Geophys. J. Int. (2003) **154,** 212–223

RESEARCH NOTE

Yet another elastic plane-wave, layer-matrix algorithm

C. H. Chapman

Schlumberger Cambridge Research, High Cross, Madingley Road, Cambridge CB3 0EL, *UK*

Accepted 2003 February 3. Received 2003 January 29; in original form 2002 August 29

SUMMARY

Many techniques have been developed to calculate the elastodynamic response of a stack of plane layers to a plane, spectral wave. These variations on the original propagator matrix method, commonly called the *Haskell matrix* method, aim to improve the efficiency and numerical stability of the algorithm. In this note, a simple variant, based on the Langer block-diagonal decomposition and second-order minors, is presented. The method is valid in isotropic media and anisotropic media with up–down symmetry on vertical planes of symmetry. The method is efficient and extremely simple to program—code in Matlab* is presented for the isotropic case.

Key words: elastic-wave theory, elastodynamics, layered media, numerical techniques, seismic anisotropy.

1 INTRODUCTION

The elastodynamic response of a stack of plane layers to a plane, spectral wave is of fundamental importance in seismology. The propagator matrix solution is commonly called the *Haskell matrix* method after the classic publication on surface waves (Haskell 1953—it is sometimes called the *Thomson–Haskell* method after the earlier publication by Thomson 1950, although that contained an error). The Haskell matrix method, and variants thereof, have been used for many studies of the reflectivity of waves, and surface or guided waves, in stacks of layers. They can be used to generalized the interface reflection/transmission coefficients used in ray theory, to model approximately the frequency-dependent effect of a layered structure at the reflector, or to model the complete response of a plane layered structure to an impulsive, point source, e.g. the reflectivity method (for instance, Fuchs & Müller 1971; Kennett 1983).

Unfortunately for elastic waves the propagator method is numerically unstable at high frequencies if waves are evanescent in some layers. The dominant evanescent behaviour of the *P* waves compared with *SV* waves, for a given frequency and wave slowness parallel to the layers, causes loss of numerical precision in any fixed word length calculation. Essentially, rounding errors in the *SV*-wave solutions grow exponentially as the *P* wave, and their independence is lost. Several solutions to this problem have been published. The independence of the solutions can be maintained by re-orthogonalizing the solutions at each interface, a method introduced by Pitteway (1965) in a related problem for radio waves. A similar approach has been used in seismology by Chapman & Phinney (1972) and Wang (1999). Alternatively, the original matrix system can be replaced by a second-order minor system, the so-called Δ-matrix method (Dunkin 1965; Thrower 1965), in which only the exponentially dominant solution is required. A faster version, the reduced Δ-matrix method (in which the sixth-order system is reduced to a fifth-order system), was developed later (Watson 1970). This method was widely used in seismology before the ray expansion method was developed by Kennett (1974, 1983). In this method, rather than finding the propagator matrix for the wavefields, the 'propagator' for the required reflection/transmission coefficients is found directly. All terms calculated represent rays in the ray expansion in the layer stack. Only exponentially small (not large) terms arise as rays must decay in their propagation direction. The Kennett algorithm is now very widely used for numerical stability, but also partly because it allows control of internal reflections and reverberations, i.e. the terms included in the ray expansion, and partly because it extends without difficulty to anisotropic media and fluid layers.

Several other methods have been developed for solving the boundary conditions of a stack of layers. Knopoff (1964) developed an alternative matrix decomposition of the complete system of equations that is numerically robust and faster than the reduced Δ-matrix system (see Schwab & Knopoff 1972, for a review of the *Knopoff* method and other publications—the method is sometimes called the *Schwab–Knopoff* method). Later Abo-Zena (1979) developed another related approach. Buchen & Ben-Hador (1996) have published a useful comparison of all the above methods.

For most purposes, any of the above methods is adequate. With improvements in computer performance, efficiency is less of an issue than it was when the algorithms were developed. Probably the most widely used algorithm, Kennett's method, is certainly not the most efficient.

*Matlab is a trademark of MathWorks, Inc.

The calculation of the *P* and *SV* phase factors across each layer with trigonometrical functions, complex if attenuation is included, is required by all algorithms and is a significant fraction of the total cost. Nevertheless, the algebra and computer code for these algorithms are not trivial. The matrix elements in the Δ-matrix and Knopoff's methods are many and varied (see, for instance, example code in Schwab & Knopoff 1972). The matrix elements in Kennett's method are reflection/transmission coefficients and these are algebraically complicated (see, for instance, Aki & Richards 1980, pp. 153–154, for the isotropic case). They have been published by many authors starting with Knott (1899) and Zoeppritz (1919) in different forms depending on the coordinate system and basic waves, but as Kennett, Kerry and Woodhouse (1978) have commented, 'these results have in many cases been marred by minor errors and misprints'.

In this note, we describe yet another algorithm. Our motivation is that it is algebraically simple and easy to program, and can be extended to transverse isotropic media. It is based on the Langer block-diagonal decomposition of the differential system, and the second-order minor method. It is numerically robust and efficient but we make no claim that it is better than any of the other algorithms. However, its implementation is certainly significantly simpler and is ideal for implementation in high-level languages such as Matlab or Java. Although the theory of the Langer block-diagonal decomposition and second-order minors is well known, the algorithm has only been described in proceedings of a school (Woodhouse 1980) and only for isotropic media. In this note, we extend the method to anisotropic media with up–down symmetry on planes of symmetry—the important example is transverse isotropic media with a vertical axis of symmetry (TIV). We describe the theory behind the algorithm and give Matlab code for the isotropic case.

2 THEORY

We first review the differential system for plane waves in anisotropic media in order to introduce our notation and discuss the important symplectic symmetries of this system and its eigensolutions. Then we specialize the system to planes of symmetry in anisotropic media with up–down symmetry, which includes isotropic and TIV media. This simplifies the equations to a fourth-order *qP*–*qSV* system that is straightforward to block-diagonalize. Using factorization and second-order minors, a simple algorithm for reflections from a stack of plane layers is obtained. In the Appendix, we give a Matlab program for the algorithm in the isotropic case.

2.1 Symplectic symmetries and eigensolutions

It is well known that in stratified elastic media, e.g. when the density and anisotropic elastic parameters are only functions of the vertical coordinate (say, x_3), then the equations of motion and the constitutive relations can be reduced to an ordinary differential system by taking the Fourier transforms with respect to time (so $\partial/\partial t \to -i\omega$) and with respect to horizontal coordinates (so $\partial/\partial x_\nu \to i\omega p_\nu$ where $\nu = 1$ or 2). With a notation that is close to that of Woodhouse (1974), we have

$$\frac{d}{dx_3}\mathbf{w} = i\omega\mathsf{A}\mathbf{w}, \tag{1}$$

where

$$\mathbf{w} = \begin{pmatrix} \mathbf{v} \\ \mathbf{t}_3 \end{pmatrix}, \tag{2}$$

and $\mathbf{v}$ and $\mathbf{t}_3$ are the particle velocity and the traction on a surface perpendicular to the x_3 axis, respectively. Woodhouse (1974) has shown how to obtain the matrix A in terms of the density, anisotropic elastic parameters (written in terms of 3×3 matrices C_{jk}, where $(\mathsf{C}_{jk})_{il} = c_{ijkl}$), and transform variables, p_ν.

In a homogeneous layer, the solutions of the differential system (1) can be expressed in terms of the eigensolutions of the matrix A. The eigenequations can be written as

$$\mathsf{AW} = \mathsf{W}\mathbf{p}_3, \tag{3}$$

where W is a matrix with the eigenvectors as columns and $\mathbf{p}_3$ is the diagonal matrix of eigenvalues, the vertical slownesses. A propagator matrix from z_B to z_A can then be written as

$$\mathsf{P}(z_\mathrm{A}, z_\mathrm{B}) = \mathsf{W}\mathrm{e}^{i\omega\mathbf{p}_3 d}\mathsf{W}^{-1}, \tag{4}$$

where $d = z_\mathrm{A} - z_\mathrm{B}$ is a layer thickness. The propagator matrix eq. (4) is sometimes known as the *Haskell matrix* (after Haskell 1953).

In isotropic and TIV media it is easy to obtain relatively simple algebraic expressions for the eigenvectors, W, and eigenvalues, $\mathbf{p}_3$. In general anisotropy, normally they must be obtained numerically, but important symmetries always follow from the symmetries of the matrix, A. These symmetries provide the inverse matrix, W^{-1}, without explicit inversion. Chapman (1994) has discussed these symplectic symmetries of the solutions. Let us denote the eigenvector columns of the matrix W by $\mathbf{w}_i$ where $i = 1, \ldots, 6$, arranged so $i = 1, \ldots, 3$ correspond to waves propagating in the positive x_3 direction and $i = 4, \ldots, 6$ in the negative direction. The eigenvectors can be written as

$$\mathbf{w}_i = (\pm 2\rho V_3)_i^{-1/2} \begin{pmatrix} \hat{\mathbf{g}}_i \\ -p_k \mathsf{C}_{3k}\hat{\mathbf{g}}_i \end{pmatrix}, \tag{5}$$

where $\hat{\mathbf{g}}_i$ is the normalized polarization for the ith eigenvector and V_3 is the x_3 component of the group velocity. The positive sign is taken for $i = 1, \ldots, 3$ and the negative sign for $i = 4, \ldots, 6$. The traction $\mathbf{t}_3$ components are found from the constitutive relations. If we define a symplectic transform of the velocity-traction vector, eq. (2), as

$$\mathbf{w}^{\ddagger} = -\begin{pmatrix}\mathbf{t}_3^{\mathrm{T}} & \mathbf{v}^{\mathrm{T}}\end{pmatrix}, \tag{6}$$

then the eigenvectors, eq. (5), satisfy the orthonormal relationship

$$\mathbf{w}_i^{\ddagger}\mathbf{w}_j = \pm\delta_{ij}, \tag{7}$$

where the sign depends on the propagation direction.

2.2 Fourth-order qP–qSV system

We now specialize our discussion to waves in symmetry planes of anisotropic media with up–down symmetries. This includes isotropic and TIV media, and some directions in more general media, e.g. symmetry planes in cubic material. Schoenberg & Protazio (1992) have considered the reflection coefficients in such media exploiting the up–down symmetry. Without loss in generality, we arrange the coordinates so the symmetry plane is the x_1–x_3 plane and take $p_2 = 0$. Henceforth, we drop the subscript on the horizontal slowness, $p = p_1$. Then the sixth-order differential system, eq. (1), separates into fourth- and second-order systems. The fourth-order system, the first, third, fourth and sixth rows and columns, describes waves with the polarization in the plane and the second-order system, the second and fifth rows and columns, has the polarization normal to the plane. We follow the normal convention of referring to these as the qP–qSV and qSH systems. Henceforth, we only consider the qP–qSV system—the vectors $\mathbf{v}$ and $\mathbf{t}_3$ are reduced to two components, and the matrix A is 4×4.

With up–down symmetry, the eigenvalues of the matrix A must exist in positive and negative pairs. We denote these by $\pm q_V$ and $\pm q_P$ for the qSV and P waves, respectively, and order them as

$$\mathsf{p}_3 = \begin{pmatrix} q_V & 0 & 0 & 0 \\ 0 & q_P & 0 & 0 \\ 0 & 0 & -q_V & 0 \\ 0 & 0 & 0 & -q_P \end{pmatrix}. \tag{8}$$

For clarity, we write the matrix of eigenvectors $\mathbf{w}_i$ as

$$\mathsf{W} = (\acute{\mathbf{w}}^V \quad \acute{\mathbf{w}}^P \quad \grave{\mathbf{w}}^V \quad \grave{\mathbf{w}}^P), \tag{9}$$

where the accent indicates the propagation direction and the superscript represents the wave type. The elements of the up and down-going eigenvectors only differ in sign and we must have

$$\mathsf{W} = \begin{pmatrix} g_1^V & g_1^P & g_1^V & g_1^P \\ g_3^V & g_3^P & -g_3^V & -g_3^P \\ \sigma_{13}^V & \sigma_{13}^P & -\sigma_{13}^V & -\sigma_{13}^P \\ \sigma_{33}^V & \sigma_{33}^P & \sigma_{33}^V & \sigma_{33}^P \end{pmatrix}. \tag{10}$$

With these specializations, we can proceed to decompose the matrix A using the Langer block-diagonalization.

2.3 Langer block-diagonalization

If we write

$$\mathbf{w} = \mathsf{L}\mathbf{y}, \tag{11}$$

and apply this transformation to the differential system (1), we obtain

$$\frac{d\mathbf{y}}{dz} = i\omega\mathsf{L}^{-1}\mathsf{A}\mathsf{L}\mathbf{y} - \mathsf{L}^{-1}\frac{d\mathsf{L}}{dz}, \tag{12}$$

where in homogeneous layers the final term in system (12) is zero. If $\mathsf{L} = \mathsf{W}$, the differential system is diagonalized, i.e.

$$\mathsf{p}_3 = \mathsf{W}^{-1}\mathsf{A}\mathsf{W}. \tag{13}$$

However, the algebra using the eigenvector matrix, W, is not trivial as all elements are non-zero, and when eigenvalues are equal, degenerates. To obtain a simpler algorithm, we follow a procedure in which the matrices have many zero or unit elements. Let us design the matrix L to block diagonalize the system, i.e.

$$\mathsf{B} = \mathsf{L}^{-1}\mathsf{A}\mathsf{L} = \mathsf{L}^{-1}\mathsf{W}\mathsf{p}_3\mathsf{W}^{-1}\mathsf{L} = \begin{pmatrix} \mathsf{B}_V & \mathbf{0} \\ \mathbf{0} & \mathsf{B}_P \end{pmatrix}. \tag{14}$$

It is convenient to write this matrix as

$$\mathsf{B} = \mathsf{B}_V \oplus \mathsf{B}_P. \tag{15}$$

We refer to this as the Langer block-diagonal decomposition as it is useful for studying the Langer asymptotic solution (Langer 1937) for inhomogeneous media where a vertical waveslowness (8), q_V or q_P, has a zero. Wasow (1965) has developed it for general differential systems, and Chapman (1974) and Woodhouse (1978) have introduced it to seismology.

A suitable form for the matrix $\mathbf{L}$ is

$$\mathbf{L} = \frac{1}{2}\mathbf{W}\begin{pmatrix} n_1 & n_2 & 0 & 0 \\ 0 & 0 & n_3 & n_4 \\ n_1 & -n_2 & 0 & 0 \\ 0 & 0 & -n_3 & n_4 \end{pmatrix} \tag{16}$$

$$= \frac{1}{2}\left[n_1(\acute{\mathbf{w}}^V + \grave{\mathbf{w}}^V) \quad n_2(\acute{\mathbf{w}}^V - \grave{\mathbf{w}}^V) \quad n_3(\acute{\mathbf{w}}^P - \grave{\mathbf{w}}^P) \quad n_4(\acute{\mathbf{w}}^P + \grave{\mathbf{w}}^P)\right], \tag{17}$$

where the normalization factors n_i are to be determined. The elements of $\mathbf{L}$ are obtained from linear combinations of the columns of $\mathbf{W}$ with revised normalizations. For the moment we just denote the re-normalization by $n_i/2$—the factor of a half is introduced into definition (17) in order to simplify the elements. Using the orthonormal relationships, eq. (7), the inverse matrix $\mathbf{L}^{-1}$ can be obtained as

$$\mathbf{L}^{-1} = \begin{pmatrix} \left(\acute{\mathbf{w}}^V - \grave{\mathbf{w}}^V\right)^{\ddagger}/n_1 \\ \left(\acute{\mathbf{w}}^V + \grave{\mathbf{w}}^V\right)^{\ddagger}/n_2 \\ \left(\acute{\mathbf{w}}^P + \grave{\mathbf{w}}^P\right)^{\ddagger}/n_3 \\ \left(\acute{\mathbf{w}}^P - \grave{\mathbf{w}}^P\right)^{\ddagger}/n_4 \end{pmatrix}. \tag{18}$$

The matrix needed in eq. (14) is

$$\mathbf{W}^{-1}\mathbf{L} = \begin{pmatrix} \grave{\mathbf{w}}^{V\ddagger} \\ \grave{\mathbf{w}}^{P\ddagger} \\ -\acute{\mathbf{w}}^{V\ddagger} \\ -\acute{\mathbf{w}}^{P\ddagger} \end{pmatrix}\mathbf{L} = \frac{1}{2}\begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & -1 & 1 \end{pmatrix}\begin{pmatrix} n_1 & 0 & 0 & 0 \\ 0 & n_2 & 0 & 0 \\ 0 & 0 & n_3 & 0 \\ 0 & 0 & 0 & n_4 \end{pmatrix}, \tag{19}$$

and an equally simple expression for $\mathbf{L}^{-1}\mathbf{W}$. Substituting these expressions in eq. (14) and simplifying, we find

$$\mathbf{B}_\nu = \begin{pmatrix} 0 & q_\nu/g_\nu \\ g_\nu q_\nu & 0 \end{pmatrix}, \tag{20}$$

where ν is V or P (no summation over ν), with

$$g_V = \frac{n_1}{n_2} \tag{21}$$

$$g_P = \frac{n_3}{n_4}. \tag{22}$$

Thus the matrix $\mathbf{L}$, eq. (17), block-diagonalizes the differential system and is clearly effective in obtaining matrices that have many zero or unit elements.

We can now investigate the elements of the matrix $\mathbf{L}$ and its inverse $\mathbf{L}^{-1}$ in more detail. Using expression (10), we have

$$\mathbf{L} = \begin{pmatrix} n_1 g_1^V & 0 & 0 & n_4 g_1^P \\ 0 & n_2 g_3^V & n_3 g_3^P & 0 \\ 0 & n_2\sigma_{13}^V & n_3\sigma_{13}^P & 0 \\ n_1\sigma_{33}^V & 0 & 0 & n_4\sigma_{33}^P \end{pmatrix}. \tag{23}$$

The inverse matrix $\mathbf{L}^{-1}$ can be obtained by two methods. First, from the definition (18), it is

$$\mathbf{L}^{-1} = -2\begin{pmatrix} \sigma_{13}^V/n_1 & 0 & 0 & g_3^V/n_1 \\ 0 & \sigma_{33}^V/n_2 & g_1^V/n_2 & 0 \\ 0 & \sigma_{33}^P/n_3 & g_1^P/n_3 & 0 \\ \sigma_{13}^P/n_4 & 0 & 0 & g_3^P/n_4 \end{pmatrix}. \tag{24}$$

Alternatively, the matrix (23) is effectively two 2×2 blocks, and so can be inverted easily. It is

$$\mathbf{L}^{-1} = \begin{pmatrix} n_4'\sigma_{33}^P & 0 & 0 & -n_4' g_1^P \\ 0 & n_3'\sigma_{13}^P & -n_3' g_3^P & 0 \\ 0 & -n_2'\sigma_{13}^V & n_2' g_3^V & 0 \\ -n_1'\sigma_{33}^V & 0 & 0 & n_1' g_1^V \end{pmatrix}, \tag{25}$$

where the factors n_i' are the corresponding n_i divided by the appropriate determinant of a 2×2 submatrix. The connections are

$$n_1 n_4' = n_1' n_4 = \left(g_1^V \sigma_{33}^P - g_1^P \sigma_{33}^V\right)^{-1} \tag{26}$$

$$n_2 n_3' = n_2' n_3 = \left(g_3^V \sigma_{13}^P - g_3^P \sigma_{13}^V\right)^{-1}. \tag{27}$$

The orthonormality relation (7) can be applied to expression (10), and together with a comparison of results (24) and (25) gives

$$\frac{n_1 n_4'}{2} = \frac{n_1' n_4}{2} = -\frac{2}{n_2 n_3'} = -\frac{2}{n_2' n_3} = -\frac{\sigma_{13}^V}{\sigma_{33}^P} = \frac{g_3^V}{g_1^P} = \frac{\sigma_{13}^P}{\sigma_{33}^V} = -\frac{g_3^P}{g_1^V}. \tag{28}$$

While the equality of all these expressions can be shown explicitly in isotropic media (it reduces to $-(q_P/q_V)^{1/2}$), the results are non-trivial in anisotropic media and would be tedious to prove explicitly. The equalities are useful as they allow factors to be removed which further simplify the matrix $\mathbf{L}$ and its inverse, creating more unit or equal elements.

Using the equalities (28), we define

$$Z_V = \frac{\sigma_{33}^V}{g_1^V} = -\frac{\sigma_{13}^P}{g_3^P} \tag{29}$$

$$Z_P = \frac{\sigma_{33}^P}{g_1^P} = -\frac{\sigma_{13}^V}{g_3^V}. \tag{30}$$

We use the notation Z as these quantities have the dimensions of impedance. They could be called *cross impedances* as they are the ratio of orthogonal components of stress and velocity. We take the subscript from the first definition of each. In isotropic media, they simplify to $2\mu p$ and $2\mu p - \rho/p$, respectively, where μ is the Lamé shear parameter. The normalization (7) is equivalent to $2(Z_P - Z_V) g_1^V g_3^V = 2(Z_V - Z_P) g_1^P g_3^P = 1$. The matrix $\mathbf{L}$, eq. (23), can then be factored as

$$\mathbf{L} = \begin{pmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 1 & 0 \\ 0 & -Z_P & -Z_V & 0 \\ Z_V & 0 & 0 & -Z_P \end{pmatrix} \begin{pmatrix} n_1 g_1^V & 0 & 0 & 0 \\ 0 & n_2 g_3^V & 0 & 0 \\ 0 & 0 & n_3 g_3^P & 0 \\ 0 & 0 & 0 & -n_4 g_1^P \end{pmatrix}$$

$$= \begin{pmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & -1 \\ -Z_P & 0 & 0 & Z_V \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 1 \\ Z_V & 0 & 0 & -Z_P \\ 0 & 0 & -1 & 0 \end{pmatrix} \begin{pmatrix} n_1 g_1^V & 0 & 0 & 0 \\ 0 & n_2 g_3^V & 0 & 0 \\ 0 & 0 & n_3 g_3^P & 0 \\ 0 & 0 & 0 & -n_4 g_1^P \end{pmatrix} \tag{31}$$

(the minus sign is introduced into the trailing normalization matrix anticipating the isotropic results and to simplify later results). An equally simple factorization is obtained for the inverse matrix $\mathbf{L}^{-1}$, eq. (25). The simplicity and repetitiveness of these expression is remarkable enough in isotropic media. In anisotropic media, it is truly unexpected.

The choice of the normalization factors n_i is arbitrary but for symmetry the natural choice is

$$n_1 = n_1' = -\frac{2}{n_2} = \frac{2}{n_2'} = \left(-2\frac{g_3^V}{g_1^V}\right)^{1/2} \tag{32}$$

$$n_3 = -n_3' = -\frac{2}{n_4} = -\frac{2}{n_4'} = \left(2\frac{g_1^P}{g_3^P}\right)^{1/2} \tag{33}$$

(because of the up–down symmetry, the expressions in parentheses are always positive for travelling waves). In isotropic media these simplify to $(2p/q_V)^{1/2}$ and $(2p/q_P)^{1/2}$, respectively. With these we obtain for the factors (21) and (22)

$$g_V = \frac{g_3^V}{g_1^V} \tag{34}$$

$$g_P = -\frac{g_1^P}{g_3^P} \tag{35}$$

(for travelling waves, these are both negative and related to minus the tangent or cotangent of the polarization angle). In isotropic media these simplify to $-p/q_V$ and $-p/q_P$, respectively. We notice that for a horizontally travelling wave, the corresponding factor g_ν is singular, but in the block matrix $\mathbf{B}_\nu$, eq. (20), one element (B_{12}) is zero and the other (B_{21}) remains finite. The trailing normalization matrix in $\mathbf{L}$, eq. (31), simplifies to

$$\left(-2 g_1^V g_3^V\right)^{1/2} \mathbf{I} \oplus \left(2 g_1^P g_3^P\right)^{1/2} \mathbf{I} = s_V \mathbf{I} \oplus s_P \mathbf{I}, \tag{36}$$

say, and similarly for the leading normalization matrix in $\mathbf{L}^{-1}$. These matrices just scale the solutions.

With the block-diagonalization, eqs (14) and (20), the propagator can be written as

$$\mathbf{P}(z_\mathrm{A}, z_\mathrm{B}) = \mathbf{L}\mathbf{X}\mathbf{L}^{-1} = \mathbf{L}(\mathbf{X}_V \oplus \mathbf{X}_P)\mathbf{L}^{-1}, \tag{37}$$

where

$$\mathbf{X}_\nu = \begin{pmatrix} \cos(\omega q_\nu d) & i/g_\nu \sin(\omega q_\nu d) \\ i g_\nu \sin(\omega q_\nu d) & \cos(\omega q_\nu d) \end{pmatrix}. \tag{38}$$

$\mathbf{X}_V$ and $\mathbf{X}_P$ are defined with the appropriate vertical slowness, q_V or q_P, and coefficients, g_V, eq. (34), or g_P, eq. (35), respectively. As the separate waves propagate independently, the scalings in $\mathbf{L}$, eq. (36), and $\mathbf{L}^{-1}$ can be ignored in the propagator. The scaling factors, $s_V s_P$, can be accumulated separately or, as the reflection coefficients only involve ratios, ignored. The propagation term in the propagator, eq. (37), can be factored as

$$\mathbf{X}_V \oplus \mathbf{X}_P = (\mathbf{X}_V \oplus \mathbf{I})(\mathbf{I} \oplus \mathbf{X}_P). \tag{39}$$

This propagator, eq. (37), is perfectly straightforward to compute but will suffer from exactly the same numerical problems as the Haskell matrix, eq. (4). To solve this problem, we use the second-order minors to find the reflection/transmission coefficients we need (Dunkin 1965; Thrower 1965). Combined with the Langer decomposition, eq. (14), this leads to a computationally simple algorithm. As far as I am aware, this has only been published in the proceedings of a school (Woodhouse 1980). The algorithm is simple in the sense that it is easy to program and requires minimal computing operations.

2.4 Reflection coefficients and second-order minors

The second-order minor method has been used by Thrower (1965) and Dunkin (1965) to overcome the numerical difficulties of the Haskell matrix method. First, we discuss the second-order minor algebra for the reflection/transmission coefficients, and then apply it to the Langer decomposition.

Second-order minors are formed from elements of a matrix by forming determinants of pairs of rows and columns. We will be concerned with the fourth-order *qP–qSV* system. The second-order minors of two 4-D vectors, $\mathbf{y}^{(1)}$ and $\mathbf{y}^{(2)}$, can be arranged as a six-dimensional vector. We write this as

$$\{\mathbf{y}^{(1)}, \mathbf{y}^{(2)}\} = \begin{pmatrix} y_1^{(1)} y_2^{(2)} - y_2^{(1)} y_1^{(2)} \\ y_1^{(1)} y_3^{(2)} - y_3^{(1)} y_1^{(2)} \\ y_1^{(1)} y_4^{(2)} - y_4^{(1)} y_1^{(2)} \\ y_2^{(1)} y_3^{(2)} - y_3^{(1)} y_2^{(2)} \\ y_2^{(1)} y_4^{(2)} - y_4^{(1)} y_2^{(2)} \\ y_3^{(1)} y_4^{(2)} - y_4^{(1)} y_3^{(2)} \end{pmatrix}, \tag{40}$$

where $\{\cdots\}$ is the operation of forming the six-dimensional vector of second-order minors. Thus $\{\mathbf{y}^{(1)}, \mathbf{y}^{(2)}\}_k = y_i^{(1)} y_j^{(2)} - y_j^{(1)} y_i^{(2)}$, where by convention we order the pairs of indices $k = 1 \leftrightarrow (i, j) = (1, 2)$, $k = 2 \leftrightarrow (i, j) = (1, 3)$, $k = 3 \leftrightarrow (i, j) = (1, 4)$, $k = 4 \leftrightarrow (i, j) = (2, 3)$, $k = 5 \leftrightarrow (i, j) = (2, 4)$ and $k = 6 \leftrightarrow (i, j) = (3, 4)$. Similarly, we form a 6×6 matrix of second-order minors from a 4×4 matrix, i.e. $\{\mathbf{X}\}_{kn} = X_{il}X_{jm} - X_{jl}X_{im}$, where $k \leftrightarrow (i, j)$ and $n \leftrightarrow (l, m)$. These matrices of second-order minors satisfy the algebra

$$\{\mathbf{X}\}\{\mathbf{Y}\} = \{\mathbf{X}\mathbf{Y}\} \tag{41}$$

$$\{\mathbf{X}\}\left\{\mathbf{y}^{(1)}, \mathbf{y}^{(2)}\right\} = \left\{\mathbf{X}\mathbf{y}^{(1)}, \mathbf{X}\mathbf{y}^{(2)}\right\}. \tag{42}$$

The second-order minors are useful because the expressions for the *qP–qSV* reflection/transmission coefficients can be expressed in terms of second-order minors of the propagator matrix. Computing the second-order minors directly, we can retain numerical accuracy as the only solution required is dominant. Computing them by combining two vectors introduces numerical difficulties as exponentially large terms may be subtracted.

Now we must show how the solution for the *qP–qSV* reflection coefficients can be written in terms of second-order minors. Let us consider a stack of layers with $z_2 > z_3 > \cdots > z_m$ with the jth interface, z_j, at the top of the jth layer. The first and mth layers are half-spaces. We consider reflections from above, i.e. the source and receiver have $x_3 > z_2$ in the first layer (half-space). In terms of the eigenvectors, $\mathbf{W}_1$ and $\mathbf{W}_m$, in the two half-spaces, the two reflection experiments with an incident *qSV* and *qP* wave can be written as

$$\mathbf{W}_1 \begin{pmatrix} R_{VV} & R_{PV} \\ R_{VP} & R_{PP} \\ 1 & 0 \\ 0 & 1 \end{pmatrix} = \mathbf{P}(x_3, z_2)\mathbf{P}(z_2, z_3)\cdots\mathbf{P}(z_{m-1}, z_m)\mathbf{W}_m \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ T_{VV} & T_{PV} \\ T_{VP} & T_{PP} \end{pmatrix}, \tag{43}$$

where, as we will only consider this one situation, we use the simple notation for the reflection/transmission coefficients where the first subscript indicates the incident ray type and the second, the generated ray type, without indicating the ray direction. Let us write

$$\overline{\mathbf{P}}(x_3, z_m) = \mathbf{W}_1^{-1}\mathbf{P}(x_3, z_2)\mathbf{P}(z_2, z_3)\cdots\mathbf{P}(z_{m-1}, z_m)\mathbf{W}_m. \tag{44}$$

Taking the second-order minors of both sides of eq. (43) and using result (42), the sixth row can be solved for $T_{VV}T_{PP} - T_{VP}T_{\mathrm{PSV}}$, which can then be eliminated from the other rows, giving

$$\begin{pmatrix} R_{VV} \\ R_{VP} \\ R_{PV} \\ R_{PP} \end{pmatrix} = \{\overline{\mathbf{P}}\}_{66}^{-1} \begin{pmatrix} \{\overline{\mathbf{P}}\}_{36} \\ \{\overline{\mathbf{P}}\}_{56} \\ -\{\overline{\mathbf{P}}\}_{26} \\ -\{\overline{\mathbf{P}}\}_{46} \end{pmatrix}. \tag{45}$$

Thus the *qP–qSV* reflection coefficients can be computed simply from second-order minors of the matrices in definitions (44). Note that only one solution, the sixth column, of the second-order minors is required. This is the dominant solution, which combines the third and fourth columns, i.e. downgoing solutions, of $\mathbf{W}_m$.

To complete the algorithm, we expand the terms in expression (45) using result (41). Thus the main term from the stack of layers is

$$\mathbf{Z} = \{\mathbf{X}_1\}\left\{\mathbf{L}_1^{-1}\right\}\left(\prod_{j=2}^{m-1}\{\mathbf{L}_j\}\{\mathbf{X}_j\}\left\{\mathbf{L}_j^{-1}\right\}\right)\{\mathbf{L}_m\}, \tag{46}$$

where

$$\left\{\overline{\mathbf{P}}\right\} = \left\{\mathbf{W}_1^{-1}\mathbf{L}_1\right\}\mathbf{Z}\left\{\mathbf{L}_m^{-1}\mathbf{W}_m\right\}. \tag{47}$$

Only the sixth column of $\{\mathbf{L}_m^{-1}\mathbf{W}_m\}$ is required, $\mathbf{x}_m$ say, which with definitions (32)–(35) is

$$\mathbf{x}_m = \frac{1}{n_1n_2n_3n_4}\begin{pmatrix} 0 \\ -n_2n_4 \\ n_2n_3 \\ n_1n_4 \\ -n_1n_3 \\ 0 \end{pmatrix} = -\frac{1}{2(g_Vg_P)^{1/2}}\begin{pmatrix} 0 \\ 1 \\ g_P \\ g_V \\ g_Vg_P \\ 0 \end{pmatrix} \tag{48}$$

(the trailing scalings, eq. (36), can be ignored as they only introduce a common factor $(s_Vs_P)_m$). Similarly using eq. (19), we can expand $\{\mathbf{W}_1^{-1}\mathbf{L}_1\}$. Then result eq. (45) can be rewritten as

$$\begin{pmatrix} R_{VV} \\ R_{VP} \\ R_{PV} \\ R_{PP} \end{pmatrix} = D^{-1}\begin{pmatrix} 0 & g_Vg_P & g_V & -g_P & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2(g_Vg_P)^{1/2}\dfrac{s_V}{s_P} \\ 2(g_Vg_P)^{1/2}\dfrac{s_P}{s_V} & 0 & 0 & 0 & 0 & 0 \\ 0 & -g_Vg_P & g_V & -g_P & 1 & 0 \end{pmatrix}_1 \mathbf{Z}\mathbf{x}_m, \tag{49}$$

where

$$D = (0 \quad g_Vg_P \quad g_V \quad g_P \quad 1 \quad 0)_1\mathbf{Z}\mathbf{x}_m, \tag{50}$$

and the leading scaling from $\{\mathbf{L}_1^{-1}\}$ in eq. (45) has been included in the result (49). Even for a single interface ($m = 2$), this reduces to a simple algorithm for the interface coefficients with

$$\mathbf{Z} = \left\{\mathbf{L}_1^{-1}\right\}\{\mathbf{L}_2\}. \tag{51}$$

With expression (51) substituted in result (49) we obtain results equal to the standard interface coefficients (e.g. Aki & Richards 1980, pp. 153–154, for the isotropic coefficients) but with a relatively simple algorithm.

2.5 Second-order minors and langer decomposition

With the Langer block-diagonal decomposition and factorization, the second-order minors are particularly simple to compute. Ignoring the trailing diagonal matrix in expression (31), which simply forms a diagonal matrix of second-order minors and reduces to a simple scaling, we have

$$\{\mathbf{L}\} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 \\ -Z_P & 0 & 0 & 0 & -Z_V & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & Z_V - Z_P & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & -Z_P & 0 & 0 & 0 & -Z_V \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 \\ -Z_V & 0 & 0 & 0 & -Z_P & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & Z_P - Z_V & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & -Z_V & 0 & 0 & 0 & -Z_P \end{pmatrix}. \tag{52}$$

Multiplication by the matrix $\{\mathbf{L}\}$, eq. (52), can be coded as four multiplications by the submatrix

$$\begin{pmatrix} 1 & 1 \\ -Z_P & -Z_V \end{pmatrix}, \tag{53}$$

with appropriate column and row indices (columns 1 and 5 with rows 1 and 2, and columns 2 and 6 with rows 5 and 6, or with columns reversed). The inverse matrix $\{\mathbf{L}^{-1}\}$ can be similarly factored and reduced to one submatrix.

The second-order minors of the propagation terms (39) are

$$\{\mathbf{X}_V \oplus \mathbf{I}\} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & c & 0 & -is/g & 0 & 0 \\ 0 & 0 & c & 0 & -is/g & 0 \\ 0 & -igs & 0 & c & 0 & 0 \\ 0 & 0 & -igs & 0 & c & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \tag{54}$$

and similarly for $\{\mathbf{I} \oplus \mathbf{X}_P\}$, where c, s and g are used as a shorthand for the appropriate cosine, sine and g_ν functions. These matrices contain the matrix (38) repeatedly, which can be coded as one function.

A Matlab program implementing this algorithm for the isotropic case is included in the Appendix. The algorithm has been validated by comparisons with interface coefficient routines (i.e. implementing Aki & Richards 1980, pp. 153–154, directly) and the Kennett algorithm for multiple layers extended to TI media (Kennett 1983), and gives identical numerical results.

3 CONCLUSIONS

In isotropic media, and on planes of symmetry in anisotropic media with up–down symmetry, the transformed wave equations reduce to a fourth-order differential system describing the *qP–qSV* waves. The Langer block-diagonal decomposition of this differential system, together with factorization and the second-order minor method, provides an alternative algorithm for numerically robust, plane-wave, layer-matrix calculations. Although the development is lengthy, the resultant algorithm is very simple and results in particularly simple computer code. The method can be used for calculations of coefficients from a single interface (the Zoeppritz coefficients) or from a multilayered stack. A Matlab version of the code for isotropic media has been given, which has been validated against single interface code and Kennett's algorithm.

REFERENCES

Abo-Zena, A., 1979. Dispersion function computations for unlimited frequency values, *Geophys. J. R. astr. Soc.,* **58,** 91–105.

Aki, K. & Richards, P.G., 1980. *Quantitative Seismology: Theory and Methods*, Freeman, San Francisco.

Buchen, P.W. & Ben-Hador, R., 1996. Free-mode surface-wave computations, *Geophys. J. Int.,* **124,** 869–887.

Chapman, C.H., 1974. The turning point of elastodynamic waves, *Geophys. J. R. astr. Soc.,* **39,** 613–621.

Chapman, C.H., 1981. Long-period corrections to body-waves: theory, *Geophys. J. R. astr. Soc.,* **64,** 321–372.

Chapman, C.H., 1994. Reflection/transmission coefficient reciprocities in anisotropic media, *Geophys. J. Int.,* **116,** 498–501.

Chapman, C.H. & Phinney, R.A., 1972. Diffracted seismic signals and their numerical solution, *Meth. Comp. Phys.,* **12,** 165–230.

Dunkin, J.W., 1965. Computation of modal solutions in layered, elastic media at high frequencies, *Bull. seism. Soc. Am.,* **55,** 335–358.

Fuchs, K. & Müller, G., 1971. Computation of synthetic seismograms with the reflectivity method and comparison with observations, *Geophys. J. R. astr. Soc.,* **23,** 417–433.

Haskell, N.A., 1953. The dispersion of surface waves on multilayered media, *Bull. seism. Soc. Am.,* **43,** 17–43.

Kennett, B.L.N., 1974. Reflections, rays and reverberations, *Bull. seism. Soc. Am.,* **64,** 1685–1696.

Kennett, B.L.N., 1983. *Seismic Wave Propagation in a Stratified Medium,* Cambridge University Press, Cambridge.

Kennett, B.L.N., Kerry, N.J. & Woodhouse, J.H., 1978. Symmetries in the reflection and transmission of elastic waves, *Geophys. J. R. astr. Soc.,* **52,** 215–230.

Knopoff, L., 1964. A matrix method for elastic wave problems, *Bull. seism. Soc. Am.,* **454,** 431–438.

Knott, C.G., 1899. On the reflexion and refraction of elastic waves, with seismological applications, *Phil. Mag., 5th ser.,* **48,** 64–97.

Langer, R.E., 1937. On the connection formulas and the solution of the wave equation, *Phys. Rev., Ser. 2,* **51,** 669–676.

Pitteway, M.L.V., 1965. The numerical calculation of wave-fields, reflexion coefficients and polarisations for long radio waves in the lower ionosphere, *Phil. Trans. R. Soc. Lond.* A, 1079,**257,** 219–241.

Schoenberg, M. & Protazio, J., 1992. 'Zoeppritz' rationalized and generalized to anisotropy, *J. Seism. Expl.,* **1,** 125–144.
Schwab, F. & Knopoff, L., 1972. Fast surface wave and free-mode computations, *Meth. Comp. Phys.,* **11,** 87–180.
Thomson, W.T., 1950. Transmission of elastic waves through a stratified solid medium, *J. appl. Phys.,* **21,** 89–93.
Thrower, E.N., 1965. The computation of dispersion of elastic waves in layered media, *J. Sound Vib.,* **2,** 210–226.
Wang, R., 1999. A simple orthonormalization method for stable and efficient computation of Green's functions, *Bull. seism. Soc. Am.,* **89,** 733–741.
Wasow, W., 1965. Asymptotic expansions of ordinary differential equations, Vol. 14, *Pure Appl. Math.,* Wiley-Interscience, New York.
Watson, T.H., 1970. A note on fast computation of Rayleigh wave dispersion in the multilayered half-space, *Bull. seism. Soc. Am.,* **60,** 161–166.
Woodhouse, J.H., 1974. Surface waves in laterally varying media, *Geophys. J. R. astr. Soc.,* **37,** 29–51.
Woodhouse, J.H., 1978. Asymptotic results for elastodynamic propagator matrices in plane stratified and spherically stratified Earth models, *Geophys. J. R. astr. Soc.,* **54,** 263–280.
Woodhouse, J.H., 1980. Efficient and stable methods for performing seismic calculations in stratified media, in *Proc. Int. School of Physics 'Enrico Fermi', Course LXXVIII. Physics of the Earth's Interior,* pp. 127–151, eds Dziewonski, A.M. & Boschi, E., North-Holland, Amsterdam.
Zoeppritz, K., 1919. Über Erdbebenwellen VII B. Über reflexion und Durchgang Seismischer Wellen durch Ünstetigkeitsflächen, *Nachr. der Königlichen Gesell. Wiss., Gottingen, math.-phys. KI.,* 66–84.

APPENDIX : MATLAB PROGRAM

The algorithm is so straightforward that we include the outline of a Matlab program for the isotropic case.

```
function [Rpp,Rpv,Rvp,Rvv] = Rcoefficients(p,dOmega,nOmega,properties,nLayers)
% LAYERS - m-file for reflection coefficients from layer stack
% INPUT:
% p = horizontal slowness
% dOmega = circular frequency increment
% nOmega = number of frequencies
%
% properties(i).Alpha = P velocities
% properties(i).Beta = S velocities
% properties(i).Rho = densities
% properties(i).Thick = layer thickness
% nLayers = number of layers (i=1 to nLayers)
%
% Coefficients are properties(1).Thick above
% interface and properties(nLayers).Thick is ignored
%
% OUTPUT:
% Rpp(nOmega) = PP coefficients
% Rpv(nOmega) = PV coefficients
% Rvp(nOmega) = VP coefficients
% Rvv(nOmega) = VV coefficients
%
% RESTRICTIONS: in this simplified version -
% p=0, Beta=0, p*Alpha=1 or p*Beta=1 are not allowed
% Code is so simply we don't use functions except for EigenFactors and
% LayerPhase which need modification for TIV or attenuating media.
% The eigen-factors are independent of frequency but with attenuation they
% can be replaced with nOmega arrays.
% Function EigenFactors needs replacing for TIV media, or with attenuation.
% Function LayerPhase needs modification for attenuation.
% Scaling solutions may be necessary to prevent overflow.
%
% eigen-factors in half-space
[qV,qP,ZV,ZP,gV,gP,sV,sP] = EigenFactors(p,dOmega,nOmega,properties(nLayers));
%
% starting solution in lower half space (48)
x1 = zeros(nOmega,1)';
x2 = ones(nOmega,1)';
x3 = gP.*x2;
x4 = gV.*x2;
```

```
x5 = gV.*x3;
x6 = x1;
%
% loop over layers
for j=nLayers-1:-1:1
    %
    % multiply by {L} for j + 1-th layer - equation (52)
    % first matrix multiply
    wa = x1+x5;
    wb = -ZV.*x1-ZP.*x5;
    wc = x2+x6;
    wd = -ZV.*x2-ZP.*x6;
    % second matrix multiply
    x1 = wc+wa;
    x2 = -ZV.*wc-ZP.*wa;
    x5 = wd+wb;
    x6 = -ZV.*wd-ZP.*wb;
    % scale
    dZ = ZV-ZP;
    x3 = dZ.*x3;
    x4 = dZ.*x4;
    %
    % set up j-th layer
    [qV,qP,ZV,ZP,gV,gP,sV,sP] = EigenFactors(p,dOmega,nOmega,properties(j));
    %
    % multiply by {L^-1} for j-th layer
    % first matrix multiply
    wa = ZV.*x1+x2;
    wb = ZV.*x5+x6;
    wc = -ZP.*x1-x2;
    wd = -ZP.*x5-x6;
    % second matrix multiply
    x1 = -ZP.*wa-wb;
    x2 = -ZP.*wc-wd;
    x5 = ZV.*wa+wb;
    x6 = ZV.*wc+wd;
    % scale
    dZ = ZV-ZP;
    x3 = dZ.*x3;
    x4 = dZ.*x4;
    %
    % multiply by X (avoid if only DC)
    if ( nOmega > 1 )
        % P phase for (I ⊕ X_P)
        [cc,iss] = LayerPhase(dOmega,nOmega,properties(j).Thick,qP);
        wa = cc.*x2-iss.*x3./gP;
        wb = cc.*x3-gP.*iss.*x2;
        wc = cc.*x4-iss.*x5./gP;
        wd = cc.*x5-gP.*iss.*x4;
        % S phase for (X_V ⊕ I)-equation (52)
        [cc,iss] = LayerPhase(dOmega,nOmega,properties(j).Thick,qV);
        x2 = cc.*wa-iss.*wc./gV;
        x4 = cc.*wc-gV.*iss.*wa;
        x3 = cc.*wb-iss.*wd./gV;
        x5 = cc.*wd-gV.*iss.*wb;
    end
```

```
end
%
% form coefficients-equation (49) and (50)
dZ = 2*sqrt(gV.*gP);
x2 = gV.*gP.*x2;
x3 = gV.*x3;
x4 = gP.*x4;
wa = x2+x3+x4+x5;
Rvv = (x2+x3-x4-x5)./wa;
Rpv = dZ.*sP.*x1./(sV.*wa);
Rvp = dZ.*sV.*x6./(sP.*wa);
Rpp = (x3+x5-x2-x4)./wa;
return;

function [qV,qP,ZV,ZP,gV,gP,sV,sP] = EigenFactors(p,dOmega,nOmega,properties)
% return eigen-factors for P and SV waves
ro = properties.Rho;
vp = properties.Alpha;
vs = properties.Beta;
% vertical slownesses
qV = sqrt((1./vs-p)*(1./vs+p));
qP = sqrt((1./vp-p)*(1./vp+p));
% cross-impedances
ZV = 2*ro*vs*vs*p;
ZP = ZV-ro/p;
% polarization (co)tangents
gV = -p/qV;
gP = -p/qP;
% scaling factors
sV = sqrt(p/ro);
sP = sV;
return;

function [cc,iss] = LayerPhase(dOmega,nOmega,thick,q)
% return cosine and i*sine phase terms
posExp = 1;
negExp = 1;
cc(1) = 1;
iss(1) = 0;
if ( nOmega > 1 )
    delExp = exp(complex(0,dOmega*thick)*q);
    for j=2:1:nOmega
        % Scalar q so use exp(i*dOmega*thick*q) increment for efficiency.
        % If q is frequency dependent (i.e. with attenuation) then
        % we will use
        % wqd = (j-1)*dOmega*thick*q(j)
        % c(j) = cos(wqd);
        % iss(j) = complex(0,1)*sin(wqd);
        posExp = posExp*delExp;
        negExp = negExp/delExp;
        cc(j) = .5*(posExp+negExp);
        iss(j) = .5*(posExp-negExp);
    end
end
return;
```

An algorithm in other object-oriented languages, e.g. Java, Fortran 90 and C++, would be very similar, and in Fortan 77 or C not much more complicated as we have only exploited arrays spanning the frequencies. A complete algorithm would need to handle some special cases:

$p = 0$, $\mu = 0$ or $q = 0$. The latter only requires the reduction

$$\mathbf{X}_\nu = \begin{pmatrix} 1 & 0 \\ -ipwd & 1 \end{pmatrix}, \tag{A1}$$

for the corresponding matrix $\mathbf{X}_\nu$, eq. (38). The former two cases require consideration of different systems. If $p = 0$, the *P–SV* waves separate and if $\mu = 0$ the system reduces to the acoustic system. However, as the numerical algorithm is robust, it is easier to handle these cases by replacing p or μ by numerically small quantities. At high frequencies it may be necessary to rescale the six-vector solution at intermediate depths in order to avoid numerical overflow (the final result, eq. 49, only contains ratios, so removing this exponential growth has no effect). Similarly, it may be necessary to subdivide layers in order to prevent overflow in a single layer.