GEOPHYSICS, VOL. 72, NO. 5 (SEPTEMBER-OCTOBER 2007); P. SM115–SM122, 5 FIGS.
10.1190/1.2750424

High-order time discretizations in seismic modeling

Jing-Bo Chen[1]

ABSTRACT

Seismic modeling plays an important role in exploration geophysics. High-order modeling schemes are in demand for practical reasons. In this context, I present three kinds of high-order time discretizations: Lax-Wendroff methods, Nyström methods, and splitting methods. Lax-Wendroff methods are based on the Taylor expansion and the replacement of high-order temporal derivatives by spatial derivatives, Nyström methods are simplified Runge-Kutta algorithms, and splitting methods comprise substeps for one-step computation. Based on these methods, three schemes with third-order and fourth-order accuracy in time and pseudospectral discretizations in space are presented. I also compare their accuracy, stability, and computational complexity, and discuss advantages and shortcomings of these algorithms. Numerical experiments show that the fourth-order Lax-Wendroff scheme is more efficient for short-time simulations while the fourth-order Nyström scheme and the third-order splitting scheme are more efficient for long-term computations.

INTRODUCTION

Seismic modeling is an important foundation of seismology. Based on the scalar wave equation or elastic wave equation, seismic modeling can produce synthetic seismograms, which play a key role in seismic interpretation and exploration. Seismic modeling methods can be classified into three main categories: direct methods, integral-equation methods, and ray-tracing methods. Each method category has its advantages and shortcomings. A good review of these methods is given by Carcione et al. (2002).

High-accuracy seismic modeling schemes become more important as computing capacity increases. Numerical schemes for direct methods of seismic modeling involve discretizations of both space and time variables. Spatial discretizations have been explored extensively, and three approaches have been developed: finite-difference methods (Alford et al., 1974; Kelly et al., 1976), pseudospectral methods (Gazdag, 1981; Kosloff and Baysal, 1982), and finite-element methods (Marfurt, 1984). Some combinations of these methods are also available, such as spectral-element methods (Komatitsch and Vilotte, 1998) and finite-volume methods (Dormy and Tarantola, 1995).

On the other hand, temporal discretizations are relatively less studied. Most of the numerical schemes use second-order finite-difference time discretization. However, for typical time-step size, time discretization exhibits severe dispersion errors for long-time simulations (Chen, 2006). Current studies in seismology usually need large-scale and long-time seismic modeling; therefore, high-order time discretizations are in demand.

Three kinds of approaches for high-order time discretizations are available: Lax-Wendroff methods, Nyström methods, and splitting methods. Lax-Wendroff methods use spatial derivatives to replace high-order time derivatives (Dablain, 1986). Nyström methods are simplified Runge-Kutta algorithms that take advantage of the special form of the wave equation (Hairer et al., 1993; Chen, 2006). Based on vector-field decomposition, splitting methods consist of several substeps for one computational step (Qin and Zhang, 1990; Yoshida, 1990).

In the next section, using pseudospectral spatial discretizations, I present the three methods for time discretization. This is followed by an analysis of their accuracy, stability, and computational complexity. I then perform numerical experiments to demonstrate the presented algorithms.

HIGH-ORDER TIME DISCRETIZATIONS

Seismic modeling is based on the scalar wave equation or the elastic wave equation. Because I am dealing mainly with time discretizations, I will use the scalar wave equation in two spatial dimensions for the sake of simplicity. For the case of an elastic wave equation or for three spatial dimensions, similar results can be obtained in a similar way. For the spatial discretization, I use pseudospectral methods for which errors mainly arise from the time discretization. Pseudospectral methods for modeling the scalar wave equation are presented by Gazdag (1981). The main point of pseudospectral methods

Manuscript received by the Editor October 29, 2006; revised manuscript received April 16, 2007; published online August 23, 2007.
[1] Chinese Academy of Sciences, Institute of Geology and Geophysics, Beijing, China. E-mail: chenjb@vip.sohu.com.

is to express the wavefield under consideration in terms of a complete set of orthogonal basis functions whose derivatives are known exactly. In practice, Fourier pseudospectral methods are usually employed.

Consider the scalar wave equation,

$$\frac{\partial^2 u}{\partial t^2} = c^2\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial z^2}\right), \tag{1}$$

where $u(x,z,t)$ is the wavefield and $c(x,z)$ is the velocity. Let $\mathbf{u} = [u_{1,1},\ldots,u_{N_x,N_z}]^T$, where T represents the transpose and $u_{i,l}$ are the wavefield values at discrete locations, i.e., $u_{i,l} \approx u(i\Delta x, l\Delta z, t)$, $i = 1,\ldots,N_x$; $l = 1,\ldots,N_z$. Here, Δx and Δz are grid increments in the x- and z-directions, and N_x and N_z are the number of grid lines in the x- and z-directions, respectively. The semidiscrete equation resulting from the pseudospectral method for equation 1 is

$$\frac{d^2\mathbf{u}}{dt^2} = c^2\mathcal{F}^{-1}\left[\mathbf{w} * \mathcal{F}(\mathbf{u})\right], \tag{2}$$

where $\mathcal{F}$ and $\mathcal{F}^{-1}$ represent 2D forward and inverse finite Fourier transforms, respectively, and $\mathbf{w} = [w_{1,1},\ldots,w_{N_x,N_z}]^T$ with $w_{i,l} = -(k_{x_i}^2 + k_{z_l}^2)$, where k_{x_i} and k_{z_l} are discrete wavenumbers in the x- and z-directions, respectively. The star $*$ denotes array multiplication between vectors. For example, suppose that $\mathbf{p} = (p_1, p_2, \ldots, p_m)$ and $\mathbf{q} = (q_1, q_2, \ldots, q_m)$, then $\mathbf{p} * \mathbf{q} = (p_1q_1, p_2q_2, \ldots, p_mq_m)$.

Lax-Wendroff methods

Based on Taylor expansions, Lax-Wendroff methods use spatial derivatives to replace high-order time derivatives (Dablain, 1986; Carcione et al., 2002):

$$\frac{u^{n+1} - 2u^n + u^{n-1}}{(\Delta t)^2} = c^2\left(\frac{\partial^2 u^n}{\partial x^2} + \frac{\partial^2 u^n}{\partial z^2}\right) + 2\sum_{j=2}^{J}\frac{(\Delta t)^{2j-2}}{(2j)!}\frac{\partial^{2j}u^n}{\partial t^{2j}}, \tag{3}$$

where $u^n \approx u(x,z,n\Delta t)$ and Δt is the time-step size. The second-order time derivative is provided by equation 1, and the higher-order time derivatives are obtained by the following recursive formula:

$$\frac{\partial^{2j}u^n}{\partial t^{2j}} = c^2\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial z^2}\right)\frac{\partial^{2j-2}u^n}{\partial t^{2j-2}}. \tag{4}$$

Scheme 3 has an accuracy of $\mathcal{O}((\Delta t)^{2J})$. Taking $J = 2$, I obtain a fourth-order scheme:

$$\frac{u^{n+1} - 2u^n + u^{n-1}}{(\Delta t)^2} = c^2\left(\frac{\partial^2 u^n}{\partial x^2} + \frac{\partial^2 u^n}{\partial z^2}\right) + \frac{c^2(\Delta t)^2}{12} \times \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial z^2}\right)\left[c^2\left(\frac{\partial^2 u^n}{\partial x^2} + \frac{\partial^2 u^n}{\partial z^2}\right)\right]. \tag{5}$$

Using pseudospectral spatial discretization, scheme 5 becomes

$$\frac{\mathbf{u}^{n+1} - 2\mathbf{u}^n + \mathbf{u}^{n-1}}{(\Delta t)^2} = c^2\mathcal{F}^{-1}\left[\mathbf{w} * \mathcal{F}(\mathbf{u}^n)\right] + \frac{c^2(\Delta t)^2}{12} \times \mathcal{F}^{-1}\left\{\mathbf{w} * \mathcal{F}\left[c^2\mathcal{F}^{-1}\left(\mathbf{w} * \mathcal{F}(\mathbf{u}^n)\right)\right]\right\}, \tag{6}$$

where the additional starting value $\mathbf{u}^n$ can be obtained through the following Taylor expansion:

$$u(n\Delta t) = \sum_{i=0}^{4}\frac{\partial^i u((n-1)\Delta t)}{\partial t^i}\frac{(\Delta t)^i}{i!}. \tag{7}$$

Here, the high-order time derivatives are obtained in the same way as in scheme 3.

Nyström methods

The basic idea of Nyström methods is as follows: First, introduce a new variable to reduce the second-order wave equation to an equivalent first-order system; then apply Runge-Kutta methods to the first-order system and simplify by taking advantage of the special form of the first-order system (Nyström, 1925; Hairer et al., 1993).

A Nyström method for equation 2 reads

$$\mathbf{V}_i = c^2\mathcal{F}^{-1}\left[\mathbf{w} * \mathcal{F}\left(\mathbf{u}^n + d_i\Delta t\mathbf{v}^n + (\Delta t)^2\sum_{j=1}^{s}a_{ij}\mathbf{V}_j\right)\right], \quad i = 1,2,\ldots,s,$$

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \Delta t\mathbf{v}^n + (\Delta t)^2\sum_{i=1}^{s}\bar{b}_i\mathbf{V}_i,$$

$$\mathbf{v}^{n+1} = \mathbf{v}^n + \Delta t\sum_{i=1}^{s}b_i\mathbf{V}_i, \tag{8}$$

where $\mathbf{v} = d\mathbf{u}/dt$, $\mathbf{u}^n \approx \mathbf{u}(n\Delta t)$, $\mathbf{v}^n \approx \mathbf{v}(n\Delta t)$, $\mathbf{u}^{n+1} \approx \mathbf{u}((n+1)\Delta t)$, $\mathbf{v}^{n+1} \approx \mathbf{v}((n+1)\Delta t)$, and d_i, a_{ij}, $\bar{b}_i$, and b_i are constants. Here, s refers to the number of the auxiliary variables $\mathbf{V}_i$ in equation 8.

Now I recall the definition about scheme order (Hairer et al., 1993). Denote $\mathbf{z} = [\mathbf{u},\mathbf{v}]^T$. Let $\mathbf{z}((n+1)\Delta t) = [\mathbf{u}((n+1)\Delta t), \mathbf{v}((n+1)\Delta t)]^T$ be the true solution to equation 2 and $\mathbf{z}^{n+1} = [\mathbf{u}^{n+1}, \mathbf{v}^{n+1}]^T$ be the numerical solution to equation 2 obtained with scheme 8. If the following inequality holds,

$$\|\mathbf{z}((n+1)\Delta t) - \mathbf{z}^{n+1}\| \le K(\Delta t)^{p+1}, \tag{9}$$

then scheme 8 has order p. Here, $\|.\|$ denotes a norm and K is a constant.

To obtain a scheme of order p, I calculate the Taylor series for both $\mathbf{z}((n+1)\Delta t)$ and $\mathbf{z}^{n+1}$ and make the two Taylor series coincide up to the term $(\Delta t)^p$. This procedure results in algebraic equations satisfied by d_i, a_{ij}, $\bar{b}_i$, and b_i, which are called order conditions. The number of algebraic equations (and therefore the coefficients) increases with the order. In deriving the Taylor series, I need to calculate the high-order derivatives of the composite functions. Usually, it is very complicated and troublesome to calculate the high-order derivatives directly. By establishing a relationship between derivatives and their graphical representations (tree), the order conditions can be derived easily. By solving these algebraic equations, I can obtain the corresponding methods. For details, see Hairer et al. (1993).

I also consider a fourth-order explicit Nyström method (Qin and Zhu, 1991):

$$\begin{aligned}
\mathbf{V}_1 &= c^2\mathcal{F}^{-1}[\mathbf{w} * \mathcal{F}(\mathbf{u}^n + d_1\Delta t\mathbf{v}^n)],\\
\mathbf{V}_2 &= c^2\mathcal{F}^{-1}[\mathbf{w} * \mathcal{F}(\mathbf{u}^n + d_2\Delta t\mathbf{v}^n\\
&\quad + a_{21}(\Delta t)^2\mathbf{V}_1)],\\
\mathbf{V}_3 &= c^2\mathcal{F}^{-1}[\mathbf{w} * \mathcal{F}(\mathbf{u}^n + d_3\Delta t\mathbf{v}^n + a_{31}(\Delta t)^2\mathbf{V}_1\\
&\quad + a_{32}(\Delta t)^2\mathbf{V}_2)],\\
\mathbf{u}^{n+1} &= \mathbf{u}^n + \Delta t\mathbf{v}^n + (\Delta t)^2(\bar{b}_1\mathbf{V}_1 + \bar{b}_2\mathbf{V}_2\\
&\quad + \bar{b}_3\mathbf{V}_3),\\
\mathbf{v}^{n+1} &= \mathbf{v}^n + \Delta t(b_1\mathbf{V}_1 + b_2\mathbf{V}_2 + b_3\mathbf{V}_3),
\end{aligned} \tag{10}$$

where

$$\begin{gathered}
d_1 = \frac{3+\sqrt{3}}{6}, \quad d_2 = \frac{3-\sqrt{3}}{6}, \quad d_3 = \frac{3+\sqrt{3}}{6},\\
\bar{b}_1 = \frac{5-3\sqrt{3}}{24},\\
\bar{b}_2 = \frac{3+\sqrt{3}}{12}, \quad \bar{b}_3 = \frac{1+\sqrt{3}}{24}, \quad b_1 = \frac{3-2\sqrt{3}}{12},\\
b_2 = \frac{1}{2}, \quad b_3 = \frac{3+2\sqrt{3}}{12},\\
a_{21} = \frac{2-\sqrt{3}}{12}, \quad a_{31} = 0, \quad a_{32} = \frac{\sqrt{3}}{6}.
\end{gathered} \tag{11}$$

Scheme 10 is an explicit algorithm; therefore, I have $a_{11} = a_{12} = a_{13} = a_{22} = a_{23} = a_{33} = 0$.

Splitting methods

Based on vector-field decomposition, splitting methods consist of several substeps for one-step computation (Yoshida, 1990; Qin and Zhang, 1990). A splitting method for equation 2 reads as follows:

$$\begin{aligned}
\mathbf{v}^{s(i)} &= \mathbf{v}^{s(i-1)} + p_i\Delta tc^2\mathcal{F}^{-1}[\mathbf{w} * \mathcal{F}(\mathbf{u}^{s(i-1)})],\\
\mathbf{u}^{s(i)} &= \mathbf{u}^{s(i-1)} + q_i\Delta t\mathbf{v}^{s(i)}, \quad i = 1,2,\ldots,I,
\end{aligned} \tag{12}$$

where the superscripts $s(i)$ denote the intermediate results and $\mathbf{u}^{s(0)} = \mathbf{u}^n \approx \mathbf{u}(n\Delta t)$, $\mathbf{v}^{s(0)} = \mathbf{v}^n \approx \mathbf{v}(n\Delta t)$, $\mathbf{u}^{s(I)} = \mathbf{u}^{n+1} \approx \mathbf{u}((n+1)\Delta t)$, and $\mathbf{v}^{s(I)} = \mathbf{v}^{n+1} \approx \mathbf{v}((n+1)\Delta t)$. This means I need to compute I substeps to obtain the results for time level $n+1$ from that of time level n. The coefficients p_i and q_i determine the accuracy of the splitting method. I take $I = 3$ and consider a third-order method (Ruth, 1983):

$$\begin{aligned}
\mathbf{v}^{s(1)} &= \mathbf{v}^n + p_1\Delta tc^2\mathcal{F}^{-1}[\mathbf{w} * \mathcal{F}(\mathbf{u}^n)],\\
\mathbf{u}^{s(1)} &= \mathbf{u}^n + q_1\Delta t\mathbf{v}^{s(1)},\\
\mathbf{v}^{s(2)} &= \mathbf{v}^{s(1)} + p_2\Delta tc^2\mathcal{F}^{-1}[\mathbf{w} * \mathcal{F}(\mathbf{u}^{s(1)})],\\
\mathbf{u}^{s(2)} &= \mathbf{u}^{s(1)} + q_2\Delta t\mathbf{v}^{s(2)},\\
\mathbf{v}^{n+1} &= \mathbf{v}^{s(2)} + p_3\Delta tc^2\mathcal{F}^{-1}[\mathbf{w} * \mathcal{F}(\mathbf{u}^{s(2)})],\\
\mathbf{u}^{n+1} &= \mathbf{u}^{s(2)} + q_3\Delta t\mathbf{v}^{n+1},
\end{aligned} \tag{13}$$

where $p_1 = 7/24$, $p_2 = 3/4$, $p_3 = -(1/24)$, $q_1 = -(2/3)$, $q_2 = -2/3$, $q_3 = 1$.

ACCURACY, STABILITY, AND COMPUTATIONAL COMPLEXITY

In this section, I discuss the accuracy, stability, and computational complexity of schemes 6, 10, and 13.

Accuracy and stability

Both schemes 6 and 10 have a fourth-order accuracy in time: $\mathcal{O}((\Delta t)^4)$. Scheme 13 has a third-order accuracy in time: $\mathcal{O}((\Delta t)^3)$. Because I use pseudospectral spatial discretization, the spatial accuracy is of exponential order $\mathcal{O}(\exp(\Delta x))$ (Fornberg, 1996). Therefore, the total accuracy of schemes 6 and 10 is $\mathcal{O}((\Delta t)^4 + \exp(\Delta x))$; for scheme 13, it is $\mathcal{O}((\Delta t)^3 + \exp(\Delta x))$.

I take $\Delta x = \Delta z = h$ and denote that the Courant number as $r = c\Delta t/h$. Using the standard spectral analysis technique (Kosloff and Baysal, 1982), I can obtain the following stability limits for the three schemes:

$$\begin{aligned}
r &< 0.7797 \text{ for scheme } 6,\\
r &< 0.5826 \text{ for scheme } 10,\\
r &< 0.5645 \text{ for scheme } 13.
\end{aligned} \tag{14}$$

Computational complexity

The main computational cost in schemes 6, 10, and 13 comes from the evaluation of pairs of fast Fourier transforms (FFTs) (forward and inverse). They involve 2, 3, and 3 pairs of FFTs, respectively. Therefore, the computational complexities of schemes 6, 10, and 13 are $4N_xN_z\log_2(N_xN_z)$, $6N_xN_z\log_2(N_xN_z)$, and $6N_xN_z\log_2(N_xN_z)$, respectively.

Scheme 6 has less computational complexity, but it needs evaluation of the additional starting values through the Taylor expansion 7. Another shortcoming of scheme 6 is that it involves spatial derivatives of the velocity explicitly. Therefore, it is necessary to discretize the spatial derivative of the velocity, and this discretization may result in some problems for discontinuous media (Kelly et al., 1976). On the other hand, schemes 10 and 13 do not need the evaluation of the additional starting values and involve no spatial derivatives of the velocity explicitly. Another advantage of schemes 10 and 13 is that they are symplectic algorithms, which have good performance for long-time simulations (Chen, 2006). See Appendix A for an introduction to symplectic algorithms.

NUMERICAL EXPERIMENTS

In this section, I perform numerical experiments to test the presented schemes. Like Gazdag (1981), I use the initial conditions:

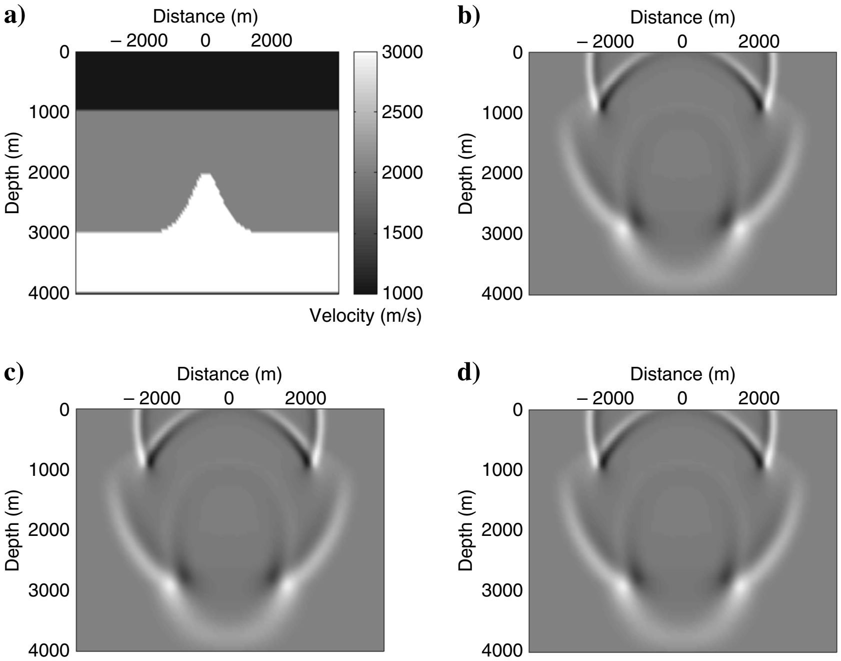


Figure 1. (a) The velocity model, and wavefields at t = 1.86 s computed with schemes 6(b), 10(c), and 13(d).

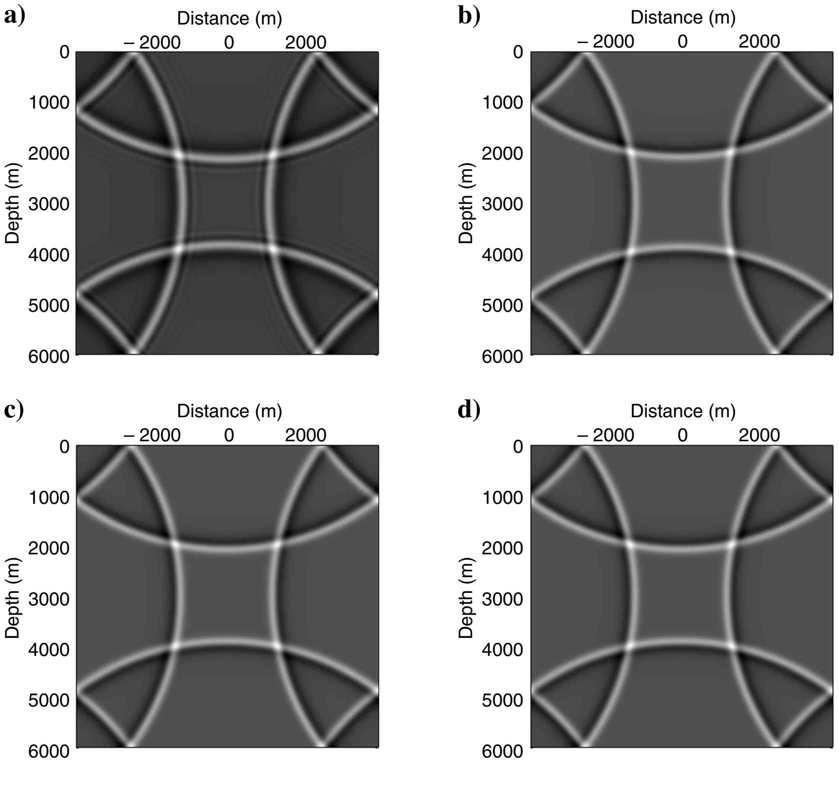


Figure 2. Impulse responses at time t = 1.8 s computed with schemes 16(a), 6(b), 10(c), and 13(d).

$$u(x,z,0) = \exp[-a^2(x^2+(z-z_0)^2)]$$
$$\text{and } \frac{\partial u(x,z,0)}{\partial t} = 0, \qquad (15)$$

where z_0 is a constant that indicates the position of the source and a is a constant determining the width of the pulse. As demonstrated in Gazdag (1981), I set $a\Delta x = 0.5$ to guarantee the reflected signal quality.

In the experiments, I take $\Delta x = \Delta z = 50$ m, $a = 0.01$ and use periodic boundary conditions. The first example is an inhomogeneous medium that consists of three regions with velocities of 1000, 2000, and 3000 m/s, respectively. There are two interfaces: one is straight; the other is curved (Figure 1a). The source is set at ($x = 0$ m, $z = 200$ m). I take $\Delta t = 0.006$. Figure 1b–d shows the wavefields at $t = 1.86$ s computed with schemes 6, 10, and 13, respectively. The wavefield consists of the incident wave, two reflected waves, and two transmitted waves. The three results have almost the same performance, and they give a qualitatively correct simulation of reflection and transmission phenomena.

The second example compares schemes 6, 10, and 13. In this case, I choose a homogeneous medium with a velocity of 3000 m/s for simplicity and clarity. I also consider a commonly used second-order scheme (Gazdag, 1981):

$$\frac{\mathbf{u}^{n+1} - 2\mathbf{u}^n + \mathbf{u}^{n-1}}{(\Delta t)^2} = c^2\mathcal{F}^{-1}\left[\mathbf{w} * \mathcal{F}(\mathbf{u}^n)\right]. \qquad (16)$$

Figure 2 shows the impulse responses at $t = 1.8$ s computed by schemes 6, 10, 13, and 16. The source is set at ($x = 0$ m, $z = 3200$ m), and $\Delta t = 0.006$. The result computed with scheme 16 starts to exhibit dispersion errors, although the other three results suffer no such errors. This is because scheme 16 has low accuracy in time. The CPU time for schemes 6, 10, and 13 is 3.93, 6.77, and 5.24 s, respectively, approximately agreeing with the computational complexity of the three schemes. The evaluation of the variable $\mathbf{v}$ in scheme 10 is more complex than that in scheme 13, although schemes 10 and 13 have the same essential computational complexity. Figure 3 shows the impulse responses at $t = 18$ s. For this computational time, the wavefront computed with scheme 16 has blurred badly, although the wavefronts computed with schemes 6, 10, and 13 are still sharp.

To reduce the dispersion errors computed with scheme 16, I can use a smaller time-step size. However, the overall computational cost (therefore, CPU time) of scheme 16 with a smaller time-step size will be much higher than scheme 10 with a larger time-step size for long-time simulations. This issue is discussed in detail in Chen (2006).

Now I perform a further comparison with the results shown in Figure 2. For this purpose, I plot the amplitude curves at a fixed point ($x = -1600$ m, $z = 1600$ m) in Figure 4a. Note a discrepancy between the amplitude curve computed with scheme 16 and the amplitude curves computed with the other schemes. However, the amplitude curves computed with schemes 6, 10, and 13 are indistinguishable in this plot. From the enlarged portion of Figure 4a, the amplitude curves computed with scheme 6 (blue curve) can be distinguished from the amplitude curves computed with scheme 10 and 13 (Figure 4b).

To make careful comparisons of the amplitude curves computed with the three schemes 6, 10, and 13, I plot their amplitude curves for a longer simulation (1000 time steps) (see Figure 4c) and its enlarged portion Figure 4d. From Figure 4d, I see that the three amplitude curves can be distinguished from one another. However, something unusual happens. The amplitude curves computed by the third-order scheme 13 (green curve) and the fourth-order scheme 10 (red curve) are closer to each other. According to the accuracy, the amplitude curves computed with fourth-order scheme 6 (blue curve) and fourth-order scheme 10 (red curve) should have been closer to each other.

To explain this phenomenon, I add an amplitude curve that is computed with scheme 6 with a smaller time-step size $\Delta t = 0.003$ s (black curve); see Figure 4e and its enlarged portion Figure 4f. In Figure 4f, the green curve is closer to the black curve than the blue curve is. This indicates that for this simulation time (6 s), the third-order scheme 13 is more accurate than the fourth-order scheme 6. I can explain this as follows: Although the fourth-order scheme 6 is more accurate than the third-order scheme 13 for one time step, the error growth of the former is faster than the later as time steps increase.

A possible explanation for the slower error growth of scheme 13 is that scheme 13 is a symplectic algorithm and symplectic algorithms have slower error growth than nonsymplectic ones (see Appendix A). Scheme 10 is also a symplectic algorithm (Qin and Zhu, 1991; Chen, 2006). The fourth-order scheme 10 is closer to the black curve than the third-order scheme 13 because it has greater accuracy. For this simulation time (6 s) and similar accuracy, scheme 10 with a time-step size of $\Delta t = 0.006$ s is more efficient than scheme 6 with a smaller time-step size of $\Delta t = 0.003$ s because their total computational complexities are 3000 and 4000 pairs of FFTs, respectively.

In the final numerical example, I test the two fourth-order schemes 6 and 10 on the Marmousi model (Figure 5). In this example, the initial conditions are taken as

$$u(x,z,0) = 0, \quad \text{and} \quad \frac{\partial u(x,z,0)}{\partial t} = 0. \tag{17}$$

I employ a commonly used Ricker wavelet with a peak frequency of 25 Hz (Reshef et al., 1988). The Ricker wavelet is placed at $x = 6000$ m and $z = 1200$ m. I set $\Delta x = 12.5$ m, $\Delta z = 4$ m, and $\Delta t = 0.0002$ s. Figure 5 illustrates wavefields at $t = 0.36$ s computed with these two schemes. The two wavefields are almost the same. However, making the same comparison as in the homogeneous case, one can see that scheme 10 is more accurate than scheme 6 for the same time-step sizes. To achieve the same accuracy with these two

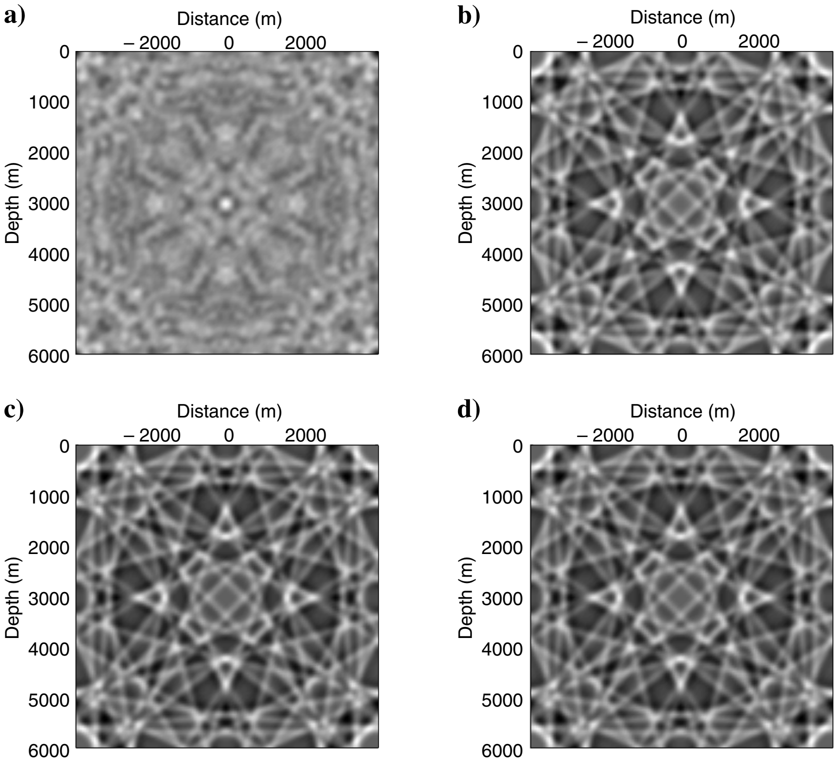
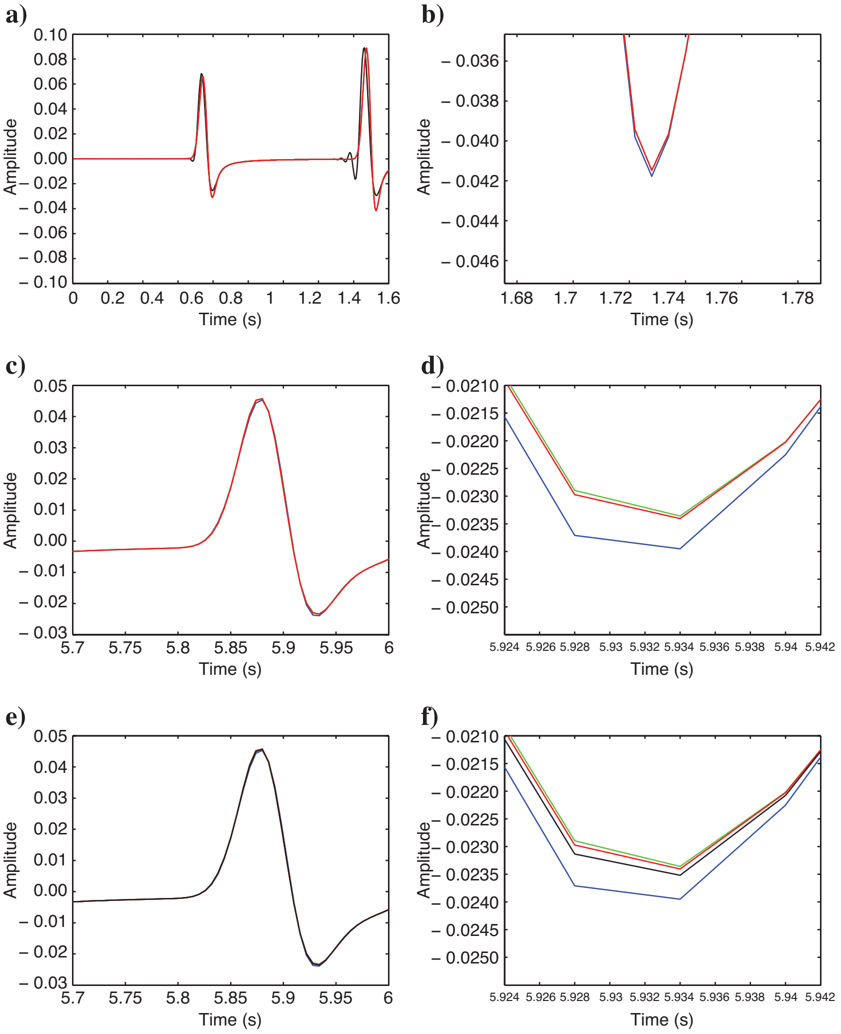


Figure 3. Impulse responses at $t = 18$ s computed with schemes 16(a), 6(b), 10(c), and 13(d).

Figure 4. Amplitude curves at a fixed point ($x = -1600$ m, $z = 1600$ m). (a) Amplitude curves computed with schemes 16 (black curve), 6 (blue curve), 10 (red curve), and 13 (green curve). (b) Enlarged portion of (a). (c) Amplitude curves computed with schemes 6 (blue curve), 10 (red curve), and 13 (green curve) for a longer simulation time (6 s). (d) Enlarged portion of (c). (e) Additional amplitude curve computed with scheme 6 with a smaller time-step size $\Delta t = 0.003$ s added (black curve). (f) Enlarged portion of (e).



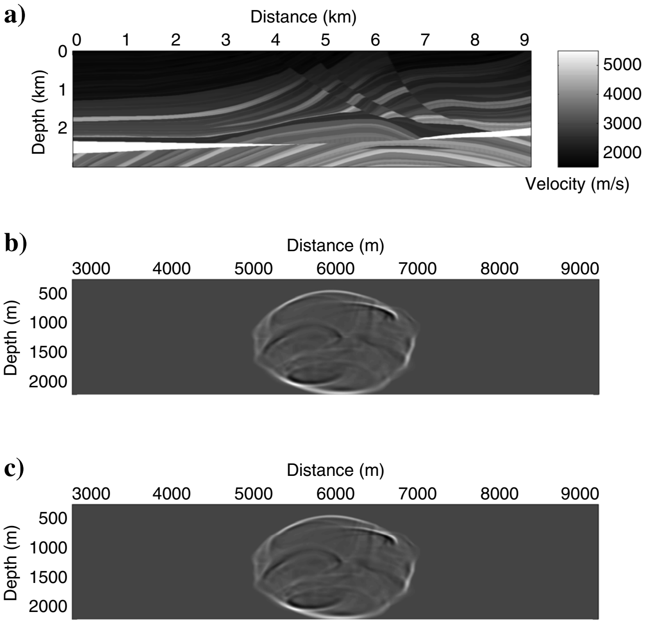


Figure 5. (a) Marmousi model, and wavefields at $t = 0.36$ s computed with schemes 6(b) and 10(c).

schemes, one must use a smaller time-step size for scheme 6. For brevity, I have not shown the corresponding plots, which are similar to those in Figure 4.

CONCLUSIONS

I have presented three modeling schemes with high-order time discretizations and spatial pseudospectral discretizations. The Lax-Wendroff scheme has better stability but needs evaluation of additional starting values. It also involves the spatial derivatives of velocity explicitly. The Nyström scheme and the splitting scheme need no additional starting values and involve no spatial derivatives of velocity explicitly. For short-time simulations, the fourth-order Lax-Wendroff scheme is more efficient. For long-time simulations, the fourth-order Nyström scheme and the third-order splitting scheme are more efficient.

ACKNOWLEDGMENTS

I would like to thank the anonymous reviewers for valuable suggestions and corrections. This work is supported by the National Natural Science Foundation of China under Grant 40474047.

APPENDIX A

SYMPLETIC ALGORITHMS FOR HAMILTONIAN SYSTEMS

In this appendix, I present the symplectic algorithms for Hamiltonian systems. For details, see Sanz-Serna and Calvo (1994) and Hairer et al. (2002).

A Hamiltonian system reads

$$\frac{d\mathbf{p}}{dt} = -\frac{\partial H(\mathbf{p},\mathbf{q})}{\partial \mathbf{q}},$$

$$\frac{d\mathbf{q}}{dt} = \frac{\partial H(\mathbf{p},\mathbf{q})}{\partial \mathbf{p}}, \qquad \text{(A-1)}$$

where $\mathbf{p} = (p_1, p_2, \ldots, p_n)$ is the generalized coordinate, $\mathbf{q} = (q_1, q_2, \ldots, q_n)$ is the generalized momentum, and $H(\mathbf{p},\mathbf{q})$ is the Hamiltonian function.

Suppose that the solution to the Hamiltonian system A-1 is $Z = F(Z_0)$, where $Z = [\mathbf{p}(t), \mathbf{q}(t)]^T$ and $Z_0 = [\mathbf{p}(0), \mathbf{q}(0)]^T$. Here, T represents the transpose of matrices. The solution $Z = F(Z_0)$ satisfies

$$\left[\frac{\partial F}{\partial Z_0}\right]\mathbf{J}\left[\frac{\partial F}{\partial Z_0}\right]^T = \mathbf{J}, \quad \text{where} \quad \mathbf{J} = \begin{bmatrix} 0 & \mathbf{I} \\ -\mathbf{I} & 0 \end{bmatrix}. \qquad \text{(A-2)}$$

Here, $[\partial F/\partial Z_0]$ is the Jacobian of the vector-valued function $F(Z_0)$ and $\mathbf{I}$ is the $n \times n$ identity matrix. A function satisfying the above equality is called a *symplectic mapping*. Therefore, the true solution of a Hamiltonian system is a symplectic mapping.

A numerical method for Hamiltonian systems is called a symplectic algorithm if the resulting numerical solution is also a symplectic mapping. The numerical solution obtained by a symplectic algorithm exactly satisfies a perturbed Hamiltonian system. This property guarantees that symplectic algorithms have slower error growth and possesses remarkable capability in preserving conservative quantities. Therefore, symplectic algorithms play an important role in high-accuracy or long-time numerical simulations. In the following, I present three kinds of symplectic algorithms.

Symplectic Runge-Kutta methods

I apply a Runge-Kutta method to the Hamiltonian system A-1 and obtain

$$\mathbf{P}_i = \mathbf{p}^n - \Delta t \sum_{j=1}^{s} a_{ij} \frac{\partial}{\partial \mathbf{q}} H(\mathbf{P}_j, \mathbf{Q}_j),$$

$$\mathbf{Q}_i = \mathbf{q}^n + \Delta t \sum_{j=1}^{s} a_{ij} \frac{\partial}{\partial \mathbf{p}} H(\mathbf{P}_j, \mathbf{Q}_j),$$

$$\mathbf{p}^{n+1} = \mathbf{p}^n - \Delta t \sum_{i=1}^{s} b_i \frac{\partial}{\partial \mathbf{q}} H(\mathbf{P}_i, \mathbf{Q}_i),$$

$$\mathbf{q}^{n+1} = \mathbf{q}^n + \Delta t \sum_{i=1}^{s} b_i \frac{\partial}{\partial \mathbf{p}} H(\mathbf{P}_i, \mathbf{Q}_i). \qquad \text{(A-3)}$$

If the coefficients in scheme A-3 satisfy

$$b_i a_{ij} + b_j a_{ji} - b_i b_j = 0, \quad i,j = 1,\ldots,s, \qquad \text{(A-4)}$$

then scheme A-3 is a symplectic algorithm.

Symplectic Nyström methods

Scheme A-3 is implicit. If the Hamiltonian function in system A-1 has the special form

$$H(\mathbf{p},\mathbf{q}) = \frac{1}{2}\mathbf{p}\mathbf{p}^T + V(\mathbf{q}), \qquad \text{(A-5)}$$

then I can obtain explicit symplectic Nyström methods. A Nyström method for system A-1 reads

$$\mathbf{Q}_i = -\frac{\partial}{\partial \mathbf{q}} V\left(\mathbf{q}^n + d_i \Delta t \mathbf{p}^n + (\Delta t)^2 \sum_{j=1}^{s} a_{ij} \mathbf{Q}_j\right),$$

$$i = 1,2,\ldots,s,$$

$$\mathbf{q}^{n+1} = \mathbf{q}^n + \Delta t \mathbf{p}^n + (\Delta t)^2 \sum_{i=1}^{s} \bar{b}_i \mathbf{Q}_i,$$

$$\mathbf{p}^{n+1} = \mathbf{p}^n + \Delta t \sum_{i=1}^{s} b_i \mathbf{Q}_i. \qquad \text{(A-6)}$$

If the coefficients in scheme A-6 satisfy

$$\bar{b}_i = b_i(1 - d_i), \quad i = 1,\ldots,s,$$

$$b_i(\bar{b}_j - a_{ij}) = b_j(\bar{b}_i - a_{ji}), \quad i,j = 1,\ldots,s, \qquad \text{(A-7)}$$

then scheme A-6 is a symplectic algorithm. Based on the above criterion, one can show that scheme 10 is a symplectic algorithm.

Symplectic splitting methods

If the Hamiltonian function in system A-1 has the separable form

$$H(\mathbf{p},\mathbf{q}) = F(\mathbf{p}) + G(\mathbf{q}), \qquad \text{(A-8)}$$

one can obtain symplectic splitting methods. A symplectic splitting method for system A-1 reads

$$\mathbf{p}^{s(i)} = \mathbf{p}^{s(i-1)} - e_i \Delta t \frac{\partial}{\partial \mathbf{q}} G(\mathbf{q}^{s(i-1)}),$$

$$\mathbf{q}^{s(i)} = \mathbf{q}^{s(i-1)} + d_i \Delta t \frac{\partial}{\partial \mathbf{p}} F(\mathbf{p}^{s(i)}), \qquad \text{(A-9)}$$

$$i = 1,2,\ldots,I,$$

where the superscripts $s(i)$ denote the intermediate results and where $\mathbf{p}^{s(0)} = \mathbf{p}^n$, $\mathbf{q}^{s(0)} = \mathbf{q}^n$, $\mathbf{p}^{s(I)} = \mathbf{p}^{n+1}$, and $\mathbf{q}^{s(I)} = \mathbf{q}^{n+1}$. I list some symplectic splitting methods as follows:

$$\text{Second order:} \quad s = 2 \; e_1 = e_2 = \frac{1}{2}, \quad d_1 = 1, \quad d_2 = 0;$$

$$\text{Third order:} \quad s = 3 \; e_1 = \frac{7}{24}, \quad e_2 = \frac{3}{4}, \quad e_3 = -\frac{1}{24},$$

$$d_1 = \frac{2}{3}, \quad d_2 = -\frac{2}{3}, \quad d_3 = 1;$$

$$\text{Fourth order:}\quad s = 4\; e_1 = e_4 = \frac{1}{2(2 - 2^{1/3})},$$

$$e_2 = e_3 = \frac{1 - 2^{1/3}}{2(2 - 2^{1/3})},$$

$$d_1 = d_3 = \frac{1}{2 - 2^{1/3}}, \quad d_2 = -\frac{2^{1/3}}{2 - 2^{1/3}}, \quad d_4 = 0.$$

Introducing a new variable $\mathbf{v} = d\mathbf{u}/dt$, I can cast the second-order equation 2 into a Hamiltonian system (Chen, 2006). Therefore, I can develop the corresponding symplectic algorithms.

REFERENCES

Alford, R. M., K. R. Kelly, and D. M. Boore, 1974, Accuracy of finite-difference modeling of the acoustic wave equation: Geophysics, **39**, 834–842.

Carcione, J. M., G. C. Herman, and A. P. E. ten Kroode, 2002, Seismic modeling: Geophysics, **67**, 1304–1325.

Chen, J. B., 2006, Modeling the scalar wave equation with Nyström methods: Geophysics, **71**, no. 5, T151–T158.

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

Dormy, E., and A. Tarantola, 1995, Numerical simulation of elastic wave propagation using a finite volume method: Journal of Geophysical Research, **100**, 2123–2133.

Fornberg, B., 1996, A practical guide to pseudospectral methods: Cambridge University Press.

Gazdag, J., 1981, Modeling of the acoustic wave equation with transform methods: Geophysics, **46**, 854–859.

Hairer, E., C. Lubich, and G. Wanner, 2002, Geometric numerical integration: Springer-Verlag.

Hairer, E., S. P. Nøsett, and G. Wanner, 1993, Solving ordinary differential equations I: Springer-Verlag.

Kelly, K. R., R. W. Ward, S. Treitel, and R. M. Alford, 1976, Synthetic seismograms: A finite-difference approach: Geophysics, **41**, 2–27.

Komatitsch, D., and J. P. Vilotte, 1998, The spectral element method: An efficient tool to simulate the seismic response of 2D and 3D geological structures: Bulletin of the Seismological Society of America, **88**, 369–392.

Kosloff, D., and E. Baysal, 1982, Forward modeling by the Fourier method: Geophysics, **47**, 1402–1412.

Marfurt, K. J., 1984, Accuracy of finite-difference and finite-element modeling of the scalar and elastic wave equations: Geophysics, **49**, 533–549.

Nyström, E. J., 1925, Üeber die numerische Integration von Differentialgleichungen: Acta Societe Scientiarum Fennicae, **50**, 1–54.

Qin, M. Z., and M. Q. Zhang, 1990, Multi-stage symplectic schemes of two kinds of Hamiltonian systems for wave equations: Computers and Mathematics with Applications, **19**, 51–62.

Qin, M. Z., and W. J. Zhu, 1991, Canonical Runge-Kutta-Nyström methods for second order ODE's: Computers and Mathematics with Applications, **22**, 85–95.

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

Ruth, R. D., 1983, A canonical integration technique: IEEE Transactions on Nuclear Science, **30**, 2669–2671.

Sanz-Serna, J. M., and M. Calvo, 1994, Numerical Hamiltonian problems: Chapman and Hall.

Yoshida, H., 1990, Construction of higher order symplectic integrators: Physics Letters A, **150**, 262–269.