$$\psi(r + \Delta r) = \Phi(r)\psi(r) \quad . \tag{1.14}$$

To propagate the solution out in range requires a representation of the propagator $\Phi(r)$.

There are three common methods of computing PE solutions: (1) the split-step Fourier (PE/SSF) method (Hardin and Tappert, 1973), (2) the implicit finite difference (IFD-PE) method (e.g., Lee and Botseas, 1982), and (3) the finite element (FEPE) method (e.g., Collins, 1988). Since the UMPE model uses the first technique, we shall isolate our discussion to the implementation of the PE/SSF method. This is easily accomplished by approximating the propagator function by

$$\Phi(r) \approx e^{-ik_0 \overline{H}_{op}(r)\Delta r} \tag{1.15}$$

where

$$\overline{H}_{op}(r) = \frac{1}{\Delta r}\int_r^{r+\Delta r} dr' H_{op}(r') \quad . \tag{1.16}$$

The validity of this approximation can be seen from the following argument. Suppose we divide Eq. (1.11) by ψ to obtain

$$\frac{\partial}{\partial r}(ln\psi) = -ik_0 H_{op}(r) \quad . \tag{1.17}$$

Integrating this yields

$$ln\psi(r + \Delta r) = ln\psi(r) - ik_0\int_r^{r+\Delta r} dr' H_{op}(r')$$

or

$$\psi(r + \Delta r) = e^{-ik_0\int_r^{r+\Delta r} dr' H_{op}(r')}\psi(r) \quad . \tag{1.18}$$

This is exactly the equation we developed in (1.14), (1.15), and (1.16). It is not formally valid, however, since H_{op} is an operator and we cannot divide (1.11) by ψ. In other words, it is only an approximation because $\left[\frac{\partial H_{op}}{\partial r}, H_{op}\right] \neq 0$. The formal solution, using a Dyson time evolution operator (Sakurai, 1985), would be

$$\psi(r + \Delta r) = \mathcal{T} e^{-ik_0 \int_r^{r+\Delta r} dr' H_{op}(r')} \psi(r) \tag{1.19}$$

where

$$\mathcal{T} e^{-ik_0 \int_r^{r+\Delta r} dr' H_{op}(r')} = 1 - ik_0 \int_r^{r+\Delta r} dr' H_{op}(r') - k_0^2 \int_r^{r+\Delta r} dr' \int_r^{r'} dr'' H_{op}(r'') H_{op}(r') + (\text{higher order terms}) \quad . \tag{1.20}$$

Thus, from Eq. (1.20), we can evaluate the first order correction to Eq. (1.18).

Assuming (1.18) to be valid, we still must evaluate $\overline{H}_{op}$ defined in Eq. (1.16). Two common approximations are

$$\overline{H}_{op} = H_{op}\left(r + \frac{1}{2}\Delta r\right) \tag{1.21}$$

and simply

$$\overline{H}_{op} = H_{op}(r) \quad . \tag{1.22}$$

These are sometimes referred to as the "centered" and "end point" schemes, respectively. The interpretation of these approximations is that, over the range step r to $r + \Delta r$, the operator (hence the environment) is sampled at either the middle or the beginning of the range step. Presumably, if Δr is small enough the differences between the solutions are negligible. The UMPE model implements Eq. (1.22) and we shall isolate the remainder of our discussion accordingly.

The operator U_{op} is simply a multiplication operator in z-space and, hence, is a diagonal matrix. The operator T_{op} is not diagonal in z-space so different depth eigenfunctions are coupled. In wavenumber space, however, the corresponding operator $\hat{T}_{op}$ is diagonal. It is desirable, therefore, to separate the application of each operator, one in z-space and one in k-space. Using the Baker-Campbell-Hausdorff expansion (Bellman, 1964), we may write

$$e^{A+B} = e^A e^B e^{[A, B] + [A, [A, B]] + [B, [B, A]] + \dots} \tag{1.23}$$

where $A = -ik_0 \Delta r T_{op}$ and $B = -ik_0 \Delta r U_{op}$. Since both T_{op} and U_{op} are small then we assume their products are of second order and negligible. Finally then, we have

$$\Phi(r) = e^{-ik_0 \Delta r T_{op}} e^{-ik_0 \Delta r U_{op}} \quad . \tag{1.24}$$

Note that this separation of H_{op} into two components, each of which is diagonal in some representation and can be applied independently of the other, is presumed by application of the SSF integration scheme. The various approximations used to separate the operator H_{op} are typically used to distinguish one type of PE/SSF model from another.

Note from Eq. (1.24) that if there are no losses present (i.e. $Im T_{op} = Im U_{op} = 0$) then

$$\|\Phi(r)\| = 1 \quad , \tag{1.25}$$

and $\Phi(r)$ is a unitary operator. Therefore, the normalization condition is

$$\|\psi(r)\| = \int_{-\infty}^{\infty} |\psi(r, z)|^2 dz = \text{constant} \quad . \tag{1.26}$$

In other words, because of the formulation of the propagator, the PE/SSF scheme is conservative. There are no intrinsic losses due to the numerical scheme.

The general algorithm behind the PE/SSF implementation is then as follows. The PE field function ψ is specified at some range r in the z-domain. A transformation is made to the k-domain followed by a multiplication of the k-space operator $e^{-ik_0 \Delta r \hat{T}_{op}}$. The result is then transformed again to the z-domain and is followed by a multiplication of the z-space operator $e^{-ik_0 \Delta r U_{op}}$. (The actual order is irrelevant since the commutation of these operators is considered insignificant but this is the order imposed in UMPE.) The final result is the field function at $r + \Delta r$. The FFT subroutine employed in the numerical code assumes the convention

$$\psi(z) = \mathrm{FFT}(\hat{\psi}(k)) \tag{1.27}$$

and

$$\hat{\psi}(k) = [\mathrm{FFT}(\psi^*(z))]^* \quad . \tag{1.28}$$

Therefore, the PE/SSF implementation can be represented by

$$\psi(r + \Delta r, z) = e^{-ik_0 \Delta r U_{op}(r, z)} \times \mathrm{FFT}\{e^{-ik_0 \Delta r \hat{T}_{op}(r, k)} \times [\mathrm{FFT}(\psi^*(r, z))]^*\} \quad , \tag{1.29}$$

where, in k-space,

$$\hat{T}_{op}(k) = \frac{1}{2}\left(\frac{k}{k_0}\right)^2 \quad . \tag{1.30}$$

Previously, we have assumed the operators took the forms defined by Eqs. (1.12) and (1.13). These forms, which followed from the derivation of the parabolic equation (1.10), are commonly referred to as the "standard PE" or SPE forms and are only one set of a number of various operator forms. To obtain other, higher order forms, we return to the original wave equation (1.2). Still ignoring the source term and the azimuthal coupling term, we now define the pressure field as

$$p(r, z) = \frac{1}{\sqrt{r}} u(r, z) \tag{1.31}$$

The function $u(r, z)$ is identical to the pressure field in two dimensions and the term $\frac{1}{\sqrt{r}}$ accounts for azimuthal spreading. Substituting (1.31) into the Helmholtz equation in two dimensions yields the far-field UNCA expression

$$\frac{\partial^2 u}{\partial r^2} + \frac{\partial^2 u}{\partial z^2} + k_0^2 n^2(r, z) u = 0 \quad . \tag{1.32}$$

We introduce the operators

$$P_{op} = \frac{\partial}{\partial r} \tag{1.33}$$

and

$$Q_{op} = \left(n^2 + \frac{1}{k_0^2}\frac{\partial^2}{\partial z^2}\right)^{1/2} \quad . \tag{1.34}$$

Eq. (1.32) then becomes

$$(P_{op}^2 + k_0^2 Q_{op}^2)u = 0 \tag{1.35}$$

which can be factored as

$$(P_{op} + ik_0 Q_{op})(P_{op} - ik_0 Q_{op})u + ik_0[P_{op}, Q_{op}]u = 0 \quad . \tag{1.36}$$

The commutator $[P_{op}, Q_{op}]$ is assumed negligible and is, in fact, exactly zero in layered media. Eq. (1.36) therefore represents the combination of incoming and outgoing waves. The outgoing wave satisfies

$$P_{op}u = ik_0 Q_{op}u \tag{1.37}$$

or

$$-ik_0^{-1}\frac{\partial u}{\partial r} = Q_{op}u \quad . \tag{1.38}$$

Eq. (1.38) is observed to have a form similar to Eq. (1.11). The trick is now to develop an approximation for Q_{op} that separates into a *z*-space operator and a *k*-space operator.

The UMPE model allows the user to choose from five different Q_{op} approximations which we shall now derive. We begin by introducing the notation

$$\varepsilon = n^2 - 1 \tag{1.39}$$

and

$$\mu = \frac{1}{k_0^2}\frac{\partial^2}{\partial z^2} \tag{1.40}$$

so

$$Q_{op} = (\mu + \varepsilon + 1)^{1/2} \quad . \tag{1.41}$$

The first approximation follows from the assumption that both ε and μ are small compared to unity. A binomial expansion then yields

$$Q_{op} \approx Q_1 = \frac{1}{2}\mu + \frac{1}{2}\varepsilon + 1 \quad . \tag{1.42}$$

Substitution of (1.39) and (1.40) shows that this is, within an additive constant, the standard PE operator,

$$Q_1 = Q_{SPE} = \frac{1}{2k_0^2}\frac{\partial^2}{\partial z^2} + \frac{1}{2}(n^2 - 1) + 1 \quad . \tag{1.43}$$

Interpreting the operator μ as $\frac{k^2}{k_0^2} = \sin^2\theta$ indicates that the assumption $\mu \ll 1$ implies $\sin^2\theta \approx \theta^2 \ll 1$, hence this is a small angle approximation. The condition $\varepsilon \ll 1$ is simply interpreted as assuming primarily stratified media as is typical of most ocean regions. In fact, since $\varepsilon = (n^2 - 1) \ll 1$, we can further approximate this by $\varepsilon \approx 2(n-1)$ to obtain

$$Q_2 = \frac{1}{2k_0^2}\frac{\partial^2}{\partial z^2} + (n-1) + 1 \quad . \tag{1.44}$$

Both of these approximations were first recognized by Tappert (1977) in his original Springer-Verlag article.

To obtain the customary formula for the PE field function ψ, we first give it the usual envelope definition

$$u(r, z) = \psi(r, z)e^{ik_0 r} \quad . \tag{1.45}$$

Substituting this into (1.38) yields

$$\frac{\partial\psi}{\partial r} = -ik_0\psi + ik_0 Q_{op}\psi \quad . \tag{1.46}$$

Application of $Q_{op} = Q_{SPE}$ produces exactly Eq. (1.10). Comparing (1.46) with (1.11) indicates the simple relationship

$$Q_{op} = 1 - H_{op} = 1 - (T_{op} + U_{op}) \tag{1.47}$$

which easily allows a split-step Fourier implementation. Equivalently then, our goal is to develop expressions for T_{op} and U_{op} based on approximations of Q_{op}.

A better approximation to Q_{op}, also introduced by Tappert (1977), assumes that only the sound speed variations are small, i.e. $\varepsilon \ll 1$. We may then write

$$Q_3 = (1+\mu)^{1/2} + \frac{1}{2}\varepsilon \quad . \tag{1.48}$$

The restriction on propagation angles has now been removed. In fact, for homogeneous media $(\varepsilon = 0)$, Eq. (1.48) is observed to be an exact representation of the original Helmholtz operator.

A higher order approximation introduced by Thomson and Chapman (1983) is actually a combination of Eqs. (1.44) and (1.48). It is based on an operator splitting by Feit and Fleck (1978) which is formally valid only when ε and μ commute. Commonly referred to as the "wide-angle" approximation (WAPE), it has the form

$$Q_4 = Q_{WAPE} = (1+\mu)^{1/2} + [(1+\varepsilon)^{1/2} - 1] \quad . \tag{1.49}$$

Invoking the operator identity

$$(1+\mu)^{1/2} = 1 + \mu[(1+\mu)^{1/2} + 1]^{-1} \tag{1.50}$$

and formulating Eq. (1.49) in terms of T_{op} and U_{op} as in Eq. (1.47) leads to

$$T_{WAPE} = -\frac{1}{k_0^2}\frac{\partial^2}{\partial z^2}\left[\left(1 + \frac{1}{k_0^2}\frac{\partial^2}{\partial z^2}\right)^{1/2} + 1\right]^{-1} \tag{1.51}$$

and

$$U_{WAPE} = -(n-1) \quad . \tag{1.52}$$

In wavenumber space, we may express (1.51) as

$$\hat{T}_{WAPE}(k) = \left(\frac{k}{k_0}\right)^2\left[\left(1 - \left(\frac{k}{k_0}\right)^2\right)^{1/2} + 1\right]^{-1} = 1 - \left[1 - \left(\frac{k}{k_0}\right)^2\right]^{1/2} \quad . \tag{1.53}$$

Note that modes with $k > k_0$ are evanescent since

$$\hat{T}_{WAPE}(k > k_0) = 1 - i\left[\left(\frac{k}{k_0}\right)^2 - 1\right]^{1/2} \quad . \tag{1.53'}$$

The final approximation we shall consider was introduced by Berman et al. (1989) and is referred to as the "modified LOGPE" or simply LOGPE. It follows from assuming the potential energy function U_{op} has the form

$$U_{LOGPE} = -ln(n) \quad . \tag{1.54}$$

Comparing the ray equations derived from the Helmholtz equation with the general formulation of the ray equations in terms of T_{op} and U_{op} leads to the definition of the kinetic energy term consistent with (1.54)

$$T_{LOGPE} = -ln\left[\cos\left(-\frac{i}{k_0}\frac{\partial}{\partial z}\right)\right] \quad . \tag{1.55}$$

In the wavenumber domain, this becomes simply

$$\hat{T}_{LOGPE}(k) = -ln\left[\cos\left(\frac{k}{k_0}\right)\right] \quad . \tag{1.55'}$$

We note that for small propagation angles $\left(\frac{k}{k_0} \ll 1\right)$,

$$T_{LOGPE} \approx -\frac{1}{2k_0^2}\frac{\partial^2}{\partial z^2} = T_{SPE} \tag{1.56}$$

while for nearly uniform media $(n \approx 1)$,

$$U_{LOGPE} \approx -\frac{1}{2}(n^2 - 1) = U_{SPE} \quad . \tag{1.57}$$

Thus, LOGPE reduces to SPE in situations where the latter approximation is valid. Unfortunately, the operator $\hat{T}_{LOGPE}(k)$ is undefined for $\frac{k}{k_0} \geq \frac{\pi}{2}$. To avoid this, a sine-squared taper function is applied over the outer $\frac{1}{8}$ range of $\frac{7\pi}{16}$ to $\frac{\pi}{2}$. For $\frac{k}{k_0} > \frac{\pi}{2}$, this function is set to zero.

Of the five PE approximations described so far, the last two are expected to yield the most accurate results. The most common PE implementation currently is the WAPE, and this is the version used in the Navy standard model (Holmes and Gainey, 1991). However, the WAPE and other so-called "higher-order" models are still not exact and may occasionally produce results that are worse than predicted by the SPE. The most famous example of this, sometimes referred to as the Porter duct problem, was defined in Test Case 7 of the PE Workshop II (Chin-Bing et al., 1993). In such instances, it is usually found that the error results from extra-sensitivity to the choice of reference sound speed. In fact, the choice of c_0 is the one ambiguous feature of all PE models, and till now we have ignored this effect. Later in this report, a scheme for computing a default value for c_0 will be discussed. However, modelers are hereby forewarned that no method is foolproof. The best approach is to vary c_0 and look for fluctuations in the calculations. Currently, the UMPE model only takes user input c_0 values.

Because of the ever present ambiguity in the selection of c_0, it is highly desirable to develop a model that is c_0-insensitive such that significant changes in the choice of c_0 will not affect the final result. Tappert (1991b) developed a rigorous definition of such a PE model. Application of the resulting code was found to eliminate the sensitivity in the Porter duct problem. However, implementation was complicated and required a transformation of the function $\psi(z)$ to a new function $\tilde{\psi}(\tilde{z})$ in a transformed space. In a range-dependent environment, such a transformation would be required at each range step thereby greatly increasing the run time. Since one of the main advantages of the PE/SSF code is the speed with which the acoustic field can be computed, a c_0-insensitive version has not been implemented in the UMPE code. Again, the user is reminded to beware of these highly sensitive (but uncommon) problems.

Two other ambiguous variables must also be introduced in the numerical implementation. As in all models, a discretization of the environment is required and defined by the mesh size $(\Delta r, \Delta z)$. Because the depth mesh influences the wavenumber increments Δk, we may define a default value for Δz, hence the transform size N, by considering a lower limit on allowable angles of propagation. Since N wavenumber values will be spread over the range $+k_{max}$ to $-k_{max}$, it follows that

$$k_{max} = \frac{N}{2}\Delta k \ , \ \ \Delta k = \frac{2\pi}{z_{\mathrm{T}}} \ , \qquad (1.58)$$

where z_T is the total computational depth, so

$$k_{max} = \frac{N\pi}{z_T} \quad . \tag{1.59}$$

Furthermore, the wavenumbers are related to the angles of propagation by

$$k = k_0 \sin\theta \quad . \tag{1.60}$$

It follows that for a given maximum angle of propagation, the minimum transform size required must satisfy

$$N_{min} \geq \frac{k_0 z_T}{\pi} \sin\theta_{max} \quad . \tag{1.61}$$

To define an upper bound on the range step size, Δr_{max}, we consider the analogy of physical optics. In the vicinity of a focus, the signal will vary significantly over a horizontal range of

$$\Delta r \approx \frac{2\mathrm{f}^2}{k_0} \tag{1.62}$$

where f is the f-number of the focusing lens,

$$\mathrm{f} = \frac{\mathrm{R}}{2\mathrm{B}} , \tag{1.63}$$

R is the focal length and 2B is the effective aperture. In the underwater acoustics problem, the focal length can be represented by the distance between convergence zones, CZ, and the effective aperture is roughly the depth of the ocean or approximately $\frac{1}{2} z_T$ (the factor of $\frac{1}{2}$ for the depth will be explained later). Then (1.63) becomes

$$\mathrm{f} = \frac{\mathrm{CZ}}{z_T} \sim \frac{1}{\sin\theta} \tag{1.64}$$

where θ is the angle of propagation. Combining (1.62) and (1.64) yields the upper limit on the range step size

$$\Delta r_{max} \leq \frac{2}{k_0 \sin^2\theta_{max}} \quad . \tag{1.65}$$

A similar analysis suggests that the maximum vertical mesh size is given by

$$\Delta z_{max} \le \frac{2\mathrm{f}}{k_0} \sim \frac{2}{k_0 \sin\theta_{max}} \tag{1.66}$$

which can be shown to yield roughly the same order for the transform size as Eq. (1.61).

From the above analysis, it is obvious that if a particular problem is known to contain only small angle propagation, the mesh size $(\Delta r, \Delta z)$ may be increased and, subsequently, the run-time will be reduced. Conversely, for problems where very large angle propagation is expected to be important a small mesh size may be required. If the user is unsure, zero values may be input for the depth transform size and the range step size which will prompt the model to use the default values defined by Eqs. (1.61) and (1.65). A value for θ_{max} is then defined by the input source angular width or $30°$, whichever is greater.

We will now review the main results of this section. The UMPE model can solve five different PE type approximations to the wave equation. The general form of each equation is

$$\frac{\partial\psi}{\partial r} = -ik_0(T_{op} + U_{op})\psi$$

and the acoustic pressure is then defined by

$$p = P_0\sqrt{\frac{R_0}{r}}\psi e^{ik_0 r} \quad .$$

The numerical algorithm employed is the split-step Fourier, or SSF, such that the solution for ψ is marched out in range according to

$$\psi(z, r + \Delta r) = e^{-ik_0\Delta r U_{op}(z, r)} \times \mathrm{FFT}\{e^{-ik_0\Delta r \hat{T}_{op}(k, r)} \times [\mathrm{FFT}(\psi^*(z, r))]^*\} \quad ,$$

where, by definition,

$$\psi(z) = \mathrm{FFT}(\hat{\psi}(k))$$

and

$$\hat{\psi}(k) = [\mathrm{FFT}(\psi^*(z))]^* \quad .$$

The five types of PE approximations and their corresponding operators are

1) SPE:

$$\hat{T}_{SPE}(k) = \frac{1}{2}\left(\frac{k}{k_0}\right)^2 ,$$

$$U_{SPE}(z) = -\frac{1}{2}[n^2(z) - 1] ;$$

2) Variant SPE:

$$\hat{T}_{VSPE}(k) = \frac{1}{2}\left(\frac{k}{k_0}\right)^2 ,$$

$$U_{VSPE}(z) = -[n(z) - 1] ;$$

3) Variant WAPE:

$$\hat{T}_{VWAPE}(k) = 1 - \left[1 - \left(\frac{k}{k_0}\right)^2\right]^{1/2} ,$$

$$U_{VWAPE}(z) = -\frac{1}{2}[n^2(z) - 1] ;$$

4) WAPE:

$$\hat{T}_{WAPE}(k) = 1 - \left[1 - \left(\frac{k}{k_0}\right)^2\right]^{1/2} ,$$

$$U_{WAPE}(z) = -[n(z) - 1] ;$$

5) LOGPE:

$$\hat{T}_{LOGPE}(k) = -ln\left[\cos\left(\frac{k}{k_0}\right)\right] ,$$

$$U_{LOGPE}(z) = -ln[n(z)] .$$

2. Boundary conditions

The only parameter introduced in the last section for any of the PE approximations was the reference sound speed c_0. In a sense, this parameter is related to the accuracy of the “small angle” approximation since it will serve to scale the relative wavenumber values $\frac{k}{k_0}$. The best choice of c_0 should then produce the most accurate results for a given problem. However, in the numerical implementation of the PE, additional parameters must be introduced to define the discretely sampled environment in a way that also yields the best results. This is primarily a concern in regions where boundaries exist. The representation of these boundaries in a PE/SSF implementation is the subject of this section.

2.1 Water/bottom interface

The UMPE model treats the bottom as a fluid of contrasting sound speed and density from that of water. In addition, the UMPE model allows for an additional bottom layer to exist on top of the basement to allow for effects of sediment layers to be included. Within either bottom volume, the PE environmental potential function, $U_{op}(z)$, is defined as before in terms of the local acoustic index of refraction, $n(z) = \frac{c_0}{c(z)}$, where $c(z)$ now includes the sound speed within the bottom volume. The effect of approximating the bottom as a fluid is the neglect of shear wave propagation. When the true bottom does support shear waves, the conversion of compressional energy incident on the interface into downward propagating shear energy is treated as a loss. In this manner, the bottom properties are replaced by equivalent fluid properties that produce the correct reflection from the interface. The treatment of shear as a loss mechanism is discussed further in a following section.

2.1.1 Sound speed discontinuity

We assume the interface between the bottom of the water column and the top of the basement, or sediment, layer is characterized by a sharp contrast in sound speed. In a numerical code with finite sampling and recurrent use of FFT's, it is desirable to use smoothly varying, continuous functions to avoid artificial reflections, aliasing, and noise from entering into the calculation. Therefore, we seek to find a smooth, continuous function of variable scale which can accurately reproduce the physical effects of a discontinuous jump in sound speed at the water/bottom interface.

We write the interface condition for the sound speed as

$$c(z) = c_w(z) + [c_b(z) - c_w(z)]H(z - z_b) \tag{2.1}$$

where we shall assume that the sound speed above the interface at $z = z_b$ has a constant value of c_w and below the interface has a value c_b. The Heaviside step function is defined by

$$H(\zeta) = \begin{cases} 0, & \zeta < 0 \\ \frac{1}{2}, & \zeta = 0 \\ 1, & \zeta > 0 \end{cases} \tag{2.2}$$

where $\zeta \equiv z - z_b$. From the theory of generalized functions (Lighthill, 1958), we may replace $H(\zeta)$ by any smooth function within a class of generalized functions that produces the same overall effect (i.e., has similar moments).

One such function satisfying the above criteria involves the hyperbolic tangent function,

$$\overline{H}(\zeta) = \frac{1}{2}\left[1 + \tanh\left(\frac{\zeta}{2L_c}\right)\right] \tag{2.3}$$

or, equivalently,

$$\overline{H}(\zeta) = (1 + e^{-\zeta/L_c})^{-1} \quad . \tag{2.3'}$$

This function has the properties

$$\overline{H}(\zeta) \to 0, \quad \zeta \ll 0; \tag{2.4a}$$

$$\overline{H}(\zeta) = \frac{1}{2}, \quad \zeta = 0; \tag{2.4b}$$

and

$$\overline{H}(\zeta) \to 1, \quad \zeta \gg 0 \,. \tag{2.4c}$$

Furthermore, the derivative of the Heaviside function is

$$H'(\zeta) = \delta(\zeta) \tag{2.5}$$

where $\delta(\zeta)$ is the Dirac-delta function and is characterized by

$$\int_{-\infty}^{\infty} \delta(\zeta) d\zeta = 1 \ . \tag{2.6}$$

Similarly, the derivative of $\bar{H}(\zeta)$ is

$$\bar{H}'(\zeta) = \bar{\delta}(\zeta) = \frac{1}{4L_c} \mathrm{sech}^2\left(\frac{\zeta}{2L_c}\right) \tag{2.7}$$

and it is easy to show that

$$\int_{-\infty}^{\infty} \bar{\delta}(\zeta) d\zeta = 1 \tag{2.8}$$

is also satisfied. This mixing function $\bar{H}(\zeta)$ is parameterized by a characteristic mixing length, L_c. It is obvious from the above analysis that

$$\lim_{L_c \to 0} \bar{H}(\zeta) = H(\zeta) \tag{2.9}$$

and

$$\lim_{L_c \to 0} \bar{\delta}(\zeta) = \delta(\zeta) \ . \tag{2.9'}$$

This limiting equality can be shown to hold for higher derivatives as well.

We have now introduced an additional parameter into our model, the sound speed mixing length, L_c. This can be adjusted by the user in an attempt to create the most realistic interface condition for reflections from a sound speed discontinuity. The UMPE code then employs Eq. (2.3') to mix the sound speed profiles above and below the interface (or interfaces) at $z = z_b$. Experience has shown that the most accurate results are gained by defining L_c as a fraction of an acoustic wavelength. Experience has also shown that the sampling in depth Δz need not necessarily be less than a wavelength in which case $L_c \sim \lambda/5$ would result in a sharp jump from c_w to c_b over one mesh point. Ironically, this is what we intended to avoid. The interpretation of this apparent contradiction is that a discontinuous jump in sound speed, hence a jump in the PE potential function $U_{op}(z)$, does not introduce much error and, in fact, yields the best representation of the reflection condition. A default minimum value for L_c has therefore been set at $L_{c_{\min}} = \frac{\Delta z}{10}$. Alternatively, and perhaps more accurately, the index of refraction could be smoothed over the interface. Experience has again shown that this produces little difference in the final result.

2.1.2 Density discontinuity

The effect of density on acoustic propagation has not yet been considered. In fact, the variation of density was ignored in the original form of the Helmholtz equation. In a fluid with a variable density ρ, the correct form of the Helmholtz equation is

$$\rho \nabla \bullet \left(\frac{1}{\rho} \nabla p \right) + k_0^2 n^2 p = 0 \quad . \tag{2.10}$$

We make the substitution

$$q = \sqrt{\frac{\rho_0}{\rho}} p \tag{2.11}$$

to obtain the equation defining the propagation of q,

$$\nabla^2 q + k_0^2 n'^2 q = 0 \tag{2.12}$$

where n' is the "effective" index of refraction given by

$$n'^2 = n^2 + \frac{1}{2k_0^2} \left[\frac{1}{\rho} \nabla^2 \rho - \frac{3}{2} \left(\frac{1}{\rho} \nabla \rho \right)^2 \right] \quad . \tag{2.13}$$

Therefore, we may solve for the pressure field $p(r,z)$, now defined by

$$p(r, z) = P_0 \sqrt{\frac{\rho R_0}{\rho_0 r}} \psi(r, z) e^{i k_0 r} \ , \tag{2.14}$$

by marching the solution of the PE function $\psi(r,z)$ out in range with the definition

$$U_{op}(z) = U_1(z) + U_2(z) \tag{2.15}$$

where $U_1(z)$ is the same environmental potential function previously defined and $U_2(z)$ accounts for the effect of the density discontinuity. For example, if the SPE approximation is employed, one can easily show that

$$U_2(z) = -\frac{1}{4k_0^2}\left[\frac{1}{\rho}\frac{\partial^2\rho}{\partial z^2} - \frac{3}{2}\left(\frac{1}{\rho}\frac{\partial\rho}{\partial z}\right)^2\right] \tag{2.16}$$

where we have assumed $\rho = \rho(z)$ only.

The reference density is defined as that of pure water, $\rho_0 = 1\ \text{g/cm}^3$, and inputs to the model for density are given as ratios, i.e. $\frac{\rho}{\rho_0}$. The UMPE model assumes each volume (water, sediment, basement) is characterized by a constant density profile. For example, with a single water/bottom interface,

$$\rho(z) = \rho_w + (\rho_b - \rho_w)H(z - z_b) \tag{2.17}$$

where $H(\zeta)$ is the Heaviside step function described previously. Obviously, the function $U_2(z)$ is non-zero only in the vicinity of the interface. As before, we wish to spread this discontinuity over some finite region in terms of smooth generalized functions. This is a more critical problem than before because $U_2(z)$ depends not on the density but on the derivatives of the density. In most current versions of PE/SSF, use is made of Eq. (2.16) to define the density potential function and the hyperbolic mixing function Eq. (2.3'), with its subsequent derivatives, to define the interface condition. However, the choice of the density mixing length L_ρ has always been somewhat ambiguous in this definition. If L_ρ is chosen too large or too small, regardless of how well the function is sampled in space, the result will not be correct. Recently, Tappert (1991a) has suggested that a better form for this function is

$$U_2(z) \approx -\frac{\varepsilon}{k_0^2}\frac{\partial^2}{\partial z^2}H(z - z_b) \tag{2.18}$$

where

$$\varepsilon = \left[\frac{1-(\rho_w/\rho_b)^{1/2}}{1+(\rho_w/\rho_b)^{1/2}}\right] . \tag{2.19}$$

For small density contrasts, this is equivalent to neglecting the second term in Eq. (2.16). The main argument used to justify this approximation is that because we are using generalized functions to represent the density discontinuity, the function $U_2(z)$ must also be defined in terms of generalized functions. However, the second term in (2.16) contains the square of a generalized function, specifically $\bar{\delta}^2(z')$, which is not a generalized function. A detailed analysis by Tappert (1991d) which attempted to remove as much singularity as possible from the solution in the vicinity of the density discontinuity showed that Eq. (2.18) is the best expression to use in the SPE approximation. Furthermore, this formulation inherently produces the best results when L_ρ is minimized, i.e. as $L_\rho \to 0$. Note that L_ρ must still be large enough such that the finite depth mesh adequately samples the function $U_2(z)$ (which is not a simple jump as U_1 was). We shall assume (2.18) can be applied for any of the PE approximations used. The main justification for this is the expectation that most bottom interacting energy will be near grazing, hence small angle reflection dominates at long ranges.

We now examine two forms of mixing function available in the UMPE model to compute the density potential function $U_2(z)$:

a) The first approximation of the Heaviside step function is the same hyperbolic tangent function used to mix the sound speeds across the interface, i.e.

$$\bar{H}(\zeta) = (1+e^{-\zeta/L_\rho})^{-1} . \tag{2.20}$$

The first derivative is

$$\bar{H}'(\zeta) = \bar{\delta}(\zeta) = \frac{1}{L_\rho}\frac{e^{-\zeta/L_\rho}}{(1+e^{-\zeta/L_\rho})^2} \tag{2.21}$$

and the second derivative is

$$\bar{H}''(\zeta) = \bar{\delta}'(\zeta) = -\frac{1}{L_\rho^2}\frac{e^{-\zeta/L_\rho}(1-e^{-\zeta/L_\rho})}{(1+e^{-\zeta/L_\rho})^3} \quad . \tag{2.22}$$

Substituting this into the density potential function (2.18) yields one of the formulas used in the UMPE model to compute the effect of a density discontinuity.

The main advantage of the hyperbolic smoothing function is that it is a C_∞ smooth function, i.e. all derivatives are continuous everywhere. The main disadvantage of this type of smoothing function is that it has "infinite" extent. Although it may quickly approach a negligible value (even zero within computer precision) away from the interface, it may still extend into undesirable regions.

One can easily show that the extrema of $\bar{\delta}'(\zeta)$ occur at $\zeta_{ex} = L_\rho \ln(2 \pm \sqrt{3}) \approx \pm 1.32 L_\rho$. At these values, $\bar{\delta}'(\zeta_{ex}) \approx \pm \frac{0.096}{L_\rho^2}$. Because it is presumed that $U_2(z)$ yields the most accurate representation for small L_ρ, for a given mesh size Δz it follows that the optimum choice of mixing length for the hyperbolic tangent function is approximately $\frac{\Delta z}{1.32}$. This produces a function with extrema exactly one depth mesh above and below the interface. For this to be properly sampled, however, the interface must also lie exactly on a mesh point. To accommodate these restrictions, a minimum value of $L_{\rho_{min}} = \frac{\Delta z}{1.32}$ is defined. When the user input mixing length is equal to or less than this, the potential function $U_2(z)$ is centered on the mesh point nearest the true interface depth. In this manner, a default condition is set if the user inputs $L_\rho = 0$. While this should produce the most accurate representation of $U_2(z)$, it may introduce an error in the depth of the interface by as much as $\frac{1}{2}\Delta z$. This error can be minimized and convergence tested by decreasing Δz (i.e., increasing the transform size).

Experience suggests that the most accurate results are achieved by requiring $L_\rho \le \lambda$. Consequently, this implies a depth mesh limit of $\Delta z_{\max} \approx 1.32\lambda \sim \lambda$. This condition may be relaxed when no density discontinuity exists. The user is encouraged to experiment with different mixing lengths and compare results with other models.

b) The second form of mixing function used completely localizes the extent to within a finite distance from the interface, but is no longer C_∞ smooth. Since we must be able to compute the second derivative, it is required to be at least C_2 smooth. Specifically, we choose a cubic spline over the finite interval $-L_\rho \le \zeta \le L_\rho$.

In designing a cubic spline approximation for $H(\zeta)$, we have used four sub-intervals of length $L_\rho/2$. Requiring continuity of the function and its first and second derivatives, we define

$$\bar{H}(\zeta) = \begin{cases} 0 & , \quad \zeta \le L_\rho \\ \frac{2}{3}\left(1 + \frac{\zeta}{L_\rho}\right)^3 & , \quad -L_\rho \le \zeta \le -\frac{L_\rho}{2} \\ \frac{1}{2} + \frac{\zeta}{L_\rho} - \frac{2}{3}\left(\frac{\zeta}{L_\rho}\right)^3 & , \quad -\frac{L_\rho}{2} \le \zeta \le \frac{L_\rho}{2} \\ 1 - \frac{2}{3}\left(1 - \frac{\zeta}{L_\rho}\right)^3 & , \quad \frac{L_\rho}{2} \le \zeta \le L_\rho \\ 1 & , \quad \zeta \ge L_\rho \end{cases} \tag{2.23}$$

The first derivative of this function is then

$$\bar{H}'(\zeta) = \bar{\delta}(\zeta) = \begin{cases} 0 & , \quad \zeta \le L_\rho \\ \frac{2}{L_\rho}\left(1 + \frac{\zeta}{L_\rho}\right)^2 & , \quad -L_\rho \le \zeta \le -\frac{L_\rho}{2} \\ \frac{1}{L_\rho}\left[1 - 2\left(\frac{\zeta}{L_\rho}\right)^2\right] & , \quad -\frac{L_\rho}{2} \le \zeta \le \frac{L_\rho}{2} \\ \frac{2}{L_\rho}\left(1 - \frac{\zeta}{L_\rho}\right)^2 & , \quad \frac{L_\rho}{2} \le \zeta \le L_\rho \\ 0 & , \quad \zeta \ge L_\rho \end{cases} \tag{2.24}$$

Note that $\bar{\delta}\left(-\frac{L_\rho}{2}\right) = \bar{\delta}\left(\frac{L_\rho}{2}\right) = \frac{1}{2L_\rho}$ and $\bar{\delta}(0) = \frac{1}{L_\rho}$, so it is obvious that

$$\int_{-\infty}^{\infty} \bar{\delta}(\zeta) d\zeta = 1 \tag{2.25}$$

and, therefore,

$$\lim_{L_\rho \to 0} \bar{\delta}(\zeta) = \delta(\zeta) \tag{2.26}$$

as required.

Finally, the second derivative is

$$\bar{H}''(\zeta) = \bar{\delta}'(\zeta) = \begin{cases} 0 & , \quad \zeta \le L_\rho \\ \frac{4}{L_\rho^2}\left(1 + \frac{\zeta}{L_\rho}\right) & , \quad -L_\rho \le \zeta \le -\frac{L_\rho}{2} \\ -\frac{4}{L_\rho^2}\left(\frac{\zeta}{L_\rho}\right) & , \quad -\frac{L_\rho}{2} \le \zeta \le \frac{L_\rho}{2} \\ -\frac{4}{L_\rho^2}\left(1 - \frac{\zeta}{L_\rho}\right) & , \quad \frac{L_\rho}{2} \le \zeta \le L_\rho \\ 0 & , \quad \zeta \ge L_\rho \end{cases} \tag{2.27}$$

Combining (2.27) with (2.18) gives the formula for computing the density potential function by employing a cubic spline polynomial smoothing function.

This function obviously has extrema at $\zeta_{ex} = \pm\frac{L_\rho}{2}$ given by $\bar{\delta}'(\zeta_{ex}) = \pm\frac{2}{L_\rho^2}$. For the same value of mixing length, the cubic spline polynomial mixing function is found to produce a representation of $\bar{\delta}'(z)$ which is narrower in depth and has a larger amplitude than the hyperbolic tangent mixing function. Furthermore, one can easily show that by defining this mixing length in such a way that the spacing of the extrema for each function agree, i.e., $L_\rho^{(poly)} = (2 \times 1.32)L_\rho^{(tanh)} = 2.64L_\rho$, the values of the extrema are $\bar{\delta}'^{(poly)}_{ex} \approx \pm\frac{0.287}{L_\rho^2} \approx 3\bar{\delta}'^{(tanh)}_{ex}$.

Thus, for the same width between extrema, the hyperbolic tangent mixing function produces a $\bar{\delta}'(z)$ representation with smaller extrema than the cubic spline polynomial mixing function. This is a consequence of the infinite extent of the former versus the finite extent of the latter coupled with the normalization condition (2.8).

As before, we define a lower limit default of $L_{\rho_{\min}} = 2\Delta z$, and when this limit is reached the mixing function is centered on the mesh nearest the true interface depth. Applying the condition $L_\rho \leq \lambda$ yields the condition $\Delta z_{\max} \sim \frac{\lambda}{2}$. This is a stronger condition than before and suggests that twice the former transform size is needed to utilize the polynomial mixing function. However, because of the finite nature of this representation, this condition may be relaxed slightly. A good rule of thumb for either mixing function is $\Delta z_{\max} \sim \lambda$ when density discontinuities are important for computing the correct reflection from the interface.

Finally, let us return for a moment to the original expression for $U_2(z)$ given by (2.16). If we ignore the singular nature of the second term and assume that the generalized functions introduced are valid, we may attempt to investigate the validity of the approximate form (2.18) by examining the relative magnitudes of the two terms in (2.16). When the hyperbolic tangent function is employed, the maximum value of the first term, up to a constant factor, has been shown to be approximately $(0.096)\frac{(\rho_b - \rho_w)}{L_\rho^2}$. Up to the same factor, the second term has a maximum value of $\frac{3}{16}\frac{(\rho_b - \rho_w)^2}{(\rho_b + \rho_w)L_\rho{}^2}$. One can then easily show that the maximum value of the second term is greater than the maximum value of the first term only when $\rho_b > 3\rho_w$. For smaller density discontinuities, this analysis suggests the approximate form for $U_2(z)$ should be adequate. Similarly, when the cubic spline polynomial is used, the first term has been shown to have a maximum of

$2\frac{(\rho_b - \rho_w)}{L_\rho^2}$ while the second term has a maximum of $\frac{3}{2}\frac{(\rho_b - \rho_w)^2}{(\rho_b + \rho_w)L_\rho^{\ 2}}$. A similar analysis shows that the second term exceeds the first term only for $\rho_b > 5\rho_w$. This implies a further improvement in accuracy when the polynomial representation of the density discontinuity is selected instead of the hyperbolic tangent.

2.2 Surface interface

The UMPE model treats the surface as a perfect reflector due to a pressure release boundary. This is a Dirichlet boundary condition defined by

$$\psi(z = 0) = 0 \ . \tag{2.28}$$

A popular technique used in PE/SSF models to achieve this is the image ocean method developed by Tappert (1977). With this method, we assume an identical image ocean overlays the real ocean for negative values of depth and, furthermore, the acoustic field is exactly equal but of opposite sign in the image ocean, i.e.

$$\psi(-z) = -\psi(z) \ . \tag{2.29}$$

The boundary condition (2.28) is then satisfied automatically.

In our numerical implementation, therefore, we must define our environmental and field arrays to be twice as long (i.e. twice as deep) as necessary to describe the real environment and real acoustic field. After each range step, the UMPE model assures this symmetry by simply imposing condition (2.29) on the image field for $z < 0$. This formulation allows direct implementation of the split-step Fourier algorithm given by Eq. (1.29) using the full FFT transformations from z-space to k-space. It should be noted that some PE/SSF models (e.g., the Navy standard model) use condition (2.28) to simplify the implementation by using only sine transforms (rather than cosine transforms since (2.29) requires the full field have odd symmetry about $z = 0$). This has the effect of reducing the necessary transform and array sizes by $\frac{1}{2}$. However, this simplification is only valid when the surface is completely flat and at depth $z = 0$. To compute the exact forward scatter due to a rough surface interface, as will be discussed in a later section, the full description of the field is required.

3. Volume attenuation

Adding attenuation to the environment effectively introduces a complex component of the sound speed or, alternatively, of the wavenumber. The simplest way to introduce this phenomenon into the wave equation is to define the equivalent index of refraction

$$n'^2 = n^2 + i\alpha \quad . \tag{3.1}$$

As before, this can be incorporated into our PE model by introducing a new potential function

$$U_{loss}(z) = -i\frac{\alpha'(z)}{k_0} = -i\frac{\alpha(z)}{2k_0} \ , \tag{3.2}$$

where $\alpha' = \frac{\alpha}{2}$ is strictly valid only when the SPE approximation is invoked. We shall assume the validity is more general, however, for attenuation values that are not exceptionally large. Based on the formulation of the algorithm, this clearly has the effect of damping the solution in the space-domain by the factor

$$e^{-ik_0\Delta r U_{loss}(z)} = e^{-\Delta r\alpha'(z)} \quad . \tag{3.3}$$

In terms of transmission loss, this reduces the field by

$$TL_\alpha = -20\log(e^{-\Delta r\alpha'}) = 8.686\Delta r\alpha'\mathrm{dB} \quad . \tag{3.4}$$

The UMPE model assumes values input for volume attenuation have units [dB/km/Hz]. Internally, these values are rescaled to have units of inverse length, where the unit of length is specified by Δr, and the field strength is reduced by the factor given in Eq. (3.3) every range step.

3.1 Water volume attenuation

There are currently many common forms of empirical formulae defining volume attenuation in sea water as a function of frequency and depth. A popular form credited to Thorp (1967) is given by

$$\alpha_T(\mathrm{dB/kyd}) = 0.003 + \frac{0.1 f^2}{1+f^2} + \frac{40 f^2}{4100+f^2} + 2.75\times10^{-4} f^2 \tag{3.5}$$

where f is the frequency in kHz. Transforming this to units of [dB/km/Hz] is accomplished by

$$\alpha'(z=0) = \alpha_T(\mathrm{dB/kyd})\left(\frac{1\ \mathrm{kyd}}{3\ \mathrm{kft}}\right)\left(\frac{1\ \mathrm{kft}}{0.3048\ \mathrm{km}}\right)\left(\frac{1}{f[\mathrm{Hz}]}\right) . \tag{3.6}$$

The dependence with depth is then given by

$$\alpha'(z) = \alpha'(z=0)[1 - 6.46\times10^{-5} z(\mathrm{m})] . \tag{3.7}$$

In the UMPE model, a particular form for $\alpha'(z)$ is encoded and the environmental potential function propagator, $e^{-ik_0 \Delta r U_{op}}$, is scaled by the factor $e^{-\Delta r \alpha'}$ at each range step. Currently, the form of $\alpha'(z)$ is considered range-independent. In fact, as of the date of this report, no clear definition of $\alpha'(z)$ has been agreed upon and the portion of the UMPE code computing $\alpha'(z)$ has been commented out.

3.2 Bottom volume attenuation

The UMPE model allows the user to input a volume attenuation coefficient for each bottom profile, i.e. α' may be range-dependent but is constant over the depth of the bottom layer. Because the volume attenuation was introduced as part of a complex index of refraction, the discontinuities between the attenuation values over an interface are smoothed in an identical manner to the sound speed smoothing previously described. This is part of the creation of the array of volume attenuation values as a function of depth, $\alpha'(z)$, for all depths. As described above, this is used to scale the values of the environmental potential function propagator.

3.3 Effective attenuation due to shear

As was briefly mentioned in a previous section, the UMPE model treats the conversion process of compressional to shear waves at the bottom interface as a loss. The physical justification is that shear waves travel more slowly and at steeper angles than the incident compressional waves which generated them. Consequently, over a given range step, they attenuate more rapidly. It is unlikely that the shear waves will exist long enough to be refracted or reflected back towards the surface to transfer energy back into compressional modes. We therefore desire to compute the fraction of incident energy which is converted to shear wave energy at some bottom interface, and then simply remove that from the problem.

Tappert (1985) derived an analytical expression valid for low grazing angle reflection at the interface of a lossy fluid and a lossless solid bottom. The solid was replaced by an equivalent fluid bottom with an effective attenuation by matching the reflection coefficients of the lossless bottom with and without shear. The effective attenuation was then a function of the bottom shear speed and represented the simplest concept of loss due to shear conversion.

More recently, Tindle and Zhang (1992) developed a more rigorous approach for the interface between a lossy fluid and a lossy solid bottom by attempting to match the parameters of an equivalent lossy fluid bottom by comparing the total reflection coefficients for each case. This provides a good approximation to both the phase and amplitude of the reflection coefficient for all angles. They showed that, in addition to an effective bottom attenuation, it is also necessary to define an effective bottom density which is smaller than the true density. Specifically, a bottom with compressional and shear speeds c_b and c_s, respectively, density ρ_b and compressional and shear attenuations α_b and α_s, respectively, can be represented by an equivalent fluid bottom with compressional speed c_b, effective density

$$\rho_b' = \rho_b\left(1 - 2k^2\frac{c_s^2}{\omega^2}\right), \tag{3.8}$$

where $\omega = 2\pi f$ is the angular acoustic frequency, and effective attenuation

$$\alpha_b{}' = \alpha_b + \frac{4k^2 c_b c_s^3\left(k^2 - \frac{\omega^2}{c_b^2}\right)}{\omega^4\left(1 - 2k^2\frac{c_s^2}{\omega^2}\right)^2}\left[2\alpha_s + \frac{c_s}{\omega}\left(k^2 - \frac{\omega^2}{c_b^2}\right)^{1/2}\left(\frac{\omega^2}{c_s^2} - k^2\right)^{1/2}\right] . \tag{3.9}$$

The value to be used for the wavenumber k remains somewhat ambiguous. For Eq. (3.9), the authors chose $k = \frac{\omega}{c_w}$. However, for Eq. (3.8) they claim for several cases investigated it was found that $k = \frac{\omega}{c_b}$ was the best choice for when the interface compressional sound speed contrast satisfied $\frac{c_b}{c_w} < 1.2$. For ratios exceeding 1.2, they suggest averaging the values of effective density obtained by defining $k = \frac{\omega}{c_b}$ and $k = \frac{\omega}{c_w}$. To avoid a discontinuous change in the value of $\rho_b{}'$, and because a sound speed ratio of 1.2 is a large contrast in most ocean environments, we shall only employ the former definition. These equations then simplify to

$$\rho_b{}' = \rho_b\left(1 - 2\frac{c_s^2}{c_b^2}\right) \tag{3.10}$$

and

$$\alpha_b{}' = \alpha_b + \frac{4c_s^3(c_b^2 - c_w^2)}{c_b(c_w^2 - 2c_s^2)^2}\left[2\alpha_s + \frac{\omega(c_b^2 - c_w^2)^{1/2}(c_w^2 - c_s^2)^{1/2}}{c_b c_w^2}\right] . \tag{3.11}$$

Note that Eq. (3.11) requires that $c_s < c_w < c_b$. These formulae are implemented in the UMPE model at each interface with the understanding that the upper layer is treated as the fluid and the lower layer is treated as the solid.

4. Rough interface forward scatter

In this section, we shall consider how the UMPE model treats both rough bottom interfaces and surface displacements. Because the UMPE model is a one-way PE model, only forward scattering can be considered. Therefore, it is best suited for small slope specular reflection and forward diffuse scatter. Backscatter from cliff-like structures can be computed with the addition of specific pieces of code for individual problems but this is not within the general framework of the current version.

For both the ocean surface and bottom interfaces, the roughness is assumed to be characterized by a power law, or fractal, 2-D spectrum at high wavenumbers or small scales, i.e.

$$W_2\left(k \gg \frac{1}{L_{corr}}\right) = \alpha k^{-\beta} \tag{4.1}$$

where L_{corr} is the correlation length of the roughness and β is the spectral exponent. (Note that our use of wavenumber is now with respect to range scales of interface roughness.) To give the full spectrum a realistically smooth structure, we assume the spectral form

$$W_2(k) = \frac{\mu}{(1 + L_{corr}^2 k^2)^{\beta/2}} \quad . \tag{4.2}$$

The normalization factor μ is defined in terms of the rms roughness σ by requiring

$$2\pi \int_0^{\infty} W_2(k) k dk = \sigma^2 \quad . \tag{4.3}$$

This leads to

$$\mu = \frac{1}{\pi}\left(\frac{\beta}{2} - 1\right)\sigma^2 L_{corr}^2 \quad . \tag{4.4}$$

For the purpose of computing the acoustic field in two dimensions (depth and range), we need only the 1-D roughness spectrum along the track of interest. Assuming the direction of propagation is along the x-axis, we need

$$W_1(k_x) = \int_{-\infty}^{\infty} W_2(\sqrt{k_x^2 + k_y^2})dk_y \quad . \tag{4.5}$$

It follows immediately from Eq. (4.3) that

$$\int_{-\infty}^{\infty} W_1(k_x)dk_x = \sigma^2 \quad . \tag{4.6}$$

An alternative formula to (4.5) for computing $W_1(k)$ along any track may be derived by converting to cylindrical coordinates giving

$$W_1(k) = 2\int_k^{\infty} \frac{K}{\sqrt{K^2 - k^2}} W_2(K)dK \quad . \tag{4.7}$$

Substituting (4.2) and (4.4) into (4.5) yields

$$W_1(k) = \gamma\sigma^2 L_{corr}(1 + L_{corr}^2 k^2)^{-\left(\frac{\beta - 1}{2}\right)} \tag{4.8}$$

with

$$\gamma = \frac{2}{\pi}\left(\frac{\beta}{2} - 1\right)\int_0^{\infty} \frac{dt}{(1 + t^2)^{\beta/2}} = \frac{\left(\frac{\beta}{2} - 1\right)\Gamma\left(\frac{1}{2}\right)}{\pi} \frac{\Gamma\left(\frac{\beta - 1}{2}\right)}{\Gamma\left(\frac{\beta}{2}\right)} \tag{4.9}$$

where $\Gamma(x)$ is the gamma function.

Within the UMPE model, we avoid computing gamma functions by assuming the 1-D spectrum has the form

$$W_1'(k) = (1 + L_{corr}^2 k^2)^{-\left(\frac{\beta - 1}{2}\right)} \quad . \tag{4.10}$$

In order to compute various stochastic realizations of roughness, we superimpose upon this a random wavenumber amplitude and phase. Because the complex amplitude of each wavenumber component, $Ae^{i\vartheta}$, should exhibit a normal distribution in the complex k-plane, the random phase of each component can be obtained from $\vartheta = 2\pi r_1$ where r_1 is a uniformly distributed random variable in the interval $(0, 1)$. The magnitude A, however, exhibits a Rayleigh distribution.

Because we are creating a realization of the power spectrum, we consider the magnitude squared which has a negative exponential distribution. The random amplitude of each component of the power spectrum can then be obtained by $A^2 = -\ln(r_2)$ where r_2 is another independent uniformly distributed random variable in the interval $(0, 1)$. This complex spectrum is then Fourier transformed to yield a roughness, $\eta(x)$, which is normalized by the actual rms value, i.e. $\frac{\eta(x)}{\langle\eta^2(x)\rangle^{1/2}}$. The result of this calculation is a single stochastic realization of an interface roughness with the proper spectral shape given by (4.10) and an rms value normalized to unity. Multiplication by the desired rms value, σ, should give us exactly the rough interface defined by (4.8) and (4.9). Furthermore, to avoid introducing interface displacements not found in actual bathymetric databases, the spectrum for bottom roughness is high-pass filtered to remove wavelengths larger than or on the order of the database resolution. (The cut-off for this filter is specified within the source code and may be edited.) The UMPE model accepts as input the total number of realizations to compute. If more than one realization is requested, the calculations of transmission loss for given depths are averaged incoherently to yield a measure of the total rms field or average power.

4.1 Water/bottom interface

The treatment of the boundary condition at bottom or sediment interfaces has already been discussed in detail in section 2. Since this treatment was independent of the actual depth of the interface, no additional calculations are necessary once the bathymetry has been specified. One should note, however, that with any type of range-dependent bathymetry, whether smooth sloping bottoms or with small scale roughness, each range step of the calculation assumes a constant environment. Recall that this follows from the approximation $\int_r^{(r+\Delta r)} H(r')dr' \approx \Delta r H(r)$ implicit in the PE/SSF algorithm. This does not imply that the effects of slopes are neglected. If the environment, hence the bathymetry, is sampled well enough and the changes occurring over one range step are relatively small (e.g. the change in bathymetry over one range step, or the slope, must be less than a wavelength, $\delta\eta \ll \lambda_0$) then diffusive effects will smear out these apparently discontinuous jumps. The combination of a number of such "jumps" over several range steps will be observed to create the anticipated effects.

4.2 Surface interface

The calculation of rough surface forward scatter is more complicated because the treatment of the surface boundary condition assumes that the surface is fixed at $z = 0$. The UMPE model provides two methods of computing rough surface scatter. The first maintains the imposed symmetry about $z = 0$ by invoking an approximation which replaces the interface displacement by a volume effect. The second method is an exact technique derived from first principles when the surface interface is not assumed to be flat.

4.2.1 Approximate surface forward scatter

We assume the surface interface is defined by

$$z - \eta(r) = 0 \tag{4.11}$$

where $\eta(r)$ is the random surface displacement, increasing downward, and is defined as a zero-mean Gaussian random function with

$$\langle \eta(r) \rangle = 0 \tag{4.12}$$

and

$$\langle \eta(r)\eta(r') \rangle = \mathrm{Re}\int_{-\infty}^{\infty} W_1(k) e^{ik(r-r')} dk \quad . \tag{4.13}$$

This definition is consistent with the interface roughness previously described.

We begin with the Helmholtz equation for the acoustic pressure with a Dirichlet boundary condition at the surface,

$$\nabla^2 p + k_0^2 n^2 p = 0 \tag{4.14}$$

and

$$p(\eta(r)) = 0 \ . \tag{4.15}$$

It can be shown that the boundary condition may formally be put into the volume as a singular index of refraction. For small surface slopes, this is approximately

$$\nabla^2 p + k_0^2 \left[n^2 - \frac{\eta(r)}{k_0^2} \delta''(z) \right] p = 0 \tag{4.16}$$

and now

$$p(0) = 0 \ . \tag{4.17}$$

Invoking the usual approximations in the standard parabolic equation to Eq. (4.16) leads to

$$i\frac{\partial\psi}{\partial r} + \frac{1}{2k_0}\frac{\partial^2\psi}{\partial z^2} - k_0[U(r,z) + \mu(r,z)]\psi = 0 \tag{4.18}$$

with

$$\psi(0) = 0 \ , \tag{4.19}$$

$U(r,z)$ is the standard PE potential function (Eq. (1.13))

$$U(r,z) = -\frac{1}{2}(n^2 - 1)$$

and $\mu(r,z)$ is given explicitly by

$$\mu(r,z) = \frac{\eta(r)}{2k_0^2} \delta''(z) \ . \tag{4.20}$$

The variable z is now interpreted as the depth increasing downward from the mean sea level at $z = 0$.

As before, we must replace the functional $\delta''(z)$ by some appropriate generalized function. The first non-vanishing moment of $\delta''(z)$ is the second moment defined by

$$\int_{-\infty}^{\infty} z^2 \delta''(z) dz = 2 \ . \tag{4.21}$$

Furthermore, the expectation value is given by

$$\langle \delta''(z) \rangle = \int_{-\infty}^{\infty} \psi^* \delta''(z) \psi dz = \int_{-\infty}^{\infty} \delta(z)[\psi^{*\prime\prime}\psi + \psi^*\psi'' + 2\psi^{*\prime}\psi']dz = 2\left|\frac{\partial}{\partial z}\psi(0)\right|^2 \qquad (4.22)$$

which follows from an integration by parts. Introducing $g(z)$ as a generalized function, we require agreement with the expectation value of (4.22), i.e.

$$\langle g(z) \rangle = \int_{-\infty}^{\infty} \psi^* g(z) \psi dz = 2\left|\frac{\partial}{\partial z}\psi(0)\right|^2 \ . \qquad (4.23)$$

Since $g(z)$ will have influence only near the surface, we expand the field in a Taylor series near $z = 0$,

$$\psi(z) \approx z\frac{\partial}{\partial z}\psi(0) + \ldots \qquad (4.24)$$

To lowest order

$$\langle g(z) \rangle \approx \int_{-\infty}^{\infty} z^2 g(z) dz \left|\frac{\partial}{\partial z}\psi(0)\right|^2 \qquad (4.25)$$

and the condition on $g(z)$ reduces to matching the second moments such that

$$\int_{-\infty}^{\infty} z^2 g(z) dz = 2 \ . \qquad (4.26)$$

This can be accommodated by a Gaussian of the form

$$g(z) = Ae^{-z^2/L_s^2} \qquad (4.27)$$

where the amplitude is defined by

$$A\int_{-\infty}^{\infty} z^2 e^{-z^2/L_s^2} dz = 2 \ . \qquad (4.28)$$

From a standard table of integrals this reduces to

$$A = \frac{4}{\sqrt{\pi}L_s^3} \ . \qquad (4.29)$$

The approximate surface forward scatter is then carried out by imposing the usual odd symmetry at $z = 0$ and adding to the environmental potential the function $\mu(r,z)$ defined by (4.20) with the generalized function given by (4.27) and (4.29). The standard PE potential function was used here for simplicity. However, the formulation of (4.20) is more general and can be added to whatever environmental potential form is desired.

It should also be mentioned at this point that for surface interface roughness the UMPE model takes input values of wind speed which may be range-dependent. An accepted empirical relationship (Neumann and Pierson, 1966) is used to determine the rms roughness

$$\sigma = \frac{(\text{wind speed})^2}{10g} \tag{4.30}$$

where $g \equiv$ acceleration of gravity. The roughness correlation length is then defined by

$$L_{corr} = 20\sigma \quad . \tag{4.31}$$

As before, the use of a generalized function is inherently ambiguous with respect to the choice of mixing length L_s in (4.27). Presumably, this function should not be much wider than a wavelength and yet be large enough to allow adequate sampling by the given depth mesh. Currently, a minimum default value of $L_{s_{\min}} = \Delta z$ has been defined. To verify some range of validity for L_s requires a comparison with exact solutions of surface scatter. A method of computing such solutions is the topic of the following section.

4.2.2 Exact surface forward scatter

We return to the standard parabolic equation and impose the surface boundary condition at the true depth of the surface displacement, i.e.

$$i\frac{\partial\psi}{\partial r} + \frac{1}{2k_0}\frac{\partial^2\psi}{\partial z^2} - k_0 U(r, z)\psi = 0 \tag{4.32}$$

with

$$\psi(z = \eta(r)) = 0 \quad . \tag{4.33}$$

Similar to the previous treatment of extending the field in an image ocean with odd symmetry about $z = 0$, we now extend the field with odd symmetry, and $U(r,z)$ with even symmetry, about $z = \eta(r)$.

The equations defining the field propagation in the real and image oceans are then (Tappert and Nghiem-Phu, 1985)

$$\text{real ocean: } i\frac{\partial\psi}{\partial r} + \frac{1}{2k_0}\frac{\partial^2\psi}{\partial z^2} - k_0 U(r, z)\psi = 0\ , \ \ (z > \eta(r)) \tag{4.34a}$$

$$\text{image ocean: } i\left(\frac{\partial\psi}{\partial r} + 2\frac{\partial\eta}{\partial r}\frac{\partial\psi}{\partial z}\right) + \frac{1}{2k_0}\frac{\partial^2\psi}{\partial z^2} - k_0 U(r, z)\psi = 0\ , \ \ (z < \eta(r))\ . \tag{4.34b}$$

It can be shown that

$$\psi(r, -z + 2\eta(r)) = -\psi(r, z) \ \Rightarrow\ \psi(r, \eta(r)) = 0\ ,$$

satisfying our required boundary condition.

To implement this, let us define the total field extending over the real and image ocean as $\tilde{\psi}(r, z)$. Then

$$\tilde{\psi}(r, z) = \begin{cases} \psi(r, z) & ,\ \ (z > \eta(r)) \\ e^{i2k_0\dot{\eta}(z-\eta)}\psi(r, z)\ , & (z < \eta(r)) \end{cases}$$

$$= \begin{cases} \psi(r, z) & ,\ \ (z > \eta(r)) \\ -e^{i2k_0\dot{\eta}(z-\eta)}\psi(r, -z + 2\eta(r))\ , & (z < \eta(r)) \end{cases} \tag{4.35}$$

and Eqs. (4.34) are transformed to a new parabolic wave equation identical in form to Eq. (4.32), but with a split environmental propagator function, i.e.

$$i\frac{\partial}{\partial r}\tilde{\psi} + \frac{1}{2k_0}\frac{\partial^2}{\partial z^2}\tilde{\psi} - k_0\tilde{U}(r, z)\tilde{\psi} = 0\ , \tag{4.36}$$

$$\tilde{U}(r, z) = \begin{cases} U(r, z) & ,\ \ (z > \eta(r)) \\ U(r, z) - 2(z - \eta)\dfrac{\partial^2}{\partial r^2}\eta & ,(z < \eta(r)) \end{cases}\ . \tag{4.37}$$

Using the FFT convention in the UMPE code,

$$\psi(z) = \mathrm{FFT}[\hat{\psi}(k)] = \sum_k \hat{\psi}_k e^{ikz}\ , \tag{4.38}$$

and employing the “frequency-shifting” theorem

$$\psi(z+z') = \sum_k \hat{\psi}_k e^{ik(z+z')} = \sum_k \hat{\psi}_k e^{ikz'} e^{ikz} = \mathrm{FFT}[\hat{\psi}(k) e^{ikz'}] \tag{4.39}$$

Eq. (4.35) becomes

$$\tilde{\psi}(r,z) = \begin{cases} \sum_k \hat{\psi}_k e^{ikz} & , \quad (z > \eta(r)) \\ -e^{i2k_0\dot{\eta}(z-\eta)} \sum_k \hat{\psi}_{-k} e^{-2ik\eta} e^{ikz} , & (z < \eta(r)) \end{cases} \tag{4.40}$$

where we have also used the identity

$$\psi(-z) = \mathrm{FFT}[\hat{\psi}(-k)] \ . \tag{4.41}$$

Eqs. (4.36), (4.37), and (4.40) are implemented in the UMPE model to compute exact surface forward scatter. To do so, an additional field array has been added to the calculation to accommodate the second formula of Eq. (4.40). At each range step, before the solution is transformed from *k*-space to *z*-space a copy of the field is made. Each formula in (4.40) is then applied to the two copies of $\hat{\psi}_k$ which produces two different fields in *z*-space. These two fields are then recombined about the position of the interface to produce the total field prior to the next range step.

Note that Eq. (4.40) is also independent of the type of PE approximation used since both (4.34a) and (4.34b) could have been written in terms of a general Q_{op} function without affecting the influence of the surface displacement. Furthermore, this formulation requires a complete Fourier transform as the symmetries involved cannot be accommodated by only a sine transform. A drawback of this technique is the necessity of additional FFT's in the SSF algorithm, thereby increasing the total run time. In this regard, the approximate forward surface scatter technique is desirable. However, to this date no thorough investigation of the validity of the approximate technique, presumably by comparing results from the two methods, has been performed.

4.3 Effect of near-surface bubbles

The presence of bubbles in the ocean volume is characterized by the probability density function of the number of bubbles per unit volume per unit radius, $n(\vec{x}, a)$. The total number of bubbles per unit volume is then

$$N(\vec{x}) = \int_0^{\infty} n(\vec{x}, a) da \tag{4.42}$$

and the void fraction, defined as the volume of bubbles per unit volume, is

$$V(\vec{x}) = \int_0^{\infty} \frac{4}{3}\pi a^3 n(\vec{x}, a) da \quad . \tag{4.43}$$

We shall assume the primary effect of the presence of bubbles is to alter the acoustic index of refraction which is now defined by

$$n(\vec{x}) = n_w(\vec{x}) + n_b(\vec{x}) \quad . \tag{4.44}$$

$n_w(\vec{x})$ is the usual index of refraction of water, and $n_b(\vec{x})$ is the effective index of refraction due to bubbles within the water volume defined by

$$n_b(\vec{x}) = \text{const} \times V(\vec{x}) \tag{4.45}$$

where the proportionality factor is some, as yet, undetermined constant. Presumably, the bubbles are organized in such a way that the average void fraction is a function of depth only, i.e.

$$V(\vec{x}) = \overline{V}(z) + \delta V(\vec{x}) \tag{4.46}$$

and we shall assume that the bubbles, hence $n_b(\vec{x})$, are concentrated near the surface. We may then expand

$$n_b(\vec{x}) \approx n_{b0}(x, y)\delta(z) + n_{b1}(x, y)\delta'(z) + n_{b2}(x, y)\delta''(z) + \ldots \tag{4.47}$$

When the PE/SSF algorithm is applied to this expansion, the odd terms vanish since $n(\vec{x})$ must be an even function about $z = 0$ (and the odd derivatives of the delta function are odd functions). The first two even terms in (4.47) are defined by

$$n_{b0}(x, y) = \int_{-\infty}^{\infty} n_b(x, y, z)dz = 2\int_{0}^{\infty} n_b(x, y, z)dz \tag{4.48}$$

and

$$n_{b2}(x, y) = \frac{1}{2}\int_{-\infty}^{\infty} z^2 n_b(x, y, z)dz = \int_{0}^{\infty} z^2 n_b(x, y, z)dz = \text{const} \times \int_{0}^{\infty} z^2 V(x, y, z)dz \ . \tag{4.49}$$

The first term, however, will also vanish since the delta function combined with the Dirichlet boundary condition at $z = 0$ has no effect. The leading term in the correction for the presence of bubbles is then $n_{b2}(x, y)\delta''(z)$.

We now recall in our development of the approximate surface forward scatter there was also a term proportional to $\delta''(z)$ defined by Eq. (4.20). It therefore seems plausible that we may incorporate the effects of near surface bubbles into our forward scattering calculations by defining an effective surface displacement,

$$\eta_{\text{eff}}(x, y) = \eta(x, y) + \text{const} \times k_0^2 \int_{0}^{\infty} z^2 V(x, y, z)dz \ . \tag{4.50}$$

Using this approach, we are then free to employ either forward surface scatter method.

There is presently no clear definition for the constant proportionality factor in Eq. (4.50). Furthermore, there is no accepted empirical relationship between wind speed and bubble void fraction. It is not clear how we should define these quantities in terms of the UMPE model input parameters. Currently, the model allows the user to input simply an effective rms roughness and correlation length characterizing the virtual surface displacement due to bubbles. With these values, a “bubble displacement” is computed using a spectrum of the same form as the rough surface but with independent amplitude and phase realizations. The total surface roughness is then formed by combining the surface and bubble displacements together.

Incidentally, it is also unclear whether each random interface realization should be independent. Specifically, should surface waves and near-surface bubbles have similar spectral shapes and/

or phases? One may also consider the correlation between interfaces of a thin sediment layer and the underlying basement. These possibilities are currently ignored.

5. Broadband travel time analysis

The calculation of the time domain arrival structure at a given range can be performed in a straightforward manner using a PE code. Recall that we began this report by assuming a form for the time-harmonic acoustic field defined by

$$P(r, z, \omega t) = p_{\omega}(r, z)e^{-i\omega t} \quad . \tag{5.1}$$

The representation of the field in the time domain is then simply

$$\tilde{P}(r, z, t) = \mathrm{FFT}[p_{\omega}(r, z)] = \sum_{\omega} p_{\omega}(r, z)e^{-i\omega t} \tag{5.2}$$

which follows by recognizing that (5.1) is a single (CW) component of the general time dependent field (5.2). In other words, by choosing a particular frequency, ω', we are computing

$$\tilde{P}(r, z, t)\delta_{\omega, \omega'} = \sum_{\omega} p_{\omega}(r, z)e^{-i\omega t}\delta_{\omega, \omega'} = P(r, z, \omega' t) \tag{5.3}$$

at time $t = 0$.

To compute the arrival time structure at some fixed range $r = R$, we need to compute the complex field $p_{\omega, R}(z)$ for many frequencies and then Fourier transform to obtain $\tilde{P}_R(z, t)$, the set of complex pressure values in time/depth space. Because the FFT assumes inputs over the frequency band $f_0 - \frac{\mathrm{BW}}{2}$ to $f_0 + \frac{\mathrm{BW}}{2}$, where BW is the bandwidth of the acoustic source and f_0 is the center frequency, it becomes computationally burdensome to consider high frequency calculations even if the frequency bandwidth of computed fields is small. Therefore, the UMPE model heterodynes the signal, shifting the center frequency to d.c In other words, we compute

$$\tilde{P}'_R(z, t) = \sum_{\omega} p_{\omega, R}(z)e^{-i(\omega - \omega_0)t} \tag{5.4}$$

where $\omega_0 = 2\pi f_0$. This allows us to place the complex pressure values $p_{\omega, R}(z)$ into frequency bins symmetrically about $\omega - \omega_0 = 0$. The inputs then extend over the band $-\frac{1}{2}\mathrm{BW}$ to $+\frac{1}{2}\mathrm{BW}$. This has no effect on the computed intensities which is evident by noting that $I = P^*P$ is independent of ω_0.

In practice, we compute the FFT of $\frac{1}{\sqrt{R}}\psi_{\omega, R}(z)$. Because this neglects the overall phase factor $e^{ik_0R} = e^{i\omega\frac{R}{c_0}}$, the time domain is heterodyned around the value $t_0 = \frac{R}{c_0}$. Arrival times are then given as values of "reduced time" or $(t - t_0)$. For time domain analysis, the UMPE model requires the user to input a center frequency, a bandwidth, and the number of frequency bins to compute (the frequency transform size). This obviously introduces some ambiguity into the absolute travel time prediction via the choice of reference sound speed. It is expected that an implementation of the c_0-insensitive model would remove this ambiguity.

To this point, we have assumed each frequency component of the total field contributes equally in amplitude. This is synonymous with assuming the frequency spectrum of the source is flat over some frequency band, i.e. a "box-car" function. It is well known that the transform of such a function will introduce side-lobes. Furthermore, since the transform size, hence the time-window, must be finite, wrap-around or aliasing may occur. To minimize these effects, the model creates a sine-squared filter which is applied to the upper and lower 1/8 of the bandwidth. This method of tapering greatly reduces the effect of side-lobes. (More about similar filters used in the model will be discussed in a later section.)

The user should still be wary of wrap-around effects. These may arise if the time window is not wide enough. To avoid this, the user should estimate the anticipated time spread of the signal before generating a solution and input the proper parameters to allow for this amount. The total length of the time window is given by

$$T = \frac{N}{BW} \tag{5.5}$$

where BW is the bandwidth of the source to be modeled and N is the number of frequencies for which the model should compute solutions to the field, or the frequency FFT size. The frequency resolution is obviously given by

$$\Delta\omega = \frac{BW}{N} = \frac{1}{T} . \tag{5.6}$$

Thus, the wider the time window needed, the finer the bandwidth must be resolved and the more total runs at different frequencies must be performed. This can obviously lead to extremely long computer run-time. Careful considerations should therefore be given before entering these parameters.

It should be noted that while the model creates a rather simple tapered box-car function to represent the frequency spectrum of the source, the code could easily be modified to accept a specific frequency spectrum defined by the user. Because the output is in dB units of transmission loss, all that would be required is a source spectrum normalized to maximum values of unity and sampled at the appropriate frequencies. This function would then serve as the filter function in the frequency domain. Furthermore, the model is unable to treat multiple interface roughness broadband calculations. If one desires to compute an ensemble average of the travel times, numerous broadband calculations for single stochastic realizations could be performed and the results combined in a post-processing routine.

Finally, there is a built-in ambiguity in the computed levels of transmission loss, or *TL* (to be defined later), in the time domain. This is due in part to variability with respect to the frequency/time transform size, beamwidth of the source function, type of source function used, and several other parameters. To obtain true *TL* levels, we can simply compute the travel time values out to the reference range of $R_0 = 1$ m. (More about this parameter in a later section.) The peak value at this range should be *TL* = 0 dB re 1m. Subtracting the actual value obtained from subsequent calculations should provide true *TL* levels. Note that this is only true when all source and broadband parameters remain the same. This method of computing absolute pulse *TL* was first described by Nghiem-Phu and Tappert (1985).

6. Acoustic particle velocity

The ability to compute acoustic particle velocities from the pressure field has recently been added as an option in the UMPE model. The motivation was simply a lack of available fully range-dependent acoustic propagation models which can produce this result. The implementation is straightforward but, as will become clear, adds a non-negligible amount of run-time to perform the necessary calculations.

We begin by stating the general form of the parabolic equation,

$$\frac{\partial\psi}{\partial r} = -ik_0 H_{op}\psi \ , \tag{6.1}$$

where ψ is related to the acoustic pressure by

$$p = P_0\sqrt{\frac{\rho R_0}{\rho_0 r}}\psi e^{ik_0 r} \ . \tag{6.2}$$

The operator H_{op} can take any of the five forms defined in section 1. By conservation of linear momentum, the acoustic particle velocity $\vec{v} = (v_r, v_z)$ is related to the pressure by

$$\vec{v} = -\frac{i}{\omega\rho_0}\nabla p \ , \tag{6.3}$$

assuming both $\vec{v}$ and p have the harmonic time dependence $e^{-i\omega t}$. Separating the components of $\vec{v}$ leads to the expressions

$$v_r = -\frac{i}{\omega\rho_0}\frac{\partial p}{\partial r} =$$

$$\frac{P_0}{\omega\rho_0}\sqrt{\frac{\rho R_0}{\rho_0 r}}\left[k_0\psi + i\left(\frac{\psi}{2r} - \frac{\partial\psi}{\partial r}\right)\right]e^{ik_0 r} = \frac{P_0}{c_0\rho_0}\sqrt{\frac{\rho R_0}{\rho_0 r}}e^{ik_0 r}\left(i\frac{\psi}{2k_0 r} + \psi - H_{op}\psi\right) \tag{6.4}$$

and

$$v_z = -\frac{i}{\omega\rho_0}\frac{\partial p}{\partial z} = -\frac{iP_0}{c_0\rho_0 k_0}\sqrt{\frac{\rho R_0}{\rho_0 r}}e^{ik_0 r}\frac{\partial\psi}{\partial z} \quad . \tag{6.5}$$

In the r-direction, it is found that the application of the operator H_{op} must be applied to the field function ψ in order to perform the calculation. As previously noted, each of our PE options separates the operator H_{op} into two parts,

$$H_{op} = T_{op}\left(\frac{\partial^2}{\partial z^2}\right) + U_{op}(n(r, z)) \quad . \tag{6.6}$$

Additionally, the vertical component of acoustic particle velocity requires application of the operator $\frac{\partial}{\partial z}$ on the ψ field. To perform these operations, we consider the wavenumber representation of the field at some arbitrary depth z defined by

$$\psi(z) = \mathrm{FFT}[\hat{\psi}(k)] = \sum_k \hat{\psi}_k e^{ikz} \ , \tag{6.7}$$

then

$$\frac{\partial}{\partial z}\psi(z) = \sum_k ik\hat{\psi}_k e^{ikz} = \mathrm{FFT}[ik\hat{\psi}(k)] \tag{6.8}$$

and

$$\frac{\partial^2}{\partial z^2}\psi(z) = -\sum_k k^2\hat{\psi}_k e^{ikz} = \mathrm{FFT}[-k^2\hat{\psi}(k)] \quad . \tag{6.9}$$

We now state explicitly the various formulas for acoustic particle velocity corresponding to the type of PE approximation chosen:

$$1)\ \ \mathrm{SPE:}\ \ T_{op} = -\frac{1}{2k_0^2}\frac{\partial^2}{\partial z^2}\ ,\ \ U_{op} = -\frac{1}{2}(n^2 - 1)\ ,$$

so

$$v_r(r, z) = \frac{P_0}{c_0\rho_0}\sqrt{\frac{\rho R_0}{\rho_0 r}}e^{ik_0 r}\left[i\frac{\psi(r, z)}{2k_0 r} + \frac{1}{2}(n^2 + 1)\psi(r, z) - \mathrm{FFT}\left(\frac{1}{2}\frac{k^2}{k_0^2}\hat{\psi}(r, k)\right)\right]\ ; \tag{6.10}$$

$$2)\ \text{VSPE:}\ T_{op} = -\frac{1}{2k_0^2}\frac{\partial^2}{\partial z^2},\ U_{op} = -(n-1),$$

so

$$v_r(r,z) = \frac{P_0}{c_0\rho_0}\sqrt{\frac{\rho R_0}{\rho_0 r}}e^{ik_0r}\left[i\frac{\psi(r,z)}{2k_0r} + n\psi(r,z) - \text{FFT}\left(\frac{1}{2}\frac{k^2}{k_0^2}\hat{\psi}(r,k)\right)\right]; \tag{6.11}$$

$$3)\ \text{VWAPE:}\ T_{op} = -\frac{\partial^2}{\partial z^2}\left[\left(1+\frac{\partial^2}{\partial z^2}\right)^{1/2} + 1\right]^{-1},\ U_{op} = -\frac{1}{2}(n^2-1),$$

so

$v_r(r,z) =$

$$\frac{P_0}{c_0\rho_0}\sqrt{\frac{\rho R_0}{\rho_0 r}}e^{ik_0r}\left[i\frac{\psi(r,z)}{2k_0r} + \frac{1}{2}(n^2+1)\psi(r,z) - \text{FFT}\left(\left[1-\sqrt{1-\frac{k^2}{k_0^2}}\right]\hat{\psi}(r,k)\right)\right]; \tag{6.12}$$

$$4)\ \text{WAPE:}\ T_{op} = -\frac{\partial^2}{\partial z^2}\left[\left(1+\frac{\partial^2}{\partial z^2}\right)^{1/2} + 1\right]^{-1},\ U_{op} = -(n-1),$$

so

$$v_r(r,z) = \frac{P_0}{c_0\rho_0}\sqrt{\frac{\rho R_0}{\rho_0 r}}e^{ik_0r}\left[i\frac{\psi(r,z)}{2k_0r} + n\psi(r,z) - \text{FFT}\left(\left[1-\sqrt{1-\frac{k^2}{k_0^2}}\right]\hat{\psi}(r,k)\right)\right]; \tag{6.13}$$

$$5)\ \text{LOGPE:}\ T_{op} = -ln\left[\cos\left(-\frac{i}{k_0}\frac{\partial}{\partial z}\right)\right],\ U_{op} = -ln(n),$$

so

$v_r(r,z) =$

$$\frac{P_0}{c_0\rho_0}\sqrt{\frac{\rho R_0}{\rho_0 r}}e^{ik_0 r}\left[i\frac{\psi(r,z)}{2k_0 r} + (1 + ln(n))\psi(r,z) + \mathrm{FFT}\left(ln\left[\cos\frac{k}{k_0}\right]\hat{\psi}(r,k)\right)\right] . \tag{6.14}$$

For any of the above options, the z-component of acoustic particle velocity is computed from

$$v_z(r,z) = \frac{P_0}{c_0\rho_0}\sqrt{\frac{\rho R_0}{\rho_0 r}}e^{ik_0 r}\mathrm{FFT}\left[\frac{k}{k_0}\hat{\psi}(r,k)\right] . \tag{6.15}$$

It is obvious from these expressions that for every range step, additional FFT's must be computed thereby increasing the overall run-time by a significant amount.

7. Source functions

In this section we shall define the initial conditions for the PE field function, $\psi(r = 0, z)$. Previously, we have assumed the relationship between ψ and the acoustic pressure, ignoring the effect of density, is of the form

$$p = P_0\sqrt{\frac{R_0}{r}}\psi e^{ik_0 r} . \tag{7.1}$$

We obviously cannot define p at range $r = 0$ from this expression. This is analogous to the undefined amplitude of a point source Green's function at the source location. Therefore, we choose to define the source amplitude relative to that at some small but finite distance from the source. Specifically, we choose

$$p(r = R_0) = P_0 . \tag{7.2}$$

Consistent with reference values used in most sonar equations, we define the reference range

$$R_0 = 1 \text{ m} \tag{7.3}$$

and the source level, SL, is related to P_0 by

$$SL = 20\log\left(\frac{P_0}{P_r}\right) \text{ dB re } P_r R_0 . \tag{7.4}$$

The dB units of SL are explicitly stated relative to a reference pressure value of $P_r = 1\ \mu\text{Pa}$ at the reference range R_0.

We are still left with the task of determining a form for the source field $\psi(r = 0, z)$. We begin by writing (7.1) as

$$\psi(r, z)e^{ik_0 r} = \frac{1}{P_0}\sqrt{\frac{r}{R_0}}p(r, z) \tag{7.5}$$

from which it follows that

$$\psi(r = 0, z) = \lim_{r \to 0}\frac{1}{P_0}\sqrt{\frac{r}{R_0}}p(r, z) \quad . \tag{7.6}$$

In the vicinity of a point source, we know the pressure field takes the form of the spherical Green's function. Since $\psi(r,z)$ is only in two dimensions, we write

$$p = \frac{a}{2\pi R}e^{ik_0 R}\ ,\ R = \sqrt{r^2 + z^2}\ ,\ a = P_0 R_0 \tag{7.7}$$

where a is defined by requiring $|p| = \dfrac{P_0}{2\pi}$ at $R = R_0$. We represent the source at $(0, z_S)$ as a point source by defining

$$\psi(r = 0, z) = \alpha\delta(z - z_S) \quad . \tag{7.8}$$

Integrating both sides of (7.6) over all depths yields

$$\alpha = a\lim_{r \to 0}\frac{1}{P_0}\sqrt{\frac{r}{R_0}}\int_{-\infty}^{\infty}\frac{1}{2\pi R}e^{ik_0 r}dz \quad . \tag{7.9}$$

Because we are ultimately interested in the solution in the far-field ($r \gg z$), we may approximate

$$R = \sqrt{r^2 + \bar{z}^2} \approx r + \frac{\bar{z}^2}{2r}\ ,\ \bar{z} = z - z_S \quad . \tag{7.10}$$

Eq. (7.9) then reduces to

$$\alpha \approx \sqrt{R_0}\lim_{r \to 0}\frac{e^{ik_0 r}}{2\pi\sqrt{r}}\int_{-\infty}^{\infty}e^{ik_0\frac{\bar{z}^2}{2r}}dz = \sqrt{R_0}\lim_{r \to 0}\frac{e^{ik_0 r}}{2\pi\sqrt{r}}\sqrt{-\frac{2\pi r}{ik_0}}$$

or

$$\alpha = \sqrt{\frac{iR_0}{2\pi k_0}} \ . \tag{7.11}$$

It is desirable to begin the calculation by specifying the source in the *k*-domain. Including the influence of the image source, a straightforward Fourier transform of (7.8) yields

$$\hat{\psi}(r=0,k) = \alpha \int_{-\infty}^{\infty} [\delta(z-z_S) - \delta(z+z_S)]e^{-ikz}dz = -2i\alpha\sin(kz_S) \ . \tag{7.12}$$

which indicates that the wavenumber representation of the starting field has a constant amplitude modulated by a phase due to the interaction of the source and its image. This constant amplitude is consistent with the notion of an omnidirectional point source which puts equal energy into all wavenumbers (i.e., all directions).

As an approximation to the delta function starting field, we shall consider a Gaussian function defined by

$$\psi(0,z) = \frac{\alpha}{w\sqrt{\pi}} e^{-(z-z_S)^2/w^2} \tag{7.13}$$

where the additional factor is required to give the correct normalization in the limit $w \to 0$. Including the image field and transforming this to the *k*-domain yields

$$\hat{\psi}(0,k) = \frac{\alpha}{w\sqrt{\pi}} \left[\int_{-\infty}^{\infty} e^{-\frac{\bar{z}^2}{w^2}} e^{-ik(\bar{z}+z_S)} d\bar{z} - \int_{-\infty}^{\infty} e^{-\frac{\bar{z}^2}{w^2}} e^{-ik(\bar{z}-z_S)} d\bar{z} \right] = -2i\alpha e^{-\frac{k^2w^2}{4}} \sin(kz_S) \ . \tag{7.14}$$

The first thing we notice about this expression is that the Gaussian normalization factor has been removed and the remaining factor is identical to that of Eq. (7.12) including the phase modulation due to the source/image interference. In this approximation, however, the constant amplitude of (7.12) has been replaced by a Gaussian of width $(2/w)$ and maximum value at $k = 0$ of unity. This may be interpreted as placing the correct amount of energy at $k = 0$ (corresponding to horizontal or $\theta = 0$ propagation) and quickly tapering off the energy placed at higher absolute wavenumbers. This is consistent with the small angle validity of the standard parabolic equation. Unfortunately, for angles significantly removed from horizontal, this type of source function creates a poor representation of an omnidirectional source.

For the wide angle PE approximations, it would seem reasonable to simply replace the Gaussian function by unity for all wavenumbers thereby equally populating all directions of propagation. However, even the wide angle approximations are assumed valid only up to angles of $40°$ or so. Additionally, the finite FFT size will restrict how large k/k_0 can be. Therefore, a smooth taper is included at high absolute wavenumber values to limit the angular width of the source function and to reduce the influence of sidelobes. Thomson and Bohun (1988) have also shown that a wide angle source needs to be modified by the factor

$$F(k) = \left(1 - \frac{k^2}{k_0^2}\right)^{-1/4}, \quad (|k| < k_0) \tag{7.15}$$

to produce the correct solution in the far-field. Note that $k = k_0$ corresponds to $\theta = 90°$, so $|k| > k_0$ represents imaginary angles of propagation (evanescent modes). It is required then that the source function be tapered within the limits of $\left|\frac{k}{k_0}\right| < 1$.

Finally, we consider the starting field appropriate for a line array of sources. Assuming the array can be approximated by a continuous line array of length L (the separation between adjacent sources must be « λ_0), it is easy to show the directivity of such a source is defined by

$$D = \frac{\sin\left(\frac{k}{k_0}w\right)}{\frac{k}{k_0}w}, \quad w = \frac{k_0 L}{2}. \tag{7.16}$$

As required from our previous comments, this has a maximum value of unity at $k = 0$ (horizontal). However, the amplitude of this type of source function must be altered to account for the directional enhancement. In other words, the source amplitude should correspond to an equivalent point source, so we must modify the source level by the directivity factor corresponding to a linear array,

$$DI = 10\log Q \tag{7.17}$$

where

$$Q = \frac{1}{4\pi}\int D^2 d\Omega \ . \tag{7.18}$$

The source level of an equivalent point source is then $SL' = SL + DI$, so the amplitude factor of the source function becomes

$$\alpha' = \alpha Q^{1/2} \ . \tag{7.19}$$

Assuming $k_0 L \gg 1$, Eq. (7.18) yields

$$Q \approx \frac{2L}{\lambda_0} = \frac{k_0 L}{\pi} \ . \tag{7.20}$$

The three types of sources described above are available options for generating a starting field in the UMPE model. Before continuing, we shall review these functions which shall now be written in the general form

$$\hat{\psi}_0(k) = \alpha f\left(\frac{k}{k_0}\right) \ . \tag{7.21}$$

The subscript "0" on the k-domain starting field is meant to indicate that this formulation applies only for a free-field source at depth $z_S = 0$. We shall consider the effects of $z_S \neq 0$ and the image source shortly. Furthermore, we now demand that both α and $f(k / k_0)$ are pure real quantities (such that $\alpha \equiv |\alpha|$). This implies even symmetry, $\hat{\psi}_0(-k) = \hat{\psi}_0(k)$. The three starting field options can now be written explicitly as

1) Gaussian source:

$$f(x) = e^{-x^2/w^2} \ , \quad (w \equiv \text{user input width}),$$

$$\alpha = \Delta k \frac{1}{2}\left(\frac{2R_0}{\pi k_0}\right)^{1/2} \ ; \tag{7.22}$$

2) wide angle source:

$$f(x) = (1 - x^2)^{-1/4} \ ,$$

$$\alpha = \Delta k \frac{1}{2}\left(\frac{2R_0}{\pi k_0}\right)^{1/2} ; \tag{7.23}$$

3) line array source:

$$f(x) = \frac{\sin(xw)}{xw}, \quad w = \frac{k_0 L}{2},$$

$$\alpha = \Delta k \frac{1}{2}\left(\frac{2R_0}{\pi k_0}\right)^{1/2}\left(\frac{k_0 L}{\pi}\right)^{1/2} . \tag{7.24}$$

Note that an additional factor of Δk has been included in the normalization. This is needed for the discrete Fourier transform which assumes both functions $\hat{\psi}(k)$ and $\psi(z)$ are unitless.

We now consider the general form for the complete k-domain starting field. If we allow "steering" of the starting field, such that the k-domain formulation is about some wavenumber $k = k_c = k_0 \sin\theta_c \neq 0$, then it is fairly straightforward to show that the k-domain starting field for a source at depth $z = z_S$ is given by

$$\hat{\psi}(k) = e^{-ikz_S}\hat{\psi}_0(k - k_c) - e^{ikz_S}\hat{\psi}_0(k + k_c) . \tag{7.25}$$

It is apparent from this formula that the complete k-domain field has odd symmetry, $\hat{\psi}(-k) = -\hat{\psi}(k)$, and therefore the z-domain field must also have odd symmetry about $z = 0$ as required by the surface boundary condition. Eq. (7.25) together with Eqs. (7.21) through (7.24) constitute the complete formulation of the available starting fields in the UMPE model.

One may note an apparent ambiguity in the prescribed implementation of a steered source when the wide-angle source is used since $\hat{\psi}(k - k_c) = \hat{\psi}(k + k_c) = 1$ (neglecting the wide-angle correction factor (7.15)). This ambiguity arises when we attempt to steer a source which represents an omnidirectional source. We have removed this ambiguity by allowing the source filter function to be steered, thereby centering the filter about $k = k_c$.

As a final note, we must address the issue of mesh point symmetry in the UMPE model. The model assumes the usual mesh symmetry in the k-domain,

$$k(i) = \begin{cases} (i-1)\Delta k\ , & 1 \le i \le \frac{N}{2}+1 \\ -(N-i+1)\Delta k\ , & \frac{N}{2}+2 \le i \le N\ . \end{cases} \tag{7.26}$$

However, in the z-domain the UMPE model employs what is sometimes referred to as the "half-mesh" symmetry defined by

$$z(i) = \begin{cases} \left(i-\frac{1}{2}\right)\Delta z\ , & 1 \le i \le \frac{N}{2} \\ -\left(N-i+\frac{1}{2}\right)\Delta z\ , & \frac{N}{2}+1 \le i \le N\ . \end{cases} \tag{7.27}$$

Another way to contrast these symmetries is by noting that $\hat{\psi}(k(i=2)) = -\hat{\psi}(k(i=N))$, $\hat{\psi}(k(i=3)) = -\hat{\psi}(k(i=N-1))$, etc. while in the z-domain, $\psi(z(i=1)) = -\psi(z(i=N))$, $\psi(z(i=2)) = -\psi(z(i=N-1))$, etc. To determine how this affects the source function, consider the following. If we computed the source function according to (7.25), the transform of this function would produce a z-space starting field $\psi(z)$ with the same "whole-mesh" symmetry. Instead of $\psi(z)$, we wish to compute $\psi\left(z+\frac{1}{2}\Delta z\right)$. Therefore, we need to make the replacement

$$\hat{\psi}(k) \rightarrow \hat{\psi}_{(1/2)}(k) = \hat{\psi}(k)e^{ik\frac{\Delta z}{2}}\ . \tag{7.28}$$

Our final form for the k-domain starting field is then Eq. (7.25) corrected for the z-domain symmetry by (7.28). Note that this is the only time such a "correction" needs to be made. Once this symmetry is established, it remains throughout the calculation. Of course, to be consistent the z-domain potential function, hence the environment and all other calculations in the z-domain, must be computed using the half-mesh symmetry.

In addition to specifying one of the analytical formulas for the starting field listed above, the user may also input a generic source function. The input format should be compared with the source code's expected format. Also, the model will assume the correct half-mesh symmetry already exists, and the source has been normalized to yield a unit source level at R_0. (The true source level is already an input parameter.)

8. Filters and sponges

As has been mentioned briefly in previous sections, there are several places within the UMPE model which require the use of a filter, or "sponge," to produce the necessary removal of energy. The function used in the model is a sine-squared function which is smooth, has a continuous derivative, and goes from zero to unity within a finite region. The most obvious need for a filter we first recognize is the radiation condition $\psi(z) \to 0$ as $z \to \pm\infty$. Because the computational depth is finite, however, we must force the field amplitude to zero at the maximum depth. Note this also serves to eliminate wrap-around between the real and image oceans. This must be done in a smooth and relatively "slow" manner to avoid reflections from the filter function itself. As is common practice in many PE/SSF models, the UMPE model applies the sine-squared filter function to the bottom quarter of the computational depth of the real ocean. The user is responsible for ensuring that this does not interfere with propagating energy that would affect down-range results, i.e. the bottom depth should be well above the region of the filter. The net effect of this bottom filter, or bottom sponge, is to remove energy which has penetrated deep within the bottom layer before it reaches the maximum computational depth.

The bottom sponge is a necessary part of the calculation to match the radiation condition and is, therefore, always computed. Occasionally, one may wish to focus attention on bottom and/or sediment interface interactions. To isolate these phenomena from interfering surface reflected energy, it is desirable to "remove" the surface. The UMPE model allows this option and accomplishes the effect by copying the bottom sponge, inverting it, and placing it at the surface interface. The result is a sine-squared filter in the upper quarter of the real ocean depth. Again, the user is responsible for ensuring that the region of interest lies between the surface and bottom sponge regions. Once these sponges have been defined, they are included in the calculation of the z-space propagator function, $e^{-ik_0 \Delta r U_{op}(z)}$, as a multiplying factor.

In addition to requiring $\psi(z) \to 0$ as $z \to \pm z_{\max}$, we must also require $\hat{\psi}(k) \to 0$ as $k \to \pm k_{\max}$. The exact same filter used to create the bottom sponge is applied to the *k*-domain propagator function, $e^{-ik_0 \Delta r \hat{T}_{op}(k)}$. This results in a sine-squared filtering of the upper quarter of the positive *k* wavenumber spectrum. This filter is then inverted symmetrically about $k = 0$ and applied again to produce filtering of the lower quarter of the negative *k* wavenumber spectrum. In this way, wrap-around in *k*-space is avoided.

Finally, we shall consider the filter applied in the creation of the source function. In addition to the center angle and angular width, the input includes the angular tapering width of the source function. With these parameters, a filter function is computed which has a value of unity over the wavenumber range corresponding to the given angular width and centered at the wavenumber corresponding to the center angle of the source. At each end of this unity function is placed a sine-squared function of wavenumber width corresponding to the given angular tapering width. This filter is applied to the final form of the source function as described in the previous section. Note that this filter has little effect if a Gaussian source function is specified since the source amplitude will already be reduced by e^{-1} before the taper begins to apply. Similarly, an array source is a narrow wavenumber source and is not significantly effected by this filter. The main role of the source filter is to force $\hat{\psi}(k) \to 0$ as $k \to \pm k_{\max}$ for the wide angle source. It also allows the wide-angle source to be steered, creating greater versatility for this source option.

9. Calculations of transmission loss

The calculation of transmission loss has previously been defined by

$$TL(r, z) = 10\log\left(\frac{|p|^2}{P_0^2}\right) = 10\log\left[\frac{\rho R_0}{\rho_0 r}|\psi(r, z)|^2\right] . \tag{9.1}$$

Since the PE/SSF algorithm generates $\psi(z)$ sampled every Δz, the values for *TL* are also sampled every Δz at each range step separated by Δr. This grid of *TL* values can be output to allow the user to view the entire *TL* field which corresponds to the total pressure field. This requires an external plotting program be used.

In those situations where we desire to determine the transmission loss at a fixed depth (for example, at a fixed receiver depth or at the sediment or bottom interface) we must somehow interpolate the gridded *TL* values to this depth. This is performed using a five-point polynomial interpolation of the $\psi(z)$ field. The UMPE model allows the user to request the *TL* values output at either bottom interface and at up to ten arbitrary depths. Since a five-point interpolation is to be used, it is required that all depths be deeper than $2\frac{1}{2}\Delta z$. Actually, depths as shallow as $\frac{1}{2}\Delta z$ are allowed but only a 3-point or 2-point interpolation is then used.

The UMPE model also allows the user to request the transmission loss to the surface. We choose to define this *TL* value differently than Eq. (9.1) since obviously this is undefined at $\psi(z = 0) = 0$. Its derivation is analogous to the calculation of rough surface scatter in section 4.2.1 where it was shown that the first non-vanishing moment of the field is given by

$$\langle \frac{1}{k_0^2}\delta''(z)\rangle = \left|\frac{1}{k_0}\frac{\partial}{\partial z}\psi(z = z_S)\right|^2 \tag{9.2}$$

where we have included the scaling factor $1/k_0^2$ to produce a quantity with dimensions of ψ^2. We therefore define the surface transmission loss in terms of the dimensionless vertical derivative of the field, i.e.

$$\psi_S(r) = -\frac{i}{k_0}\frac{\partial\psi}{\partial z}\bigg|_{z = z_S(r)} . \tag{9.3}$$

This is effectively the measure of the field observed by a dipole receiver. Considering the wavenumber representation as in section 6, we see that (9.3) may be computed analytically by

$$\psi_S = \sum_k \left(\frac{k}{k_0}\right) \hat{\psi}_k e^{ikz_s} \tag{9.4}$$

where the summation is over all of wavenumber space.

10. Future upgrades

In the preceding sections, the physics of the UMPE model has been derived and numerical algorithms have been developed to produce solutions of the underwater acoustic pressure field. The ultimate goal of any acoustics model is the ability to take only environmental and source parameter inputs and return an accurate solution. However, as is typical of most numerical models, several ambiguous variables have been introduced into the UMPE model. Default values have been defined for all but the reference sound speed which would hopefully lead to an efficient and accurate result. A method of computing a default reference sound speed is described below which would eliminate the need for a "best guess" by the user. The corresponding accuracy of the solution provided by this method is still unclear.

Other upgrades to consider might follow from the introduction of new physics (e.g., internal wave scattering, effects of currents, backscattering or reverberation). The ability to compute the full 4-D field (3 spatial dimensions and time) is currently being investigated. To achieve this requires the inclusion of azimuthal coupling in the UMPE model. A basic description of the formulation of such a model is also given below.

Finally, one may consider various improvements in the numerical implementation that increase the speed and efficiency of the calculations. The most obvious extension of the model is a vectorized version of the source code allowing compilation on an array processor. This is also currently being developed. A large increase in speed may eventually allow implementation of the c_0-insensitive model mentioned in the first section of this report. However, the desire to compute long-range, multiple realization, multiple frequency, multiple coupled-azimuth solutions which would tend to absorb any increase in speed may outweigh the desire for what may be only slight improvements in accuracy.

10.1 Automatic c_0 selection

Recall that the choice of c_0, hence k_0, scales the wavenumbers or propagation angles according to $k = k_0 \sin\theta$. It also serves to scale the acoustic index of refraction

$n(r, z) = \dfrac{c_0}{c(r, z)}$ and, consequently, the environmental potential function $U_1(r, z)$. Tappert (1991c) has suggested one way to define a default reference sound speed by requiring

$$\frac{1}{z_b}\int_0^{z_b} U_1(r, z)dz = 0 \tag{10.1}$$

where z_b is the maximum depth of the water column. From this expression it is obvious that the reference sound speed may now be range-dependent. If, for example, we choose to employ the standard PE definition

$$U_1(r, z) = \frac{1}{2}\left(1 - \frac{c_0^2}{c^2(r, z)}\right) \tag{10.2}$$

then Eq. (10.1) leads to the defining equation for c_0 given by

$$\frac{1}{c_0^2} = \frac{1}{z_b}\int_0^{z_b} \frac{dz}{c^2(r, z)} \quad . \tag{10.3}$$

If instead we choose the wide-angle PE definition

$$U_1(r, z) = \left(1 - \frac{c_0}{c(r, z)}\right) \quad , \tag{10.4}$$

the defining equation for c_0 becomes

$$\frac{1}{c_0} = \frac{1}{z_b}\int_0^{z_b} \frac{dz}{c(r, z)} \quad . \tag{10.5}$$

Finally, employing the LOGPE definition for the environmental potential,

$$U_1(r,z) = -ln\left[\frac{c_0}{c(r,z)}\right] \tag{10.6}$$

leads to

$$ln[c_0] = \frac{1}{z_b}\int_0^{z_b} ln[c(r,z)]dz \quad . \tag{10.7}$$

If the sound speed profile or the bottom bathymetry is range-dependent, these definitions produce a range-dependent reference sound speed. Consequently, the reference wavenumber is now also range-dependent and so is the phase space potential function $\hat{T}_{op}\left(\frac{k}{k_0}\right)$. Thus, this approach requires the recalculation of the phase space propagator each time c_0 is updated, increasing the total run-time. Again, a trade-off between speed and accuracy may be required.

10.2 Azimuthal coupling

To develop a model that incorporates azimuthal coupling, we begin by returning to the original wave equation in cylindrical coordinates defined by Eq. (1.2) without a source term. Substituting (1.31) for the pressure field and keeping all terms leads to

$$\frac{\partial^2 u}{\partial r^2} + \frac{\partial^2 u}{\partial z^2} + \frac{1}{r^2}\frac{\partial^2 u}{\partial \varphi^2} + \left[k_0^2 n^2(r,z) + \frac{1}{4r^2}\right]u = 0 \quad . \tag{10.8}$$

Previously, both terms with the factors $\frac{1}{r^2}$ were dropped in the far-field approximation and azimuthal coupling was considered insignificant. If azimuthal coupling is important, the corresponding far-field expression is simply

$$\frac{\partial^2 u}{\partial r^2} + \frac{\partial^2 u}{\partial z^2} + \frac{1}{r^2}\frac{\partial^2 u}{\partial \varphi^2} + k_0^2 n^2(r,z)u = 0 \quad . \tag{10.9}$$

As before, we introduce the operators

$$P_{op} = \frac{\partial}{\partial r} \tag{10.10}$$

and

$$Q'_{op} = \left[n^2 + \frac{1}{k_0^2}\left(\frac{\partial^2}{\partial z^2} + \frac{1}{r^2}\frac{\partial^2}{\partial \varphi^2}\right)\right]^{1/2} \tag{10.11}$$

so (10.9) can be written in the more general form

$$(P_{op}^2 + k_0^2 Q'^2_{op})u = 0 \ . \tag{10.12}$$

Factoring (10.12) into incoming and outgoing waves produces the defining first order differential equation for the outgoing energy

$$-ik_0^{-1}\frac{\partial u}{\partial r} = Q'_{op}u \ . \tag{10.13}$$

We again define

$$\varepsilon = n^2 - 1 \tag{10.14}$$

and

$$\mu = \frac{1}{k_0^2}\frac{\partial^2}{\partial z^2} \tag{10.15}$$

and introduce

$$\nu = \frac{1}{k_0^2 r^2}\frac{\partial^2}{\partial \varphi^2} \tag{10.16}$$

such that

$$Q'_{op} = (\mu + \varepsilon + \nu + 1)^{1/2} \ . \tag{10.17}$$

If we assume that the effect of azimuthal coupling is relatively small (since $\frac{\partial}{\partial \varphi}c(r, z, \varphi) \ll \frac{\partial}{\partial z}c(r, z, \varphi)$, in general) then we may expect $\nu \ll \mu + \varepsilon$. Therefore, (10.17) may be approximated by

$$Q'_{op} = (\mu + \varepsilon + 1)^{1/2} + \frac{1}{2}\nu = Q_{op} + \frac{1}{2}\nu \ . \tag{10.18}$$

We now define the PE field function $\psi(r, z, \varphi)$ according to Eq. (1.45) and obtain

$$\frac{\partial\psi}{\partial r} = ik_0\left(Q_{op} + \frac{1}{2}\nu - 1\right)\psi = -ik_0\left(T_{op} + U_{op} - \frac{1}{2k_0^2 r^2}\frac{\partial^2}{\partial\varphi^2}\right)\psi \tag{10.19}$$

where the last step follows by replacing Q_{op} by the kinetic and potential energy operators as in section 1. These operators can now take any of the approximate forms previously defined.

The effect of including azimuthal coupling is to add a new operator to the propagation equation which we shall define as

$$V_{op} = -\frac{1}{2k_0^2 r^2}\frac{\partial^2}{\partial\varphi^2} \ . \tag{10.20}$$

Analogous to the operator T_{op} in z-space, this operator is not diagonal in φ-space and is therefore not a simple multiplication operator. Just as T_{op} coupled vertical angles of propagation, V_{op} produces a coupling of azimuthal angles. Following the same reasoning as before, we define a variable s as the "wavenumber" analog to the azimuthal bearing φ according to

$$\psi(r, z, \varphi) = \int \hat{\psi}(r, z, k_0 s) e^{ik_0 s\varphi} d(k_0 s) = \mathrm{FFT}[\hat{\psi}(r, z, k_0 s)] \ . \tag{10.21}$$

The propagator that couples azimuths is then $e^{-ik_0\Delta r\hat{V}_{op}(r, s)}$ where

$$\hat{V}_{op}(r, s) = \frac{s^2}{2r^2} \ . \tag{10.22}$$

As before, we assume the commutator of U_{op} and V_{op} is negligible. Note, however, that the commutator of T_{op} and V_{op} is identically zero so the order of their operations is arbitrary.

We now must examine more closely the explicit calculation of the variable s. To do so, we continue with the analogy to the vertical wavenumber variable k. The incremental values of vertical wavenumber were defined by

$$k = n\Delta k \ , \ n = -\left(\frac{N}{2} - 1\right), \frac{N}{2} \tag{10.23}$$

where N is the transform size of the FFT between z-space and k-space, and with

$$\Delta k = \frac{2\pi}{z_{\mathrm{T}}} = \frac{\pi}{\mathrm{H}} \tag{10.24}$$

where $z_T = 2H$ is the total computational depth. (The depth of the water column plus bottom layers is H and the total computational depth includes the real and image ocean or 2H.) Since $\Delta z = \frac{2H}{N}$, it follows that

$$\Delta k \Delta z = \frac{2\pi}{N} \quad . \tag{10.25}$$

Therefore, if we define M as the transform size between φ-space and s-space then

$$s = m\Delta s \; , \; m = -\left(\frac{M}{2} - 1\right), \frac{M}{2} \tag{10.26}$$

with

$$k_0 \Delta s = \frac{2\pi}{\varphi_T} \tag{10.27}$$

and φ_T is the total azimuthal interval in radians (e.g., 2π, $\frac{\pi}{2}$, etc.). The incremental values of azimuth are obviously given by

$$\Delta\varphi = \frac{\varphi_T}{M} \tag{10.28}$$

so the azimuthal analog to (10.25) is

$$k_0 \Delta s \Delta\varphi = \frac{2\pi}{M} \quad . \tag{10.29}$$

To compute the 3-D propagation of a point source, we simply create identical starting fields for each bearing at $\varphi = m\Delta\varphi$. For $\varphi_T < 2\pi$, the entire z-space starting fields should be tapered near the ends, or outermost bearings, near $\varphi = \pm\frac{M}{2}\Delta\varphi$. A 2-D FFT between $\psi(r, z, \varphi)$ and $\hat{\psi}(r, k, s)$ is then used in conjunction with a split-step algorithm analogous to Eq. (1.29) with the operator substitutions

$$U_{op}(r, z) \rightarrow U_{op}(r, z, \varphi) \tag{10.30}$$

and

$$\hat{T}_{op}(r,k) \rightarrow \hat{T}_{op}(r,k) + \hat{V}_{op}(r,s) \quad . \tag{10.31}$$

The above discussion outlines the basic formulation of a fully 3-D PE/SSF model. Naturally, there are subtle issues that complicate the implementation of these equations. These are currently being addressed and it is hoped that a successful source code will be available within the next year.

11. Numerical implementation and organization

The source code for the UMPE model has been separated into various subroutines which are tasked with the major components of the SSF algorithm. The names of the main program and subroutines with brief descriptions are given below. Each has standard Fortran 77 form and can be compiled on Sun workstations using the included "makefile" script.

pemp.f - This is the main program of the UMPE model. It handles the majority of the input/output. It also forms the basic SSF algorithm and implements this over many range steps and for multiple frequencies and multiple rough interface realizations if required. The surface interface boundary condition is also imposed within this program.

initk.f - The source function is created in the wavenumber domain. The various source parameters required for this calculation are read by the main program and passed to this subroutine through a common block.

phase1.f - The k-domain propagator function is computed and passed to the main program. The tapering filter applied over the outer 1/8 of the positive and negative wavenumbers, and the real and image ocean depths, is created in this subroutine and used in the creation of the z-domain propagator. Because this version of the UMPE model assumes constant range step sizes and a constant reference sound speed c_0, this function does not change with range and can be computed prior to entering the main range loop.

phase2.f - The z-domain propagator function is computed and passed to the main program. Because this function contains all the information about the environment, the environmental data is input exclusively to this subroutine. This reduces the need to pass data into the subroutine from the main program. The depth interpolation of the sound speed profile as well as the range interpolation of the environment is performed here. If shear is present in either bottom layer, equivalent fluid properties are computed. The water column, sediment layer, and basement layer are then combined to form the environmental potential function and, subsequently, the z-space propagator.

ssi.f - Called by phase2.f, this subroutine performs the depth interpolation of the sound speed profile using a simple 1-2-1 filter. This smoothing may be enhanced by increasing the number of iterations of application of the filter.

etagen.f - Called by phase2.f if stochastically rough interfaces are requested. For exact surface scatter, the first derivative of the surface roughness is also computed.

ftime.f - When a broadband analysis is requested, the main program outputs to a single file the complex field function $\psi(z)$ at the appropriate range for each frequency. This subroutine is called by the main program after all frequencies have been computed to perform the necessary Fourier analysis and output the travel time calculations.

fft842.f - Performs all of the Fourier transforms required in the model. It computes only a forward FFT. The maximum complex array size allowed is 4096 points as written but can be upgraded to larger array sizes if one is careful to be consistent throughout the source code.

tloss.f - Determines values of *TL* at arbitrary depths by performing a five-point interpolation of the field. The square modulus of the field function $\psi(z)$ at any given depth is passed back to the main program.

ave.f - Performs range averages of several output data options at the end of the main program.

datime.f, qtime.f, qdate.f - Used for timing purposes, these subroutines allow the main program to output a time and date stamp at the beginning and end of each run.

There are three input files for the UMPE model. The names of these files with brief descriptions are given below. Sample listings can be found in the following section of examples. Only the name of the first file must remain the same as it is assumed by the main program.

pefiles.inp - This file contains the names of the various I/O files and flags indicating which data should be computed and output. The first two files named contain the various run parameters for the model and the environmental data, respectively (these names may be changed). These will be discussed in more detail below. The remaining filenames define the names of various output files to be created by the main program (these names may also be changed). Each filename or set of associated filenames is preceded by one or more integers. The first integer indicates whether the following data file should be computed and output (0=no, 1=yes). The second integer, if it exists, specifies the frequency number (during broadband calculations) for which this calculation should be made. The user should bear in mind the ordering of the frequency calculations begins with the center frequency, increases to the maximum followed directly by the minimum and increases back to the center. The third and fourth integers, if either exists, specifies the roughness realization and range step number, respectively, for which this calculation should be made. A set of filenames are each associated with the same function but are defined to display different characteristics. In the order that they appear in pefiles.inp, the output filenames and a brief description of their content are as follows:

tlb.dat, tlbave.dat - *TL* to the bottom (basement) interface and its associated range average. The output is a two column ascii file of range, *TL* values. The range units are [km] while the *TL* units are [dB re 1m].

tlbs.dat, tlbsave.dat - *TL* to the bottom sediment interface and its associated range average. The output is a two column ascii file of range, *TL* values. The range units are [km] while the *TL* units are [dB re 1m].

tls.dat, tlsave.dat - *TL* to the surface interface and its associated range average (as defined by Eqs. (9.1) and (9.4)). The output is a two column ascii file of range, *TL* values. The range units are [km] while the *TL* units are [dB re 1m].

apvelrr.bin - The real part of the range component of acoustic particle velocity. The output is an unformatted binary file of acoustic particle velocity values. The ordering is standard row-column such that the first row contains the values at all requested depths for the initial output range and the last row contains the values at all requested depths for the final output range. The velocity units are [m/s].

apvelri.bin - The imaginary part of the range component of acoustic particle velocity. The output is an unformatted binary file of acoustic particle velocity values. The ordering is standard row-column such that the first row contains the values at all requested depths for the initial output range and the last row contains the values at all requested depths for the final output range. The velocity units are [m/s].

apvelzr.bin - The real part of the depth component of acoustic particle velocity. The output is an unformatted binary file of acoustic particle velocity values. The ordering is standard row-column such that the first row contains the values at all requested depths for the initial output range and the last row contains the values at all requested depths for the final output range. The velocity units are [m/s].

apvelzi.bin - The imaginary part of the depth component of acoustic particle velocity. The output is an unformatted binary file of acoustic particle velocity values. The ordering is standard row-column such that the first row contains the values at all requested depths for the initial output range and the last row contains the values at all requested depths for the final output range. The velocity units are [m/s].

prssrr.bin - The real part of the complex acoustic pressure. The output is an unformatted binary file of acoustic particle velocity values. The ordering is standard row-column such that the first row contains the values at all requested depths for the initial output range and the last row contains the values at all requested depths for the final output range. The pressure units are [Pa].

prssri.bin - The imaginary part of the complex acoustic pressure. The output is an unformatted binary file of acoustic particle velocity values. The ordering is standard row-column such that the first row contains the values at all requested depths for the initial output range and the last row contains the values at all requested depths for the final output range. The pressure units are [Pa].

phase1r.dat, phase1i.dat - The real and imaginary parts of the k-domain propagator function. The output is a two column ascii file of wavenumber, function values. The wavenumber units are [1/m]. (The propagator is unitless.)

phase2r.dat, phase2i.dat, ssp.dat, potent.dat - The real and imaginary parts of the z-domain propagator, the total interpolated sound speed profile, and the total potential function. Each output is a two column ascii file of depth, function values. The depth units are [m], the sound speed units are [m/s]. (The propagator and potential are unitless.)

psikr.dat, psiki.dat, psizr.dat, psizi.dat, psizt.dat - The real, imaginary parts of the k-domain field function $\hat{\psi}(k)$ and the real, imaginary, and total amplitude of the z-domain field function $\psi(z)$. Each output is a two column ascii file of either wavenumber or depth, function values.

tlgmt.bin, botgmt.dat, sedgmt.dat, surfgmt.dat - The complete range-depth field of *TL* data, the bottom (basement) interface depth, the sediment interface depth, and the surface interface depth. The first file is an unformatted binary file of transmission loss values. The ordering is standard row-column such that the first row contains the values at all requested depths for the initial output range and the last row contains the values at all requested depths for the final output range. The *TL* units are [dB re 1m]. The remaining files are two column ascii files of range [km], interface depth [m] values.

finfield.dat - A two column file specifying the real and imaginary parts of the field function $\psi(z)$ at the end of the calculation for all computational depths.

The first two files named in pefiles.inp contain all the input data for the UMPE model. As examples, we shall refer to them here as perun.inp and peenv.inp. Brief descriptions of their contents are as follows:

perun.inp - Note that each line has a brief description included to remind the user what each parameter defines. These descriptions should not be changed. The code assumes the data is given in this order and follows the 33rd column of each line (a colon exists at the 33rd column). Otherwise, the data is given in free format with the exception that integer inputs are assumed to be given as integers. The parameters are:

Title - An ascii title used to simply remind the user of the problem for which these parameters were defined.

Type of PE approx. (1-5) - The integer choice corresponds to the types of approximations described in section 1 of this report.

Type of density smoothing (1,2) - The integer choice defines the type of density mixing function to be employed, either a hyperbolic tangent (1) or a cubic spline (2), as described in section 2.1.2.

Type of starting field (1-3) - The integer choice defines the type of starting field to be employed, either a Gaussian (1), a wide-angle (2), or a vertical array (3), as described in section 7.

Type of input data units (1,2) - The integer choice defines the type of input data units, either standard MKS (1) (with ranges given in [km] rather than [m]) or English units (2) (with ranges given in [nm] rather than [ft]). Either set of units must be used consistently throughout both input data files. All output files are given in MKS units.

Apply surface filter? - The integer choice (0=no, 1=yes) determines whether a surface filter will be applied to remove surface reflections as described in section 8.

Approx. or exact surf? - The integer choice (0=approx, 1=exact) determines whether the approximate or exact method of computing surface scatter should be used as described in section 4.2.

# of roughness realizations - Integer number of stochastic interface roughness realizations for which calculations should be made. Outputs of *TL* at single depths or at interfaces are then averaged incoherently to yield a measure of the mean square pressure.

Range average *TL* curves? - The integer choice (0=no, 1=yes) determines whether the output of *TL* at single depths or at interfaces should be range averaged and output to a separate file.

Range to average over - Real number [km, nm] defining the range interval to perform a running range average if requested by preceding choice.

SL - Real source level [dB re mPa re (m, ft)].

Source depth - Real depth of source [(m, ft)].

Ref. sound speed - Real reference sound speed to be used in calculations [(m, ft)/s].

Center frequency - Real center frequency of source [Hz].

Number of frequencies - Integer number of frequencies for broadband calculations.

Frequency bandwidth - Real frequency bandwidth of source for broadband calculations [Hz].

FFT transform size - Integer number defining the size of the arrays in the PE/SSF algorithm (default = 0).

Range step size - Real size of range mesh for calculations [(km, nm)] (auto = 0.0).

Maximum range - Real max range of calculation [(km, nm)].

Max computational depth - Real max depth of calculation [(m, ft)].

Central source angle - Real angle describing steering of source function [deg].

Source array length - Real length of vertical array (used only when vertical array source is chosen) [(m, ft)].

Rms source width - Real angle defining width of source function [deg].

Source taper width - Real angle defining width of source function taper [deg].

Density mix length - Real number of depth meshes defining size of density mixing function (auto = 0.0).

Sspeed mix length - Real number of depth meshes defining size of sound speed mixing function (auto = 0.0).

Surf loss fctn width - Real number of depth meshes defining size of surface loss function used for the approximate rough surface scatter (auto = 0.0).

TL depths, *TL* depths - Integer number of depths (<= 10), and real depths that follow [(m, ft)], for which calculations of *TL* should be computed and output.

BB ranges, BB ranges - Integer number of ranges (<= 4), and real ranges that follow [(km, nm)], for which broadband calculations of time-domain *TL* field data should be computed and output.

BB extraction depth - Real single depth of *TL* -vs- time data to be extracted from each of the above BBTL ranges [(m, ft)].

Min, max output range, nskip - Real minimum and maximum ranges of any requested output fields [(km, nm)] and integer factor of computational range steps to skip in making the output file. (Note maximum total number of output ranges is 1000.)

Min, max output depth, nskip - Real minimum and maximum depths of any requested output fields [(m, ft)] and integer factor of computational depth mesh points to skip in making the output file.

peenv.inp - The following environmental parameters are contained in this file.

ISEED - An integer seed for random number generators.

NSS - Integer number of different sound speed profiles to follow (must be at least one). Each profile has the following format.

ZV, NSSDA - Real range of current profile (first profile must be at range 0.0) and integer number of sound speed values in depth given for current profile which follow.

SSD, SS - Real depth of this sound speed value and real sound speed value.

NB - Integer number of bottom profiles to follow (must be at least one). Each profile has the following format.

ZB, BD, BV, BG, RDEN, BLKMI, SIGBOT, ALCBO, BSWS, BSWLKMI - Range of this profile, depth, sound speed at top of this profile, sound speed gradient for this profile, density ratio w.r.t. water, attenuation, rms roughness of interface, correlation length of interface roughness, shear wave sound, speed, and shear wave attenuation. (All parameters are real.)

NBS - Integer number of sediment profiles to follow (may be zero). Each profile has the following format.

ZBS, BSD, BSV, BSG, RDENS, BSLKMI, SIGBOTS, ALCBOS, BSSWS, BSSWLKMI -

Range of this profile, sediment thickness, sound speed at top of this profile, sound speed gradient for this profile, density ratio w.r.t. water, attenuation, rms roughness of interface, correlation length of interface roughness, shear wave sound speed, and shear wave attenuation. (All parameters are real.) Note that sediment profiles are defined in terms of sediment thicknesses and not true depth. The sediments are layered on top of the bottom profiles.

NS - Integer number of surface roughness profiles to follow (may be zero). Each profile has the following format.

ZS, SIGSUR - Range of this profile, wind speed. (Internally, wind speed is converted into an rms roughness and correlation length scale as described in section 4.2)

NSB - Integer number of equivalent bubble surface roughness profiles to follow (may be zero). Each profile has the following format.

ZSB, SIGSURB - Range and equivalent rms roughness of this profile.

12. Examples

In this section, we present several examples of input files and output solutions. The first three were taken from problems defined in the PE Workshop II proceedings (Chin-Bing et al., 1993) and are designed to exhibit the accuracy of the UMPE model when default values are selected. Each of these has reference solutions with which we can compare the results of the model. The last three problems were chosen to exhibit some of the versatile features of the model.

Example 1: The first example taken from the PE Workshop II proceedings consists of a simple fluid half-space. The solution displays a simple pattern due to the interference of the image source and the real source. This problem is sometimes referred to as the Lloyd's mirror problem. It is listed as Test Case 1 in the workshop proceedings. The output requested from the UMPE model is the transmission loss at a single depth (defined in perun.inp) and the complete *TL* field data (requested in pefiles.inp). Note that since the UMPE model assumes at least one bottom interface, we simply define the bottom with fluid properties equivalent to the overlying fluid. In Fig. 1, the *TL* field data is displayed and the expected interference pattern can easily be seen. The comparison of the *TL* reference solution with the model prediction is shown in Fig. 2. (All figures have been created with the GMT plotting package (Wessel and Smith, 1992).)

Input files for Test Case 1 of PE Workshop II:

<u>perun.inp</u>

```
TITLE                                   :PEII test case 1
TYPE OF PE APPROX. (1-5)                :4
TYPE DENSITY SMOOTHING (1,2)            :2
TYPE OF STARTING FIELD (1-3)            :2
TYPE OF INPUT DATA UNITS (1,2)          :1
```

```
APPLY SURFACE FILTER? (0,1)              :0
APPROX OR EXACT SURF? (0,1)              :0
# OF ROUGHNESS REALIZATIONS              :1
RANGE AVERAGE TL CURVES? (0,1)           :0
RANGE TO AVERAGE OVER (km)               :1.0
SL(dB re uPa re 1m,1yd)                  :200.000
SOURCE DEPTH (m,ft)                      :350.000
REF. SOUND SPEED (m/s,ft/s)              :1500.00
CENTER FREQUENCY (Hz)                    :40.000
NUMBER OF FREQUENCIES                    :1
FREQUENCY BANDWIDTH (Hz)                 :32.0000
FFT TRANSFORM SIZE                       :0
RANGE STEP SIZE (km,nm)                  :0.0
MAXIMUM RANGE (km,nm)                    :10.000
MAX COMPUTATIONAL DEPTH (m,ft)           :6000.00
CENTRAL SOURCE ANGLE (DEG)               :0.
SOURCE ARRAY LENGTH (m,ft)               :20.0000
RMS SOURCE WIDTH (DEG)                   :80.0000
SOURCE TAPER WIDTH (DEG)                 :10.00000
DENSITY MIX LENGTH (# DELD)              :0.0
SSPEED MIX LENGTH (# DELD)               :0.0
SURF LOSS FCTN WIDTH (# DELD)            :0.0
# TL DEPTHS, TL DEPTHS (m,ft)            :1  3990.000
# BB RANGES, BB RANGES (km,nm)           :1  100.000
BB EXTRACTION DEPTH (m,ft)               :1000.000
MIN, MAX OUTPUT RANGE, NSKIP             :0.00 10.00 1
MIN, MAX OUTPUT DEPTH, NSKIP             :0.00 6000.00 1
```

<u>peenv.inp</u>

```
3000
1
0.0 2
  0.   1500.00
 10000.0  1500.
 1
0.0    3990.0  1500.00  0.0  1.0   .000  00.0  200.0  0.0  .000
 1
0.0       0.  1500.00  0.0  1.0   .000  00.0  200.0  0.0  .000
 1
0.0     0.0
 1
0.0     0.0
```

Example 2: This problem is listed as Test Case 2 in the workshop proceedings. It is characterized by a water column of varying depth overlying a denser fluid bottom. The interface forms an upslope-downslope configuration. Only transmission loss at two depths is requested as output. The results are compared with the reference solutions in Fig. 3.

Input files for Test Case 2 of PE Workshop II:

<u>perun.inp</u>

```
TITLE                                   :PEII test case 2
TYPE OF PE APPROX. (1-5)                :4
TYPE DENSITY SMOOTHING (1,2)            :2
TYPE OF STARTING FIELD (1-3)            :2
TYPE OF INPUT DATA UNITS (1,2)          :1
APPLY SURFACE FILTER? (0,1)             :0
APPROX OR EXACT SURF? (0,1)             :0
# OF ROUGHNESS REALIZATIONS             :1
RANGE AVERAGE TL CURVES? (0,1)          :0
RANGE TO AVERAGE OVER (km)              :1.0
SL(dB re uPa re 1m,1yd)                 :200.000
SOURCE DEPTH (m,ft)                     :100.000
REF. SOUND SPEED (m/s,ft/s)             :1500.00
CENTER FREQUENCY (Hz)                   :25.000
NUMBER OF FREQUENCIES                   :1
FREQUENCY BANDWIDTH (Hz)                :32.0000
FFT TRANSFORM SIZE                      :0
RANGE STEP SIZE (km,nm)                 :0.0
MAXIMUM RANGE (km,nm)                   :7.000
MAX COMPUTATIONAL DEPTH (m,ft)          :800.00
CENTRAL SOURCE ANGLE (DEG)              :0.
SOURCE ARRAY LENGTH (m,ft)              :20.0000
RMS SOURCE WIDTH (DEG)                  :60.0000
SOURCE TAPER WIDTH (DEG)                :20.00000
DENSITY MIX LENGTH (# DELD)             :0.0
SSPEED MIX LENGTH (# DELD)              :0.0
SURF LOSS FCTN WIDTH (# DELD)           :0.0
# TL DEPTHS, TL DEPTHS (m,ft)           :2  20.000  150.000
# BB RANGES, BB RANGES (km,nm)          :1  100.000
BB EXTRACTION DEPTH (m,ft)              :1000.000
MIN, MAX OUTPUT RANGE, NSKIP            :0.00 7.00 1
MIN, MAX OUTPUT DEPTH, NSKIP            :0.00 800.00 1
```

<u>peenv.inp</u>

```
 3000
1
0.0 2
  0.   1500.00
 10000.0  1500.
 3
0.0    200.0  1700.00  0.0  1.5   .294  00.0  200.0  0.0  .000
3.5     25.0  1700.00  0.0  1.5   .294  00.0  200.0  0.0  .000
7.0    200.0  1700.00  0.0  1.5   .294  00.0  200.0  0.0  .000
 1
0.0      0.  1500.00  0.0  1.0   .000  00.0  200.0  0.0  .000
 1
0.0     0.0
 1
0.0     0.0
```

Example 3: The problem is listed as Test Case 7 in the workshop proceedings and is sometimes referred to as the "Porter's duct" problem. The environment consists of a typical deep ocean sound speed channel with a shallow surface duct. The bottom is a simple fast fluid. The output requested from the UMPE model is the transmission loss at a single depth and the complete *TL* field data. Additionally, we requested as output the environmental propagator functions. This allows us to plot the extrapolated sound speed profile used by the model. In Fig. 4, the *TL* field data is displayed and the sound speed profile is shown in Fig. 5. The comparison of the *TL* reference solution with the model prediction is shown in Fig. 6 for two different values of reference sound speed. For this particular problem, the solution is found to be highly sensitive to changes in c_0 when the wide angle approximation is chosen.

Input files for Test Case 7 of PE Workshop II:

<u>perun.inp</u>

```
TITLE                                  :PEII test case 7
TYPE OF PE APPROX. (1-5)               :4
TYPE DENSITY SMOOTHING (1,2)           :2
TYPE OF STARTING FIELD (1-3)           :2
TYPE OF INPUT DATA UNITS (1,2)         :1
APPLY SURFACE FILTER? (0,1)            :0
APPROX OR EXACT SURF? (0,1)            :0
# OF ROUGHNESS REALIZATIONS            :1
RANGE AVERAGE TL CURVES? (0,1)         :0
RANGE TO AVERAGE OVER (km)             :1.0
SL(dB re uPa re 1m,1yd)                :200.000
SOURCE DEPTH (m,ft)                    :25.000
REF. SOUND SPEED (m/s,ft/s)            :1483.00
CENTER FREQUENCY (Hz)                  :80.000
NUMBER OF FREQUENCIES                  :1
FREQUENCY BANDWIDTH (Hz)               :32.0000
FFT TRANSFORM SIZE                     :0
RANGE STEP SIZE (km,nm)                :0.0
MAXIMUM RANGE (km,nm)                  :150.000
MAX COMPUTATIONAL DEPTH (m,ft)         :6000.00
CENTRAL SOURCE ANGLE (DEG)             :0.
SOURCE ARRAY LENGTH (m,ft)             :20.0000
RMS SOURCE WIDTH (DEG)                 :60.0000
SOURCE TAPER WIDTH (DEG)               :20.00000
DENSITY MIX LENGTH (# DELD)            :0.0
SSPEED MIX LENGTH (# DELD)             :0.0
SURF LOSS FCTN WIDTH (# DELD)          :0.0
```

```
# TL DEPTHS, TL DEPTHS (m,ft)            :1  100.000
# BB RANGES, BB RANGES (km,nm)           :1  100.000
BB EXTRACTION DEPTH (m,ft)               :1000.000
MIN, MAX OUTPUT RANGE, NSKIP             :0.00 150.00 12
MIN, MAX OUTPUT DEPTH, NSKIP             :0.00 6000.00 1
```

<u>peenv.inp</u>

```
 3000
1
0.0 20
  0.    1497.00
  250.    1502.00
  300.    1485.00
  375.    1478.00
  425.    1477.00
  500.    1476.00
  600.    1476.50
  700.    1477.00
  810.    1478.00
  900.    1479.00
  1000.   1480.00
  1100.   1481.00
  1180.   1482.00
  1340.   1484.00
  1600.   1487.00
  1800.   1490.00
  2500.   1498.70
  3000.   1506.80
  4000.   1523.90
 10000.0  1523.90
 1
0.0    4000.0 1523.80  0.0  1.0   .066  00.0  200.0  0.0  .000
 1
0.0       0.  1500.00  0.0  1.0   .000  00.0  200.0  0.0  .000
 1
0.0     0.0
 1
0.0     0.0
```

Example 4: This example is chosen to display the effects of rough bottom interfaces on acoustic propagation. We have defined the environment as a typical deep ocean water column with a thin sediment layer overlying a hard bottom. The average sediment depth is 500 m. The rms roughness of the sediment interface is 10 m while the rms roughness of the bottom interface is 50 m. Both roughness spectra have been filtered from 10 km to 20 km to remove length scales longer than 20 km (this number is set within the subroutine etagen.f). The *TL* field data has been output and is plotted in Fig. 7. The transmission loss to each interface has been averaged over 1 km intervals and these curves are displayed in Fig. 8.

Input files for Example 4:

perun.inp

```
TITLE                                   :Rough bottom interfaces
TYPE OF PE APPROX. (1-5)                :4
TYPE DENSITY SMOOTHING (1,2)            :2
TYPE OF STARTING FIELD (1-3)            :2
TYPE OF INPUT DATA UNITS (1,2)          :1
APPLY SURFACE FILTER? (0,1)             :0
APPROX OR EXACT SURF? (0,1)             :0
# OF ROUGHNESS REALIZATIONS             :1
RANGE AVERAGE TL CURVES? (0,1)          :1
RANGE TO AVERAGE OVER (km)              :1.0
SL(dB re uPa re 1m,1yd)                 :200.000
SOURCE DEPTH (m,ft)                     :250.000
REF. SOUND SPEED (m/s,ft/s)             :1500.00
CENTER FREQUENCY (Hz)                   :100.000
NUMBER OF FREQUENCIES                   :1
FREQUENCY BANDWIDTH (Hz)                :32.0000
FFT TRANSFORM SIZE                      :1024
RANGE STEP SIZE (km,nm)                 :0.01
MAXIMUM RANGE (km,nm)                   :100.000
MAX COMPUTATIONAL DEPTH (m,ft)          :6000.00
CENTRAL SOURCE ANGLE (DEG)              :0.
SOURCE ARRAY LENGTH (m,ft)              :20.0000
RMS SOURCE WIDTH (DEG)                  :60.0000
SOURCE TAPER WIDTH (DEG)                :20.00000
DENSITY MIX LENGTH (# DELD)             :0.0
SSPEED MIX LENGTH (# DELD)              :0.0
SURF LOSS FCTN WIDTH (# DELD)           :0.0
# TL DEPTHS, TL DEPTHS (m,ft)           :1  100.000
# BB RANGES, BB RANGES (km,nm)          :1  100.000
BB EXTRACTION DEPTH (m,ft)              :1000.000
MIN, MAX OUTPUT RANGE, NSKIP            :0.00 100.00 10
MIN, MAX OUTPUT DEPTH, NSKIP            :0.00 6000.00 2
```

peenv.inp

```
 3000
1
0.0 30
 0.000000E+00   1.520840E+03
 1.000000E+01   1.520660E+03
 2.000000E+01   1.520280E+03
 3.000000E+01   1.519330E+03
 5.000000E+01   1.512440E+03
 7.500000E+01   1.505780E+03
```

```
 1.000000E+02   1.503170E+03
 1.250000E+02   1.501060E+03
 1.500000E+02   1.498310E+03
 2.000000E+02   1.491900E+03
 2.500000E+02   1.489080E+03
 3.000000E+02   1.486760E+03
 4.000000E+02   1.482140E+03
 5.000000E+02   1.479350E+03
 6.000000E+02   1.478510E+03
 7.000000E+02   1.478650E+03
 8.000000E+02   1.479040E+03
 9.000000E+02   1.479640E+03
 1.000000E+03   1.480270E+03
 1.100000E+03   1.481050E+03
 1.200000E+03   1.481890E+03
 1.300000E+03   1.482820E+03
 1.400000E+03   1.483740E+03
 1.500000E+03   1.484660E+03
 1.750000E+03   1.487460E+03
 2.000000E+03   1.490590E+03
 2.500000E+03   1.497990E+03
 3.000000E+03   1.506030E+03
 3.500000E+03   1.514420E+03
 1.000000E+04   1.514420E+03
 1
0.0    3500.0  2000.00  0.0  3.0   .100  50.0 1000.0  0.0  .000
 1
0.0     500.0  1700.00  0.0  1.5   .010  10.0  200.0  0.0  .000
 1
0.0     0.0
 1
0.0     0.0
```

Example 5: In this example, we exhibit the model's ability to compute the exact surface scatter from a large Gaussian displacement of the surface. (This displacement was defined analytically and included by specifically editing parts of the phase2.f subroutine. This is not available in the generic version but does illustrate how one may easily modify the code to include various features.) Note that this calculation could not be performed properly with the approximate method since the small surface displacement approximation is invalid. The water column is treated as a homogeneous half-space with the "bottom" having properties identical to the overlying fluid. In Fig. 9 we display the *TL* field data, and the surface transmission loss, TL_s, as defined by Eqs. (9.1) and (9.4), is shown in Fig. 10.

Input files for Example 5:

perun.inp

```
TITLE                                   :Gaussian surface displacement
TYPE OF PE APPROX. (1-5)                :4
TYPE DENSITY SMOOTHING (1,2)            :2
TYPE OF STARTING FIELD (1-3)            :2
TYPE OF INPUT DATA UNITS (1,2)          :1
APPLY SURFACE FILTER? (0,1)             :0
APPROX OR EXACT SURF? (0,1)             :1
# OF ROUGHNESS REALIZATIONS             :1
RANGE AVERAGE TL CURVES? (0,1)          :0
RANGE TO AVERAGE OVER (km)              :1.0
SL(dB re uPa re 1m,1yd)                 :200.000
SOURCE DEPTH (m,ft)                     :75.000
REF. SOUND SPEED (m/s,ft/s)             :1500.00
CENTER FREQUENCY (Hz)                   :250.000
NUMBER OF FREQUENCIES                   :1
FREQUENCY BANDWIDTH (Hz)                :32.0000
FFT TRANSFORM SIZE                      :1024
RANGE STEP SIZE (km,nm)                 :0.01
MAXIMUM RANGE (km,nm)                   :5.000
MAX COMPUTATIONAL DEPTH (m,ft)          :1500.00
CENTRAL SOURCE ANGLE (DEG)              :0.
SOURCE ARRAY LENGTH (m,ft)              :20.0000
RMS SOURCE WIDTH (DEG)                  :40.0000
SOURCE TAPER WIDTH (DEG)                :20.00000
DENSITY MIX LENGTH (# DELD)             :0.0
SSPEED MIX LENGTH (# DELD)              :0.0
SURF LOSS FCTN WIDTH (# DELD)           :0.0
# TL DEPTHS, TL DEPTHS (m,ft)           :0  100.000
# BB RANGES, BB RANGES (km,nm)          :1  100.000
BB EXTRACTION DEPTH (m,ft)              :1000.000
MIN, MAX OUTPUT RANGE, NSKIP            :0.00 5.00 1
MIN, MAX OUTPUT DEPTH, NSKIP            :0.00 1500.00 1
```

peenv.inp

```
 3000
1
0.0 2
 0.0     1500.00
10000.0   1500.00
 1
0.0    1200.0  1500.00  0.0  1.0   .000   0.0  200.0  0.0  .000
 1
0.0       0.0  1500.00  0.0  1.0   .000   0.0  200.0  0.0  .000
 1
0.0     0.0
 1
0.0     0.0
```

Example 6: In our final example, the broadband capabilities of the model are displayed. The environment we have chosen consists of a typical deep ocean sound speed profile overlying a hard, lossy bottom and is range-independent. As output we have selected the time domain *TL* field and a slice at the sound channel axis at 600 m. The normalization of the *TL* levels was performed after the calculation with the peak level after one range step. Because a range step is 10 m, the values are rescaled by subtracting $TL_{min}(r = 10 \text{ m}) - 20\log\left(\frac{r = 10 \text{ m}}{R_0}\right)$. The number of frequencies was chosen to yield a time window width of roughly 5 sec. A more narrow source function was defined to avoid high angle propagation that would arrive significantly later than the water-bourne energy. The time domain *TL* field, displayed in Fig. 11, exhibits the typical caustic structure for deep sound channel propagation. The first figure shows the full time window and the second expands the view to examine the earliest arrivals. Source angles beyond roughly $15°$ have long enough path lengths to introduce relatively long travel times. Such arrivals can be seen to arrive as much as 4 sec later than the horizontal axial arrival and are observed to create a wrap-around effect in the calculation. This provides a clear example of the need for a sufficient time window width. The time arrival pattern observed at the axis of the sound channel is displayed in Fig. 12. Again, an expanded view of the earliest arrivals is provided. Significant side-lobe energy is present due to the high ratio of bandwidth to center frequency.

Input files for Example 6:

<u>perun.inp</u>

```
TITLE                                :Deep ocean, broadband, range-independent
TYPE OF PE APPROX. (1-5)             :4
TYPE DENSITY SMOOTHING (1,2)         :2
TYPE OF STARTING FIELD (1-3)         :2
TYPE OF INPUT DATA UNITS (1,2)       :1
APPLY SURFACE FILTER? (0,1)          :0
APPROX OR EXACT SURF? (0,1)          :0
# OF ROUGHNESS REALIZATIONS          :1
RANGE AVERAGE TL CURVES? (0,1)       :0
RANGE TO AVERAGE OVER (km)           :1.0
SL(dB re uPa re 1m,1yd)              :200.000
SOURCE DEPTH (m,ft)                  :600.000
REF. SOUND SPEED (m/s,ft/s)          :1478.51
CENTER FREQUENCY (Hz)                :50.000
NUMBER OF FREQUENCIES                :256
FREQUENCY BANDWIDTH (Hz)             :50.0000
```

FFT TRANSFORM SIZE	:1024
RANGE STEP SIZE (km,nm)	:0.01
MAXIMUM RANGE (km,nm)	:100.000
MAX COMPUTATIONAL DEPTH (m,ft)	:6500.00
CENTRAL SOURCE ANGLE (DEG)	:0.
SOURCE ARRAY LENGTH (m,ft)	:20.0000
RMS SOURCE WIDTH (DEG)	:20.0000
SOURCE TAPER WIDTH (DEG)	:10.00000
DENSITY MIX LENGTH (# DELD)	:0.0
SSPEED MIX LENGTH (# DELD)	:0.0
SURF LOSS FCTN WIDTH (# DELD)	:0.0
# TL DEPTHS, TL DEPTHS (m,ft)	:0 100.000
# BB RANGES, BB RANGES (km,nm)	:2 0.01 100.000
BB EXTRACTION DEPTH (m,ft)	:600.000
MIN, MAX OUTPUT RANGE, NSKIP	:0.00 100.00 20
MIN, MAX OUTPUT DEPTH, NSKIP	:0.00 6500.00 2

peenv.inp

```
 3000
1
0.0 32
 0.000000E+00   1.520840E+03
 1.000000E+01   1.520660E+03
 2.000000E+01   1.520280E+03
 3.000000E+01   1.519330E+03
 5.000000E+01   1.512440E+03
 7.500000E+01   1.505780E+03
 1.000000E+02   1.503170E+03
 1.250000E+02   1.501060E+03
 1.500000E+02   1.498310E+03
 2.000000E+02   1.491900E+03
 2.500000E+02   1.489080E+03
 3.000000E+02   1.486760E+03
 4.000000E+02   1.482140E+03
 5.000000E+02   1.479350E+03
 6.000000E+02   1.478510E+03
 7.000000E+02   1.478650E+03
 8.000000E+02   1.479040E+03
 9.000000E+02   1.479640E+03
 1.000000E+03   1.480270E+03
 1.100000E+03   1.481050E+03
 1.200000E+03   1.481890E+03
 1.300000E+03   1.482820E+03
 1.400000E+03   1.483740E+03
 1.500000E+03   1.484660E+03
 1.750000E+03   1.487460E+03
 2.000000E+03   1.490590E+03
 2.500000E+03   1.497990E+03
 3.000000E+03   1.506030E+03
 3.500000E+03   1.514420E+03
 4.000000E+03   1.523210E+03
 4.500000E+03   1.532230E+03
 1.000000E+04   1.532230E+03
 1
0.0    4500.0  1700.00  0.0  2.0   .200  00.0  200.0  0.0  .000
 1
```

```
0.0      0.  1500.00  0.0  1.0   .000  00.0  200.0  0.0  .000
 1
0.0     0.0
 1
0.0     0.0
```

Acknowledgments

We thank Dr. Lan Nghiem-Phu and Dr. David J. Thomson for many helpful discussions during the development of this model. This work was supported by the Office of Naval Research (ONR Grant No. N00014-93-1-0062 and N00014-89-J-1156).

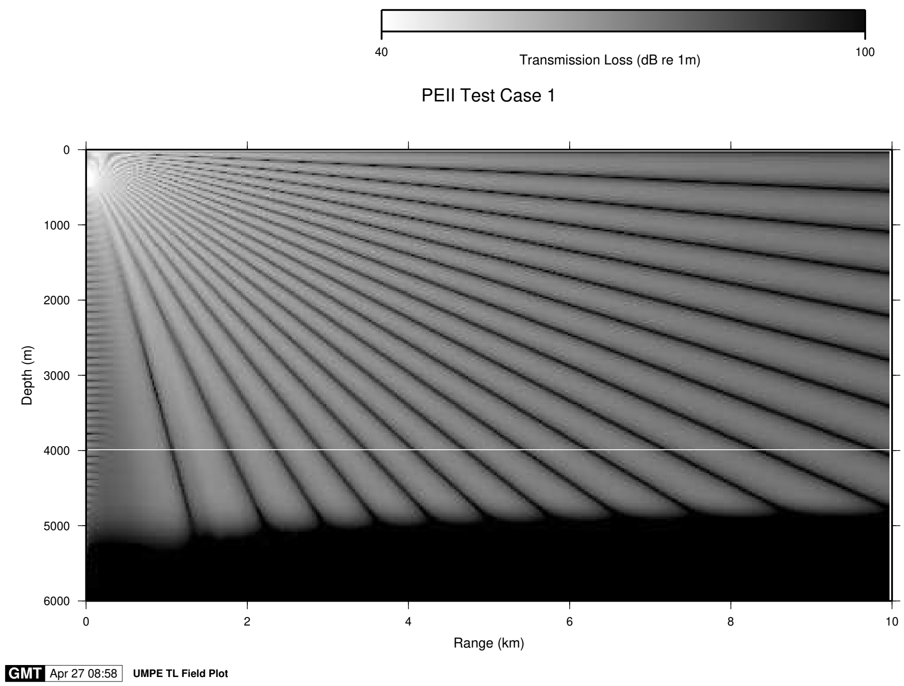


Figure 1: Transmission loss field plot calculated from UMPE model for Test Case 1 of PE Workshop II.

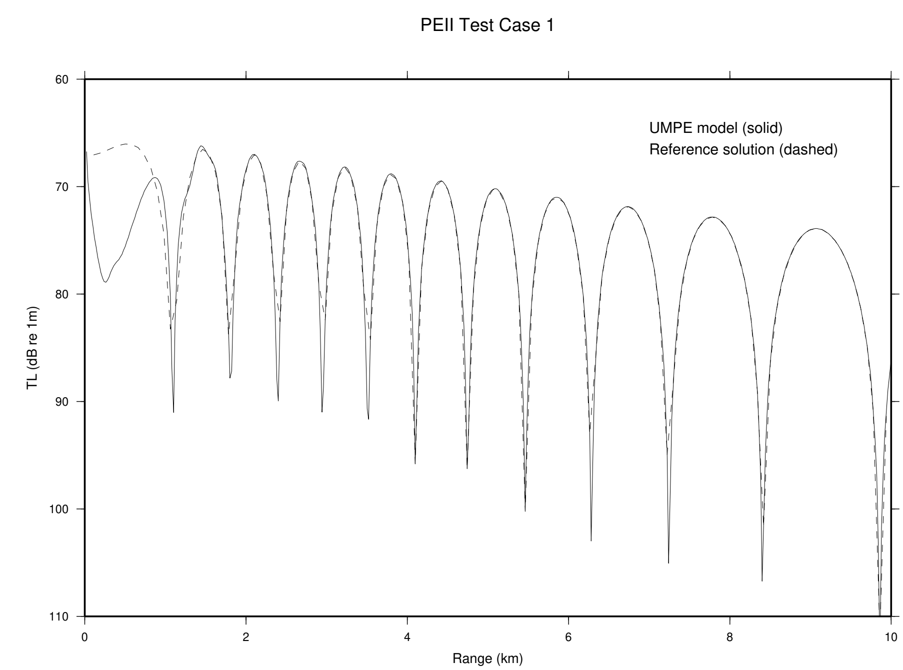
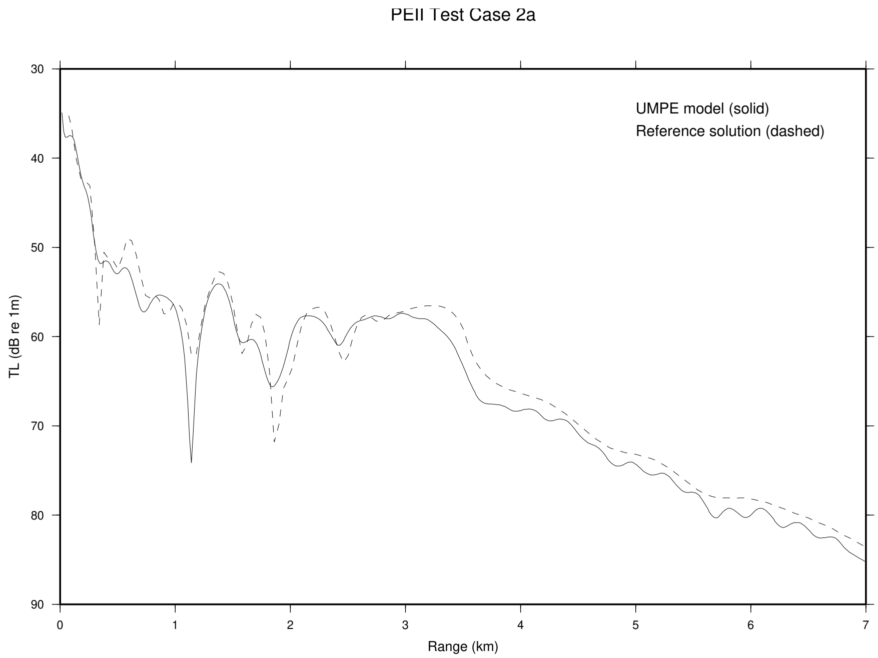


Figure 2: Comparison of *TL* line plots from UMPE model (solid curve) and reference solution (dashed curve) for Test Case 1 of PE Workshop II.



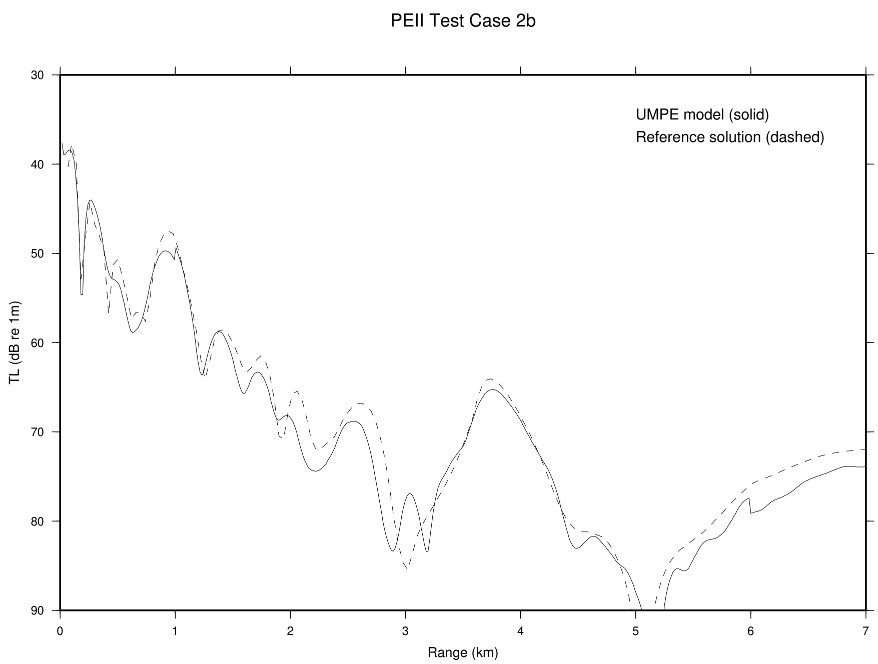


Figure 3: Comparison of *TL* line plots at two different depth from UMPE model (solid curves) and reference solutions (dashed curves) for Test Case 2 of PE Workshop II.

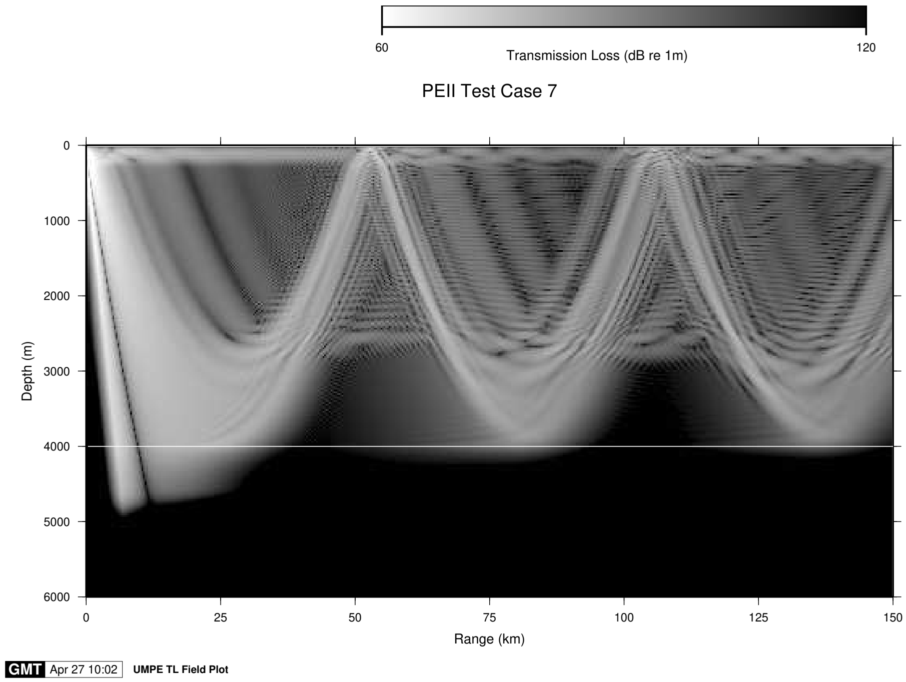


Figure 4: Transmission loss field plot calculated from UMPE model for Test Case 7 of PE Workshop II.

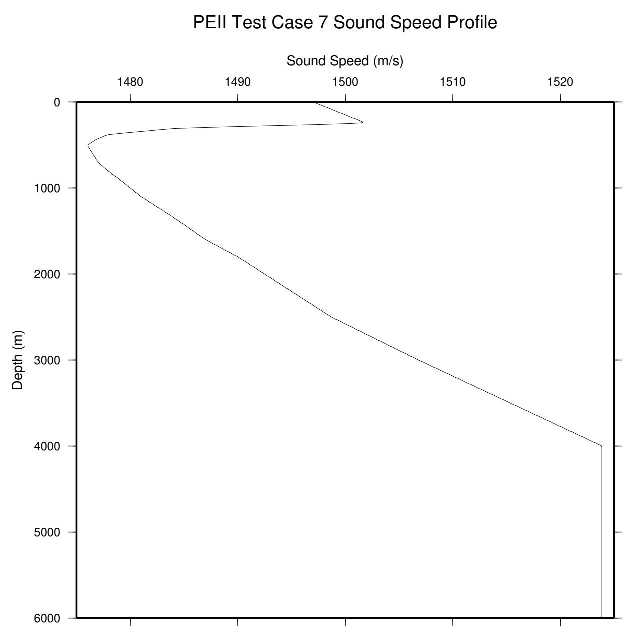
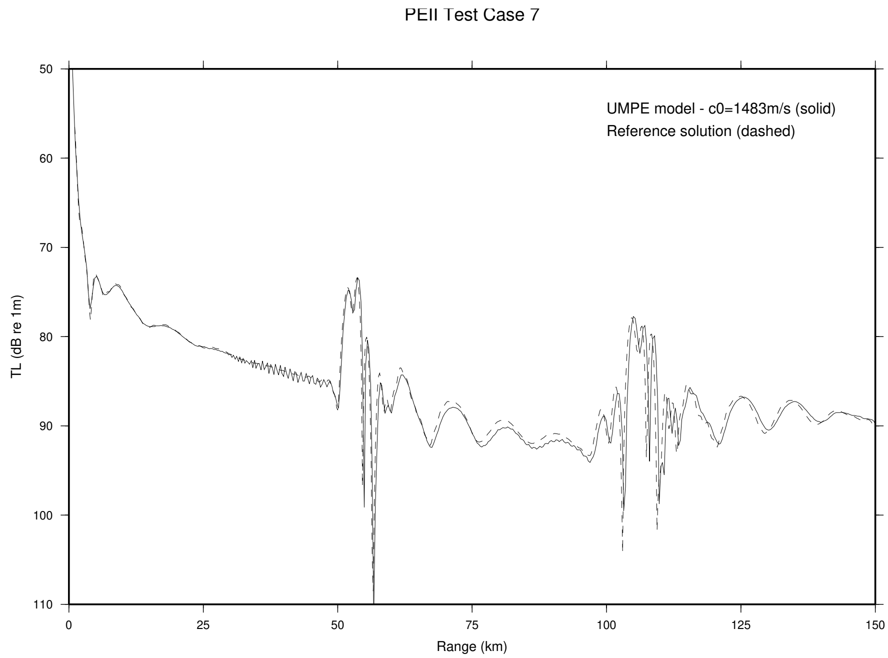


Figure 5: Extrapolated sound speed profile used by the UMPE model for Test Case 7 of PE Workshop II.



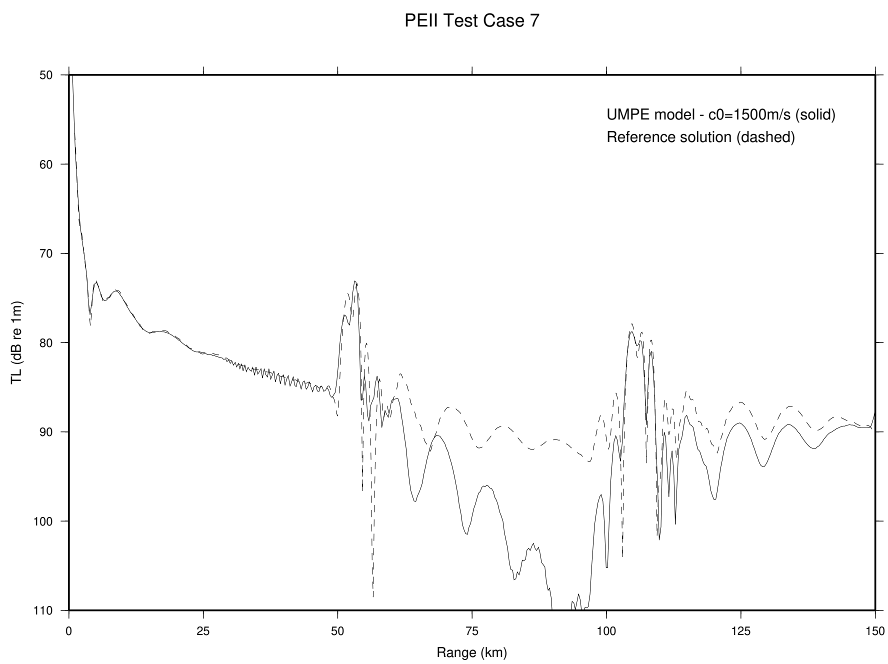


Figure 6: Comparison of *TL* line plots at two different reference sound speeds from UMPE model (solid curves) and reference solution (dashed curves) for Test Case 7 of PE Workshop II.

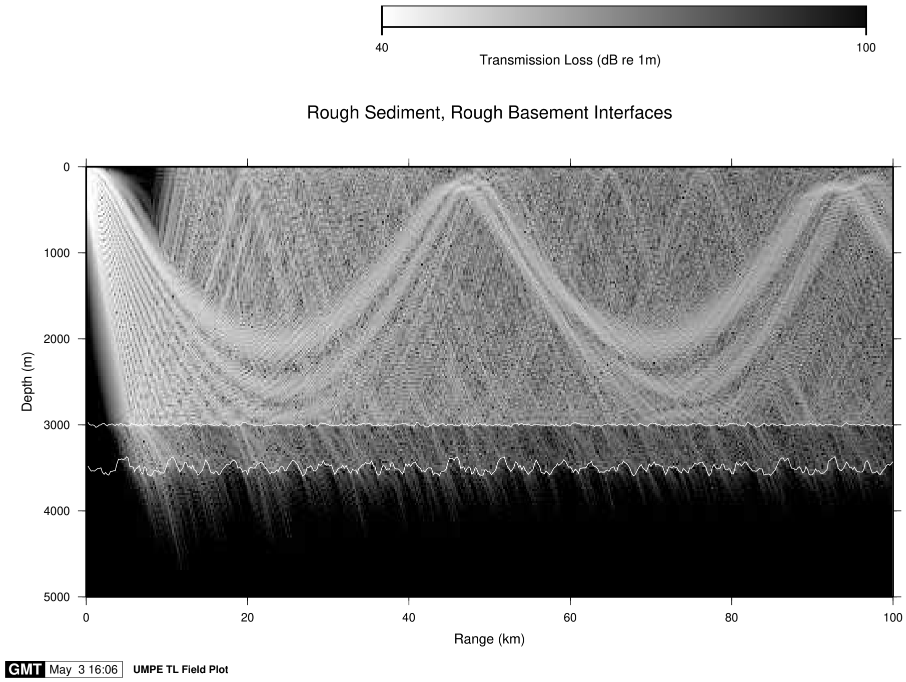


Figure 7: Transmission loss field plot displaying effects of forward scatter due to two rough bottom interfaces.

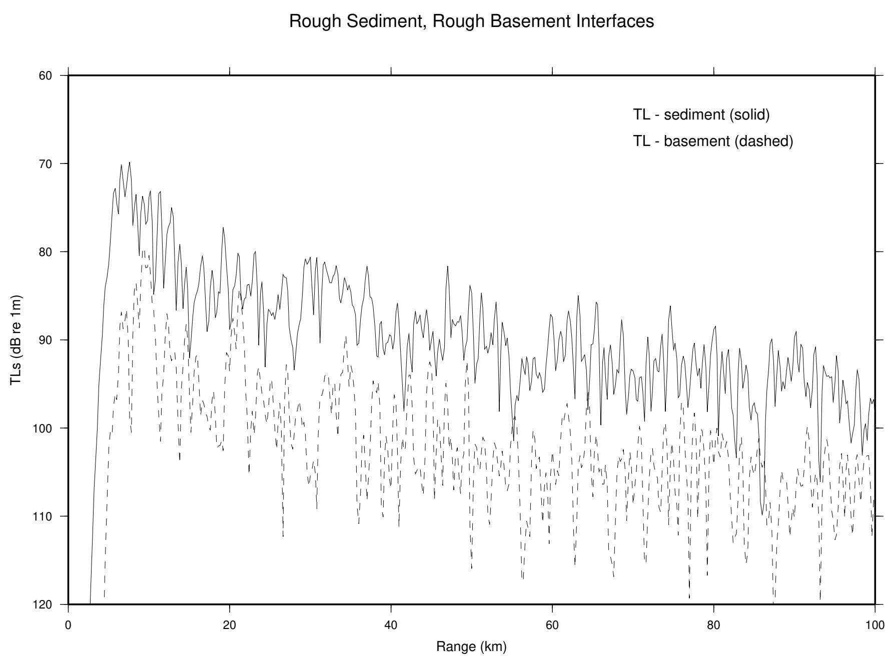


Figure 8: Plots of transmission loss computed to each rough interface.

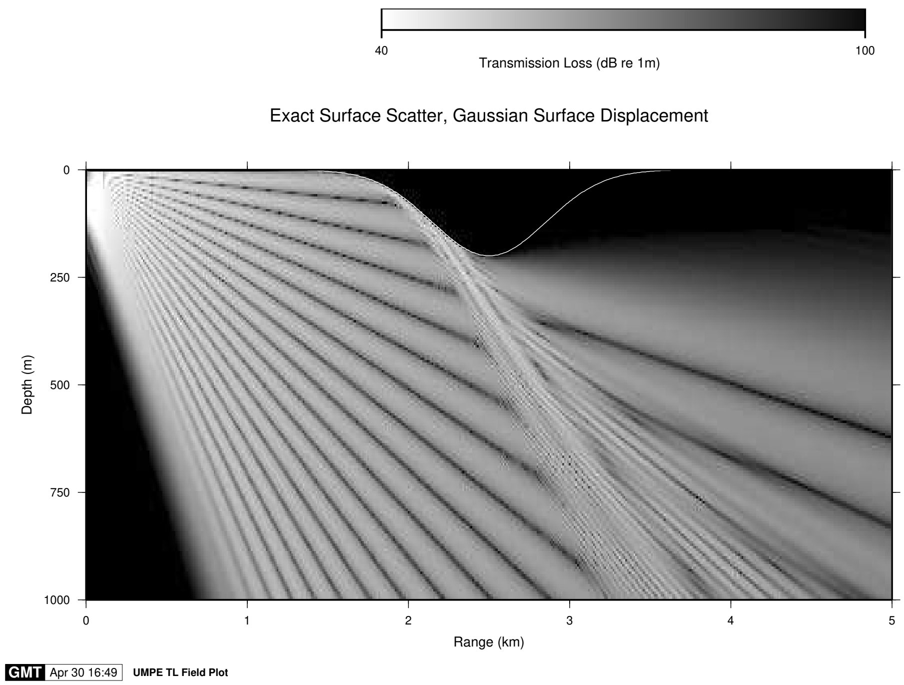


Figure 9: Transmission loss field plot displaying effects of exact surface scattering from a Gaussian surface displacement.

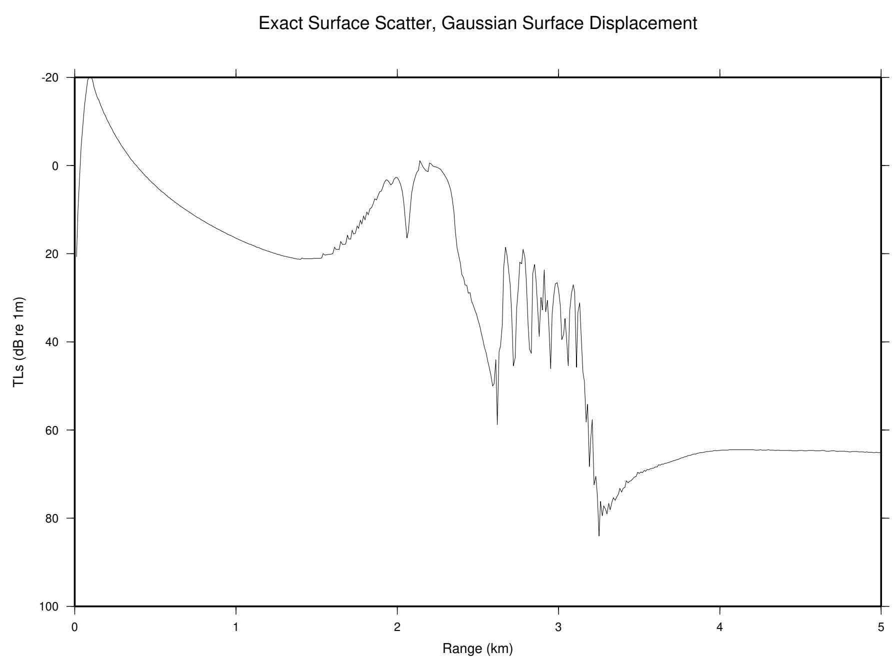


Figure 10: Plot of TL_s in presence of Gaussian surface displacement as defined by Eqs. (9.1) and (9.3) of the text.

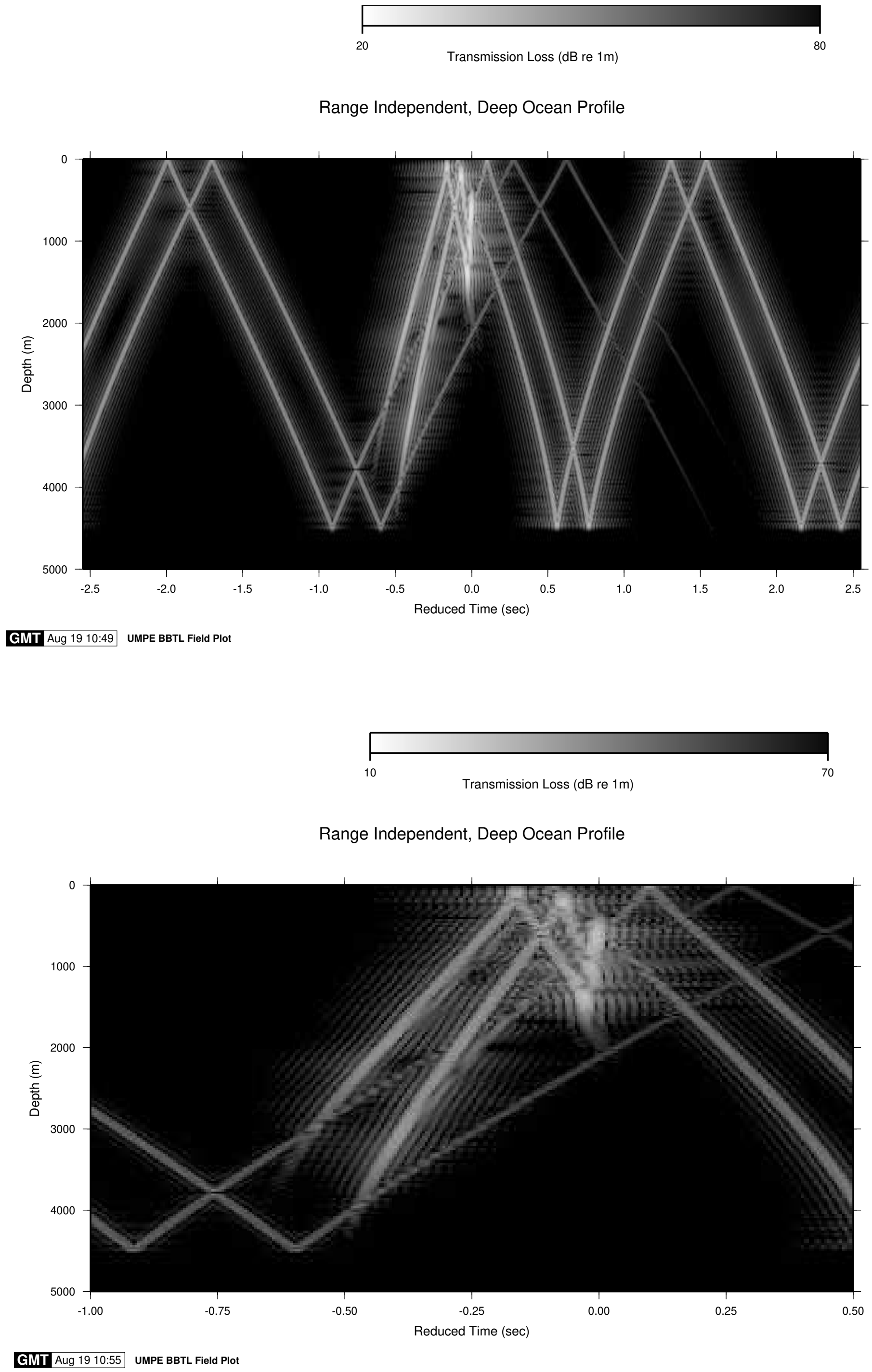
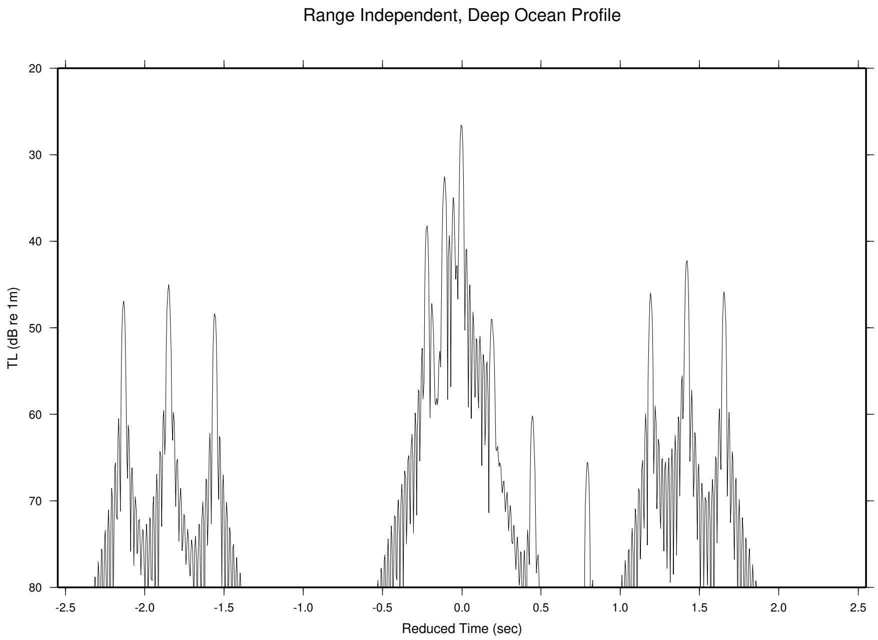


Figure 11: Transmission loss field plots in the time domain computed from a broadband UMPE model run in a typical deep ocean profile. The upper figure displays the total time window computed while the lower figure shows an expanded view of the earliest arrivals.



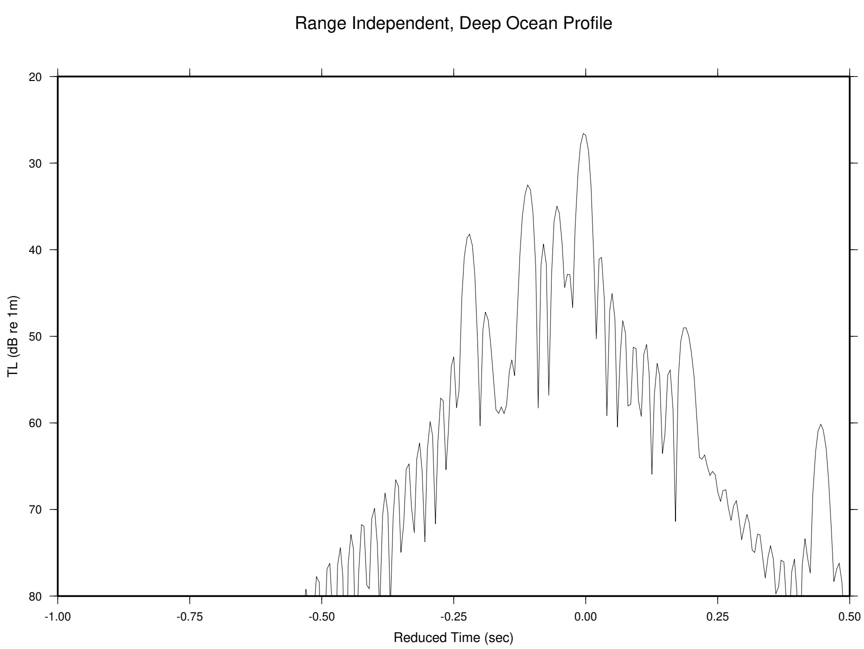


Figure 12: Time arrival patterns of *TL* at the channel axis for a typical deep ocean profile. The upper figure displays the total time window computed while the lower figure shows an expanded view of the earliest arrivals.

References

Bellman, R. (**1964**). *Perturbation Techniques in Mathematics, Physics, and Engineering* (Holt, Rinehart, and Winston, Inc., New York) pp. 38-39.

Berman, D. H., Wright, E. B. and Baer, R. N. (**1989**). "An optimal PE-type wave equation," J. Acoust. Soc. Am. **86**, 228-233.

Chin-Bing, S. A., King, D. B., Davis, J. A., and Evans, R. B., eds. (**1993**). "PE Workshop II: Proceedings of the 2nd Parabolic Equation Workshop, May 6-9, 1991," Naval Research Laboratory - Stennis Space Center, NRL Book Contribution.

Collins, M. D. (**1988**). "FEPE User's Guide," NORDA TN-365, Naval Ocean Research and Development Activity, Stennis Space Center, MS.

Feit, M. D. and Fleck, J. A. (**1978**). "Light propagation in graded-index fibers," Appl. Opt. **17**, 3990-3998.

Hardin, R. H. and Tappert, F. D. (**1973**). "Applications of the split-step Fourier method to the numerical solution of nonlinear and variable coefficient wave equations," SIAM Rev. **15**, 423.

Holmes, E. S. and Gainey, L. A. (**1991**). "Software product specification for the PE model, v.3.2," OAML-SPS-22 Naval Oceanography Office, Bay St. Louis, MS.

Lee, D. and Botseas, G. (**1982**). "IFD: An implicit finite-difference computer model for solving the parabolic equation," Rep. TR-6659, Naval Underwater Systems Center, New London, CT.

Lighthill, M. J. (**1958**). *Introduction to Fourier analysis and generalised functions* (Cambridge University Press, New York) pp. 79.

Neumann, G. and Pierson, W. J., Jr. (**1966**). *Principle of Physical Oceanography* (Prentice Hall, Englewood Cliffs, NJ) p. 351.

Nghiem-Phu, L. and Tappert, F. D. (**1985**). "Modeling of reciprocity in the time domain using the parabolic equation method," J. Acoust. Soc. Am. **78**, 164-171.

Sakurai, J. J. (**1985**). *Modern Quantum Mechanics* (Benjamin/Cummings, Menlo Park, CA), pp. 72-73.

Smith, K. B., Brown, M. G., and Tappert, F. D. (**1992**). "Ray chaos in underwater acoustics," J. Acoust. Soc. Am. **91**, 1939-1949.

Tappert, F. D. (**1974**). "Parabolic equation method in underwater acoustics," J. Acoust. Soc. Am. Suppl. 1 **55**, S34.

Tappert, F. D. (**1977**). "The parabolic approximation method," in Lecture Notes in Physics, Vol. 70, *Wave Propagation and Underwater Acoustics*, edited by J. B. Keller and J. S. Papadakis (Springer-Verlag, New York), pp. 224-287.

Tappert, F. D. and Nghiem-Phu, L (**1985**). "A new split-step Fourier algorithm for solving the parabolic wave equation with rough surface scattering," J. Acoust. Soc. Am. Suppl. 1 **77**, S101.

Tappert, F. D. (**1985**). "Parabolic equation modeling of shear waves," J. Acoust. Soc. Am. **78**, 1905-1906.

Tappert, F. D. (**1991a**). "Density in PE/SSF," unpublished lecture notes.

Tappert, F. D. (**1991b**). "Sensitivity to c_0 for Parabolic-Type Approximations," unpublished lecture notes.

Tappert, F. D. (**1991c**). "Automatic c_0 Selection for SPE," unpublished lecture notes.

Tappert, F. D. (**1991d**). "The c_0-Independent PE Model: Analysis of Sound Speed Discontinuity," unpublished lecture notes.

Thomson, D. J. and Chapman, N. R. (**1983**). "A wide-angle split-step algorithm for the parabolic equation," J. Acoust. Soc. Am. **74**, 1848-1854.

Thomson, D. J. and Bohun, C. S. (**1988**). "A wide-angle initial field for the parabolic equation models," J. Acoust. Soc. Am. Suppl. 1 **83**, S118.

Thorp, W. H. (**1967**). "Analytic description of the low-frequency attenuation coefficient," J. Acoust. Soc. Am. **42**, 270.

Tindle, C. T. and Zhang, Z. Y. (**1992**). "An equivalent fluid approximation for a low shear speed ocean bottom," J. Acoust. Soc. Am. **91**, 3248-3256.

Wessel, P. and Smith, W. H. F. (**1992**). *The GMT-SYSTEM Version 2.1 Technical Reference and Cookbook.*