Wave field extrapolation using a new multiplication/convolution method

WORK IN PROGRESS

Christiaan C. Stolk

ABSTRACT

In this paper we consider numerical methods for the acoustic equation, based on evolution in one of the space variables. There is a range of methods that compute alternatively in the Fourier and in the space domain, such as phase shift plus interpolation (PSPI) and generalized screens (GS). In this paper we derive a more general class of such methods using the one-way wave equation. We first discuss the approximation of the square root operator. The evolution in depth is done by a "preconditioned" ODE solver, where we use an easily computable approximate solution and solve a differential equation for the difference with the exact solution. In this setup the errors from the approximation to the square root operator and the evolution can be analyzed separately. Some features of our method are that it allows an arbitrary range of medium velocities (by higher order interpolation before exponentiation), large propagation angle and small error, where of course there is a trade off with computational cost. Dampening of waves with near horizontal propagation angles is introduced explicitly. We also propose adaptive depth stepping.

1. INTRODUCTION

Methods for solving the acoustic equation by evolution in depth are important in applications, particularly in seismic data processing. There are many different methods that mix computations in the Fourier and in the space domain. In this class of methods one can distinguish between methods with finite-difference terms, such as Fourier finite-difference (FFD) (Ristow and Ruhl, 1994), or Fourier finite-difference plus interpolation (FFDPI) (Biondi, 2002), and methods that combine different terms computed in the Fourier domain by space dependent weight functions. The latter include for instance phase shift plus interpolation (PSPI) (Gazdag and Sguazzero, 1984), split-step Fourier (SSF) (Stoffa et al., 1990), extended local Born (Huang et al., 1999), generalized screens (GS) ((de Hoop et al., 2000)). There is a large flexibility in the setup of such methods, in particular for methods of the second type. It appears that this flexibility is not fully

exploited in methods available in the literature. In this paper we derive a new generalized screen method that has a better control of error and large angle behavior. We start from a regularized single square root equation, that includes explicitly the zeroeth order terms that determine the amplitude behavior. The latter is derived by known pseudodifferential techniques.

We will have the following setup. The direction of evolution will be called the vertical direction and denoted by $z \in \mathbb{R}$, while the other directions will be denoted by x. The dimension of space will be denoted by n. This work is valid for any $n \geq 2$. Let $U = U(z, x, t)$ be the acoustics wave field, $\nu = \nu(z, x)$ the medium slowness (inverse of the local wave speed) and $F = F(z, x, t)$ a volume source. The acoustic equation with constant density is

$$\left(-\nu^2 \partial_t^2 + \partial_x^2 + \partial_z^2\right) U = F. \tag{1}$$

We will generally consider this with a condition at $z = 0$

$$U(0, \cdot\,, \cdot\,) = U_0, \tag{2}$$

and with the condition $U = 0$ for $t < 0$, and aim to compute the wave field for $z > 0$.

We will assume we are in a regime of geometrical optics, with wave length short compared to the length scale of the medium variations. Then up- and downgoing waves can be distinguished and the only coupling occurs for turning rays, where the vertical component of velocity changes sign. It is well known that by using pseudodifferential operators, the original acoustic equation can be transformed to a system of equations, such that this property becomes apparent. We then obtain two indepent equations for the up- and downgoing waves by *removing the coupling term, and instead adding a pseudodifferential term that will dampen the wave fronts with near horizontal propagation directions* (section 2). This results in the pseudodifferential equation given in (4). (Note that variations in the medium coefficient with length scale comparable to the wave-length lead to reflections, coupling up- and downgoing waves. Hence some assumption is necessary if we want to approximate solutions to (1).)

In general, after discretization, a pseudodifferential operator leads to a full matrix on basis of position coordinate x and also on k, and its computation is expensive. On the other hand when the coefficient ν is constant it is diagonal in the Fourier domain and its exponential (a set of complex phase factors) can be computed easily over large step size. This leads to an easily computable approximative solution when the coefficient varies not too much around some value ν_0. The idea is to obtain a fast and more accurate method by adding correction terms to such an approximate solution (GS) or by combining two such approximative solutions (PSPI). The expressions for the wave field propagated a small distance in depth are of the form

$$\sum \text{ multiplications and convolutions of } u_0. \tag{3}$$

This is evaluated by performing the convolution as a multiplication in the Fourier domain, using FFT's.

In this paper we consider approximations of the square root operator based on (3). The square root symbol, which can be considered as a function of ν and k, will be approximated by interpolation over ν. We consider piecewise Lagrange interpolation, Hermite interpolation and interpolation by cubic splines. We thus generalize an idea from generalized screens where a Taylor series approximation is taken around a single support slowness ν_0. Our approach allows in particular for larger opening angles and slowness variations. We discuss the resulting errors and give examples. We also discuss the discretization (section 3).

To solve the evolution equation we observe that the difference of the true solution with an exponential approximate solution as described above satisfies a differential equation. We propose to solve this equation by a numerical ODE solver (Runge-Kutta since this is a convenient method). In this approach an error estimate is available, even during the computation (in certain schemes), so that the stepsize can be chosen adaptively during the computation.

Some of the features that we believe are advantages of this method are the following. By increasing the number of ν support points in the interpolation the operator can be approximated over an arbitrary range of medium velocities (before domain decomposition/averaging exponentials) and the error can be made small up to large angles (where of course there is trade off with computational cost). We explicitly introduce dampening of waves with near horizontal propagation angles that are not correctly propagated.

The organization of the paper is as follows. In section 2 we present the derivation of the one-way wave equation. We include the zeroeth order terms that determine amplitude. In section 3 we discuss the approximation of the square root operator B. Then in section 4 we discuss the solution of the evolution equation. We give some numerical examples in section 5. We end with a discussion in section 6. In Appendix A some basic facts about pseudodifferential operators are given that are relevant for the numerical approximation. In Appendix B some functions are listed that are used to construct smooth approximations to the square root.

2. THE SINGLE SQUARE ROOT EQUATION

In this section we derive a regularized one-way wave equation, given by

$$\frac{\partial u}{\partial z} = (s_{\mathrm{p}}\mathrm{i}B - s_{\mathrm{d}}C)\, u + f. \tag{4}$$

This equation describes the propagation of wave fronts as long as the angle of propagation with the vertical is smaller than some maximal angle, and includes a dampening term to suppress the (incorrectly propagated) remainder. The square root operator B, which is a pseudodifferential operator, is defined in (9), and lemma 1. The operator C (also pseudodifferential) is the dampening term described on page 6. The wave field u and the source f are related to the original wave field U and source F by (18). The constants $s_{\mathrm{p}} = \pm 1$ and $s_{\mathrm{d}} = \pm 1$ denote respectively the time direction of propagation (sign of $\frac{\partial t}{\partial z}$) and the direction of numerical evaluation (direction of damping). The equation is

derived in several steps from the first order system equivalent to (1). The final step is a modification, where the term coupling up and down propagating waves is removed and the dissipative term is added. Throughout the discussion we keep track of the zeroeth order term in the square root equation, to have correct highest order amplitudes. The pseudodifferential operator technique and the decoupling into first order equations are classical in the mathematical literature (for the latter see e.g. (Taylor, 1975)).

We will use the following convention for the Fourier transform

$$\mathcal{F}_x u(k_x) = (2\pi)^{-(n-1)} \int_{\mathbb{R}^{n-1}} \mathrm{e}^{-\mathrm{i}k_x \cdot x} u(x)\, \mathrm{d}x, \tag{5}$$

$$\mathcal{F}_t u(\omega) = (2\pi)^{-1} \int_{\mathbb{R}} \mathrm{e}^{-\mathrm{i}\omega t} u(x)\, \mathrm{d}t. \tag{6}$$

2.1. FIRST ORDER SYSTEM AND PSEUDODIFFERENTIAL SQUARE ROOT OPERATOR

The acoustic equation (1) corresponds to a first order system in z for the pressure U and $V = \frac{\partial U}{\partial z}$, that is given by

$$\frac{\partial}{\partial z} \begin{pmatrix} U \\ V \end{pmatrix} = \begin{pmatrix} 0 & I \\ -A & 0 \end{pmatrix} \begin{pmatrix} U \\ V \end{pmatrix} + \begin{pmatrix} 0 \\ F \end{pmatrix}, \tag{7}$$

where A is the second order differential operator defined by

$$A = -\nu(x,z)^2 \frac{\partial^2}{\partial t^2} + \sum_j \frac{\partial^2}{\partial x_j^2}.$$

Associated with A is its symbol $A(z, x, k_x, \omega) = \nu(x,z)^2\omega^2 - |k_x|^2$.

We will look for a pseudodifferential operator square root of A, $B^2 = A$. Clearly B must have principal symbol $B(z, x, k_x, \omega) = \sqrt{A(z, x, k_x, \omega)}$, which is not smooth near the zeroes of A. Therefore, for each z we consider the set of (x, t, k_x, ω) where A is strictly positive. Given $\epsilon > 0$, such a set is given by

$$|\nu^{-1}\omega^{-1}k_x| \leq 1 - \epsilon, \tag{8}$$

This will correspond to angles of propagation (with the vertical) up to θ where θ is given by $\sin(\theta) = 1 - \epsilon$.

We will denote by $b = b(z, x, k_x, \omega)$ the first and highest order symbol of a square root of A on (8). Outside (8) we assume that b is real and smooth. *There the symbol b must be chosen* such that the resulting numerical algorithm is optimal, we discuss this further in section 3. As for the choice we sign, on (8) we set

$$b(z, x, k_x, \omega) = -\omega\sqrt{\nu^2 - \omega^{-2}k_x^2}. \tag{9}$$

When the coefficient ν depends on x the square root operator also has a lower-order part, that depends on the quantization, i.e. the way we associate an operator with b. We

summarize the results we need in the following lemma (see Appendix A for the definitions of the different quantizations).

Lemma 1: *There is a pseudodifferential operator $B(z,x,D_x,D_t)$ such that $B^2 = A$ modulo $S^{-\infty}$ on (8). Its symbol is given on (8) by*

$$B(z,x,k_x,\omega) = b(z,x,k_x,\omega) + \tfrac{1}{2}\mathrm{i}b(z,x,k_x,\omega)^{-1}\sum_{j=1}^{n-1}\frac{\partial b}{\partial k_{x_j}}\frac{\partial b}{\partial x_j} + \mathrm{order}(-1). \tag{10}$$

The operator $B(z,x,D_x,D_t)$ is also given by $\mathrm{Op}_R\, B_R$, where

$$B_R(z,x,k_x,\omega) = b(z,x,k_x,\omega) - \tfrac{1}{2}\mathrm{i}b(z,x,k_{x_j},\omega)^{-1}\sum_{j=1}^{n-1}\frac{\partial b}{\partial k_{x_j}}\frac{\partial b}{\partial x_j} + \mathrm{order}(-1). \tag{11}$$

and by $\frac{1}{2}(\mathrm{Op}_L\, B_M + \mathrm{Op}_R\, B_M)$, where

$$B_M(z,x,k_x,\omega) = b(z,x,k_x,\omega) + \mathrm{order}(-1). \tag{12}$$

The proof (a standard construction order by order that is often useful) can be found in the appendix.

2.2. TRANSFORMATION OF THE SYSTEM AND SINGLE SQUARE ROOT EQUATION

Let Q be an invertible matrix pseudodifferential operator. Define (u_+,u_-) by

$$\begin{pmatrix} U \\ V \end{pmatrix} = Q\begin{pmatrix} u_+ \\ u_- \end{pmatrix}, \begin{pmatrix} 0 \\ F \end{pmatrix} = Q\begin{pmatrix} f_+ \\ f_- \end{pmatrix}.$$

The transformed version of (7) reads

$$\frac{\partial}{\partial z}\begin{pmatrix} u_+ \\ u_- \end{pmatrix} = Q^{-1}\begin{pmatrix} 0 & I \\ -A & 0 \end{pmatrix} Q\begin{pmatrix} u_+ \\ u_- \end{pmatrix} + \frac{\partial Q^{-1}}{\partial z} Q\begin{pmatrix} u_+ \\ u_- \end{pmatrix} + \begin{pmatrix} f_+ \\ f_- \end{pmatrix}. \tag{13}$$

For the choice of Q, note that the eigenvalues of the symbol $\begin{pmatrix} 0 & I \\ -A & 0 \end{pmatrix}$ are given by

$$\text{eigenvalues}: \ \pm\,\mathrm{i}A(z,x,k_x,\omega)^{1/2}, \ \text{eigenvectors}: \ \begin{pmatrix} 1 \\ \pm\mathrm{i}A(z,x,k_x,\omega)^{1/2} \end{pmatrix}.$$

Like the square root B, we choose Q only on (8). On this set we set Q to highest order such that its columns are eigenvectors, with some freedom in the normalization (the number s, this will be chosen below). By $\widetilde{A^s}$ we denote a pseudodifferential operator equal to A^s

microlocally on (8), smooth outside this set and with symbol $A^s = (\nu^2\omega^2 - k_x^2)^{s/2}$ on (8). Similarly $\widetilde{B^{-1}}$ is given by B^{-1} microlocally on (8) etc.

$$Q(z, x, k_x, \omega) = \widetilde{A^{-s}} \begin{pmatrix} 1 & 1 \\ \mathrm{i}B & -\mathrm{i}B \end{pmatrix} + \text{l.o.t.}, \tag{14}$$

$$Q(z, x, k_x, \omega)^{-1} = \frac{1}{2}\widetilde{A^s} \begin{pmatrix} 1 & -\mathrm{i}\widetilde{B^{-1}} \\ 1 & \mathrm{i}\widetilde{B^{-1}} \end{pmatrix} . + \text{l.o.t.}, . \tag{15}$$

Outside (8) we still require that Q is an invertible pseudodifferential operator of order $\begin{pmatrix} -2s & -2s \\ 1-2s & 1-2s \end{pmatrix}$. This leads to the following equation for (u_+, u_-)

$$\frac{\partial}{\partial z}\begin{pmatrix} u_+ \\ u_- \end{pmatrix} = \begin{pmatrix} \mathrm{i}B & 0 \\ 0 & -\mathrm{i}B \end{pmatrix}\begin{pmatrix} u_+ \\ u_- \end{pmatrix} + \begin{pmatrix} 2s-\frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & 2s-\frac{1}{2} \end{pmatrix} \widetilde{A^{-1}}\widetilde{\frac{\partial A}{\partial z}}\begin{pmatrix} u_+ \\ u_- \end{pmatrix} + R\begin{pmatrix} u_+ \\ u_- \end{pmatrix} + \begin{pmatrix} f_+ \\ f_- \end{pmatrix}. \tag{16}$$

Here R is the sum of a lower order part, that is a pseudodifferential operator of order -1, and a contribution that is a pseudodifferential matrix of order 1 supported outside (8).

To obtain correct highest order amplitudes, the operators on the right hand side of (16) must be computed for the two highest order terms, i.e. first and zeroeth order terms, while for the operator Q we are interested only in the highest term.

The second term in (17) is of zeroeth order. Its off-diagonal part can be removed on (8) by adding lower order terms to Q, see (Taylor, 1975). This results in the following equation

$$\frac{\partial}{\partial z}\begin{pmatrix} u_+ \\ u_- \end{pmatrix} = \begin{pmatrix} \mathrm{i}B & 0 \\ 0 & -\mathrm{i}B \end{pmatrix}\begin{pmatrix} u_+ \\ u_- \end{pmatrix} + (2s-\frac{1}{2})\widetilde{A^{-1}}\widetilde{\frac{\partial A}{\partial z}}\begin{pmatrix} u_+ \\ u_- \end{pmatrix} + R\begin{pmatrix} u_+ \\ u_- \end{pmatrix} + \begin{pmatrix} f_+ \\ f_- \end{pmatrix}. \tag{17}$$

We take only the part of (17) inside (8), describing the propagating hyperbolic modes. That is we consider a modification of (17) where

1. *the operator R, that couples u_+ and u_- is removed.*
2. *an additional dissipative term given by a pseudodifferential operator C is added.* We require that its symbol $C \geq 0$ is smooth, equal to zero on (8), and $C > 0$ where $B \neq A^{1/2}$.

Then a wave front is propagated correctly as long as its angle of propagation with the vertical is smaller than θ. For larger angles it is suppressed by the damping term given by C. There is no unique prescription for C, otherwise then the positivity requirement, and this freedom should be exploited to make for good properties of the final numerical scheme.

Summary We now give our one-way wave equation. *We will set* $s = \frac{1}{4}$ *so that the second term on the r.h.s. of (17) vanishes.* Denote by $s_\mathrm{p} = \pm 1$ the sign of $\frac{\partial z}{\partial t}$ along the propagation direction of the wave front. For $s_\mathrm{p} = +1$ we set $u = u_+$, and we assume $u_- = 0$. Similarly, for $s_\mathrm{p} = -1$, we set $u = u_-$, and we assume $u_+ = 0$. We denote $f = f_\pm$ corresponding to s_p. Then we have

$$u = \widetilde{A^s} U \qquad\qquad f = -s_\mathrm{p} \tfrac{1}{2} \mathrm{i} \widetilde{A^s} \widetilde{B^{-1}} F. \tag{18}$$

By $s_\mathrm{d} = \pm 1$ we denote the direction of the decay of the wave field outside (8), which should equal the direction of numerical evaluation. With the modifications we hence obtain the following equation from (17)

$$\frac{\partial u}{\partial z} = (s_\mathrm{p} \mathrm{i} B - s_\mathrm{d} C)\, u + f.$$

3. APPROXIMATION OF THE SQUARE ROOT OPERATOR BY MULTIPLICATIONS AND CONVOLUTIONS

3.1. THE METHOD

The application of pseudodifferential operators is in general expensive, because the operator is not local in space coordinates, nor in Fourier coordinates. Direct evaluation of (A-1) leads to an n dimensional Fourier transform for each x. In this case, we first note that the operator B decouples when Fourier transformed with respect to the time variable. Equation (4) can be Fourier transformed to the frequency domain, and solved separately for each frequency.

From now on we will therefore assume that u is Fourier transformed with respect to time, $u = u(z, x, \omega)$.

Consider the approximation of $\mathrm{Op}_L\, b$ (which gives a correct highest order contribution to the square root operator, see Lemma 1) that is now given by

$$\mathrm{Op}_L\, b\, u(z, x, \omega) = (2\pi)^{-(n-1)} \int_{\mathbb{R}^{n-1}} \int_{\mathbb{R}^{n-1}} b(z, x, k_x, \omega) \mathrm{e}^{\mathrm{i}(x-y)\cdot k_x} u(z, y, \omega)\, \mathrm{d}k_x\, \mathrm{d}y.$$

Consider an approximation of b of the following form

$$b(z, x, k_x, \omega) \approx \sum_{j=1}^{K} w_j(x; z, \omega) g_j(k_x; z, \omega). \tag{19}$$

Then the application of $\mathrm{Op}_L\, b$ becomes

$$\mathrm{Op}_L\, b\, u \approx \sum_{j=1}^{K} w_j \mathcal{F}_x^{-1}(g_j \mathcal{F}_x u). \tag{20}$$

A single term in the sum is hence obtained by first a transformation to Fourier space, multiplication in Fourier space by g, then inverse Fourier transformation back to ordinary

space, and multiplication by w. For $\mathrm{Op}_R\, b$ we have a similar formula with the order of multiplication in position and Fourier domain interchanged

$$\mathrm{Op}_R\, b\, u \approx \sum_{j=1}^{K} \mathcal{F}_x^{-1}\left(g_j \mathcal{F}_x(w_j u)\right). \tag{21}$$

To have correct zeroeth order term in the square root operator (see lemma 1) we will use the symmetrized form $\frac{1}{2}(\mathrm{Op}_L\, b + \mathrm{Op}_R\, b)$ for the square root operator. (Then the square root operator is automatically selfadjoint, which is good. Also we don't know an approximation to (10) or (11), comparable to the approximation for (19) that we will find below.)

The problem is to find K, w_j and g_j such that (19) is a good approximation. We observe that b depends on x only through the medium slowness $\nu(x)$. This makes it possible to use the following interpolation approach. Let $S = \{\nu_1, \ldots, \nu_K\}$ be a set of suitably chosen support slownesses. By standard numerical interpolation methods we can find $w_j(\nu)$ such that

$$b(\nu) \approx \sum_{j=1}^{K} w_j(\nu) b(\nu_j) \tag{22}$$

is a good approximation to $b(\nu)$. Derivatives can also be included. Let $\alpha = \alpha_1, \ldots, \alpha_K$ be a vector of integers ≥ 1. There are $w_{j,j'}, 0 \leq j' < \alpha_j$ such that

$$b(\nu) \approx \sum_{j=1}^{K} \sum_{j'=0}^{\alpha_j - 1} w_{j,j'}(\nu) \partial_\nu^{j'} b(\nu_j) \tag{23}$$

is a good approximation to $b(\nu)$. We will consider piecewise Lagrange or cubic spline interpolation for (22) and piecewise Hermite interpolation for (23). All of these are standard, see textbooks on numerical analysis such as (Stoer and Bulirsch, 2002; Kincaid and Cheney, 1996). The simplest case of Hermite interpolation is $K = 1$ (Taylor series approximation), which will roughly lead to generalized screens. With Hermite interpolation the derivatives up to the specified order can be made continuous.

In Lagrange interpolation a single polynomial is found that has the same values as a function f at the ν_j. For Hermite interpolation the derivatives up to order $\alpha_j - 1$ at the ν_j must also agree. The unique polynomial of order $K - 1$ determined by S, and a list of values $F = \{f(\nu_j) \,|\, \nu_j \in S\}$ will be denoted by $P_{\mathrm{Lagrange}}(\nu; S_l, F)$. Similarly the unique polynomial of order $-1 + \sum_{j=1}^{K} \alpha_j$ determined by S, α and a list of values $F = \{\partial_\nu^{j'} f(\nu_j) \,|\, \nu_j \in S, 0 \leq j' < \alpha_j\}$ will benoted by $P_{\mathrm{Hermite}}(\nu; S_l, \alpha_l, F)$. Denote by e_j a list of values which is zero except for the j-th element, and similarly for $e_{j,j'}$.

In *piecewise Lagrange or Hermite interpolation* we determine first the number l such that $\nu \in]\nu_l, \nu_{l+1}]$ (with $\nu_0 = -\infty$, $\nu_{K+1} = \infty$). For each l we have a subset $S_l \subset S$ of support points. The weights w_i for piecewise Lagrange interpolation are given by

$$w_j(\nu) = \begin{cases} P_{\mathrm{Lagrange}}(\nu; S_l, e_j) & \nu_j \in S_l, \\ 0 & \nu_j \notin S_l. \end{cases} \tag{24}$$

For piecewise Hermite interpolation, let α_l denote a number of derivatives for each $\nu \in S_k$. We then have

$$w_{j,j'}(\nu) = \begin{cases} P_{\text{Hermite}}(\nu; S_l, \alpha_l, e_{j,j'}) & \nu_j \in S_l \text{ and } 0 \leq j' < \alpha_{l,j}. \\ 0 & \nu_j \notin S_l \text{ or } j' \geq \alpha_{l,j}. \end{cases} \tag{25}$$

The following formula for the error is well known. Let $n+1 = \sum_{j=1}^{K} \alpha_j$ (where α_j is 1 for Lagrange interpolation) and let f be $n+1$ times differentiable. Then the error satisfies

$$f(\nu) - P_{\text{Hermite}}(\nu; S, \alpha) = \frac{f^{(n+1)}(\omega)}{(n+1)!} \prod_{j \in S} (\nu - \nu_j)^{\alpha_j},$$

where ω is some unknown points in the interval $I(S \cup \{\nu\}) := ([\min(S \cup \{\nu\}), \max(S \cup \{\nu\})]$. A function obtained by piecewise interpolation (with constant order, i.e. constant number of points used for interpolation) converges to f if the stepsize decreases.

In *cubic spline interpolation* we take $\alpha_j = 1$ and then there is a polynomial $P_{\text{cubicspline},l}$ on each interval $\nu \in]\nu_l, \nu_{l+1}]$, such that the resulting function as well as its first and second derivatives are continuous. We take

$$w_i(\nu) = P_{\text{cubicspline},l}(\nu; S, e_j). \tag{26}$$

If the first derivatives at the end are also used, then there is an error estimate that gives the size of the error in terms of distance between grid points etc., see (Stoer and Bulirsch, 2002, section 2.4.3).

The interpolation works for all k_x, ω such that $b(\nu, k_x, \omega)$ is a number of times continuously differentiable w.r.t. ν on the interval of ν considered. It therefore is natural to choose b to be regularized on a scale (given by ϵ above) comparable with the distance between the ν_j (parameter ϵ above).

3.2. ERROR CRITERION

There are several quantities we must choose to obtain an approximation of the square root operator. They are the regularized symbol b (must be chosen outside the set (8)), the type of interpolation to be used in the approximation (19), the support point S and (in piecewise Lagrange interpolation), the subset of support points used for each ν interval. The first criterion is that the approximated symbol (8) has propagation velocities and arrival times close to those of the correct symbol. We will thus give the expressions for the velocities.

The error in the propagation velocity can be decomposed into a part parallel to the correct velocity, and a part normal to the correct velocity. The first of these corresponds roughly to the error in travel time, so we have naturally the following primary criterion.

Criterion 1 The inline velocity error is small up to some maximal angle.

As a secondary error criterion we have

Criterion 2 The orthogonal velocity error is small up to some maximal angle.

The example in subsection 3.3 suggests the following additional criterion. This concerns the waves in the dampened region that are in principle suppressed.

Criterion 3 The velocity in the region just outside (8) is not much faster than the correct wave front.

How important this last criterionis in must still be further studied (it is important in particular when velocity decreases for increasing depth.)

In the following we will denote

$$p = \frac{k_x}{-\omega} \tag{27}$$

We will consider approximations to the square root of the form

$$-\omega\nu f(\nu, \frac{k_x}{-\omega}), \tag{28}$$

where $f = f(\nu, p)$ is a given function. Along a ray of geometrical optics we have

$$\frac{\mathrm{d}t}{\mathrm{d}z} = -\frac{\partial b}{\partial \omega} = \nu f(\nu, \frac{k_x}{-\omega}) - \nu \frac{k_x}{-\omega}\partial_p f(\nu, \frac{k_x}{-\omega}) = \nu f(\nu, p) - \nu p \partial_p f(\nu, p), \tag{29}$$

$$\frac{\mathrm{d}x}{\mathrm{d}z} = -\frac{\partial b}{\partial k_x} = -\nu \partial_p f(\nu, \frac{k_x}{-\omega}) = -\nu \partial_p f(\nu, p). \tag{30}$$

The velocities $v_z = \frac{\partial z}{\partial t}$ and $v_x = \frac{\partial x}{\partial t}$ are given by

$$v_z = \frac{1}{\mathrm{d}t/\mathrm{d}z} \qquad\qquad v_x = \frac{\mathrm{d}x/\mathrm{d}z}{\mathrm{d}t/\mathrm{d}z}. \tag{31}$$

3.3. EXAMPLES OF APPROXIMATED DISPERSION RELATIONS (SYMBOLS)

We will assume that the regularized square root b is a scaled version of a function of a single variable. Let $f(k)$ be a function on $\mathbb{R}$, that is equal to $\sqrt{1-k^2}$ on $[-1+\epsilon, 1-\epsilon]$. We will choose b of the form

$$b(\nu, k_x, \omega) = -\omega\nu f(\frac{k_x}{-\omega\nu}).$$

We will suppose the ν_i are all determined by ν_1 and ν_2/ν_1 according to

$$\nu_j = \nu_1(\nu_2/\nu_1)^{j-1}.$$

We will consider an example with order 2 piecewise Lagrange interpolation where

$$S_l = \{\nu_l, \nu_{l+1}, \nu_{l+2}\}$$

(for $1 \leq l \leq K-2$, we will not consider ν such that l is outside this range).

The general approach is the following

1. Start with a family of functions $f_{a,b}$, such that for

$$|k| \lesssim 1 - a$$

the function $f_{a,b}$ is equal to the square root, and regularized outside this region, where b denote possible additional parameters.

2. Fix a maximal angle, or equivalently ϵ in (8).

3. For each a, b we search for a maximum error (based on Criterion 1 and/or Criterion 2) as a function of ν, p, $|p/\nu| < 1 - \epsilon$, ν in a given interval. This maximum error should be minimal, this gives a choice of a, b.

4. A more sophisticated approach would be to predetermine a maximal error and to search for maximal value of ν_2/ν_1, so that the minimal number of FFT's is necessary in the evaluation of (20). This involves adjusting a, b and ν_2/ν_1.

We will now apply this to two families of functions, up to step 3. From now on in this section we will denote by f the exact square root, in scaled coordinates

$$f(k) = \sqrt{1 - k^2}.$$

We study two classes of functions to approximate the square root, where in each class we look for an optimal choice. We fix the $\nu_2/\nu_1 = 1.1$. The maximal k is given by 0.9 (maximal angle 64 degrees). We study the relative error in velocity (for the computations we can set $\nu = 1$, and vary the ν_1).

The first function we take from a class such that a small imaginary part to the square root, localized around $|k| = 1$. The main parameter is called a and describes the width of the regularization, while there is another parameter b to describe its size.

$$\widetilde{f}_{a,b}(k) = \mathrm{Re}\,\sqrt{1 - k^2 + \mathrm{i}a^2 b(h_{(3)}(k-1) + h_{(3)}(k+1))}.$$

We computed the maximal error for an array of values of a, b. A choice close to optimal for both the parallel and the orthogonal part of the error is $a = 0.25, b = 1.4$, with error 0.002, and 0.026. We denote this choice by f_1.

The second function is obtained by modifying the derivative of the square root, such that for $k > 1 - a$ (for some a), it goes to a constant instead of to $-\infty$. The derivative of the square root is given by

$$f'(k) = -k(1 - k^2)^{-1/2} \tag{32}$$

We define a scaled version of $h_{(2)}$ (see Appendix B), denoted $h_{(2),\tilde{a},\tilde{b}}$, that is the identity on $]-\infty, \tilde{a}]$, and constant equal to $\tilde{b}$ for $y > \tilde{b} + (\tilde{b} - \tilde{a})$. It is given by

$$h_{(2),\tilde{a},\tilde{b}}(y) = \tilde{a} + (\tilde{b} - \tilde{a}) h_{(2)}(\frac{y - \tilde{a}}{\tilde{b} - \tilde{a}})$$

Now let $\tilde{a} = 1 - a, \tilde{b} = 1 - ba$. We choose $\widehat{f}_{a,b}$ such that its derivative is given by a modification of (32)

$$\widehat{f}_{a,b}{}'(k) = f'(h_{(2),\tilde{a},\tilde{b}}(k))$$

That is

$$\widehat{f}_{a,b}(k) = 1 + \int_0^k f'(h_{(2),\tilde{a},\tilde{b}}(\tilde{k}))\,\mathrm{d}\tilde{k}.$$

Again, for an array of choices for the parameter we computed the maximal error. A roughly optimal choice is given by $a = 0.22, b = 0.3$ with error given approximately by 0.001 and 0.017. We set $f_2 = \widehat{f}_{0.22,0.3}$.

We compared the exact square root and the two optimized choices f_1, f_2. A plot of f, f_1, f_2 is given in Figure 1. The corresponding velocity curves are given in Figure 2. We see from Figure 2 that f_1 has a wavefront that propagates faster than the correct wave front. Although it appears that the function f_2 does not have such a wave front, this is not quite correct, since we will want f_2 to become constant for some larger value of k, but at least the phenomenon, if it appears will appear at larger k. The incorrect part of the wave front will be in the region with damping, but for f_2 it is much further in this region, which is clearly preferable to make the method more robust.

An illustration of the behavior of the error (for different values of ν_1) is given in Figures 3 and 4. The errors are significantly smaller for f_2, as was apparent already from the numbers we mentioned (maximal inline error (travel time) 0.001 for f_2, 0.002 for f_1, maximal orthogonal errors 0.017 and 0.026).

A conclusion would be that, at least in this example a function that "continuous downward below zero" like f_2 offers better behavior of the velocity curve (with regard to the fast erroneous wave front further in the k-region), as well as the possibility for smaller errors.

3.4. IMAGINARY PART

The dampening part can be included using the same approximation (19). The main thing to check is that the interpolation does not lead to negative damping contribution (blow up instead of damping), and that sufficient damping is present.

3.5. DISCRETIZATION

We consider here the discretization in the lateral coordinates x and k_x. Suppose first $n = 2$, so that x is of dimension 1. Suppose data is given on an evenly spaced grid with N points and distance h, on the interval $x_{\min} = 0, x_{\max} = Nh$. In the Fourier domain we have N points with grid distance $\frac{2\pi}{Nh}$, covering the interval $[-k_{x,\max}, k_{x,\max}]$ where

$$k_{x,\max} = \frac{\pi}{h},$$

We have to deal with the periodic nature of both position space and Fourier space (periodicity in frequency domain related to aliasing, in space domain sometimes called

FIG. 1. The functions f, f_1, f_2

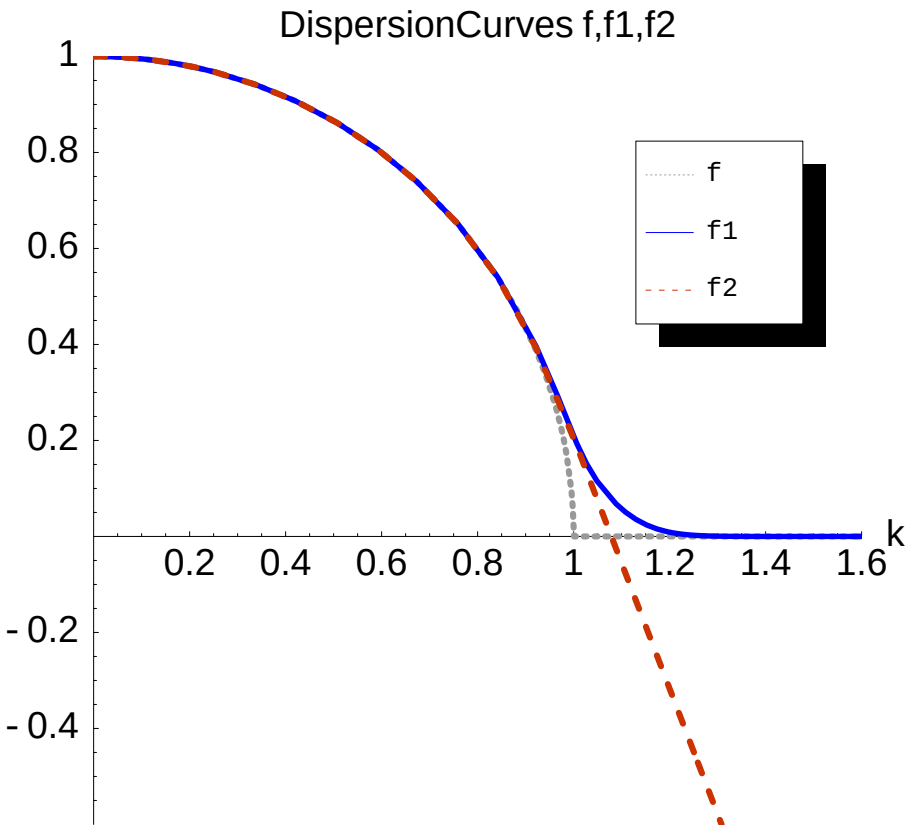


FIG. 2. Velocity curves associated with the functions f, f_1, f_2

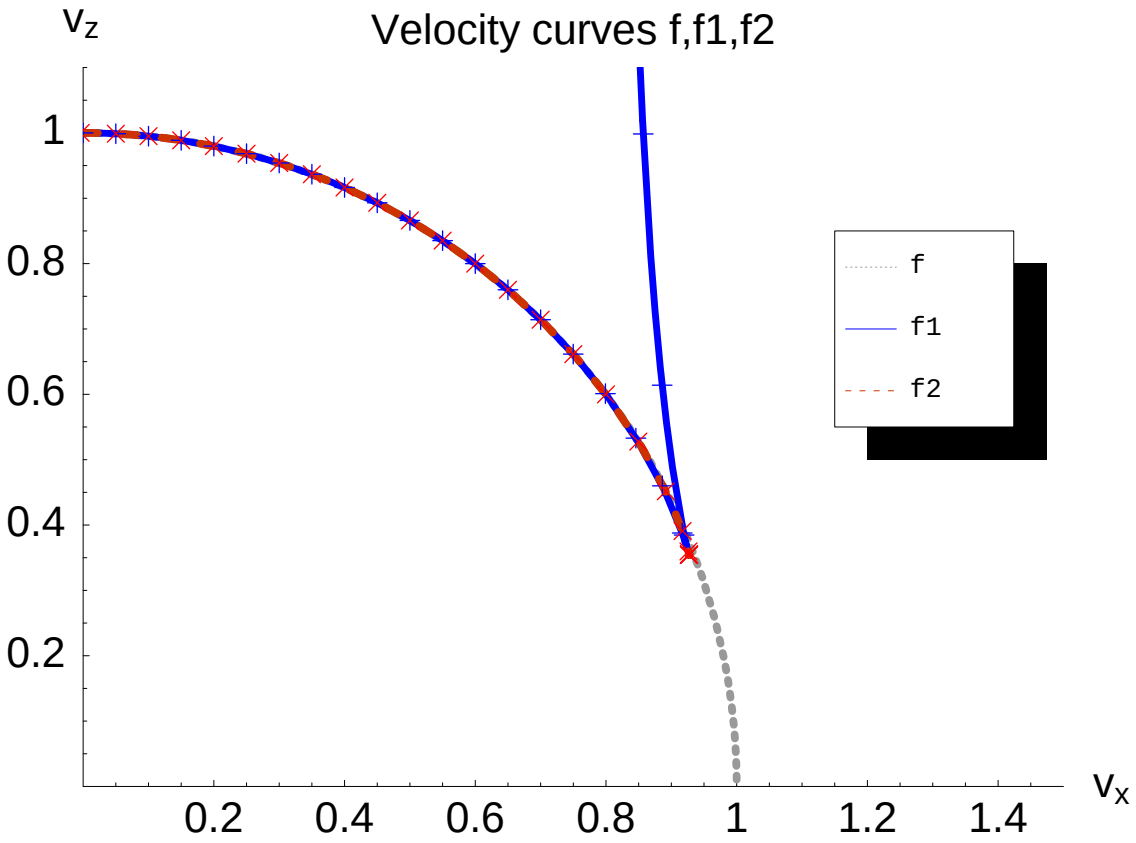


FIG. 3. Errors in velocity parallel (inline) and orthogonal to correct velocity for f_1

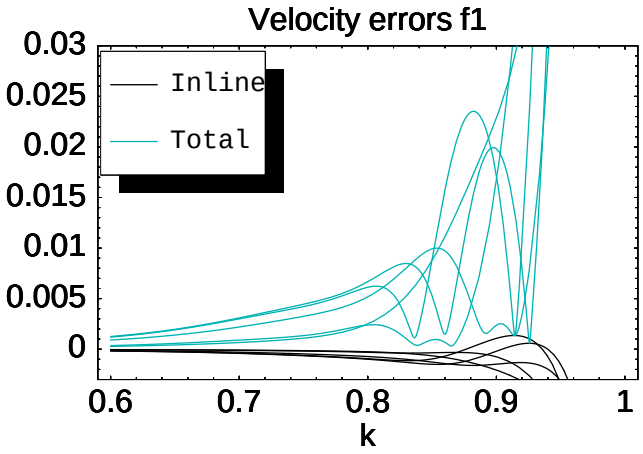
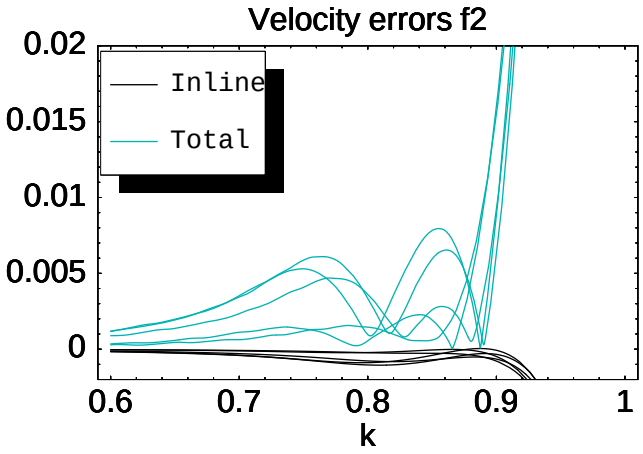


FIG. 4. Errors in velocity parallel (inline) and orthogonal to correct velocity for f_2



"wraparound"). The symbols for the propagating and dissipative contributions $b(\nu(x), k_x)$ and $c(\nu(x), k_x)$ must therefore satisfy

1. $b(\nu(x), k_x)$ and $c(\nu(x), k_x)$ must be smooth as periodic functions on $[0, x_{\max}] \times [-k_{x,\max}, k_{x,\max}]$, i.e. values and derivatives at $x = 0$ and at $x = x_{\max}$ must agree, same for $k_{x,\max}$ and $k_{x,\max}$.

2. $c > 0$ at $k_x = \pm k_{x,\max}$

3. An additional dissipative term must be added, that is nonzero around $x = 0$ and $x = x_{\max}$. In the depth evolution below we will implement this by an explicit exponential decay (and not solve a differential equation for this).

It appears that the width of the frequency band should be related to the scale of the variations in the medium, since the variations in the medium cause the "spreading" in the Fourier domain. The size of the x-interval around $x = 0, x = x_{\max}$ where the space dissipative term is non-zero should probably be related to the maximal value of $\frac{\partial x}{\partial z}$, and the maximal stepsize in the depth stepping algorithm ("fast" signal must not be able to pass through it).

3.6. SUMMARY

The action of $s_{\mathrm{p}}\mathrm{i}B(z,x,D_x,\omega) - s_{\mathrm{d}}C(z,x,D_x,\omega)$ is hence approximated by a discrete operator

$$M(z)\widehat{u} = \tfrac{1}{2}\sum_{j=1}^{K} w_j(\nu(z,\cdot))\mathcal{F}^{-1}\big((\mathrm{i}s_{\mathrm{p}}\widehat{b}(\nu_j) - s_{\mathrm{d}}\widehat{c}(\nu_j))\mathcal{F}\widehat{u}\big) + \tfrac{1}{2}\sum_{j=1}^{K} \mathcal{F}^{-1}\big(\mathrm{i}s_{\mathrm{p}}\widehat{b}(\nu_j) - s_{\mathrm{d}}\widehat{c}(\nu_j))\mathcal{F}(w_j(\nu(z,\cdot))\widehat{u})\big). \quad (33)$$

The differential equation reads

$$\frac{\partial \widehat{u}}{\partial z} = M(z)\widehat{u}. \quad (34)$$

4. SOLUTION BY NUMERICALLY SOLVING A "PRECONDITIONED" ODE

4.1. THE "PRECONDITIONED" ODE

When the coefficient ν is equal to a constant ν_0, then the system (33) is diagonal in the Fourier domain, with matrix $M_0 = \mathrm{i}s_{\mathrm{p}}\widehat{b}(\nu_j) - s_{\mathrm{d}}\widehat{c}(\nu_j)$. The solution to (34) is given by an exponential

$$\widehat{u}(z) = \mathrm{e}^{(z-z_0)M_0}\widehat{u}(z_0), \quad (35)$$

that is straightforward to compute in the Fourier domain. When ν varies not too much about ν_0, then (35) is an approximate solution to (34) and the difference

$$\widehat{u}(z) - \mathrm{e}^{(z-z_0)M_0}\widehat{u}(z_0) \quad (36)$$

should vary slowly. Now define

$$\widehat{v}(z; M_0, z_0) = \mathrm{e}^{-(z-z_0)M_0}\widehat{u}. \quad (37)$$

Then (36) implies that

$$\widehat{v}(z) - \widehat{u}(z_0) \quad (38)$$

should vary slowly. In fact $\widehat{v}$ satisfies the differential equation

$$\frac{\partial \widehat{v}}{\partial z}(z) = \mathrm{e}^{-(z-z_0)M_0}(M(z) - M_0)\mathrm{e}^{(z-z_0)M_0}\widehat{v}(z). \quad (39)$$

Here, if ν is in an interval close to ν_0 the matrix $M - M_0$ is small. Also we will assume $z - z_0$ is small, so that the r.h.s. of (39) is smaller than the r.h.s. of the original ODE (34). Equation (39) is of course valid for any M_0. For this to be useful, M_0 should simply be diagonal in the Fourier domain and such that $M - M_0$ is small.

We could call (39) a preconditioned system[1],because it is somewhat similar to the idea of preconditioning (modifying a system wit a smart guess so that convergence becomes better).

Another advantage of (39) is that the maximal error is not at $k \approx 0$, but will more likely be at large values of $|k|$, with near horizontal propagation, since there the difference $M - M_0$ tends to be the largest.

We propose to solve (39) by a Runge-Kutta method. Because these are one-steps methods it is not too difficult to change from $v(z; M_0, z_0)$ to some $v(z; M_1, z_1)$, and in addition they do not require special startup procedures (as appear to be needed in multistep methods).

When space decay included in the operator M, then $M - M_0$ is generally no longer small, and the "preconditioning" appears too difficult to apply. Therefore we propose to apply an exponential decay factor at each step, where the exponent is non-zero only around the boundary of the domain.

4.2. VARIABLES AND SEQUENCE OF STEPS INVOLVED IN THE COMPUTATION

In the Runge-Kutta approach in general the Runge-Kutta steps will not necessarily be equal to the grid points in depth. We distinguish the depth stepping algorithm from the output algorithm.

Depth stepping

The main variables here are the depths for the Runge-Kutta depth stepping z_j, the unpreconditioned values of $\widehat{u}$ at these depths denoted by $\widehat{u}_j$, and the $\widehat{v}_{j+1,j}$, that will denote the approximation to $v(z_{j+1}; M_j, z_j)$. The $\widehat{u}_j, \widehat{v}_{j+1,j}$ will be stored in Fourier domain.

Computational steps are

Step 1 $\widehat{v}_{j+1,j} = \text{RK-step}(z_{j+1}, z_j, \widehat{u}_j, \ldots)$
Step 2 If adaptive stepping: check error estimate, if needed change z_{j+1} and redo step 1
Step 3 Multiply by exponential space-decay factor, non-zero exponent around boundary
Step 4 $\text{Interpolated\_output}(\widehat{v}_{j+1,j}, \widehat{u}_j, z_{j+1}, z_j, M_j)$
Step 5 $\widehat{u}_{j+1} = \mathrm{e}^{(z_{j+1}-z_j)M_j}\widehat{v}_{j+1,j}$, $j = j+1$, go back to step 1

Number of FFT's (the dominant contribution to the cost for application to migration) is given by[2] $2Kn_{\text{RK}}$, where n_{RK} is the number of function evaluations per RK-step and K is the number of terms in multiplication/convolution approximation.

Output

The output involves interpolation between $\widehat{v}_{j+1,j}$ and $\widehat{u}_j$, to obtain $\widehat{v}_{\text{out}}$ at intermediate

[1] I still have to check for official terminology.

[2] Assumes symmetrized selfadjoint form, use that one of $\mathcal{F}$, $\mathcal{F}^{-1}$ can be done outside the sum, and that weights add up to one $\sum_{j=1}^{K} w_j = 1$

depths, multiplication according to

$$\widehat{u}_{\text{out}} = \mathrm{e}^{(z_{\text{out}}-z_j)M_j}\widehat{v}_{\text{out}},$$

and then if needed Fourier transformation back to the space domain. The latter would be the dominant cost.

5. NUMERICAL EXAMPLES

We hope to address this in a future version. Numerical examples so far show good behavior in smooth models with large lateral velocity variations. There is not yet a good comparison in speed, but the method is similar to generalized screens.

6. DISCUSSION

Discussion will also be included in a future version.

Appendix A. LEFT AND RIGHT PSEUDODIFFERENTIAL OPERATORS

In this appendix we use the following notation. We let $x \in \mathbb{R}^n$, $D_x = -\mathrm{i}\partial/\partial x$. By α, β we denote multi-indices, $\alpha = (\alpha_1, \ldots, \alpha_n)$, $\partial_x^\alpha = \partial_{x_1}^{\alpha_1} \ldots \partial_{x_n}^{\alpha_n}$, $\alpha! = \alpha_1! \ldots \alpha_n!$, $|\alpha| = \sum_{j=1}^n \alpha_j$.

A (standard) symbol of order m is a function $a \in C^\infty(\mathbb{R}^n \times \mathbb{R}^n)$ that satisfies

$$\left|\partial_{k_x}^\alpha \partial_x^\beta a(x, k_x)\right| \leq C_{\alpha,\beta}(1 + |k_x|)^{m-|\alpha|},$$

the set of such symbols is denoted S^m or $S^m(\mathbb{R}^n \times \mathbb{R}^n)$, we denote $S^{-\infty} = \bigcap_m S^m$.

With such a symbol we can define a pseudodifferential operator. There are multiple ways to do this (quantizations). The standard or left quantization is given by

$$\mathrm{Op}_L Au = (2\pi)^{-n} \int_{\mathbb{R}^n} \int_{\mathbb{R}^n} A(x, k_x)\mathrm{e}^{\mathrm{i}(x-y)\cdot k_x} u(y)\, \mathrm{d}k_x\, \mathrm{d}y \tag{A-1}$$

We will use standard notation $A(x, D) = \mathrm{Op}_L A$. This corresponds to first multiplying in Fourier space, and then in position space. Alternatively we have the following definition.

$$\mathrm{Op}_R Au = (2\pi)^{-n} \int_{\mathbb{R}^n} \int_{\mathbb{R}^n} A(y, k_x)\mathrm{e}^{\mathrm{i}(x-y)\cdot k_x} u(y)\, \mathrm{d}k_x\, \mathrm{d}y \tag{A-2}$$

that is equal to $(\mathrm{Op}_L \overline{A})^*$. The symmetric choice is to replace $A(x, k_x)$ in (A-1) by $A((x + y)/2, k_x)$ which leads to the Weyl quantification. Unlike the previous choices it it selfadjoint when a is real. This will not be considered here, because there seems to be no approximation method comparable to the one discussed in section 3. To have a selfadjoint operator associated with a real symbol we will consider $\frac{1}{2}(\mathrm{Op}_L A + \mathrm{Op}_R A)$.

The adjoint of a pseudodifferential operator is again a pseudodifferential operator with (left-) symbol

$$\sum_\alpha \frac{(-\mathrm{i})^{|\alpha|}}{\alpha!}\partial_{k_x}^\alpha \partial_x^\alpha \overline{a}(x,k_x), \qquad \text{modulo } S^{-\infty}. \tag{A-3}$$

The product of two pseudodifferential operators is again a pseudodifferential operator. The (left-) symbol of the operator $\mathrm{Op}_L A \,\mathrm{Op}_L B$ is given by

$$A\#B = A\#_L B = \sum_\alpha \frac{(-\mathrm{i})^{|\alpha|}}{\alpha!}\frac{\partial^\alpha A}{\partial k_x^\alpha}\frac{\partial^\alpha B}{\partial x^\alpha}, \qquad \text{modulo } S^{-\infty}. \tag{A-4}$$

Most of the operators we will consider are polyhomogeneous, meaning that $A(x,k_x) = \sum_{k<=m} A_m$ where each A_k is homogeneous of degree k in k_x for $|k_x|$ greater than some constant C.

Proof of Lemma 1 It is sufficient to find a sequence of symbols $B^{(k)} \in S^1$, $k = 0,1,\ldots,$ such that $R^{(k)} := B^{(k)2} - A$ is in S^{1-k} on (8). We let $B^{(0)}(z,x,k_x,\omega) = \sqrt{A(z,x,k_x,\omega)}$, then

$$B^{(0)2} - A = (-\mathrm{i})\sum_{i=1}^{n-1}\frac{\partial B^{(0)}}{\partial k_x}\frac{\partial B^{(0)}}{\partial x} + \text{ l.o.t.}$$

Now if we have such $B^{(k)}$, then we can set

$$B^{(k+1)} = B^{(k)} - \frac{1}{2B^{(0)}}R^{(k)},$$

which proves the statement for B. Similarly we can prove the other statements.

Appendix B. SMOOTH CUTOFF AND SOME OTHER FUNCTIONS

We give here some functions that are used in the main text. A smooth "cutoff" function $h_{(1)}(y)$, $y \in \mathbb{R}$, that is zero for $y < 0$ and 1 for $y > 1$ is given by

$$h_{(1)}(y) = \begin{cases} 0 & y \le 0 \\ \frac{\mathrm{e}^{-1/y}}{\mathrm{e}^{-1/y}+\mathrm{e}^{-1/(1-y)}} & y \in [0,1] \\ 1 & y \ge 1. \end{cases} \tag{B-1}$$

A function that is the identify $h_{(2)}(y) = y$ for $y < 0$, and becomes constant equal to 1 for $y > 2$ is given by

$$h_{(2)}(y) = \int_0^y (1 - h^{(1)}(\tilde{y}/2))\,\mathrm{d}\tilde{y}. \tag{B-2}$$

A localized bump at $y = 0$ is given by

$$h_{(3)}(y) = \mathrm{e}^{-(\sqrt{\pi}y)^2}. \tag{B-3}$$

REFERENCES

Biondi, B. (2002). Stable wide-angle fourier finite-difference downward extrapolation of 3-d wavefields. *Geophysics*, 67(03):872–882.

de Hoop, M. V., Le Rousseau, J. H., and Wu, R.-S. (2000). Generalization of the phase-screen approximation for the scattering of acoustic waves. *Wave Motion*, 31(1):43–70.

Gazdag, J. and Sguazzero, P. (1984). Migration of seismic data by phase-shift plus interpolation. *Geophysics*, 49(02):124–131.

Huang, L. Y., Fehler, M. C., and Wu, R. S. (1999). Extended local born fourier migration method. *Geophysics*, 64(5):1524–1534.

Kincaid, D. and Cheney, W. (1996). *Numerical analysis*. Brooks/Cole Publishing Co., Pacific Grove, CA, second edition. Mathematics of scientific computing.

Ristow, D. and Ruhl, T. (1994). Fourier finite-difference migration. *Geophysics*, 59(12):1882–1893.

Stoer, J. and Bulirsch, R. (2002). *Introduction to numerical analysis*, volume 12 of *Texts in Applied Mathematics*. Springer-Verlag, New York, third edition. Translated from the German by R. Bartels, W. Gautschi and C. Witzgall.

Stoffa, P. L., Fokkema, J. T., de Luna Freire, R. M., and Kessinger, W. P. (1990). Split-step fourier migration. *Geophysics*, 55(04):410–421.

Taylor, M. E. (1975). Reflection of singularities of solutions to systems of differential equations. *Comm. Pure Appl. Math.*, 28:457–478.