Stanford Exploration Project, Report 129, May 6, 2007, pages 33–43

A tutorial on optimizing time domain finite-difference schemes: "Beyond Holberg"

John T. Etgen

ABSTRACT

Time-domain finite-difference solutions to the wave equation are a standard method for modeling seismic wave propagation, but they see limited use in data processing because they are expensive to compute. Given the resurgence of interest in reverse-time migration, anything that improves the performance and accuracy of the finite-difference method may be worthwhile. Additionally, the design problem for optimal finite-difference operators is a "simple and approachable" problem for teaching non-linear least squares. The standard solution for improving the finite-difference method is to use long spatial difference operators. However, there are computational reasons why it might be preferable to use spatially compact stencils, assuming they can be made accurate enough.

INTRODUCTION

All finite-difference solutions to wave equations in media with more than one spatial dimension have numerical phase and group velocities that differ from the desired phase and group velocities of the "true" solution. The errors manifest themselves as "dispersion"; different frequencies propagate at different speeds. This leads to progressively greater distortion of the propagating waveforms with increasing time and propagation distance. For example, the standard second-order in time, second-order in space finite-difference approximation to the 2D or 3D acoustic wave equation needs a large degree of oversampling of the spatial axes (ten points per wavelength or more) to achieve remotely reasonable accuracy. Thus, despite its simplicity, it's almost never used in practice. Finite difference methods that use spatial difference operators with more points, so called high-order finite-difference methods (Dablain, 1986; Etgen, 1986, 1988), are commonly used to improve the accuracy and efficiency of the method, as well as reduce its memory requirements, which is particularly important in 3D (Wu et al., 1996). Most original high-order finite-difference methods used a Taylor-series approach to derive spatial difference coefficients, while employing second-order differences in time to preserve the simplicity of a 3-point time marching scheme. Dablain (1986) is probably the earliest, most thorough description of this method. In the same paper, Dablain showed how to create a time marching algorithm that has fourth-order accuracy in time by applying the spatial difference operator in a two-term cascade and weighting the terms of the cascade appropriately to cancel the leading order error of the time discretization. The ultimate quest for accuracy often combines higher-order time expansion with the Fourier pseudo-spectral method (Reshef

et al., 1988) for the spatial derivatives. That approach can be generalized to many terms, ultimately leading to orthogonal-polynomial series solutions that have spectral accuracy (Tal-Ezer et al., 1987). The complexity involved in implementing these "very accurate" methods means that they don't see much large-scale practical application, particularly for reverse-time migration. They are also difficult to implement as "out-of-core" solutions or on a cluster of machines using message passing; techniques that are often necessary for large 3D problems. So, we return to the finite-difference method in the hope of creating an accurate scheme that is computationally flexible and efficient. Difference operators derived from Taylor-series expansion are "conservatve" in that they have very high accuracy for low wavenumbers, but in achieving that, they waste some potential accuracy improvements at higher wavenumbers. In a classic paper, Holberg (1987) shows how to find improved spatial difference operators using optimization techniques. Most modern high-order finite-difference techniques use his method to create spatial-difference coefficients. But I believe there are a couple of fundamental shortcomings in Holberg's design; so I revisited the design problem for optimum finite-difference schemes. In one space dimension, the accuracy of the second-order in time, second-order in space finite-difference method improves the closer the time step size is to the stability limit, $v\Delta t/\Delta x = 1$. It's an interesting curiosity that the two sources of error cancel exactly when taking the time-step size at exactly the stability limit. While this exact cancellation does not exist for the second-order finite-difference method in any higher number of spatial dimensions, it does point out that there is an opportunity to "tune" the spatial-difference coefficient design by trading it off against the errors generated by the time discretization. I also felt that the classical "equiripple" filter design technique that Holberg used leads to unacceptably large errors at low and moderate wavenumbers, paricularly when using relatively compact difference operators.

FINITE DIFFERENCE STENCILS BY LEAST SQUARES

The approach I take here begins in the same vein as Holberg; I set up an optimization problem to find spatial difference coefficients that minimize the dispersion of the resulting finite-difference scheme. First we need an expression for the numerical "velocity" of the finite-difference scheme as a function of frequency and propagation direction exporessed in terms of the finite-difference coefficients. There are 2 commonly used (and ultimately equivalent) approaches to accomplish this. The first approach is to substitute the expression for a trial plane-wave solution into the difference expression. The second approach, which I follow here, is to take a few liberties with Fourier transforms and compute the multi-dimensional Fourier transform of the difference scheme directly. To illustrate this, let's take the 2D acoustic wave equation discretized with second-order finite-differences.

$$\begin{aligned}
&\tfrac{1}{\Delta t^2}(U(x,z,t+\Delta t)-2U(x,z,t)+U(x,z,t-\Delta t)) = \\
&\tfrac{v^2}{\Delta x^2}(U(x+\Delta x,z,t)-2U(x,z,t)+U(x-\Delta x,z,t))+ \\
&\tfrac{v^2}{\Delta z^2}(U(x,z+\Delta z,t)-2U(x,z,t)+U(x,z-\Delta z,t)).
\end{aligned}$$

Moving Δt and recalling how to transform a series of impulses, write the Fourier transform of the difference equation as:

$$(2\cos(\omega\Delta t)-2)U(\omega,k_x,k_z) =$$
$$\frac{v^2\Delta t^2}{\Delta x^2}(2\cos(k_x\Delta x)-2)U(\omega,k_x,k_z)+\frac{v^2\Delta t^2}{\Delta z^2}(2\cos(k_z\Delta z)-2)U(\omega,k_x,k_z).$$

The numerical phase velocity is given simply by the ratio:

$$V_{phase} = \frac{\omega(k_x,k_z,v,\Delta x,\Delta z,\Delta t)}{\sqrt{k_x^2+k_z^2}} \tag{1}$$

where I have left the dependence on all possible variables in the expression. At this point, I make my first deviation from the method that Holberg used. Holberg used group velocity in his expressions and ultimately matched the numerical group velocity of the difference scheme to the desired velocity. In the method here, I use phase velocity. One of the nice things about using phase velocity is that it is straight-forward to make simulated "far field signatures" of your finite-difference method without going to the trouble of coding the actual finite-difference algorithm. One possible criticism of using phase velocities is that they tend to show less variability than group velocities, that is, you might think that phase velocities understate the dispersion. Since phase velocities are computed from the ratio of ω and $|\mathbf{k}|$, phase velocities are more sensitive to the lower wavenumbers (and thus the lower frequencies) than group velocities which are computed from the slope of ω as a function of $\mathbf{k}$. I'll leave it as an exercise for the reader to show whether weighting functions would make the the two approaches equivalent. The second and more important difference from Holberg's derivation is that I do not consider the error of the time discretization to be negligible. Near the begining of his paper, Holberg assumes that computational time step sizes will be small enough to not contribute to the error of the scheme. For long (hence hopefully accurate) spatial difference operators, the error from time discretization is often the *largest* error at moderate frequencies/wavenumbers. Better still, the error from the time discretization is usually opposite in sense from that of the space discretization, so the two can be balanced against each other for improved accuracy. Making time steps coarser tends to lead to phase velocities that are too fast while inaccuracy in the spatial difference operator leads to phase velocities that are too slow. It's that fact that leads to the curious situation in 1D where the two sources of error can be balanced exactly. Although there's no way to balance the errors exactly in 2D or 3D, we should use an error expression that captures the contributions from both the time and space discretizations and balances them against each other to the best possible effect. I begin by creating an expression for the sum over the wavenumber domain of the squared differences between the numerical phase velocity of the difference scheme and the desired phase velocity;

$$Err = \sum_{\mathbf{k}} W(\mathbf{k})\left[V_{phase}(\mathbf{k}) - V_{true}\right]^2. \tag{2}$$

Then we need the expression for V_{phase} as a function of the coefficients we wish to find. I express the difference operator for a second spatial derivative with respect to x as:

$$\frac{\partial^2 U}{\partial x^2} \approx \frac{1}{\Delta x^2}\sum_{i=1}^{n} a_i\left[U(x+i\,\Delta x)-2U(x)+U(x-i\,\Delta x)\right]. \tag{3}$$

You'll notice that the difference operator, chosen this way, is essentially a series of weighted and gapped standard second difference operators. I did this to preserve the correct behavior of the approximate second derivative at zero wavenumber. Using the corresponding expressions for the differences in z (and in 3D y), Fourier transforming them, and using the expression from equation 2 for the Fourier transform of the time difference expression write:

$$2\cos(\omega\Delta t)-2=\frac{v^2\Delta t^2}{\Delta x^2}\sum_{i=1}^{n}a_i\left[2\cos(ik_x\Delta x)-2\right]+\frac{v^2\Delta t^2}{\Delta z^2}\sum_{i=1}^{n}b_i\left[2\cos(ik_z\Delta z)-2\right]\ . \quad (4)$$

To compress the notation a bit, introduce:

$$\begin{aligned}F_x(\mathbf{a})&=\textstyle\sum_{i=1}^{n}a_i\left[2\cos(ik_x\Delta x)-2\right]\\ F_z(\mathbf{b})&=\textstyle\sum_{i=1}^{n}b_i\left[2\cos(ik_z\Delta z)-2\right]\ .\end{aligned}$$

Finally I arrive at a reasonably compact expression for V_{phase} as a function of the coefficients I seek and other parameters such as velocity, time step size and grid spacing.

$$V_{phase}(\mathbf{a},\mathbf{b})=\frac{\arccos\left[\frac{v^2\Delta t^2}{2\Delta x^2}F_x(\mathbf{a})+\frac{v^2\Delta t^2}{2\Delta z^2}F_z(\mathbf{b})+1\right]}{\Delta t|\mathbf{k}|}. \quad (5)$$

To find the coefficient vectors **a** and **b**, differentiate the sum squared error (equation 4) with respect to each a_i and b_i and set each of those equations equal to zero. Of course if Δx and Δz are the same, we can use the same coefficients for each, but for now I'll keep the expressions general because we want the method to work when $\Delta x\neq\Delta z$.

$$\begin{aligned}\frac{\partial Err}{\partial a_1}&=0\\ \frac{\partial Err}{\partial a_2}&=0\\ \vdots&=\vdots\\ \frac{\partial Err}{\partial a_n}&=0\\ \frac{\partial Err}{\partial b_1}&=0\\ \frac{\partial Err}{\partial b_2}&=0\\ \vdots&=\vdots\\ \frac{\partial Err}{\partial b_n}&=0\end{aligned}$$

This is a system of non-linear equations in the unknowns a_i and b_i. To solve them, I use a Gauss Newton iterative method which leads to the following system of linear equations (notation compressed a bit further):

$$\begin{aligned}\frac{\partial^2 Err}{\partial a_1^2}\Delta a_1\quad&\frac{\partial^2 Err}{\partial a_1a_2}\Delta a_2\quad\cdots\quad\frac{\partial^2 Err}{\partial a_1b_n}\Delta b_n&=&\quad-\frac{\partial Err}{\partial a_1}\\ \frac{\partial^2 Err}{\partial a_1a_2}\Delta a_1\quad&\frac{\partial^2 Err}{\partial a_2^2}\Delta a_2\quad\cdots\quad\frac{\partial^2 Err}{\partial a_2b_n}\Delta b_n&=&\quad-\frac{\partial Err}{\partial a_2}\\ \vdots\quad&\vdots\quad\vdots\quad\vdots&=&\quad\vdots\\ \frac{\partial^2 Err}{\partial a_1b_n}\Delta a_1\quad&\frac{\partial^2 Err}{\partial a_2b_n}\Delta a_2\quad\cdots\quad\frac{\partial^2 Err}{\partial b_n^2}\Delta b_n&=&\quad-\frac{\partial Err}{\partial b_n}.\end{aligned}$$

I solve these equations using the conjugate gradient method, and perform outer iterations until the gradient of the sum squared error doesn't decrease, which usually takes 3 or 4 iterations.

APPLICATION

We use finite-difference methods to propagate waves in heterogeneous media, so the spatial and temporal grid sizes are determined by the range of velocities in the model and the frequency content of the waves we wish to propagate. The maximum frequency and the minimum velocity combine with the accuracy of our scheme to limit the maximum spatial grid size that can be used. Then, given the spatial grid sizes, the maximum velocity in the model sets the maximum time step size that we can use for stability reasons. Because of all this, each model (each unique combination of $v_{min}, v_{max}, \Delta x, \Delta z$, and Δt) will have it's own set of optimum coefficients. In practice I just compute a table of coefficients for the case at hand and build precomputed "updating stencils". This wastes memory somewhat (maybe more than somewhat), but in principle can lead to a highly efficient code.

Right answer to the wrong question

Figure 1 shows the relative phase velocity error, clipped at 1%, for a Taylor-coefficient, sixth-order in space, second-order in time scheme for the 2D acoustic wave equation operating for the case $v = .33, dt = 0.3, \Delta x = 1.0, \Delta z = 1.0$. The spatial stencil for this scheme occupies the central point and 3 points to the sides on each spatial axis. I chose this particular combination of v and Δt to simulate the situation where waves are propagating in the low-velocity portion of a model that has a v_{max}:v_{min} ratio of 3:1. This is where dispersion tends to be the greatest, so we want as much accuracy from our stencil as possible. In some sense, models with a large v_{max}:v_{min} ratio are the most aggravating for the finite-difference method. Fine spatial sampling has to be used because of the low velocities, but then *really* small time steps have to be used because of the high velocities. As you might recall, accuracy tends to be better when you can run larger time steps since the error from the time differencing will partially cancel the error from the spatial differences. White areas of the wavenumber plane in Figure 1 have numerical phase velocities less than .99 times the correct phase velocity. Grey-colored areas have phase velocities within +/- 1% of the correct value, with neutral grey "perfect". Black areas (which there are none in this figure), would be wavenumbers that have phase velocities more than 1% greater than the correct value. This scheme is only accurate (to 1% anyway) out to about 45% of spatial Nyquist, requiring a sammple rate of about 4.4 points per shortest wavelength. This plot shows you the behavior of a finite-difference scheme with Taylor-series derived coefficients quite well; excellent accuracy at zero wavenumber and a flatish response out to some cutoff, then rapidly increasing error.

So, the hope is to use the optimization method described above to create a set of spatial differencing coefficients that will push the 1% error cutoff (or whatever error cutoff you wish) to higher wavenumbers, which enables the scheme to propagate higher frequencies on the same grid for the same cost. Figure 2 shows the relative phase velocity error for just such an "opti-

mized" 7 point second-difference operator that occupies the same stencil as the Taylor method shown in Figure 1, for the same velocity, time-step and spatial sample rates. The weight term used in the objective function (equation 4) was $1/|\mathbf{k}|$ for $\mathbf{k}$ out to 80% of Nyquist with a superimposed taper down to zero weight at 90% of Nyquist. This is a pretty good illustration of "trying to do too much" and least squares methods doing exactly what you told them to do, not necessarily what you *really* want them to do. It's probably important to state that the least squares method is not failing per se. Figure 3 shows a graph of the relative phase velocity error along the k_z axis and we can see the approximate "equiripple" behavior of the error, typical of a least squares solution over the wavenumber band we allowed in the specification of the error function. This behavior arises because the error at the high wavenumbers is naturally larger than at low wavenumbers. With only a few coefficients, there's no way to correct that error without compromising the behavior at low wavenumbers. But in fact, if we were only going to propagate a short distance, this scheme has less than 1% phase velocity error out to almost 80% of Nyquist.

Asking a better question

We need a little more "intelligence" in what we ask least squares to do. In the expression for the sum squared error, the standard approach is to fit a perfectly flat V_{phase}. Even though I took some care, as described above, to weight the objective function appropriately over wavenumbers, and restrict the fit to a range of $\mathbf{k}$; I found that trying to fit a "perfect" V_{phase} leads to difficulties. Rather than fit a flat V_{phase}, I fit my "low-order" compact stencil response to the respose of a higher-order method. In what follows and in practice, I fit to the respose of a much higher-order method with a small time-step size, and Taylor-series derived spatial difference coefficients. In addition to fitting a higher-order Taylor response, I also weight the error at each individual wavenumber by $1/|\mathbf{k}|^{1+\epsilon}$, where $0 < \epsilon \leq .5$, just to bias the fit to the low wavenumbers a bit. Now, I admit this is a somewhat "personal" choice, biased by my desire to preserve tight error bounds on low and mid-range wavenumbers. I purposefully do not want an equiripple phase velocity respose, because a compact stencil just has too few degrees of freedom to keep the ripples to an acceptable level.

Figure 4 shows a plot of relative phase velocity error for a new set of "optimum" coefficients designed to preserve "good" behavior at low and moderate wavenumbers while improving the response at higher wavenumbers compared to Taylor coefficients. This scheme is very accurate out to 55% of Nyquist, requiring about 3.6 points per shortest wavelength. This phase velocity spectrum looks reminiscent of the phase velocity spectrum from a higher-order Taylor method, more of a "pseudo-maxi-flat" response than an "equiripple response", which is exactly what I wanted. Whether this particular scheme is acceptable or not depends on the amount of oversampling we can tolerate which depends on the architecture of the computer we plan to use. The method can design larger (or down to two degrees of freedom smaller, since this one only has three) stencils, giving me the ability to adapt to whatever size stencil works best on any particular machine.

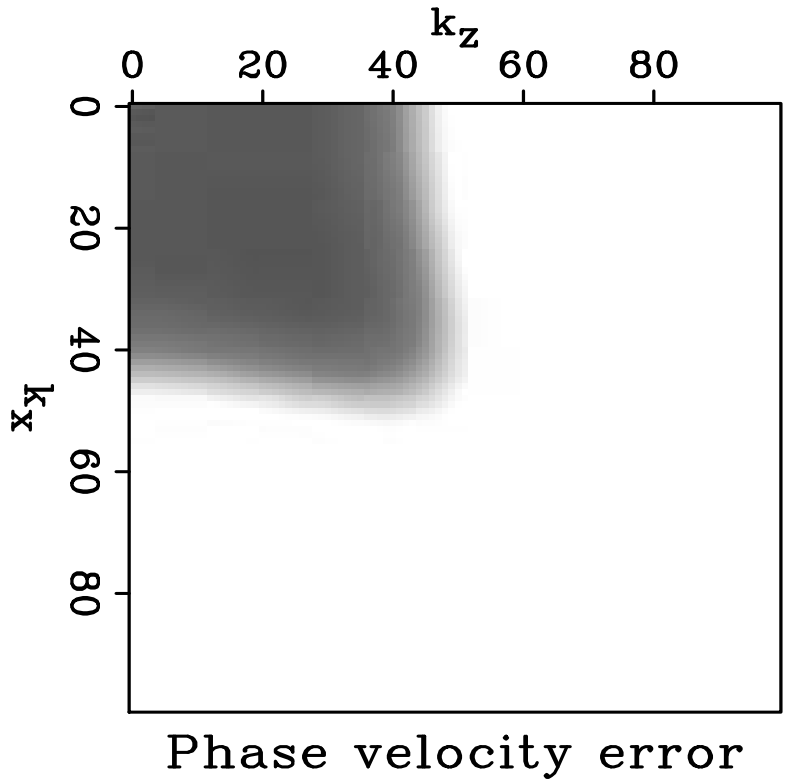

Figure 1: Relative phase velocity error clipped at 1% for the standard sixth-order in space, second-order in time finite-difference solution to the 2D acoustic wave equation. Grey areas of the plot have phase velocity close to the desired phase velocity. White areas have relative phase velocity error > 1% slow. john1-Vphase_taylor6 [ER]

CONCLUSION

In principle, one shouldn't ignore the error caused by the time discretization of a finite-difference method; instead, use it to trade against the error from the spatial difference operator. Designing spatial finite-difference operators this way leads to spatially compact finite-difference stencils that achieve higher accuracy than schemes based on Taylor-series expansion (obviously). I think it's still an open question as to whether this method is better than a method that uses optimized spatial finite-difference coefficients that are not a function of the local velocity (Holberg's method). For spatially compact operators I think it's important to include the effect of the time discretization, and in this case the method presented here is the method of choice. For long spatial operators, which are used when one needs to use as few points per shortest wavelength as possible, the amount of storage required would probably be discouraging, but in principle, the method presented here is more accurate.

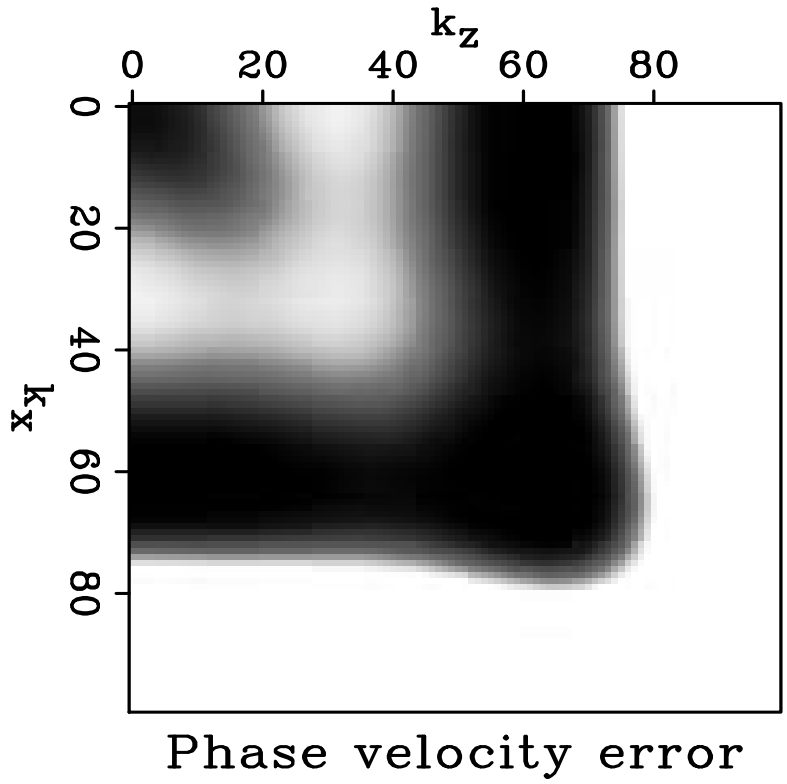

Figure 2: Relative phase velocity error clipped at 1% for an optimized finite-difference stencil with seven points on each spatial axis. Grey areas of the plot have nearly correct phase velocity. White areas of the plot have phase velocity too slow by >1%; Black areas of the plot have phase velocity >1% too fast. The optimization criterion was "too aggressive", resulting in an unacceptable fit . john1-Vphase_opt_7pt_naive [ER]

ACKNOWLEDGMENTS

I thank BP for permission to publish this material.

REFERENCES

Dablain, M. A., 1986, The application of high-order differencing to the scalar wave equation: Geophysics, **51**, 54–66.

Etgen, J., 1986, High order finite-difference reverse time migration with the two way nonreflecting wave equation: SEP– **48**, 133–146.

—, 1988, Evaluating finite difference operators applied to wave simulation: SEP– **57**, 243–258.

Holberg, O., 1987, Computational aspects of the choice of operator and sampling interval for

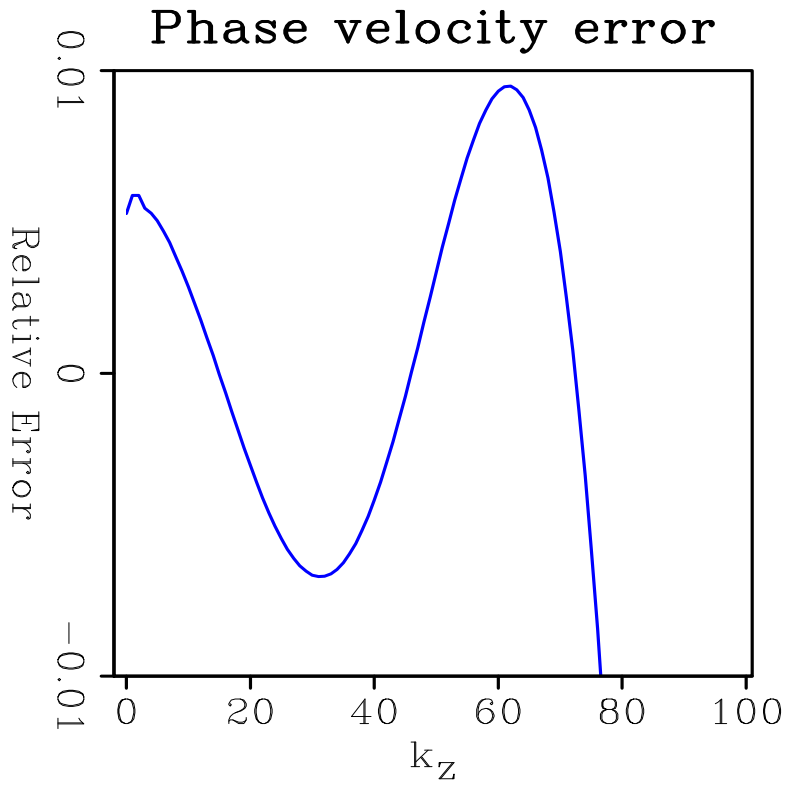

Figure 3: Graph of relative phase velocity error along the k_z axis showing approximate equiripple error. This amount of error is probably too large, although the fitting range is quite high, out to almost 80% of Nyquist. john1-Vphase_opt_7pt_naive_graph [ER]

numerical differentiation in large-scale simulation of wave phenomena: Geophys. Prosp., **35**, 629–655.

Reshef, M., Kosloff, D., Edwards, M., and Hsiung, C., 1988, Three-dimensional acoustic modeling by the Fourier method: Geophysics, **53**, 1175–1183.

Tal-Ezer, H., Kosloff, D., and Koren, Z., 1987, An accurate scheme for seismic forward modelling: Geophys. Prosp., **35**, 479–490.

Wu, W. J., Lines, L. R., and Lu, H. X., 1996, Analysis of higher-order, finite-difference schemes in 3-D reverse-time migration: Geophysics, **61**, 845–856.

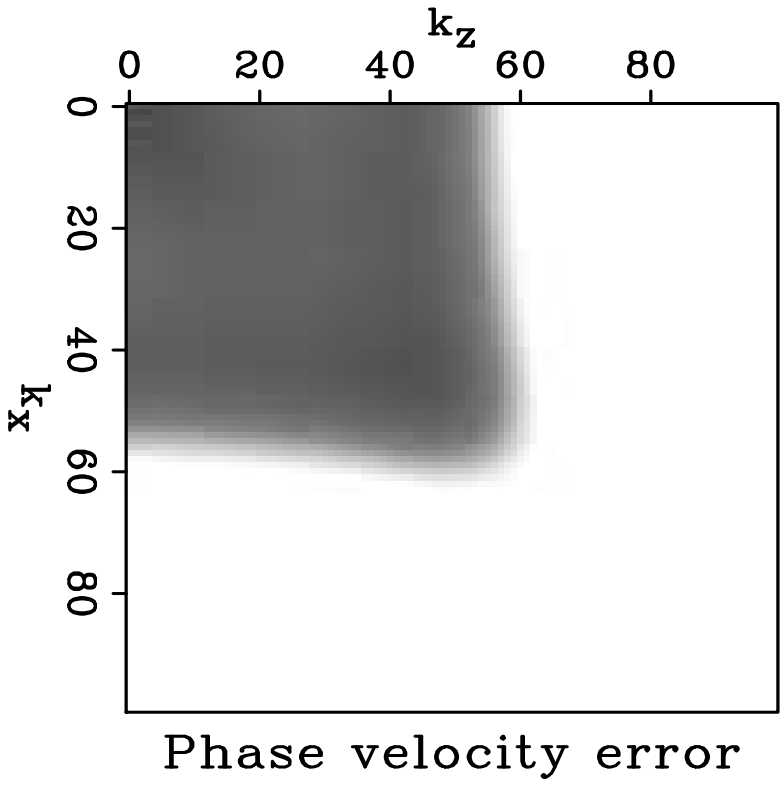

Figure 4: Relative phase velocity error clipped at 1% for an optimized finite-difference stencil with seven points on each spatial axis. Although the range of wavenumbers fit is smaller than in Figure 2, this set of difference coefficients can propagate accurately waves that are sampled at about 3.6 points per wavelength. john1-Vphase_opt_7pt_improved [ER]