

Chapter 9

Numerical methods

In those pieces of [scientific] apparatus I see not only devices to make the forces of nature serviceable in new ways, no, I view them with much greater respect; I dare say that I see in them the true devices for unveiling the essence of things.

Ludwig Boltzmann (1886) (commenting on Lord Kelvin's idea to found a mathematical institute for computations (Broda, 1983)).

Seismic numerical modeling is a technique for simulating wave propagation in the earth. The objective is to predict the seismogram that a set of sensors would record, given an assumed structure of the subsurface. This technique is a valuable tool for seismic interpretation and an essential part of seismic inversion algorithms.

To solve the equation of motion by direct methods, the geological model is approximated by a numerical mesh; that is, the model is discretized in a finite numbers of points. These techniques are also called grid methods and full-wave equation methods, since the solution implicitly gives the full wave field. Direct methods do not have restrictions on the material variability and can be very accurate when a sufficiently fine grid is used. Although they are more expensive than analytical and ray methods in terms of computer time, the technique can easily handle the implementation of different rheologies. Moreover, the generation of snapshots can be an important aid in interpretation.

Finite-differences (FD), pseudospectral (PS) and finite-element (FE) methods are considered in this chapter. The main aspects of the modeling are introduced as follows: (a) time integration, (b) calculation of spatial derivatives, (c) source implementation, (d) boundary conditions, and (e) absorbing boundaries. All these aspects are discussed and illustrated in the next sections, using the acoustic and SH equations of motion.

9.1 Equation of motion

Consider the lossless acoustic and SH equations of motion which describe propagation of compressional and pure shear waves, respectively.

The pressure formulation for inhomogeneous media can be written as

$$-L^2 p + f = \partial_{tt}^2 p, \quad -L^2 = \rho c^2 \partial_i (\rho^{-1} \partial_i) \quad (9.1)$$

where x_i , $i = 1, 2, 3$ are Cartesian coordinates, $p(x_i)$ is the pressure, $c(x_i)$ is the velocity

of the compressional wave, $\rho(x_i)$ is the density and $f(x_i, t)$ is the body force. Repeated indices imply summation over the number of spatial dimensions.

The propagation of SH waves is a two-dimensional phenomenon, with the particle velocity, say v_2 , perpendicular to the plane of propagation. Euler's equation and Hooke's law yield the particle-velocity/stress formulation of the SH equation of motion,

$$\partial_t \underline{\mathbf{v}} = \mathbf{H} \cdot \underline{\mathbf{v}} + \underline{\mathbf{f}}, \quad (9.2)$$

where

$$\underline{\mathbf{v}} = (v_2, \sigma_{32}, \sigma_{12})^\top, \quad \underline{\mathbf{f}} = (f, 0, 0)^\top, \quad (9.3)$$

$$\mathbf{H} \cdot \underline{\mathbf{v}} = \mathbf{A} \cdot \partial_1 \underline{\mathbf{v}} + \mathbf{B} \cdot \partial_3 \underline{\mathbf{v}}, \quad (9.4)$$

$$\mathbf{A} = \begin{pmatrix} 0 & 0 & \rho^{-1} \\ 0 & 0 & 0 \\ \mu & 0 & 0 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 0 & \rho^{-1} & 0 \\ \mu & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad (9.5)$$

σ denotes stress and μ is the shear modulus (see Chapter 1). The form (9.2) is representative of most of the equations of motion used in seismic wave propagation, regardless of the stress-strain relation. The solution to equation (9.2) subject to the initial condition $\underline{\mathbf{v}}(0) = \underline{\mathbf{v}}_0$ is formally given by

$$\underline{\mathbf{v}}(t) = \exp(t\mathbf{H}) \cdot \underline{\mathbf{v}}_0 + \int_0^t \exp(\tau\mathbf{H}) \cdot \underline{\mathbf{f}}(t - \tau) d\tau, \quad (9.6)$$

where $\exp(t\mathbf{H})$ is called the evolution operator, because application of this operator to the initial condition array (or to the source array) yields the solution at time t . We refer to \mathbf{H} as the propagation matrix. The SH and acoustic differential equations are hyperbolic (Jain, 1984, p. 251; Smith, 1985, p. 4), since the field has a finite velocity.

The standard variational formulation used in finite-element methods is written in terms of the pressure. To obtain the variational formulation, consider a volume Ω bounded by a surface S . The surface S is divided into S_p , where pressure boundary conditions are defined, and S_{dp} , where normal accelerations (or pressure fluxes) are given. Assume a small pressure variation δp that is consistent with the boundary conditions. The variational principle is obtained by multiplying equation (9.1) by δp , and integrating over the volume Ω and by parts (using the divergence theorem),

$$\int_{\Omega} \frac{1}{\rho} \partial_i \delta p \partial_i p \, d\Omega = - \int_{\Omega} \frac{\delta p}{\rho c^2} \partial_{tt}^2 p \, d\Omega + \int_{\Omega} \frac{f \delta p}{\rho c^2} \, d\Omega + \int_{S_{dp}} \frac{\delta p}{\rho} \partial_i p n_i \, dS, \quad (9.7)$$

where n_i are the components of the normal to the surface S . This formulation is equivalent to a Galerkin procedure (Zienkiewicz, 1977, p. 70; Hughes, 1987, p. 7).

9.2 Time integration

The numerical solution of the equation of motion requires the discretization of the time variable using finite differences. (An exception to this is the spectral methods discussed later). The basic idea underlying FD methods is to replace the partial derivatives by approximations based on Taylor-series expansions of functions near the point of interest.

Forward and backward difference approximations of the time derivatives (Smith, 1985, p. 7) lead to explicit and implicit FD schemes, respectively. Explicit means that the wave field at a present time is computed from the wave field at past times. On the other hand, in implicit methods the present values depend on past and future values. Unlike explicit methods, implicit methods are unconditionally stable, but lead to extensive computations due to the need to carry out large matrix inversions. In general, the differential formulation of the equation of motion is solved with explicit algorithms, since the time step is determined by accuracy criteria rather than by stability criteria (Emmerman, Schmidt and Stephen, 1982)

Equations of motion used in seismic exploration and seismology can be expressed as $\partial_t \mathbf{v} = \mathbf{H} \cdot \mathbf{v}$, where \mathbf{H} is the propagation matrix containing the material properties and spatial derivatives (e.g., equations (1.47) and (9.2)). Assume constant material properties and a plane-wave kernel of the form $\exp(i\mathbf{k} \cdot \mathbf{x} - i\omega_c t)$, wherein \mathbf{k} is the real wavenumber vector, \mathbf{x} is the position vector and ω_c is a complex frequency. Substitution of the plane-wave kernel into the equation of motion yields an eigenvalue equation for the eigenvalues $\lambda = -i\omega_c$. For the acoustic and SH equations of motion, these eigenvalues lie on the imaginary axis of the λ -plane. For instance, in 1-D space, the eigenvalues corresponding to equation (9.2) are $\lambda = \pm ikc$, where c is the shear-wave velocity.

In seismic modeling, there are other equations of interest in which eigenvalues might lie in the left-hand λ -plane. We describe some of these below. Consider an anelastic medium described by a viscoelastic stress-strain relation. Wave attenuation is governed by material relaxation times, which quantify the response time of the medium to a perturbation. (Lossless (elastic) solid materials respond instantaneously, i.e., the relaxation time is zero.) For a viscoelastic medium with moderate attenuation, the eigenvalues have a small negative real part, meaning that the waves are attenuated. In addition, when solving the equations in the time domain, there are eigenvalues with a large negative part and close to the real axis that are approximately given by minus the reciprocal of the relaxation times corresponding to each attenuation mechanism. Then, the domain of the eigenvalues has a T shape (see Tal-Ezer, Carcione and Kosloff, 1990). If the central frequency of these relaxation peaks is close to the source frequency band, or equivalently, if the related eigenvalues are close to the imaginary axis of the λ -plane, an explicit scheme performs very efficiently.

In order to determine the efficiency of an explicit scheme applied to porous media, it is critical to understand the roles of the eigenvalues. For porous media, the eigenvalue corresponding to the slow wave at seismic frequencies (a quasi-static mode) has a very large negative part, which is related to the location of the Biot relaxation peaks, usually beyond the sonic band for pore fluids like water and oil (Carcione and Quiroga-Goode, 1996). When the modulus of the eigenvalues is very large compared to the inverse of the maximum propagation time, the differential equation is said to be stiff (Jain, 1984, p. 72; Smith, 1985, p. 198). Although the best algorithm would be an implicit method, the problem can still be solved with explicit methods (see below).

Denote the discrete time by $t = ndt$, where dt is the time step, and n is a non-negative integer. Time and space discretization of the equation of motion with an explicit scheme – forward time difference only – leads to an equation of the form $\mathbf{v}^{n+1} = \mathbf{G} \cdot \mathbf{v}^n$, where \mathbf{G} is called the amplification matrix. The Neumann condition for stability requires $\max|g_j| \leq 1$, where g_j are the eigenvalues of \mathbf{G} (Jain, 1984, p. 418). This condition does

not hold for all dt when explicit schemes are used, and we note that implicit schemes do not have any restrictions on the time step. For instance, explicit fourth-order Taylor and Runge-Kutta methods require $dt|\lambda_{\max}| < 2\sqrt{2}$ (Jain, 1984, p. 71), implying very small time steps for very large eigenvalues. Implicit methods are A-stable (Jain, 1984, p. 118), meaning that the domain of convergence is the left open-half λ -plane. However, stability does not mean accuracy and, therefore, the time step must comply with certain requirements.

9.2.1 Classical finite differences

Evaluating the second time derivative in equation (9.1) at $(n+1)dt$ and $(n-1)dt$ by a Taylor expansion, and summing both expressions, we obtain

$$\partial_{tt}^2 p^n = \frac{1}{dt^2} \left[p^{n+1} + p^{n-1} - 2p^n - 2 \sum_{l=2}^L \frac{dt^{2l}}{(2l)!} \frac{\partial^{2l} p^n}{\partial t^{2l}} \right]. \quad (9.8)$$

The wave equation (9.1) provides the high-order time derivatives, using the following recursion relation

$$\frac{\partial^{2l} p^n}{\partial t^{2l}} = -L^2 \frac{\partial^{2l-2} p^n}{\partial t^{2l-2}} + \frac{\partial^{2l-2} f^n}{\partial t^{2l-2}}. \quad (9.9)$$

This algorithm, where high-order time derivatives are replaced by spatial derivatives, is often referred to as the Lax-Wendroff scheme (Jain, 1984, p. 415; Smith, 1985; p. 181; Dablain, 1986; Blanch and Robertsson, 1997). A Taylor expansion of the evolution operator $\exp(dt \mathbf{H})$ is equivalent to a Lax-Wendroff scheme.

The dispersion relation connects the frequency with the wavenumber and allows the calculation of the phase velocity corresponding to each Fourier component. Time discretization implies an approximation of the dispersion relation, which in the continuous case is $\omega = ck$ for equations (9.1) and (9.2). Assuming constant material properties and a 1-D wave solution of the form $\exp(ikx - i\bar{\omega}ndt)$, where k is the wavenumber and $\bar{\omega}$ is the FD angular frequency, we obtain the following dispersion relation

$$\frac{2}{dt} \sin\left(\frac{\omega dt}{2}\right) = ck \sqrt{1 - 2 \sum_{l=2}^L (-1)^l \frac{(ckdt)^{2l-2}}{(2l)!}}. \quad (9.10)$$

The FD approximation to the phase velocity is $\bar{c} = \bar{\omega}/k$. Using (9.10) with second-order accuracy (neglect $O(dt^2)$ terms), we find that the FD phase velocity is

$$\bar{c} = \frac{c}{|\text{sinc}(\theta)|}, \quad \theta = \bar{f}dt, \quad (9.11)$$

where $\bar{\omega} = 2\pi\bar{f}$ and $\text{sinc}(\theta) = \sin(\pi\theta)/(\pi\theta)$. Equation (9.11) indicates that the FD velocity is greater than the true phase velocity. Since $\bar{\omega}$ should be a real quantity, thus avoiding exponentially growing solutions, the value of the sine function in (9.10) must be between -1 and 1 . This constitutes the stability criterion. For instance, for second-order time integration this means $ckdt/2 \leq 1$. The maximum phase velocity, c_{\max} , and the maximum wavenumber (i.e. the Nyquist wavenumber, π/dx_{\min}) must be considered. Then, the condition is

$$dt \leq s \left(\frac{dx_{\min}}{c_{\max}} \right), \quad s = \frac{2}{\pi}. \quad (9.12)$$

A rigorous demonstration, based on the amplification factor, is given by Smith (1985, p. 70; see also Celia and Gray, 1992, p. 232). In n -D space, $s = 2/(\pi\sqrt{n})$, and for a fourth-order approximation ($L=2$) in 1-D space, $s = 2\sqrt{3}/\pi$. Equation (9.12) indicates that stability is governed by the minimum grid spacing and the higher velocities.

Let us consider the presence of attenuation. As we have seen in previous chapters, time-domain modeling in lossy media described by viscoelastic stress-strain relations requires the use of memory variables, one for each relaxation mechanism. The introduction of additional differential equations for these field variables avoids the numerical computation of the viscoelastic convolution integrals. The differential equation for a memory variable e in viscoelastic modeling has the form

$$\frac{\partial e}{\partial t} = ae - be, \quad b > 0, \quad (9.13)$$

(see Section 2.7), where ϵ is a field variable, for instance, the dilatation, and a and b are material properties – b is approximately the central angular frequency of the relaxation peak. Equation (9.13) can be discretized by using the central differences operator for the time derivative ($dt(\partial e/\partial t)^n = e^{n+1/2} - e^{n-1/2}$) and the mean value operator for the memory variable ($2e^n = e^{n+1/2} + e^{n-1/2}$). The approximations are used in the Crank-Nicolson scheme (Smith, 1985, p. 19). This approach leads to an explicit algorithm

$$e^{n+1/2} = \frac{2dta}{2+bdt} e^n + \left(\frac{2-bdt}{2+bdt} \right) e^{n-1/2} \quad (9.14)$$

(Emmerich and Korn, 1987). This method is robust in terms of stability, since the coefficient of $e^{n-1/2}$, related to the viscoelastic eigenvalue of the amplification matrix, is less than 1 for any value of the time step dt . The same method performs equally well for wave propagation in porous media (Carcione and Quiroga-Goode, 1996).

9.2.2 Splitting methods

Time integration can also be performed using the method of dimensional splitting, also called Strang's scheme (Jain, 1984, p. 444; Bayliss, Jordan and LeMesurier, 1986; Mufti, 1985; Vafidis, Abramovici and Kanasewich, 1992). Let us consider equation (9.2). The 1-D equations $\partial_t \mathbf{v} = \mathbf{A} \cdot \partial_1 \mathbf{v}$ and $\partial_t \mathbf{v} = \mathbf{B} \cdot \partial_3 \mathbf{v}$ are solved by means of one-dimensional difference operators \mathbf{L}_1 and \mathbf{L}_3 , respectively. For instance, Bayliss, Jordan and LeMesurier (1986) use a fourth-order accurate predictor-corrector scheme and the splitting algorithm $\mathbf{v}^{n+2} = \mathbf{L}_1 \cdot \mathbf{L}_3 \cdot \mathbf{L}_3 \cdot \mathbf{L}_1 \cdot \mathbf{v}^n$, where each operator advances the solution by a half-step. The maximum allowed time step is larger than for unsplit schemes, since the stability properties are determined by the 1-D schemes.

Splitting is also useful when the system of differential equations is stiff. For instance, Biot's poroelastic equations can be partitioned into a stiff part and a non-stiff part, such that the evolution operator can be expressed as $\exp(\mathbf{H}_r + \mathbf{H}_s)t$, where r indicates the regular matrix and s the stiff matrix. The product formulas $\exp(\mathbf{H}_r t) \cdot \exp(\mathbf{H}_s t)$ and $\exp(\frac{1}{2}\mathbf{H}_s t) \cdot \exp(\mathbf{H}_r t) \cdot \exp(\frac{1}{2}\mathbf{H}_s t)$ are first- and second-order accurate, respectively. The stiff part can be solved analytically and the non-stiff part with a standard explicit method (Carcione and Quiroga-Goode, 1996; Carcione and Seriani, 2001). Strang's scheme can be shown to be equivalent to the splitting of the evolution operator for solving the poroelastic equations.

9.2.3 Predictor-corrector methods

Predictor-corrector schemes of different orders find wide application in seismic modeling (Bayliss, Jordan and LeMesurier, 1986; Mufti, 1985; Vafidis, Abramovici and Kanasewich, 1992; Dai, Vafidis and Kanasewich, 1995). Consider equation (9.2) and the first-order approximation

$$\underline{\mathbf{v}}_1^{n+1} = \underline{\mathbf{v}}^n + dt\mathbf{H} \cdot \underline{\mathbf{v}}^n, \quad (9.15)$$

known as the forward Euler scheme. This solution is given by the intersection point between the tangent of $\underline{\mathbf{v}}$ at $t = ndt$ and the line $t = (n+1)dt$. A second-order approximation can be obtained by averaging this tangent with the predicted one. Then the corrector is

$$\underline{\mathbf{v}}^{n+1} = \underline{\mathbf{v}}^n + \frac{dt}{2}(\mathbf{H} \cdot \underline{\mathbf{v}}^n + \mathbf{H} \cdot \underline{\mathbf{v}}_1^{n+1}). \quad (9.16)$$

This algorithm is the simplest predictor-corrector scheme (Celia and Gray, 1992, p. 64). A predictor-corrector MacCormack scheme, second-order in time and fourth-order in space, is used by Vafidis, Abramovici and Kanasewich (1992) to solve the elastodynamic equations.

The Runge-Kutta method

The Runge-Kutta method is popular because of its simplicity and efficiency. It is one of the most powerful predictor-correctors methods, following the form of a single predictor step and one or more corrector steps. The fourth-order Runge-Kutta approximation for the solution of equation (9.2) is given by

$$\underline{\mathbf{v}}^{n+1} = \underline{\mathbf{v}}^n + \frac{dt}{6}(\Delta_1 + 2\Delta_2 + 2\Delta_3 + \Delta_4), \quad (9.17)$$

where

$$\begin{aligned}\Delta_1 &= \mathbf{H}\underline{\mathbf{v}}^n + \underline{\mathbf{f}}^n \\ \Delta_2 &= \mathbf{H}\left(\underline{\mathbf{v}}^n + \frac{dt}{2}\Delta_1\right) + \underline{\mathbf{f}}^{n+1/2} \\ \Delta_3 &= \mathbf{H}\left(\underline{\mathbf{v}}^n + \frac{dt}{2}\Delta_2\right) + \underline{\mathbf{f}}^{n+1/2} \\ \Delta_4 &= \mathbf{H}(\underline{\mathbf{v}}^n + dt\Delta_3) + \underline{\mathbf{f}}^{n+1}.\end{aligned}$$

The stability region extends to $\lambda_{\max} = -2.78$ on the negative real axis and $\lambda_{\max} = \pm i(2\sqrt{2})$ on the imaginary axis, where $\lambda = -i\omega_c$ are the eigenvalues of matrix \mathbf{H} (Jain, 1984, p. 71). Hence, the time step is determined by the relation $dt|\lambda_{\max}| < 2\sqrt{2}$.

9.2.4 Spectral methods

As mentioned before, a Taylor expansion of the evolution operator $\exp(dt\mathbf{H})$ is equivalent to a Lax-Wendroff scheme. Increasing the number of terms in equation (9.8), allows one the use of a larger time step with high accuracy. However, Taylor expansions and Runge-Kutta methods are not the best in terms of accuracy. The evolution operator in equation (9.6) can be expanded in terms of Chebyshev polynomials as

$$\underline{\mathbf{v}}(t) = \sum_{k=0}^M C_k J_k(tR) Q_k \left(\frac{\mathbf{H}}{R}\right) \cdot \underline{\mathbf{v}}_0, \quad (9.18)$$

where $C_0 = 1$ and $C_k = 2$ for $k \neq 0$, J_k is the Bessel function of order k , and Q_k are modified Chebyshev polynomials. R should be chosen larger than the absolute value of the eigenvalues of \mathbf{H} (Tal-Ezer, Kosloff and Koren, 1987). This technique allows the calculation of the wave field with large time steps. Chebyshev expansions are optimal since they require the minimum number of terms. The most time consuming part of a modeling algorithm is the evaluation of the terms $-L^2 p$ in equation (9.1) or $\mathbf{H} \cdot \mathbf{v}$ in equation (9.2), due to the computation of the spatial derivatives. A Taylor-expansion algorithm needs $N = t_{\max}/dt$ of such evaluations to compute the solution at time t_{\max} . On the other hand, the number of evaluations using equation (9.18) is equal to the number of terms in the Chebyshev expansion. Numerical tests indicate that M is comparable to N for second-order finite differencing, but the error of the Chebyshev operator is practically negligible for single-precision programming (Tal-Ezer, Kosloff and Koren, 1987). This means that there is no numerical dispersion due to the time integration.

When the wave equation is second-order in time as in equation (9.1), the REM method (rapid-expansion method) is twice as efficient since the expansion contains only even order Chebyshev functions (Kosloff, Queiroz Filho, Tessmer and Behle, 1989). A similar algorithm for the viscoelastic wave equation is developed by Tal-Ezer, Carcione and Kosloff (1990).

The Chebyshev expansion can also be used for solving parabolic equations (Tal-Ezer, 1989). Let us consider the 2-D electromagnetic diffusion equation (8.419). This equation has the form (9.2) with $\mathbf{v} = H_2$, $\mathbf{H} = \hat{\mu}^{-1}(\partial_1 \hat{\sigma}^{-1} \partial_1 + \partial_3 \hat{\sigma}^{-1} \partial_3)$ and $\mathbf{f} = -\hat{\mu} \partial_t M_2 + (\partial_3 J_1 - \partial_1 J_3)$. The eigenvalue equation in the complex λ -domain ($\lambda = -i\omega_C$), corresponding to matrix \mathbf{H} , is

$$\lambda \left(\lambda + \frac{k_1^2 + k_3^2}{\hat{\mu} \hat{\sigma}} \right) = 0. \quad (9.19)$$

The eigenvalues are therefore zero and real and negative, and the maximum (Nyquist) wavenumber components are $k_1 = \pi/dx$ and $k_3 = \pi/dz$ for the grid spacings dx and dz .

The evolution operator in equation (8.419) can be expanded in terms of Chebyshev polynomials as

$$\mathbf{v}(t) = \sum_{k=0}^M C_k \exp(-bt) I_k(tR) Q_k(\mathbf{F}) \cdot \mathbf{v}_0, \quad (9.20)$$

where

$$\mathbf{F} = \frac{1}{b}(\mathbf{H} + b\mathbf{I}), \quad (9.21)$$

b is the absolute value of the largest eigenvalue of \mathbf{H} , and I_k is the modified Bessel function of order k . The value of b is equal to $(\pi^2/\hat{\mu}\hat{\sigma})(1/dx^2 + 1/dz^2)$. As Tal-Ezer (1989) has shown, the polynomial order should be $O(\sqrt{bt})$ (his equation (4.13)). It can be shown that $M = 6\sqrt{bt}$ is enough to obtain stability and accuracy (Carcione, 2006). The main code (Fortran 77) for solving equation (8.419) is given in the appendix (Section 9.9.1). The spatial derivatives are calculated with the staggered Fourier method (see Section 9.3.2). The complete computer program can be downloaded from <http://software.seg.org> (Carcione, 2006).

These methods are said to have spectral accuracy, in the sense that the error of the approximation tends exponentially to zero when the degree of the approximating polynomial increases.

9.2.5 Algorithms for finite-element methods

In the FE method, the field variables are evaluated by interpolation from nodal values. For a second-order isoparametric method (Zienkiewicz, 1977, p. 178; Hughes, 1987, p. 118), the interpolation can be written as

$$p(x_i) = \Phi^\top \cdot \underline{\mathbf{p}}, \quad (9.22)$$

where $\underline{\mathbf{p}}$ is a column array of the values $p(x_i)$ at the nodes and Φ^\top is a row array of spatial interpolation functions, also referred to as shape and basis functions. The approximation to (9.7) is obtained by considering variations δp according to the interpolation (9.22). Since $\delta p = \Phi^\top \cdot \delta \underline{\mathbf{p}}$, and $\delta \underline{\mathbf{p}}$ is arbitrary, the result is a set of ordinary differential equations at the nodal pressures $\underline{\mathbf{p}}$ (Zienkiewicz, 1977, p. 531; Hughes, 1987, p. 506):

$$\underline{\mathbf{K}} \cdot \underline{\mathbf{p}} + \underline{\mathbf{M}} \cdot \partial_{tt}^2 \underline{\mathbf{p}} + \underline{\mathbf{S}} = 0, \quad (9.23)$$

where $\underline{\mathbf{K}}$ is the stiffness matrix, $\underline{\mathbf{M}}$ is the mass matrix, and $\underline{\mathbf{S}}$ is the generalized source matrix. These matrices contain volume integrals that are evaluated numerically. The matrix $\underline{\mathbf{M}}$ is often replaced by a diagonal lumped mass matrix, such that each entry equals the sum of all entries in the same row of $\underline{\mathbf{M}}$ (Zienkiewicz, 1977, p. 535). In this way, the solution can be obtained with an explicit time-integration method, such as the central difference method (Serón, Sanz, Kindelan and Badal, 1990). This technique can be used with low-order interpolation functions, for which the error introduced by the algorithm is relatively low. When high-order polynomials – including Chebyshev polynomials – are used as interpolation functions, the system of equations (9.23) is generally solved with implicit algorithms. In this case, the most popular algorithm is the Newmark method (Hughes, 1987, p. 490; Padovani, Priolo and Seriani, 1994; Serón, Badal and Sabadell, 1996).

Finally, numerical modeling can be performed in the frequency domain. The method is very accurate but expensive when using differential formulations, since it involves the solution of many Helmholtz equations (Jo, Shin and Suh, 1996). It is more often used in FE algorithms (Marfurt, 1984; Santos, Douglas, Morley and Lovera, 1988; Kelly and Marfurt, 1990).

9.3 Calculation of spatial derivatives

The algorithm used to compute the spatial derivatives usually gives its name to the modeling method. The following sections briefly review these algorithms.

9.3.1 Finite differences

Finite-differences methods use the so-called homogeneous and heterogeneous formulations to solve the equation of motion. In the first case, the motion in each homogeneous region is described by the equation of motion with constant acoustic parameters. For this method, boundary conditions across all interfaces must be satisfied explicitly. The heterogeneous formulation implicitly incorporates the boundary conditions by constructing finite-difference representations using the equation of motion for heterogeneous media.

The homogeneous formulation is of limited use, since it can only be used efficiently for simple geometries. Conversely, the heterogeneous formulation makes it possible to assign different acoustic properties to every grid point, providing the flexibility to simulate a variety of complex subsurface models, e.g., random media, velocity gradients, etc.

In general, staggered grids are used in heterogeneous formulations to obtain stable schemes for large variations of Poisson ratio (Virieux, 1986). In staggered grids, groups of field variables and material properties are defined on different meshes separated by half the grid spacing (Fornberg, 1996, p. 91). The newly computed variables are centered between the old variables. Staggering effectively divides the grid spacing in half, thereby increasing the accuracy of the approximation.

Seismic modeling in inhomogeneous media requires the calculation of first derivatives. Consider the following approximation with an odd number of points, suitable for staggered grids:

$$\frac{\partial p_0}{\partial x} = w_0(p_{\frac{1}{2}} - p_{-\frac{1}{2}}) + \dots + w_l(p_{l+\frac{1}{2}} - p_{-l-\frac{1}{2}}), \quad (9.24)$$

with l weighting coefficients w_l . The antisymmetric form guarantees that the derivative is zero for even powers of x . Let us test the spatial derivative approximation for $p = x$ and $p = x^3$. Requiring that equation (9.24) be accurate for all polynomials up to order 2, we find the approximation $(p_{\frac{1}{2}} - p_{-\frac{1}{2}})/dx$, while for fourth-order accuracy (the leading error term is $O(dx^4)$) the weights are obtained from $w_0 + 3w_1 = 1/dx$ and $w_0 + 27w_1 = 0$, giving $w_0 = 9/(8dx)$, and $w_1 = -1/(24dx)$ (Fornberg, 1996, p. 91).

To obtain the value of the derivative at $x = jdx$, substitute subscript 0 with j , $l + \frac{1}{2}$ with $j + l + \frac{1}{2}$ and $-l - \frac{1}{2}$ with $j - l - \frac{1}{2}$. Fornberg (1996, p. 15) provides an algorithm for computing the weights of first and second spatial derivatives for the general case, i.e., approximations which need not be evaluated at a grid point such as centered and one-sided derivatives. He also shows that the FD coefficients w_l in equation (9.24) are equivalent to those of the Fourier PS method when $l + 1$ approaches the number of grid points (Fornberg, 1996, p. 34).

Let us now study the accuracy of the approximation by considering the dispersion relation. Assuming constant material properties and a 1-D wave solution of the form $\exp(i\bar{k}jdx - i\omega t)$, the second-order approximation gives the following FD dispersion relation and phase velocity:

$$\omega^2 = c^2 \bar{k}^2 \text{sinc}^2(\psi), \quad \bar{c} = c|\text{sinc}(\psi)|, \quad \psi = \bar{K}dx, \quad (9.25)$$

where $\bar{k} = 2\pi\bar{K}$. The spatial dispersion acts in the sense opposite to temporal dispersion (see equation (9.11)). Thus, the FD velocity is smaller than the true phase velocity.

Staggered grids improve accuracy and stability, and eliminate non-causal artifacts (Madariaga, 1976; Virieux, 1986; Levander, 1988; Özdenvar and McMechan, 1997; Carcione and Helle, 1999). Staggered grid operators are more accurate than central differences operators in the vicinity of the Nyquist wavenumber (e.g., Kneib and Kerner, 1993). The particle-velocity/stress formulation in staggered grids constitutes a flexible modeling technique, since it allows us to freely impose boundary conditions and is able to directly yield all the field variables (Karrenbach, 1998).

However, there is a disadvantage in using staggered grids for anisotropic media of symmetry lower than orthorhombic. Staggering implies that the off-diagonal stress and

strain components are not defined at the same location. When evaluating the stress-strain relation, it is necessary to sum over a linear combination of the elasticity constants (c_{IJ} , $I, J = 1, \dots, 6$) multiplied by the strain components. Hence, some terms of the stress components have to be interpolated to the locations where the diagonal components are defined (Mora, 1989). The elasticity constants associated with this interpolation procedure are c_{IJ} , $I = 1, 2, 3$, $J > 3$, c_{45} , c_{46} and c_{56} .

A physical criterion is to compute the weights w_l in equation (9.24) by minimizing the relative error in the components of the group velocity $v_g = \partial\omega/\partial k$. This procedure, combined with grid staggering and a convolutional scheme, yields an optimal differential operator for wave equations (Holberg, 1987). The method is problem dependent, since it depends on the type of equation of motion. Igel, Mora and Riollet (1995) obtain high accuracy with operators of small length (eight points) in the anisotropic case. The treatment of the P-SV case and more details about the finite-difference approximation can be found in Levander (1989).

The modeling algorithm can be made more efficient by using hybrid techniques, for instance, combining finite differences with faster algorithms such as ray tracing methods (Robertsson, Levander and Holliger, 1996) and integral-equation methods (Kummer, Behle and Dorau, 1987). In this way, modeling of the full wave field can be restricted to the target (e.g., the reservoir) and propagation in the rest of the model (e.g., the overburden) can be simulated with faster methods.

Irregular interfaces and variable grid spacing are easily handled by FE methods, since, in principle, grid cells can have any arbitrary shape. When using FD and PS algorithms, an averaging method can be used to reduce spurious diffractions arising from an inappropriate modeling of curved and dipping interfaces (the so-called staircase effect). Muir, Dellinger, Etgen and Nichols (1992) use effective media theory based on Backus averaging to find the elasticity constants at the four grid points of the cell. The modeling requires an anisotropic rheological equation. Zeng and West (1996) obtain satisfactory results with a spatially weighted averaging of the model properties. Similarly, algorithms based on rectangular cells of varying size allow the reduction of both staircase diffractions and the number of grid points (Moczo, 1989; Opršal and Zahradník, 1999). When the grid points are not chosen in a geometrically regular way, combinations of 1-D Taylor series cannot be used and 2-D Taylor series must be applied (Celia and Gray, 1992, p. 93).

A finite-differences code (Fortran 77) for solving the SH-wave equation of motion for anisotropic-viscoelastic media is given in the appendix (Section 9.9.2) and a program for solving Maxwell's equations is given in Section 9.9.3. The latter is based on the acoustic-electromagnetic analogy. Both codes use a fourth-order staggered approximation for computing the spatial derivatives. The error of this approximation is $3 dx^4/640$, compared to $dx^4/60$ for the approximation on a regular grid (Fornberg, 1996, p. 91).

9.3.2 Pseudospectral methods

The pseudospectral methods used in forward modeling of seismic waves are mainly based on the Fourier and Chebyshev differential operators. Gazdag (1981), first, and Kosloff and colleagues, later, applied the technique to seismic exploration problems (e.g., Kosloff and Baysal, 1982; Reshef, Kosloff, Edwards and Hsiung, 1988). Mikhailenko (1985) combined transform methods with FD and analytical techniques.

The sampling points of the Fourier method are $x_j = x_{\max}$, $j = 0, \dots, N_1$, where x_{\max} is the maximum distance and N_1 is the number of grid points. For a given function $f(x)$, with Fourier transform $\tilde{f}(k_1)$, the first and second derivatives are computed as

$$\widetilde{\partial_1 f} = ik\tilde{f}, \quad \widetilde{\partial_1 \partial_1 f} = -k^2\tilde{f}, \quad (9.26)$$

where k is the discrete wavenumber. The transform \tilde{f} to the wavenumber domain and the transform back to the space domain are calculated by the fast Fourier transform (FFT). Staggered operators that evaluate first derivatives between grid points are given by

$$D_1^\pm f = \sum_{k=0}^{k(N_1)} ik \exp(\pm ikdx/2) \tilde{f}(k) \exp(ikx), \quad (9.27)$$

where $k(N_1) = \pi/dx$ is the Nyquist wavenumber. The standard differential operator is given by the same expression, without the phase shift term $\exp(\pm ikdx/2)$. The standard operator requires the use of odd-based FFT's, i.e., N_1 should be an odd number. This is because even transforms have a Nyquist component which does not possess the Hermitian property of the derivative (Kosloff and Kessler, 1989). When $f(x)$ is real, $\tilde{f}(k)$ is Hermitian (i.e., its real part is even and imaginary part is odd). If N_1 is odd, the discrete form of k is an odd function; therefore, $ik\tilde{f}(k)$ is also Hermitian and the derivative is real (see the appendix (Section 9.9.4)).

On the other hand, the first derivative computed with the staggered differential operator is evaluated between grid points and uses even-based Fourier transforms. The approximation (9.27) is accurate up to the Nyquist wavenumber. If the source spectrum is negligible beyond the Nyquist wavenumber, we can consider that there is no significant numerical dispersion due to the spatial discretization. Hence, the dispersion relation is given by equation (9.10), which for a second-order time integration can be written as

$$\bar{\omega} = \frac{2}{dt} \sin^{-1} \left(\frac{ckdt}{2} \right). \quad (9.28)$$

Because k should be real to avoid exponentially growing solutions, the argument of the inverse sine must be less than one. This implies that the stability condition $k_{\max}cdt/2 \leq 1$ leads to $\alpha \equiv cdt/dx \leq 2/\pi$, since $k_{\max} = \pi/dx$ (α is called the Courant number). Generally, a criterion $\alpha < 0.2$ is used to choose the time step (Kosloff and Baysal, 1982). The Fourier method has periodic properties. In terms of wave propagation, this means that a wave impinging on the left boundary of the grid will return from the right boundary (the numerical artifact called wraparound). The Fourier method is discussed in detail in the appendix (Section 9.9.4).

The Chebyshev method is mainly used in the particle-velocity/stress formulation to model free-surface, rigid and non-reflecting boundary conditions at the boundaries of the mesh. Chebyshev transforms are generally computed with the FFT, with a length twice that used by the Fourier method (Gottlieb and Orszag, 1977, p. 117). Since the sampling points are very dense at the edges of the mesh, the Chebyshev method requires a one-dimensional stretching transformation to avoid very small time steps (see equation (9.12)). Because the grid cells are rectangular, mapping transformations are also used to model curved interfaces to obtain an optimal distribution of grid points (Fornberg, 1988;

Carcione, 1994b) and model surface topography (Tessmer and Kosloff, 1994). The Fourier and Chebyshev methods are accurate up to the maximum wavenumber of the mesh that corresponds to a spatial wavelength of two grid points – at maximum grid spacing for the Chebyshev operator. This fact makes these methods very efficient in terms of computer storage – mainly in 3-D space – and makes Chebyshev technique highly accurate for simulating Neumann and Dirichlet boundary conditions, such as stress-free and rigid conditions (Carcione, 1994b). Examples of its use in domain decomposition is given in Carcione (1996a) and Carcione and Helle (2004) to model wave propagation across fractures and at the ocean bottom, respectively. The Chebyshev method is discussed in detail in the appendix (Section 9.9.5).

9.3.3 The finite-element method

The FE method has two advantages over FD and PS methods, namely, its flexibility in handling boundary conditions and irregular interfaces. On the basis of equation (9.22), consider the 1-D case, with uniform grid spacing dx , and an element whose coordinates are X_1 and X_2 ($X_2 - X_1 = dx$) and whose nodal pressures are P_1 and P_2 . This element is mapped into the interval $[-1, 1]$ in a simplified coordinate system (the reference Z -system). Denote the physical variable by x and the new variable by z . The linear interpolation functions are

$$\phi_1 = \frac{1}{2}(1 - z), \quad \phi_2 = \frac{1}{2}(1 + z). \quad (9.29)$$

If the field variable and the independent (physical) variable are computed using the same interpolation functions, one has the so-called isoparametric method (Hughes, 1987, p. 20). That is,

$$p = \phi_1 P_1 + \phi_2 P_2, \quad x = \phi_1 X_1 + \phi_2 X_2. \quad (9.30)$$

Assembling the contributions of all the elements of the stiffness matrix results in a central second-order differencing operator if the density is constant. When the density is variable, the stiffness matrix is equivalent to a staggered FD operator (Kosloff and Kessler, 1989).

FE methods have been used to solve problems in seismology, in particular, propagation of Love and Rayleigh waves in the presence of surface topography (Lysmer and Drake, 1972; Schlue, 1979). FE applications for seismic exploration require, in principle, more memory and computer time than the study of surface waves (soil-structure interaction). In fact, the problem of propagation of seismic waves from the surface to the target (the reservoir) involves the storage of large matrices and much computer time. During the 70s and the 80s, efforts were made to render existing low-order FE techniques efficient rather than proposing new algorithms. In the 90s, Serón, Sanz, Kindelan and Badal (1990) and Serón, Badal and Sabadell (1996) further developed the computational aspects of low-order FE to make them more efficient for seismic exploration problems.

When high-order FE methods are used, we must be aware that besides the physical propagation modes, there are parasitic modes (Kelly and Marfurt, 1990). These parasitic modes are non-physical solutions of the discrete dispersion relation obtained from the Neumann stability analysis. For instance, for a 2D cubic element grid, there are ten modes of propagation – two corresponding to the P and SV waves, and eight parasitic modes of propagation. High-order FE methods became more efficient with the advent of the spectral

element method (SPEM) (Seriani, Priolo, Carcione and Padovani, 1992; Padovani, Priolo and Seriani, 1994; Priolo, Carcione and Seriani, 1994; Komatitsch and Vilotte, 1998; Komatitsch, Barnes and Tromp, 2000). In this method, the approximation functional space is based on high-order orthogonal polynomials having spectral accuracy; that is, the rate of convergence is exponential with respect to the polynomial order. Consider the 2-D case and the acoustic wave equation. The physical domain is decomposed into non-overlapping quadrilateral elements. On each element, the pressure field $p(z_1, z_2)$, defined on the square interval $[-1,1] \times [-1,1]$ in the reference system Z , is approximated by the following product

$$p(z_1, z_2) = \sum_{i=0}^N \sum_{j=0}^N P_{ij} \phi_i(z_1) \phi_j(z_2), \quad (9.31)$$

where P_{ij} are the nodal pressures, and ϕ_i are Lagrangian interpolants satisfying the relation $\phi_i(\zeta_k) = \delta_{ik}$ within the interval $[-1,1]$ and identically zero outside. Here δ_{ik} denotes the Kronecker delta and ζ stands for z_1 and z_2 . The Lagrangian interpolants are given by

$$\phi_j(\zeta) = \frac{2}{N} \sum_{n=0}^N \frac{1}{c_j c_n} T_n(\zeta_j) T_n(\zeta), \quad (9.32)$$

where T_n are Chebyshev polynomials, ζ_j are the Gauss-Lobatto quadrature points, and $c_0 = c_N = 0$, $c_n = 1$ for $1 \leq n \leq N$. The Chebyshev functions are also used for the mapping transformation between the physical world X and the local system Z . Seriani, Priolo, Carcione and Padovani (1992) use Chebyshev polynomials from eighth-order to fifteenth-order. This allow up to three points per minimum wavelength without generating parasitic or spurious modes. As a result, computational efficiency is improved by about one order of magnitude compared to low order FE methods. If the meshing of a geological structure is as regular as possible (i.e., with a reasonable aspect ratio for the elements), the matrices are well conditioned and an iterative method such as the conjugate gradient uses less than eight iterations to solve the implicit system of equations.

9.4 Source implementation

The basic seismic sources are a directional force, a pressure source, and a shear source, simulating, for instance, a vertical vibrator, an explosion, or a shear vibrator. Complex sources, such as earthquakes sources, can be represented by a set of directional forces (e.g., a double couple (Aki and Richards, 1980, p. 82)).

Consider the so-called elastic formulation of the equation of motion, that is, P and S wave propagation (Kosloff, Reshef and Loewenthal, 1984). A directional force vector has components $f_i = a(x_i)h(t)\delta_{im}$, where a is a spatial function (usually a Gaussian), $h(t)$ is the time history, δ denotes the Kronecker delta function, and m is the source direction. A pressure source can be obtained from a potential of the form $\phi = a(x_i)h(t)$ as $f_i = \partial_i \phi$. A shear source is of the form $\mathbf{f} = \text{curl } \mathbf{A}$, where \mathbf{A} is a vector potential. In the (x, y) -plane $\mathbf{A} = (0, 0, A)$ with $A = a(x_i)h(t)$. In particle-velocity/stress formulations, the source can be introduced as described above or in the stress-strain relations, such that a pressure source implies equal contributions to σ_{11} , σ_{22} and σ_{33} at the source location, and shear

sources result from a stress tensor with zero trace (e.g., Bayliss, Jordan and LeMesurier, 1986).

Introducing the source in a homogeneous region by imposing the values of the analytical solution should handle the singularity at the source point. Many FD techniques (Kelly, Ward, Treitel and Alford, 1976; Virieux, 1986) are based on the approach of Alterman and Karal (1968). The numerical difficulties present in the vicinity of the source point are solved by subtracting the field due to the source from the total field due to reflection, refraction and diffractions in a region surrounding the source point. This procedure inserts the source on the boundary of a rectangular region. The direct source field is computed analytically.

On the other hand, when solving the particle-velocity/stress formulation with pseudospectral (PS) algorithms and high-order FD methods (Bayliss, Jordan and LeMesurier, 1986), the source can be implemented in one grid point in view of the accuracy of the differential operators. Numerically (in 1-D space and uniform grid spacing), the strength of a discrete delta function in the space domain is $1/dx$, where dx is the grid size, since each spatial sample is represented by a sinc function with argument x/dx . (The spatial integration of this function is precisely dx .) The introduction of the discrete delta will alias the wavenumbers beyond the Nyquist (π/dx) to the lower wavenumbers. However, if the source time-function $h(t)$ is band-limited with cut-off frequency f_{\max} , the wavenumbers greater than $k_{\max} = 2\pi f_{\max}/c_{\min}$ will be filtered, where c_{\min} is the minimum wave velocity in the mesh. Moreover, since the equation of motion is linear, seismograms with different time histories can be implemented by convolving $h(t)$ with only one simulation using $\delta(t)$ as a source – a discrete delta with strength $1/dt$.

The computation of synthetic seismograms for simulating zero-offset (stacked) seismic sections requires the use of the exploding-reflector concept (Loewenthal, Lu, Roberson and Sherwood, 1976) and the so-called non-reflecting wave equation (Baysal, Kosloff and Sherwood, 1984). A source proportional to the reflection coefficients is placed on the interfaces and is initiated at time zero. All the velocities must be divided by two to get the correct arrival times. The non-reflecting condition implies a constant impedance model to avoid multiple reflections, which are, in principle, absent from stacked sections and constitute unwanted artifacts in migration processes.

9.5 Boundary conditions

Free-surface boundary conditions are the most important in seismic exploration and seismology. They also play an important role in the field of non-destructive evaluation for the accurate sizing of surface breaking cracks (Saffari and Bond, 1987). While in FE methods the implementation of traction-free boundary conditions is natural – simply do not impose any constraint at the surface nodes – FD and PS methods require a special boundary treatment.

Some restrictions arise in FE and FD modeling when large values of the Poisson ratio occur at a free surface. Consider first the free-surface boundary conditions. The classical algorithm used in FD methods (e.g., Kelly, Ward, Treitel and Alford, 1976) includes a fictitious line of grid points above the surface, uses one-sided differences to approximate normal derivatives, and employs central differences to approximate tangential derivatives. This simple low-order scheme has an upper limit of $c_P/c_S \leq 0.35$, where c_P and c_S are the

P-wave and S-wave velocities. The use of a staggered differential operator and radiation conditions of the paraxial type (see below) is effective for any variation of Poisson ratio (Virieux, 1986).

The traction-free condition at the surface of the earth can be achieved by using the Fourier PS method and including a wide zone on the lower part of the mesh containing zero values of the stiffnesses – the so-called zero-padding technique (Kosloff, Reshef and Loewenthal, 1984). While for small angles of incidence this approximation yields acceptable results, for larger angles of incidence, it introduces numerical errors. Free-surface and solid-solid boundary conditions can be implemented in numerical modeling with non-periodic PS operators by using a boundary treatment based on characteristics variables (Kosloff, Kessler, Queiroz Filho, Tessmer, Behle and Strahilevitz, 1990; Kessler and Kosloff, 1991; Carcione, 1991; Tessmer, Kessler, Kosloff and Behle, 1992; Igel, 1999). This method is proposed by Bayliss, Jordan and LeMesurier (1986) to model free-surface and non-reflecting boundary conditions. The method is summarized below (Tessmer, Kessler, Kosloff and Behle, 1992; Carcione, 1994b).

Consider the algorithm for the SH equation of motion (9.2). Most explicit time integration schemes compute the operation $\mathbf{H} \cdot \mathbf{v} \equiv (\mathbf{v})^{\text{old}}$ where \mathbf{H} is defined in equation (9.2). The array $(\mathbf{v})^{\text{old}}$ is then updated to give a new array $(\mathbf{v})^{\text{new}}$ that takes the boundary conditions into account. Consider the boundary $z = 0$ (e.g., the surface) and that the wave is incident on this boundary from the half-space $z > 0$. Compute the eigenvalues of matrix \mathbf{B} : $\pm\sqrt{\mu/\rho} = \pm c$ and 0 (see equation (9.4)). Compute the right eigenvectors of matrix \mathbf{B} , such that they are the columns of a matrix \mathbf{R} . Then, $\mathbf{B} = \mathbf{R} \cdot \Lambda \cdot \mathbf{R}^{-1}$, with Λ being the diagonal matrix of the eigenvalues. If we define the characteristics array as $\mathbf{c} = \mathbf{R}^{-1} \cdot \mathbf{v}$, and consider equation (9.2) corresponding to the z -direction:

$$\partial_t \mathbf{c} = \Lambda \cdot \partial_3 \mathbf{c}, \quad (9.33)$$

the incoming and outgoing waves are decoupled. Two of the characteristics variables, components of array \mathbf{c} , are $v_2 + \sigma_{32}/I$ and $v_2 - \sigma_{32}/I$, with $I = \rho c$. The first variable is the incoming wave and the second variable is the outgoing wave. Equating the new and old outgoing characteristics and assuming stress-free boundary conditions ($\sigma_{32} = 0$), the update of the free-surface grid points is

$$\begin{pmatrix} v \\ \sigma_{12} \\ \sigma_{32} \end{pmatrix}^{\text{new}} = \begin{pmatrix} 1 & 0 & I^{-1} \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} v \\ \sigma_{12} \\ \sigma_{32} \end{pmatrix}^{\text{old}}. \quad (9.34)$$

It can be shown that this application of the method of characteristics is equivalent to a paraxial approximation (Clayton and Engquist, 1977) in one spatial dimension. Robertsson (1996) presents a FD method that does not rely on mapping transformations and, therefore, can handle arbitrary topography, although it must have a staircase shape. The free-surface condition is based on the method of images introduced by Levander (1986). This method is accurate and stable for high values of the Poisson ratio. An efficient solution to the staircase problem is given by Moczo, Bystrický, Kristek, Carcione, and Bouchon (1997), who propose a hybrid scheme based on the discrete-wavenumber, FD and FE methods. These modeling algorithms include attenuation based on memory-variable equations (Emmerich and Korn, 1987; Carcione, Kosloff and Kosloff, 1988d).

9.6 Absorbing boundaries

The boundaries of the numerical mesh may produce non-physical artifacts that disturb the physical events. These artifacts are reflections from the boundaries or wraparounds as in the case in the Fourier method. There are two main techniques used in seismic modeling to avoid these artifacts: the sponge method and the paraxial approximation.

The classical sponge method uses a strip along the boundaries of the numerical mesh, where the field is attenuated (Cerjan, Kosloff, Kosloff and Reshef, 1985; Kosloff and Kosloff, 1986). Considering the pressure formulation, we can write equation (9.1) as a system of coupled equations as

$$\partial_t \begin{pmatrix} p \\ q \end{pmatrix} = \begin{pmatrix} -\xi & 1 \\ -L^2 & -\xi \end{pmatrix} \cdot \begin{pmatrix} p \\ q \end{pmatrix} + \begin{pmatrix} 0 \\ f \end{pmatrix}, \quad (9.35)$$

where ξ is an absorbing parameter. The solution to this equation is a wave traveling without dispersion, but whose amplitude decreases with distance at a frequency-independent rate. A traveling pulse will, thus, diminish in amplitude without a change of shape. An improved version of the sponge method is the perfectly matched-layer method or PML method used in electromagnetism and interpreted by Chew and Liu (1996) as a coordinate stretching. It is based on a – non-physical – modification of the wave equation inside the absorbing strips, such that the reflection coefficient at the strip/model boundary is zero. The improvement implies a reduction of nearly 75 % in the strip thickness compared to the classical method.

The sponge method can be implemented in FE modeling by including a damping matrix \mathbf{D} in equation (9.23),

$$\mathbf{K} \cdot \mathbf{p} + \mathbf{D} \cdot \partial_t \mathbf{p} + \mathbf{M} \cdot \partial_{tt}^2 \mathbf{p} + \mathbf{S} = 0, \quad (9.36)$$

with $\mathbf{D} = \alpha \mathbf{M} + \beta \mathbf{K}$, where α and β are the damping parameters (e.g., Sarma, Mallick and Gadalinglajkar, 1998).

The paraxial approximation method is another technique used to avoid undesirable non-physical artifacts. One-way equations and the method based on characteristics variables discussed in the previous section are particular cases. For approximations based on the one-way wave equation (paraxial) concept, consider the acoustic wave equation on the domain $x \geq 0$. At the boundary $x = 0$, the absorbing boundary condition has the general form

$$\left\{ \prod_{j=1}^J [(\cos \alpha_j) \partial_t - c \partial_1] \right\} p = 0, \quad (9.37)$$

where $|\alpha_j| < \pi/2$ for all j (Higdon, 1991). Equation (9.37) provides a general representation of absorbing boundary conditions (Keys, 1985; Randall, 1988). The reason for the success of equation (9.37) can be explained as follows. Suppose that a plane wave is hitting the boundary at an angle α and a velocity c . In 2-D space, such a wave can be written as $p(x \cos \alpha + z \sin \alpha + ct)$. When an operator of the form $(\cos \alpha) \partial_t - c \partial_1$ is applied to this plane wave, the result is zero. The angles α_j are chosen to take advantage of a priori information about directions from which waves are expected to reach the boundary.

Consider now the approach based on characteristics variables and apply it to the SH equation of motion (9.2) in the plane $z = 0$. The outgoing characteristic variable is

$v_2 - \sigma_{32}/I$ (see the previous section). This mode is left unchanged (new = old), while the incoming variable $v_2 + \sigma_{32}/I$ is set to zero (new = 0). Then, the update of the boundary grid points is

$$\begin{pmatrix} v \\ \sigma_{12} \\ \sigma_{32} \end{pmatrix}^{\text{new}} = \frac{1}{2} \begin{pmatrix} 1 & 0 & I^{-1} \\ 0 & 2 & 0 \\ I & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} v \\ \sigma_{12} \\ \sigma_{32} \end{pmatrix}^{\text{old}}. \quad (9.38)$$

These equations are exact in one dimension, i.e., for waves incident at right angles. Approximations for the 2-D case are provided by Clayton and Engquist (1977).

9.7 Model and modeling design – Seismic modeling

Modeling synthetic seismograms may have different purposes – for instance, to design a seismic experiment (Özdenvar, McMechan and Chaney, 1996), to provide for structural interpretation (Fagin, 1992) or to perform a sensitivity analysis related to the detectability of a petrophysical variable, such as porosity, fluid type, fluid saturation, etc. Modeling algorithms can also be part of inversion and migration algorithms.

Designing a model requires the joint collaboration of geologists, geophysicists and log-analysts when there is well information about the study area. The geological modeling procedure generally involves the generation of a seismic-coherence volume to define the main reservoir units and the incorporation of fault data of the study area. Seismic data require the standard processing sequence and pre-stack depth migration supported by proper inversion algorithms when possible. A further improvement is achieved by including well-logging (sonic- and density-log) information. Since the logs have a high degree of detail, averaging methods are used to obtain the velocity and density field at the levels of seismic resolution.

In planning the modeling with direct methods, the following steps are to be followed:

1. From the maximum source frequency and minimum velocity, find the constraint on the grid spacing, namely,

$$dx \leq \frac{c_{\min}}{2f_{\max}}. \quad (9.39)$$

The equal sign implies the maximum allowed spacing to avoid aliasing; that is, two points per wavelength. The actual grid spacing depends on the particular scheme. For instance, $O(2,4)$ FD schemes require 5 to 8 grid points per minimum wavelength.

2. Find the number of grid points from the size of the model.
3. Allocate additional wavelengths for each absorbing strip at the sides, top and bottom of the model. For instance, the standard sponge method requires four wavelengths, where the wavelength is $\lambda_d = 2c_{\max}/f_d$ and f_d is the dominant frequency of the seismic signal.
4. Choose the time step according to the stability condition (9.12) and accuracy criteria. Moreover, when possible, test the modeling algorithm against the analytical solutions and perform seismic-reciprocity tests to verify its correct performance.
5. Define the source-receiver configuration.

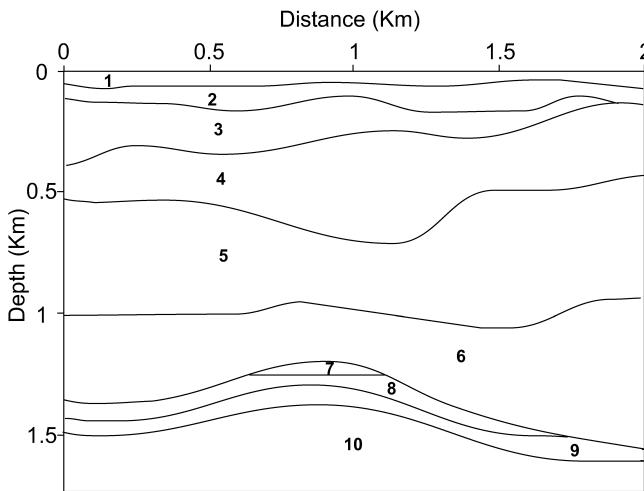


Figure 9.1: Geological model.

Consider the model shown in Figure 9.1 with the properties indicated in Table 9.1. The low velocities and low quality factors of medium 7 simulate a sandstone subjected to an excess pore pressure. All the media have a Poisson ratio equal to 0.2, except medium 7 which has a Poisson ratio of 0.3, corresponding to an overpressured condition. The modeling algorithm (Carcione, 1992a) is based on a fourth-order Runge-Kutta time-integration scheme and the Fourier and Chebyshev methods, which are used to compute the spatial derivatives along the horizontal and vertical directions, respectively. This allows the modeling of free-surface boundary conditions. Since the mesh is coarse (two points per minimum wavelength), Zeng and West's averaging method (Zeng and West, 1996) is applied to the slownesses to avoid diffractions due to the staircase effect – the density and the relaxation times are arithmetically averaged. The mesh has 135×129 points, with a horizontal grid spacing of 20 m, and a vertical dimension of 2181 m with a maximum vertical grid spacing of 20 m. Stress-free and non-reflecting boundary conditions of the type (9.34) and (9.38) are applied at the top and bottom boundaries, respectively. In addition, absorbing boundaries of the type (9.35) of length 18 grid points are implemented at the side and bottom boundaries. The source is a vertical force (a Ricker wavelet) applied at 30 m depth, with a maximum frequency of 40 Hz. The wave field is computed by using a time step of 1 ms with a maximum time of 1 s – the total wall-clock time is 120 s in an Origin 2000 with 4 CPU's. The seismogram recorded at the surface is shown in Figure 9.2, where the main event is the Rayleigh wave (ground-roll) traveling with velocities between the shear velocities of media 1 and 2, approximately. The reflection event corresponding to the anticlinal structure can be clearly seen between 0.6 s and 0.8 s.

Medium	c_P (km/s)	c_S (km/s)	Q_P	Q_S	ρ g/cm ³
1	2.6	1.6	80	60	2.1
2	3.2	1.96	100	78	2.3
3	3.7	2.26	110	85	2.3
4	4	2.45	115	90	2.4
5	4.3	2.63	120	92	2.5
6	4.5	2.75	125	95	2.6
7	3.2	1.7	30	25	2.3
8	4.6	2.82	150	115	2.6
9	4.8	2.94	160	120	2.7
10	5.4	3.3	220	170	2.8

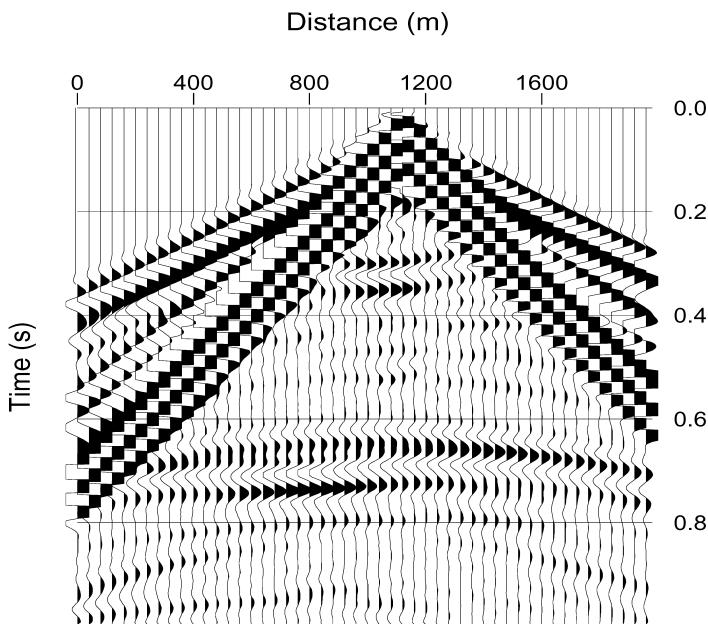
Table 9.1. Material properties

Figure 9.2: Seismogram of the vertical particle velocity.

Forward numerical modeling is a powerful method to aid in the interpretation of seismic surveys. Carcione, Finetti and Gei (2003) use ray tracing, the non-reflecting wave equation and the exploding-reflector approach to interpret low signal-to-noise ratio deep-crust seismic sections. Synthetic seismograms are useful to recognize patterns associated with different types of structures, and predicting some of the drawbacks when interpreting migrated and unmigrated sections of a given complex structure.

Another useful application is seismic characterization. Carcione and Gei (2003) use rock physics, seismic theory and numerical modeling of wave propagation to analyze the seismic response of an antarctic subglacial lake. Optimal seismic surveys can be planned on the basis of this type of investigations.

9.8 Concluding remarks

The direct methods discussed in this chapter (finite-difference, pseudospectral methods and finite-element methods) do not impose restrictions on the type of stress-strain relation, boundary conditions or source-type. In addition, they allow general material variability. For instance, the numerical solution of wave propagation in an anisotropic poro-viscoelastic medium – appropriate for reservoir environments – is not particularly difficult in comparison with simple cases, such as the acoustic wave equation describing the propagation of dilatational waves. Many of the complex stress-strain relations handled by direct methods cannot be solved by integral-equations or asymptotic methods without simplifying assumptions. However, direct methods for solving these equations are certainly more expensive in terms of computer time and storage.

Finite differences are simple to program and are efficient when compared to alternative methods, under fairly mild accuracy requirements. In this sense, a good choice can be a second-order in time, fourth-order in space FD algorithm. Pseudospectral methods can be more expensive in some cases, but guarantee higher accuracy and relatively lower background noise when staggered differential operators are used. These operators are also suitable when large variations of Poisson ratio are present in the model (e.g., a fluid/solid interface). In three dimensions, pseudospectral methods require a minimum of grid points, compared to finite differences, and can be the best choice when limited computer storage is available. However, if a dense grid is required for physical reasons (e.g., fine layering, scattering inhomogeneities, etc.) the FD algorithm can be more convenient.

Without a doubt, the best algorithm to model surface topography and curved interfaces is the finite-element method. With the use of spectral interpolators, this algorithm can compete with earlier techniques with respect to accuracy and stability. However, this approach may prove to be unstable for large variations of the Poisson ratio. Finite-element methods are best suited for engineering problems where interfaces have well defined geometrical features, in contrast with geological interfaces. Moreover, model meshing is not intensively required as is the case in seismic inversion algorithms. Use of non-rectangular grids, mainly in 3-D space, is one of the main disadvantages of finite-element methods, because of the topological problems to be solved when constructing the model. Finite-element methods are, however, preferred for seismic problems involving the propagation of surface waves in situations of complex topography.

9.9 Appendix

9.9.1 Electromagnetic-diffusion code

The following Fortran-77 computer program implements the simulation of the initial-value problem corresponding to the TM equation (8.419) using the expansion (9.20) and the Fourier PS method (equation (9.27)). The same program can be used for the TE equation (8.420) if the conductivity is interchanged with the magnetic permeability and vice versa.

The model is homogeneous, but the properties are defined as arrays, so the program can be used for a general inhomogeneous medium. The first-order spatial derivative computed with the staggered differential operator uses even-based Fourier transforms. The spectral coefficients of the Fourier expansion are computed by the Fast Fourier Transform (FFT) using the algorithm introduced by Temperton (1988), requiring the number of grid points be composed of prime factors formed with 2, 3, 4, 5, 7, 8, 9, 11, 13, 16 and 17.

The routine for the modified Bessel functions is taken from Zhang and Jin (1996), who provide a floppy disk with the program. This code exceeds the dynamic range of the computer (Origin 300) for arguments larger than 700, but a small modification allows the calculation of $\exp(-bt)I_k(bt)$, which poses no difficulties. The main equations describing the algorithm are indicated in the comments.

```

c _____
c Electromagnetic diffusion equation – Magnetic field
c _____
c Section 8.10.2 : Differential equation (8.419)
c Section 8.10.2 : Analytical solution (equation (8.437))
c Section 9.2.4 : Time integration (equation (9.20))
c Section 9.3.2 : Spatial derivatives (equation (9.27))
c _____
parameter (nxt=120, nzt=120, nbes=5000, na=20)
dimension Hy(nxt,nzt),Hy1(nxt,nzt),Hyt(nxt,nzt)
dimension Ex(nxt,nzt),Ez(nxt,nzt)
dimension bk(nbdes)
dimension ifaxx(10),akx(nxt),cox(nxt),six(nxt)
dimension ifaxz(10),akz(nzt),coz(nzt),siz(nzt)
real*8 btd,Ik(0:nbdes),Ikp(0:nbdes),Kk(0:nbdes),Kkp(0:nbdes)
real mu(nxt,nzt),kbar
dimension sigma(nxt,nzt),wx(na),wz(na)
common/rec/b,fac
data pi/3.14150265/
c _____
open(4,file='SNP')
c
c INPUT DATA ——
c Number of grid points
c (These numbers should be even and composed of primes factors)
nx=nxt
nz=nzt
c Grid spacings
dx=10.
dz=dx
c Initial-condition parameters
kbar=0.1

```

```

Dk=0.5*kbar
c Model
c Reference magnetic permeability and conductivity
amu0=4.*pi*1.e-7
s0=0.001
amx=0.
do 1 i=1,nx
do 1 j=1,nz
    mu(i,j)=amu0
    sigma(i,j)=s0
    a=1./(mu(i,j)*sigma(i,j))
1      amx=amax1(amx,abs(a))
     a=amx
c Propagation time
t=3.e-6
c If iab=1 apply absorbing boundaries
iab=1
c
c INITIAL CONDITION -----
x0=0.5*nx*dx
z0=0.5*nz*dx
do 5 i=1,nx
do 5 j=1,nz
    x=(i-1)*dx
    z=(j-1)*dz
    arg=-0.25*Dk*Dk*((x-x0)**2+(z-z0)**2)
    arg1=kbar*(x-x0)
    arg2=kbar*(z-z0)
5      Hy(i,j)=exp(arg)*cos(arg1)*cos(arg2)
c -----
c Wavenumber components and phase shifts for staggering
call wn(akx,cox,six,nx,dx)
call wn(akz,coz,siz,nz,dz)
c Vector-FFT factors
call spfa17(ifaxx,nx)
call spfa17(ifaxz,nz)
c
c ABSORBING BOUNDARIES -----
if(iab.eq.1) then
    nab=18
    gam=1.e+6
    alp=0.1
c Weights for the absorbing strips
    call wgt(wx,nab,gam,alp)
    call wgt(wz,nab,gam,alp)
c Define properties of the bottom strip
    do 10 i=1,nx
        do 10 j=nz-nab+1,nz
            mu(i,j)=mu(i,1)
10           sigma(i,j)=sigma(i,1)
        endif
c -----
        do 11 i=1,nx
        do 11 j=1,nz
            Hy1(i,j)=0.
11           Hyt(i,j)=0.

```

```

b=a*pi*pi*(1./dx/dx+1./dz/dz)
bt=b*t
c
c EXPANSION COEFFICIENTS -----
M=6.*sqrt(bt)
btd=bt
c Ik = exp(-bt) Ik(bt)
call IKNA(M,btd,NM,Ik,Ikp,Kk,Kkp)
bk(1)=Ik(0)
bk(2)=2.*Ik(1)
do 15 k=3,M+1
k1=k-1
15      bk(k)=2.*Ik(k1)
c
c TIME EVOLUTION -----
c First two terms
fac=1.
call cheb(Hy,Hy1,mu,sigma,nx,nz,
&           ifaxx,akx,cox,six,
&           ifaxz,akz,coz,siz,
&           wx,wz,nab,iab)
c
do 20 i=1,nx
do 20 j=1,nz
** Eq. (9.20) **
20      Hyt(i,j)=Hyt(i,j)+bk(1)*Hy(i,j)+bk(2)*Hy1(i,j)
c
c Terms 2,..,M
fac=2.
do 25 k=3,M+1
call cheb(Hy1,Hy,mu,sigma,nx,nz,
&           ifaxx,akx,cox,six,
&           ifaxz,akz,coz,siz,
&           wx,wz,nab,iab)
c
do 30 i=1,nx
do 30 j=1,nz
      Hyt(i,j)=Hyt(i,j)+bk(k)*Hy(i,j)
      hh=Hy1(i,j)
      Hy1(i,j)=Hy(i,j)
30      Hy(i,j)=hh
c
25      continue
c
c COMPUTE ELECTRIC FIELD -----
c
c Spatial derivatives: Dx(+) and Dz(+)
call difx(Hyt,Ez,+1,0,ifaxx,akx,cox,six,nx,nz)
call difz(Hyt,Ex,+1,0,ifaxz,akz,coz,siz,nx,nz)
do 35 i=1,nx
do 35 j=1,nz
      Ex(i,j)=-Ex(i,j)/sigma(i,j)
35      Ez(i,j)=Ez(i,j)/sigma(i,j)
c
c WRITE SNAPSHOTS -----
write(4,*)nx,nz

```

```

do 40 i=1,nx
40      write(4,*)(Hyt(i,j),Ex(i,j),Ez(i,j),j=1,nz)
c _____
      write(6,100)t*1.e+6
100     format(1x,'Propagation time: ',F4.0,' microsec.')
      write(6,101)bt
101     format(1x,'Argument of Bessel functions, bt: ',F5.0)
      write(6,102)M
102     format(1x,'Chebyshev polynomial degree, M: ',I4)
      write(6,103)bk(M)/bk(1)
103     format(1x,'ratio bM/b0: ',E15.7)
c
      stop
      end
c End of main program -----
c
c SUBROUTINES
c _____
c SPATIAL DERIVATIVES AND RECURSION EQUATION -----
c
      subroutine cheb(Hy,Hy1,mu,sigma,nx,nz,
      &           ifaxx,akx,cox,six,
      &           ifaxz,akz,coz,siz,
      &           wx,wz,nab,iab)
      dimension Hy(nx,nz),Hy1(nx,nz)
      dimension a1(nx,nz),a2(nx,nz),a3(nx,nz)
      dimension ifaxx(10),akx(nx),cox(nx),six(nx)
      dimension ifaxz(10),akz(nz),coz(nz),siz(nz)
      dimension sigma(nx,nz),wx(nab),wz(nab)
      real mu(nx,nz)
      common/rec/b,fac
c
c Spatial derivatives: Dx(+) and Dz(+) *** Eq. (9.27) ***
      call difx(Hy,a1,+1,0,ifaxx,akx,cox,six,nx,nz)
      call difz(Hy,a2,+1,0,ifaxz,akz,coz,siz,nx,nz)
c
      do 5 i=1,nx
      do 5 j=1,nz
         5      a1(i,j)=a1(i,j)/sigma(i,j)
         5      a2(i,j)=a2(i,j)/sigma(i,j)
c
c Spatial derivatives: Dx(-) and Dz(-) *** Eq. (9.27) ***
      call difx(a1,a3,-1,0,ifaxx,akx,cox,six,nx,nz)
      call difz(a2,a3,-1,1,ifaxz,akz,coz,siz,nx,nz)
c a3 = ((1/sigma) Hy,x).x + ((1/sigma) Hy,z).z
      c
      do 10 i=1,nx
      do 10 j=1,nz
         10     a3(i,j)=a3(i,j)/mu(i,j)
c
c Apply absorbing boundaries
      if(iab.eq.1) then
         call ab(a3,Hy,wx,wz,nab,nx,nz)
      endif
c Recursion equation
      do 15 i=1,nx

```

```

      do 15 j=1,nz
      GN=a3(i,j)
** Eq. (9.21) **
15      FN=(GN+b*Hy(i,j))/b
      Hy1(i,j)=fac*FN-Hy1(i,j)
c
      return
      end
c _____
c
c Subroutines
c _____
c
c Modified Bessel functions (Zhang and Jin, 1996)
c   subroutine IKNA(n,x,nm,bi,di,bk,dk)
c
c Wavenumber components and phase shifts for staggering
c   subroutine wn(ak,co,si,n,d)
c
c x-derivative
c   subroutine difx(a1,a2,isg,iopt,ifaxx,akx,cox,six,nx,nz)
c z-derivative
c   subroutine difz(a1,a2,isg,iopt,ifaxz,akz,coz,siz,nx,nz)
c
c Vector FFT (Temperton, 1988)
c   subroutine spfa17(ifax,n)
c   subroutine pfa17(a,b,ifax,inc,jump,n,lot,isign,ierr)
c
c Absorbing boundaries (Kosloff and Kosloff, 1986)
c   subroutine wgt(w,nab,gam,alp)
c   subroutine ab(a1,a2,wx,wz,nab,nx,nz)
c _____
c The complete computer program can be downloaded
c from http://software.seg.org

```

9.9.2 Finite-differences code for the SH-wave equation of motion

The following Fortran program solves the inhomogeneous anisotropic and viscoelastic SH-wave equation of motion, which is given in Section 4.5.3. The time discretization of Euler's equation (1.46)₁ has second-order accuracy, and it is based on equation (9.8) (the first three terms on the right-hand side):

$$u_2^{n+1} = 2u_2^n - u_n^{n-1} + dt^2 \rho^{-1} (D_1^- \sigma_6 + D_3^- \sigma_4)^n + f_2^n, \quad (9.40)$$

where $\partial_t u_2 = v_2$, $\sigma_6 = \sigma_{12}$ and $\sigma_4 = \sigma_{23}$. The strain components are obtained as

$$e_4 = D_3^+ u_2, \quad \text{and} \quad e_6 = D_1^+ u_2, \quad (9.41)$$

where D^- and D^+ represent staggered spatial-derivative operators of order 4. The different signs imply a shift of half the grid size, to obtain the acceleration at the same points of the displacement (Carcione, 1999c).

The discretization of the memory-variable equations (4.149)₄ and (4.149)₆ is based on

equation (9.14). For example, the first equation is

$$e_{23}^{n+1/2} = \left(\frac{2 dt \tau_\sigma^{(2)} \varphi_2}{2\tau_\sigma^{(2)} + dt} \right) e_4^n + \left(\frac{2\tau_\sigma^{(2)} - dt}{2\tau_\sigma^{(2)} + dt} \right) e_{23}^{n-1/2}, \quad (9.42)$$

where e_{23} denotes the memory variable, and $\varphi_2 = (\tau_\epsilon^{(2)})^{-1} - (\tau_\sigma^{(2)})^{-1}$.

On a regular grid, the field components and material properties are represented at the same grid points. On a staggered grid, variables and material properties are defined at half-grid points, as shown by Carcione (1999c). Material properties at half-grid points should be computed by averaging the values defined at regular points (not implemented in this program). The averaging is chosen in such a way to reduce the error between the numerical solution corresponding to an interface aligned with the numerical grid and the equivalent solution obtained with a regular grid. Minimum ringing amplitudes are obtained for the arithmetic average of the density and relaxation times, and the geometric average of the shear moduli.

In particular, the program solves the reflection-transmission problem of Section 6.1, for a source of 25 Hz central frequency. The mesh has 120×120 points and a grid spacing of 10 m. A snapshot of the displacement u_2 at 250 ms is shown in Figure 9.3.

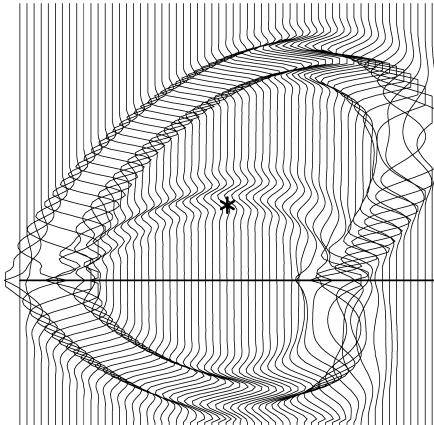


Figure 9.3: Snapshot of the SH-wave displacement, corresponding to the reflection-transmission problem studied in Section 6.1. The star indicates the location of the source.

The comments in the program indicate the different equations used in the simulation.

```
c _____
c Anisotropic, viscoelastic SH-wave propagation
c _____
c Section 4.4 : Plane-wave analysis
c Section 4.5.3 : Differential equations
c Section 4.6 : Analytical solution
c Section 6.1 : Reflection-transmission problem
c Section 8.2.1 : Time integration
c Section 8.3.1 : Spatial derivatives
```

```

c _____
c O(2,4) finite-difference scheme
c _____
c parameter (nxt=120, nzt=120, nstept=500)
c field variables
c u2: displacement
c e4 and e6: strain components
c s4 and s6: stress components
c e23 and e12: memory variables
    dimension u2(nxt,nzt),u(nxt,nzt),s4(nxt,nzt),s6(nxt,nzt)
    dimension e23(nxt,nzt),e12(nxt,nzt)
c material properties
    dimension ts2(nxt,nzt),phi2(nxt,nzt),ts4(nxt,nzt),phi4(nxt,nzt)
    dimension c44(nxt,nzt),c66(nxt,nzt),c46(nxt,nzt),rho(nxt,nzt)
c
    dimension seis(nxt,nstept)
    dimension ab(30)
    dimension f(1000)
c _____
c open(10,file='SNAP')
c open(15,file='SEIS')
    dx=10.
    dz=10.
    dt=0.001
    nx=120
    nz=120
    nstep=250
    pi=3.14159265
c snapshots every nsp steps
    nsp=nstep
c source location and central frequency
    ix=60
    iz=60
    freq=25.
c MODEL _____
c central frequency of relaxation peaks
    f0=25.
    tau=1./(2.*pi*f0)
    do i=1,nx
        do j=1,nz
c upper layer _____
    ** Eqs. (6.22)-(6.24) **
    rho(i,j)=2000.
    c44(i,j)=9.68e+9
    c66(i,j)=12.5e+9
    c46(i,j)=-0.5*sqrt(c44(i,j)*c66(i,j))
c loss in vertical direction
    Q2=30.
    ** Eq. (6.20) **
    ts2(i,j)=(tau/Q2)*(sqrt(Q2*Q2+1.)-1.)
    te2=(tau/Q2)*(sqrt(Q2*Q2+1.))+1.)
    phi2(i,j)=1./te2-1./ts2(i,j)
c loss in horizontal direction
    Q4=40.
    ts4(i,j)=(tau/Q4)*(sqrt(Q4*Q4+1.))-1.)
    te4=(tau/Q4)*(sqrt(Q4*Q4+1.))+1.)

```

```

phi4(i,j)=1./te4-1./ts4(i,j)
c lower layer _____
  if(j.ge.80) then
** Eqs. (6.22)-(6.24) **
  rho(i,j)=2500.
  c44(i,j)=19.6e+9
  c66(i,j)=25.6e+9
  c46(i,j)=0.5*sqrt(c44(i,j)*c66(i,j))
c loss in vertical direction
  Q2=60.
  ts2(i,j)=(tau/Q2)*(sqrt(Q2*Q2+1.)-1.)
  te2=(tau/Q2)*(sqrt(Q2*Q2+1.))+1.)
  phi2(i,j)=1./te2-1./ts2(i,j)
c loss in horizontal direction
  Q4=80.
  ts4(i,j)=(tau/Q4)*(sqrt(Q4*Q4+1.)-1.)
  te4=(tau/Q4)*(sqrt(Q4*Q4+1.))+1.)
  phi4(i,j)=1./te4-1./ts4(i,j)
  endif
  end do
  end do
c _____
  do i=1,nx
  do j=1,nz
  u2(i,j)=0.
  u(i,j)=0.
  end do
  end do
  do n=1,nstep
  f(n)=0.
  end do
c _____
c absorbing parameters
  r=0.99
  nab=12
  do i=1,nab
  ab(i)=r**i
  ab(i)=1.
  end do
c source's wavelet
** Eq. (2.233) **
  call wavelet(f,freq,nw,dt)
c finite-differences weights
** Eq. (9.24) **
  x1=9./(8.*dx)
  x2=-1./(24.*dx)
  z1=9./(8.*dz)
  z2=-1./(24.*dz)
c TIME STEPPING _____
  do 10 n=1,nstep
    if(mod(n,10).eq.0) print *,n
c apply absorbing boundaries _____
c horizontal strips
  do 11 j=1,nab
  j2=j+2
  j3=nz-j-1

```

```

    sab=ab(nab+1-j)
    do 11 i=3,nx-2
    u2(i,j2)=u2(i,j2)*sab
11      u2(i,j3)=u2(i,j3)*sab
c vertical strips
    do 12 i=1,nab
    i2=i+2
    i3=nx-i-1
    sab=ab(nab+1-i)
    do 12 j=3,nz-2
    u2(i2,j)=u2(i2,j)*sab
12      u2(i3,j)=u2(i3,j)*sab
c
    do 13 i=3,nx-2
    do 13 j=3,nz-2
c strains
** Eqs. (9.24) and (9.41) **
c i-3/2 → i-2
c i-1/2 → i-1
c i+1/2 → i
c i+3/2 → i+1
    e4=z1*(u2(i,j)-u2(i,j-1))+z2*(u2(i,j+1)-u2(i,j-2))
    e6=x1*(u2(i,j)-u2(i-1,j))+x2*(u2(i+1,j)-u2(i-2,j))
c memory-variable equations
    f1=2.*ts2(i,j)-dt
    f2=2.*ts2(i,j)+dt
    ee=e23(i,j)
** Eqs. (4.149)4 and (9.42) **
    e23(i,j)=(2.*dt*ts2(i,j)*phi2(i,j)*e4+f1*e23(i,j))/f2
    e23(i,j)=0.5*(e23(i,j)+ee)
    f1=2.*ts4(i,j)-dt
    f2=2.*ts4(i,j)+dt
    ee=e12(i,j)
** Eqs. (4.149)6 and (9.42) **
    e12(i,j)=(2.*dt*ts4(i,j)*phi4(i,j)*e6+f1*e12(i,j))/f2
    e12(i,j)=0.5*(e12(i,j)+ee)
c stresses
** Eq. (4.150) **
    s4(i,j)=c44(i,j)*(e4+e23(i,j))+c46(i,j)*e6
    s6(i,j)=c66(i,j)*(e6+e12(i,j))+c46(i,j)*e4
13      continue
    do 14 i=3,nx-2
    do 14 j=3,nz-2
** Eq. (9.24) **
c i-3/2 → i-1
c i-1/2 → i
c i+1/2 → i+1
c i+3/2 → i+2
    ds4=z1*(s4(i,j+1)-s4(i,j))+z2*(s4(i,j+2)-s4(i,j-1))
    ds6=x1*(s6(i+1,j)-s6(i,j))+x2*(s6(i+2,j)-s6(i-1,j))
c acceleration
    acc=(ds4+ds6)/rho(i,j)
c source
    source=0.

```

```

if(n.le.nw.and.i.eq.ix.and.j.eq.iz) source=f(n)
c Euler's equation
** Eqs. (1.46)1 and (9.40) **
u(i,j)=2.*u2(i,j)-u(i,j)+dt*dt*acc+source
14    continue
c update of displacement
do 15 i=3,nx-2
do 15 j=3,nz-2
uu=u2(i,j)
u2(i,j)=u(i,j)
u(i,j)=uu
15    continue
c
c write snapshot
if(mod(n,nsp).eq.0) then
print *, 'write snapshot',n
write(10,*)nx,nz,dx,dz
do i=1,nx
write(10,*)(u2(i,j),j=1,nz)
end do
endif
c load seismogram at j=25
do i=1,nx
seis(i,n)=u2(i,25)
end do
10    continue
c write seismogram
write(15,*)nx,nstep,dx,dt
do i=1,nx
write(15,*)(seis(i,j),j=1,nstep)
end do
close(10)
close(15)
stop
end
c
c WAVELET
subroutine wavelet(f,fb,nw,dt)
dimension f(nw)
** Eq. (2.233) **
pi=3.14159265
wb=2.*pi*fb
t0=6./(5.*fb)
Dw=0.5*wb
nw=2.*t0/dt
do n=1,nw
t=(n-1)*dt
D=t-t0
f(n)=exp(-Dw*Dw*D*D/4.)*cos(wb*D)
end do
return
end

```

9.9.3 Finite-differences code for the SH-wave and Maxwell's equations

The following Fortran program can be used to solve the SH-wave and Maxwell's equations in inhomogeneous media. The SH-wave differential equations for isotropic media, based on Maxwell's viscoelastic model, can be rewritten from equations (8.26)-(8.28) in the particle-velocity/stress formulation:

$$\begin{aligned}\partial_t v_2 &= \frac{1}{\rho} (\partial_1 \sigma_{12} + \partial_3 \sigma_{23} + f_2) \\ \partial_t \sigma_{23} &= \mu \left(\partial_3 v_2 - \frac{1}{\eta} \sigma_{23} \right) \\ \partial_t \sigma_{12} &= \mu \left(\partial_1 v_2 - \frac{1}{\eta} \sigma_{12} \right),\end{aligned}\quad (9.43)$$

where $c_{44} = c_{66} = \mu$ is the shear modulus, $\tau_{44} = \tau_{66} = 1/\eta$, and η is the shear viscosity.

On the other hand, the TM Maxwell's equations are

$$\begin{aligned}\partial_t H_2 &= \frac{1}{\hat{\mu}} [\partial_1 E_3 + \partial_3 (-E_1) - M_2] \\ \partial_t (-E_1) &= \frac{1}{\hat{\epsilon}} [\partial_3 H_2 - \hat{\sigma}(-E_1)] \\ \partial_t E_3 &= \frac{1}{\hat{\epsilon}} (\partial_1 H_2 - \hat{\sigma} E_3),\end{aligned}\quad (9.44)$$

where $\hat{\epsilon}_{11} = \hat{\epsilon}_{33} = \hat{\epsilon}$ is the dielectric permittivity, and $\hat{\sigma}_{11} = \hat{\sigma}_{33} = \hat{\sigma}$ is the conductivity.

Equations (9.43) and (9.44) are mathematically analogous for the following correspondence:

$$\begin{array}{ccc}v_2 & \Leftrightarrow & H_2 \\ \sigma_{23} & \Leftrightarrow & -E_1 \\ \sigma_{12} & \Leftrightarrow & E_3 \\ \mu & \Leftrightarrow & 1/\hat{\epsilon} \\ \rho & \Leftrightarrow & \hat{\mu} \\ f_2 & \Leftrightarrow & -M_2.\end{array}\quad (9.45)$$

The program is written by using the field variables and material properties of the SH-wave equation. Maxwell's equation can easily be solved by using the correspondence (9.45). The time discretization has fourth-order accuracy, and it is based on the Runge-Kutta approximation (9.17), while the spatial derivatives are computed with the fourth-order staggered operator (9.24). In terms of the staggered operators, equations (9.43) become

$$\begin{aligned}\partial_t v_2 &= \frac{1}{\rho} (D_1^- \sigma_{12} + D_3^- \sigma_{23} + f_2) \\ \partial_t \sigma_{23} &= \mu \left(D_3^+ v_2 - \frac{1}{\eta} \sigma_{23} \right) \\ \partial_t \sigma_{12} &= \mu \left(D_1^+ v_2 - \frac{1}{\eta} \sigma_{12} \right),\end{aligned}\quad (9.46)$$

where D^- and D^+ represent staggered spatial-derivative operators of order 4. The different signs imply a shift of half the grid size, to obtain the acceleration at the same points of the particle velocity (Carcione, 1999c). The averaging of the material properties is performed as indicated in Carcione (1999c).

The program solves the isotropic version of the reflection-transmission problem illustrated in Section 6.1. The mesh has 120×120 points and a grid spacing of 10 m. A snapshot of the particle velocity v_2 at 250 ms is shown in Figure 9.4.

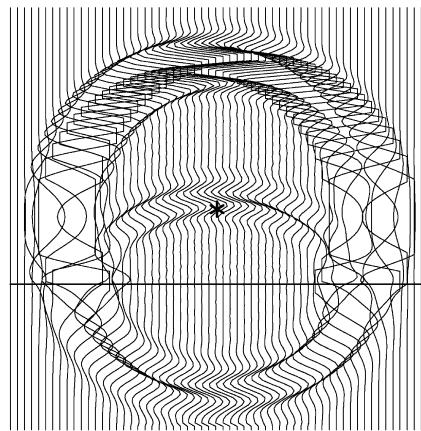


Figure 9.4: Snapshot of the SH-wave particle velocity, corresponding to the reflection-transmission problem studied in Section 6.1. The star indicates the location of the source.

```

c _____
c Isotropic, viscoelastic SH-wave propagation
c _____
Section 4.4 : Plane-wave analysis
Section 4.5.3 : Differential equations
Section 4.6 : Analytical solution
Section 6.1 : Reflection-transmission problem
Section 9.2.3 : Time integration
Section 9.3.1 : Spatial derivatives
c _____
c TM Maxwell's equations
c _____
Section 8.2.1 : Plane-wave analysis
Section 8.2 : Differential equations
Section 4.6 : Analytical solution
Section 8.4 : Reflection-transmission problem
Section 9.2.3 : Time integration
Section 9.3.1 : Spatial derivatives
c _____
c Acoustic-electromagnetic analogy
c _____
** Eq. (9.45) **
c v2 <-> H2
c s23=s32 <-> -E1
c s12 <-> E3
c mu <-> inverse of the permittivity
c rho <-> magnetic permeability
c eta <-> inverse of the conductivity
c _____

```

```

c Electromagnetic example (units: cm and ns)
c _____
c Velocity: 20 cm/ns (light velocity = 30 cm/ns)
c Dielectric permittivity (vacuum): 8.85 1.e-12
c Magnetic permeability (vacuum): 4 pi 1.e+23
c Conductivity: 0.001*1.e+21 (0.001 S/m)
c Frequency: 0.2 (200 MHz)
c dx: 10 (10 cm)
c dz: 10 (10 cm)
c _____
c O(4,4) finite-difference scheme
c _____
parameter (nxt=120, nzt=120, nstept=500)
c field variables
c v2: particle velocity
c s12 and s23: stress components
    dimension v2(nxt,nzt),s12(nxt,nzt),s32(nxt,nzt)
    dimension v2a(nxt,nzt),s12a(nxt,nzt),s32a(nxt,nzt)
    dimension v2t(nxt,nzt),s12t(nxt,nzt),s32t(nxt,nzt)
c material properties
c mu: shear modulus
c rho: density
c eta: Maxwell viscosity
    dimension mu(nxt,nzt),rho(nxt,nzt),eta(nxt,nzt)
    common/fd-weights/x1,x2,z1,z2
    real mu
c
    dimension ab(30)
    dimension f(nstept)
c _____
open(10,file='SNAP')
    dx=10.
    dz=10.
    dt=0.001
    nx=120
    nz=120
    nstep=250
    pi=3.14159265
c snapshots every nsp steps
    nsp=nstep
c source location and central frequency
    ix=60
    iz=60
    freq=25.
c MODEL _____
    do i=1,nx
        do j=1,nz
c upper layer _____
** Eqs. (6.22)-(6.24) **
    rho(i,j)=2000.
    mu(i,j)=9.68e+9
c quality factor at source central frequency
** Eqs. (8.42) and (8.57) **
    Q=5.
    eta(i,j)=Q*mu(i,j)/(pi*freq)
c lower layer _____

```

```

if(j.ge.80) then
rho(i,j)=2500.
mu(i,j)=19.6e+9
Q=10.
eta(i,j)=Q*mu(i,j)/(pi*freq)
endif
end do
end do

```

```

c _____
do i=1,nx
do j=1,nz
v2(i,j)=0.
s12(i,j)=0.
s32(i,j)=0.
end do
end do
do n=1,nstept
f(n)=0.
end do

```

```

c _____
c absorbing parameters
r=0.99
nab=12
do i=1,nab
ab(i)=r**i
ab(i)=1.
end do
c source's wavelet
** Eq. (2.233) **
call wavelet(f,freq,nw,dt)
c finite-differences weights
** Eq. (9.24) **
x1=9./(8.*dx)
x2=-1./(24.*dx)
z1=9./(8.*dz)
z2=-1./(24.*dz)

```

```

c TIME STEPPING _____
do 10 n=1,nstep
if(mod(n,10).eq.0) print *,n

```

```
c apply absorbing boundaries _____
```

```

c horizontal strips
do 11 j=1,nab
j2=j+2
j3=nz-j-1
sab=ab(nab+1-j)
do 11 i=3,nx-2
v2(i,j2)=v2(i,j2)*sab
11   v2(i,j3)=v2(i,j3)*sab

```

```

c vertical strips
do 12 i=1,nab
i2=i+2
i3=nx-i-1
sab=ab(nab+1-i)
do 12 j=3,nz-2
v2(i2,j)=v2(i2,j)*sab
12   v2(i3,j)=v2(i3,j)*sab

```

```

c
c Runge-Kutta method
** Eq. (9.17) **
do 13 i=1,nx
do 13 j=1,nz
v2t(i,j)=v2(i,j)
s12t(i,j)=s12(i,j)
s32t(i,j)=s32(i,j)
v2a(i,j)=v2(i,j)
s12a(i,j)=s12(i,j)
s32a(i,j)=s32(i,j)
13 continue
** Eq. (9.4) **
call H(v2a,s12a,s32a,mu,rho,eta,nx,nz)
c
c D1
do 14 i=1,nx
do 14 j=1,nz
v2t(i,j)=v2t(i,j)+dt*v2a(i,j)/6.
s12t(i,j)=s12t(i,j)+dt*s12a(i,j)/6.
s32t(i,j)=s32t(i,j)+dt*s32a(i,j)/6.
v2a(i,j)=v2(i,j)+0.5*dt*v2a(i,j)
s12a(i,j)=s12(i,j)+0.5*dt*s12a(i,j)
s32a(i,j)=s32(i,j)+0.5*dt*s32a(i,j)
14 continue
if(n.le.nw) then
v2t(ix,iz)=v2t(ix,iz)+dt*f(n)/6.
v2a(ix,iz)=v2a(ix,iz)+0.5*dt*f(n)
endif
c
call H(v2a,s12a,s32a,mu,rho,eta,nx,nz)
c
c D2
do 15 i=1,nx
do 15 j=1,nz
v2t(i,j)=v2t(i,j)+dt*v2a(i,j)/3.
s12t(i,j)=s12t(i,j)+dt*s12a(i,j)/3.
s32t(i,j)=s32t(i,j)+dt*s32a(i,j)/3.
v2a(i,j)=v2(i,j)+0.5*dt*v2a(i,j)
s12a(i,j)=s12(i,j)+0.5*dt*s12a(i,j)
s32a(i,j)=s32(i,j)+0.5*dt*s32a(i,j)
15 continue
if(n.le.nw) then
v2t(ix,iz)=v2t(ix,iz)+dt*f(n+1)/3.
v2a(ix,iz)=v2a(ix,iz)+0.5*dt*f(n+1)
endif
c
call H(v2a,s12a,s32a,mu,rho,eta,nx,nz)
c
c D3
do 16 i=1,nx
do 16 j=1,nz
v2t(i,j)=v2t(i,j)+dt*v2a(i,j)/3.
s12t(i,j)=s12t(i,j)+dt*s12a(i,j)/3.
s32t(i,j)=s32t(i,j)+dt*s32a(i,j)/3.
v2a(i,j)=v2(i,j)+dt*v2a(i,j)

```

```

s12a(i,j)=s12(i,j)+dt*s12a(i,j)
s32a(i,j)=s32(i,j)+dt*s32a(i,j)
16 continue
if(n.le.nw) then
v2t(ix,iz)=v2t(ix,iz)+dt*f(n+1)/3.
v2a(ix,iz)=v2a(ix,iz)+dt*f(n+1)
endif
c
call H(v2a,s12a,s32a,mu,rho,eta,nx,nz)
c
c D4
do 17 i=1,nx
do 17 j=1,nz
v2t(i,j)=v2t(i,j)+dt*v2a(i,j)/6.
s12t(i,j)=s12t(i,j)+dt*s12a(i,j)/6.
s32t(i,j)=s32t(i,j)+dt*s32a(i,j)/6.
17 continue
if(n.le.nw) then
v2t(ix,iz)=v2t(ix,iz)+dt*f(n+2)/6.
endif
c
do 18 i=1,nx
do 18 j=1,nz
v2(i,j)=v2t(i,j)
s12(i,j)=s12t(i,j)
s32(i,j)=s32t(i,j)
18 continue
c
c write snapshot
if(mod(n,nsp).eq.0) then
print *, 'write snapshot',n
write(10,*)nx,nz,dx,dz
do i=1,nx
write(10,*)(v2(i,j),j=1,nz)
end do
endif
10 continue
close(10)
stop
end
c
subroutine H(v2,s12,s32,mu,rho,eta,nx,nz)
dimension v2(nx,nz),s12(nx,nz),s32(nx,nz)
dimension mu(nx,nz),rho(nx,nz),eta(nx,nz)
dimension v2a(nx,nz)
common/fd-weights/x1,x2,z1,z2
real mu
c
do 1 i=1,nx
do 1 j=1,nz
v2a(i,j)=0.
1 continue
c
** Eq. (9.4) **
c
do 2 i=3,nx-2

```

```

do 2 j=3,nz-2
  v2a(i,j)=v2(i,j)
c momentum conservation
** Eqs. (9.24) and (9.46)1 **
c i-3/2 → i-1
c i-1/2 → i
c i+1/2 → i+1
c i+3/2 → i+2
  ds4=z1*(s32(i,j+1)-s32(i,j))+z2*(s32(i,j+2)-s32(i,j-1))
  ds6=x1*(s12(i+1,j)-s12(i,j))+x2*(s12(i+2,j)-s12(i-1,j))
c acceleration
** Eq. (9.46)1 **
  v2(i,j)=(ds4+ds6)/rho(i,j)
2      continue
c
  do 3 i=3,nx-2
    do 3 j=3,nz-2
c strains and stresses
** Eqs. (9.24), and (9.46)2 and (9.46)3 **
c i-3/2 → i-2
c i-1/2 → i-1
c i+1/2 → i
c i+3/2 → i+1
  e4=z1*(v2a(i,j)-v2a(i,j-1))+z2*(v2a(i,j+1)-v2a(i,j-2))
  e6=x1*(v2a(i,j)-v2a(i-1,j))+x2*(v2a(i+1,j)-v2a(i-2,j))
  s32(i,j)=mu(i,j)*(e4-s32(i,j))/eta(i,j)
  s12(i,j)=mu(i,j)*(e6-s12(i,j))/eta(i,j)
3      continue
c
  return
end
c
c WAVELET
  subroutine wavelet(f,fb,nw,dt)
  dimension f(nw)
** Eq. (2.233) **
  pi=3.14159265
  wb=2.*pi*fb
  t0=6./(5.*fb)
  Dw=0.5*wb
  nw=2.*t0/dt
  do n=1,nw
    t=(n-1)*dt
    D=t-t0
    f(n)=exp(-Dw*D*D*D/4.)*cos(wb*D)
  end do
  return
end

```

9.9.4 Pseudospectral Fourier Method

The Fourier PS method is a collocation technique in which a continuous function $u(x)$ is approximated by a truncated series

$$u_N(x) = \sum_{r=0}^{N-1} \tilde{u}_r \phi_r(x) \quad (9.47)$$

of known expansion functions ϕ_r , wherein the spectral (expansion) coefficients are chosen such that the approximate solution u_N coincides with the solution $u(x)$ at a discrete set x_0, x_1, \dots, x_{N-1} of sampling or collocation points,

$$u_N(x_j) = u(x_j), \quad j = 0, \dots, N - 1. \quad (9.48)$$

The collocation points are defined by equidistant sampling points

$$x_j = jdx, \quad (9.49)$$

where dx is the grid spacing. The expansion functions are defined by

$$\phi_r(x) = \exp(ikx), \quad (9.50)$$

with

$$k_r = \frac{2\pi r}{Ndx}, \quad r = 0, \dots, N - 1 \quad (9.51)$$

being the wavenumber. Thus,

$$\phi_r(x_j) = \exp(2\pi irj/N). \quad (9.52)$$

Since the functions ϕ are periodic, the Fourier PS method is appropriate for problems with periodic boundary conditions – for example, a wave which exits the grid on one side, and reenters it on the opposite side. The coefficients \tilde{u}_r are implicitly defined by

$$u(x_j) = \sum_{r=0}^{N-1} \tilde{u}_r \exp(2\pi irj/N) \quad j = 0, \dots, N - 1. \quad (9.53)$$

The sequence of $u(x_j)$ is the inverse discrete Fourier transform of the sequence of \tilde{u}_r . This set of equations is equivalent to

$$\tilde{u}_r = \frac{1}{N} \sum_{j=0}^{N-1} u(x_j) \exp(-2\pi irj/N) \quad r = 0, \dots, N - 1. \quad (9.54)$$

The computation of differential operators by the Fourier method conveniently reduces to a set of multiplications of the different coefficients \tilde{u}_r , with factors ik_r , since

$$\partial_1 \phi_r(x) = ik_r \phi_r(x), \quad (9.55)$$

so that

$$\partial_1 u_N(x) = \sum_{r=0}^{N-1} ik_r \tilde{u}_r \phi_r(x). \quad (9.56)$$

The spectral coefficients \tilde{u}_r are computed by the Fast Fourier Transform (FFT). Examples of efficient algorithms are the mixed-radix FFT (Temperton, 1983) and the prime factor

FFT (Temperton, 1988). The steps of the calculation of the first partial derivative are as follows:

$$u(x_j) \rightarrow \text{FFT} \rightarrow \tilde{u}_r \rightarrow ik_r \tilde{u}_r \rightarrow \text{FFT}^{-1} \rightarrow \partial_1 u(x_j). \quad (9.57)$$

The method is infinitely accurate up to the Nyquist wavenumber, which corresponds to a spatial wavelength of two grid points. This means that if the source is band-limited, the algorithm is free of numerical dispersion provided that the grid spacing is chosen $dx \leq c_{\min}/(2f_{\max})$ with f_{\max} being the cut-off frequency of the source and c_{\min} the minimum phase velocity in the mesh. The wavenumber can be expressed in the more convenient form

$$k_\nu = \begin{cases} \frac{2}{N} k_{\text{Nyq}} \nu & \text{for } \nu = 0, \dots, \frac{N}{2}, \\ -\frac{2}{N} k_{\text{Nyq}} (N - \nu) & \text{for } \nu = \frac{N}{2} + 1, \dots, N - 1, \end{cases} \quad (9.58)$$

where for N odd, $N/2$ represents truncation to the closest integer, and $k_{\text{Nyq}} = \pi/dx$ is the Nyquist wavenumber. For example, $N = 5$ has wavenumbers

$$\left(0, \frac{2}{5}, \frac{4}{5}, -\frac{4}{5}, -\frac{2}{5} \right) k_{\text{Nyq}}, \quad (9.59)$$

and $N = 6$ has wavenumbers

$$\left(0, \frac{2}{6}, \frac{4}{6}, 1, -\frac{4}{6}, -\frac{2}{6} \right) k_{\text{Nyq}}. \quad (9.60)$$

We see that when N is even, the wavenumber operator contains the Nyquist wavenumber; hence, k_ν is an odd function in the periodic sense only for N odd, since $k_\nu = -k_{N-\nu}$. When N is even, the Nyquist wavenumber breaks the antisymmetry.

We shall see now that when computing first-order derivatives, the number of grid points must be odd. Indeed, it is well known that when $u(x)$ is real, its continuous Fourier transform $\tilde{u}(k)$ is Hermitian, i.e. its real part is even and its imaginary part is odd (Bracewell, 1965, p. 16), and vice versa, if $\tilde{u}(k)$ is Hermitian, its inverse transform is real. Similar properties hold for discrete Fourier transform. Indeed, for N odd,

$$\tilde{u}(k) = \text{even} + i \text{odd}. \quad (9.61)$$

Then,

$$ik\tilde{u}(k) = i \text{odd} + \text{even} \quad (9.62)$$

is also Hermitian, and $\partial_1 u$ is real. Conversely, when N is even, $ik\tilde{u}(k)$ is not Hermitian because of the Nyquist wavenumber.

We now give some numerical tricks when using the FFT for computing partial derivatives.

1. It is possible to compute the derivatives of two real functions $\partial_1 f$ and $\partial_1 g$ by two complex FFT's in the following way: put f into the real part and g into the imaginary part and compute the direct FFT at k_r :

$$\sum_j [f_j^e + f_j^o + i(g_j^e + g_j^o)] (\cos \theta_{jr} - i \sin \theta_{jr}), \quad (9.63)$$

where summations go from 0 to $N - 1$. The functions have been split into even and odd parts (e and o , respectively), and θ_{jr} is an abbreviation of $k_r x_j$. Terms like $\sum f_j^o \cos \theta_{jr}$ vanish since summation of an odd function is zero – note that the cosine is even and the sine is odd. Then, equation (9.63) reduces to

$$\sum_j f_j^e \cos \theta_{jr} + g_j^o \sin \theta_{jr} + i(g_j^e \cos \theta_{jr} - f_j^o \sin \theta_{jr}). \quad (9.64)$$

Now, multiply by ik_r , and transform back to the space domain. At point x_i , this gives

$$\sum_r \sum_j ik_r [f_j^e \cos \theta_{jr} + g_j^o \sin \theta_{jr} + i (g_j^e \cos \theta_{jr} - f_j^o \sin \theta_{jr})] (\cos \theta_{ir} + i \sin \theta_{ir}). \quad (9.65)$$

Since many of the terms vanish, the result is

$$\sum_r \sum_j k_r (f_j^o \sin \theta_{jr} \cos \theta_{ir} + f_j^e \sin \theta_{ir} \cos \theta_{jr}) + ik_r (g_j^o \sin \theta_{jr} \cos \theta_{ir} + g_j^e \sin \theta_{ir} \cos \theta_{jr}). \quad (9.66)$$

By applying the same arguments to each single function, it can be easily shown that the real and imaginary parts of (9.66) are the derivatives of f and g at x_i respectively.

2. It is possible to compute two FFT's from one complex FFT, where, by real and imaginary FFT's, we mean

$$\tilde{f}_R = \sum_j f_j \cos \theta_{jr}, \quad (9.67)$$

and

$$\tilde{f}_I = \sum_j f_j \sin \theta_{jr}. \quad (9.68)$$

As before, we take a complex FFT of $F = f + ig$, which gives

$$\tilde{F} = \tilde{f}_R + i(\tilde{g}_R - \tilde{f}_I). \quad (9.69)$$

Since f and g are real functions, their transforms are Hermitian; hence,

$$\tilde{f}_R(k) = \tilde{f}_R(-k) \quad \tilde{g}_R(k) = \tilde{g}_R(-k) \quad (9.70)$$

$$\tilde{f}_I(k) = -\tilde{f}_I(-k) \quad \tilde{g}_I(k) = -\tilde{g}_I(-k) \quad (9.71)$$

Using these properties, we note that

$$\frac{1}{2} [\tilde{F}_R(-k) + \tilde{F}_R(k)] = \tilde{f}_R, \quad (9.72)$$

and

$$\frac{1}{2} [\tilde{F}_I(-k) + \tilde{F}_I(k)] = \tilde{g}_R, \quad (9.73)$$

i.e, the two desired real transforms.

9.9.5 Pseudospectral Chebyshev Method

When a function is not periodic, the Fourier method is not convenient for implementing free-surface and rigid boundary conditions. The reason is that the basis functions of the Fourier expansion are periodic. Satisfactory results are obtained with orthogonal polynomials, such as Chebyshev or Legendre polynomials. We consider the Chebyshev basis, because, as we shall see later, the derivative can be computed by using the FFT routine. The function $u(\zeta)$, $-1 \leq \zeta \leq 1$ is expanded into Chebyshev polynomials $T_n(\zeta)$ as

$$u(\zeta_j) = \sum_{n=0}^{N'} a_n T_n(\zeta_j), \quad (9.74)$$

where

$$T_n(\zeta_j) = \cos n\theta_j, \quad (9.75)$$

with

$$\zeta_j = \cos \theta_j, \quad \theta_j = \frac{\pi j}{N}, \quad j = 0, \dots, N, \quad (9.76)$$

denoting the Gauss-Lobatto collocation points. \sum' halves the first and last terms. The partial derivative of order q is given by

$$\frac{\partial^q u(\zeta)}{\partial \zeta^q} = \sum_{n=0}^N a_n^{(q)} T_n(\zeta), \quad (9.77)$$

(Gottlieb and Orszag, 1977, p. 117), where

$$c_{n-1} a_{n-1}^{(q)} - a_{n+1}^{(q)} = 2na_n^{(q-1)}, \quad n \geq 1, \quad (9.78)$$

with $c_0 = 2$, $c_n = 1$ ($n > 0$). Hence, defining $a_n = a_n^{(0)}$ and $b_n = a_n^{(1)}$, the first-order derivative is equal to

$$\frac{\partial u}{\partial \zeta} = \sum_{n=0}^N b_n T_n(\zeta), \quad (9.79)$$

where

$$b_{n-1} = b_{n+1} + 2na_n, \quad n = N, \dots, 2, \quad b_{N+1} = b_N = 0. \quad (9.80)$$

We consider the domain $[0, z_{\max}]$ and want to interpolate $u(z)$ in this domain. The transformation

$$z_j = \frac{z_{\max}}{2} (\zeta_j + 1) \quad (9.81)$$

maps the domain $[-1, 1]$ onto the physical domain $[0, z_{\max}]$. The Gauss-Lobatto points have maximum spacing at the center of the numerical grid, with

$$dz_{\max} = \frac{z_{\max}}{2} \left\{ \cos \left(\frac{N}{2} \frac{\pi}{N} \right) - \cos \left[\left(\frac{N}{2} + 1 \right) \frac{\pi}{N} \right] \right\} = \frac{z_{\max}}{2} \sin \left(\frac{\pi}{N} \right). \quad (9.82)$$

Note that $d\zeta_{\max} = \sin(\pi/N)$. In wave problems, we determine the maximum grid spacing according to the Nyquist criterion, $dz \leq c_{\min}/(2f_{\max})$. The spatial derivative is

$$\frac{\partial u}{\partial z} = \frac{\partial u}{\partial \zeta} \frac{\partial \zeta}{\partial z} = \frac{2}{z_{\max}} \frac{\partial u}{\partial \zeta} = \frac{1}{dz_{\max}} \sin \left(\frac{\pi}{N} \right) \frac{\partial u}{\partial \zeta}. \quad (9.83)$$

This is a transformation from the physical domain to the Chebyshev domain.

Now, let us see how to calculate $\partial u / \partial \zeta$. The expansion of $u(\zeta)$ and its coefficients can be written as

$$u(\zeta_j) = \sum_{n=0}^N a_n \cos \left(\frac{\pi nj}{N} \right) \quad (9.84)$$

and

$$a_n = \frac{2}{N} \sum_{j=0}^N u(\zeta_j) \cos \left(\frac{\pi nj}{N} \right). \quad (9.85)$$

The coefficients a_n can be evaluated by using a FFT routine. Let us define $N' = 2N$ and $u(\zeta_j) = 0$ for $j = 1 + N'/2, \dots, N' - 1$. Then

$$a_n = \frac{4}{N'} \sum_{j=0}^{N'-1} u(\zeta_j) \cos\left(\frac{2\pi nj}{N'}\right), \quad (9.86)$$

is a real Fourier transform that can be calculated by complex FFT's as described in the previous section. Afterwards, we get the b_n 's from the a_n 's by using the recursion equation (9.78) and again, the calculation of (9.79) is carried out with a real Fourier transform. However, the Chebyshev method, as presented so far, is impractical, because the grid spacing at the extremes of the domain is very fine. When the number of grid points is doubled, the grid spacing decreases by a factor of two. Hence, when solving the problem with an explicit time marching scheme, the conventional Chebyshev differential operator requires time steps of the order $O(N^{-2})$. A new algorithm developed by Kosloff and Tal-Ezer (1993), based on a coordinate transformation, allows time steps of order $O(N^{-1})$, which are those required also by the Fourier method. The new N sampling points are defined by

$$z_j = z_{\max} \left[\frac{g(\zeta_j) - g(-1)}{g(1) - g(-1)} \right], \quad j = 0, \dots, N, \quad (9.87)$$

where $g(\zeta)$ is a grid stretching function that stretches the very fine Chebyshev grid near the boundary in order to have a minimum grid size of the order $O(N^{-1})$, thus requiring a less severe stability condition. A suitable stretching function is

$$g(\zeta) = -\frac{1}{\sqrt{|p|}} \arcsin\left(\frac{2p\zeta + q}{\sqrt{q^2 - 4p}}\right), \quad (9.88)$$

where $p = 0.5\alpha^{-2}(\beta^{-2} + 1) - 1$, and $q = 0.5\alpha^{-2}(\beta^{-2} - 1)$. Since

$$\frac{dg}{d\zeta} = \frac{1}{\sqrt{1 + q\zeta + p\zeta^2}}, \quad (9.89)$$

it can be seen that the amount of grid stretching at $\zeta = -1$ is $dg/d\zeta = \alpha$, and that the stretching at $z = 1$ is $dg/d\zeta = \alpha\beta$. The spatial derivative is

$$\frac{\partial u}{\partial z} = \frac{\partial u}{\partial \zeta} \frac{\partial \zeta}{\partial z} = \left[\frac{g(1) - g(-1)}{z_{\max}} \right] \sqrt{1 + q\zeta + p\zeta^2} \frac{\partial u}{\partial \zeta}. \quad (9.90)$$

In many cases, we need to sample the function at the equidistant points

$$z_j^{(e)} = j\Delta z, \quad \Delta z = \frac{z_{\max}}{N}. \quad (9.91)$$

The corresponding points in the Chebyshev domain are, from (9.87),

$$\zeta'_j = g^{-1} \left\{ \left[\frac{g(1) - g(-1)}{z_{\max}} \right] z_j + g(-1) \right\}. \quad (9.92)$$

The values of the function at equidistant points in the physical space are given by

$$u(z_j^{(e)}) = u(\zeta'_j). \quad (9.93)$$

To obtain these values, we compute the spectral coefficients a_n of $u(\zeta_j)$, and then

$$u(z_j^{(e)}) = \sum_{n=0}^{N'} a_n T_n(\zeta'_j). \quad (9.94)$$