Received: 23 February 2023 | Accepted: 14 July 2023

DOI: 10.1111/1365-2478.13411

Geophysical Prospecting | EAGE EUROPEAN ASSOCIATION OF GEOSCIENTISTS & ENGINEERS

ORIGINAL ARTICLE

Temporal dispersion correction for wave-propagation modelling with a series approach

W. A. Mulder[1,2]

[1]Shell Global Solutions International B.V., Den Haag, The Netherlands

[2]Department of Geoscience & Engineering, Delft University of Technology, Delft, The Netherlands

Correspondence
W. A. Mulder, Carel van Bylandtlaan 16, 2596 HR Den Haag, The Netherlands.
Email: Wim.Mulder@shell.com

Abstract

Temporal dispersion correction of second-order finite-difference time stepping for numerical wave propagation modelling exploits the fact that the discrete operator is exact but for the wrong frequencies. Mapping recorded traces to the correct frequencies removes the numerical error. Most of the implementations employ forward and inverse Fourier transforms. Here, it is noted that these can be replaced by a series expansion involving higher time derivatives of the data. Its implementation by higher-order finite differencing can be sensitive to numerical noise, but this can be suppressed by enlarging the stencil. Tests with the finite-element method on a homogeneous acoustic problem with an exact solution show that the method can achieve the same accuracy as higher-order time stepping, similar to that obtained with Fourier transforms. The same holds for an inhomogeneous problem with topography where the solution on a very fine mesh is used as reference. The series approach costs less than dispersion correction with the Fourier method and can be used on the fly during the time stepping. It does, however, require a wavelet that is sufficiently many times differentiable in time.

KEYWORDS
computing aspects, modelling, seismics, wave

INTRODUCTION

Discretization errors affect numerical modelling of wave propagation for seismic applications. For the spatial part, the errors can be kept small by using sufficiently many points per wavelength in a finite-difference code. Higher-order schemes are usually more efficient, except in very heterogeneous earth models. The same holds for finite elements that tend to have a better accuracy than finite differences in the presence of rough topography or large contrasts in the subsurface properties, but only if the element boundaries follow the sharp interfaces.

The temporal error is likewise controlled by the size of the time step. A smaller time step reduces the error but also increases the number of required time steps and, therefore, the computational cost. For time stepping with a second-order finite-difference scheme in time, Stork (2013) showed that the temporal error can be removed from recorded traces by dispersion correction. The numerical scheme produces a solution that is exact for the wrong frequency. When mapped to the correct frequency, the error almost completely disappears. Since Stork's paper, several implementations of this idea were proposed (Anderson et al., 2015; Dai et al., 2014; Koene et al., 2018; Li et al., 2016; Mittet, 2017; Qin et al., 2017; Xu et al., 2017; Wang & Xu, 2015b).

The method of Koene et al. (2018) performs the Fourier interpolation by means of a slow Fourier transform, followed

by a fast inverse one. When applying the scheme, a problem was encountered when requiring very high accuracy, in particular with low frequencies in the source wavelet and in two dimensions where the Green function has slow decay (Mulder, 2023). The dispersion correction suffers from the abrupt truncation of the recorded traces at the maximum recording time. The obvious solution of extending the maximum time of the simulation in combination with a smooth taper will ameliorate the problem, but the required extra number of time steps increases the compute cost. Also, the Fourier interpolation has a quadratic cost in the number of time samples, which can be lowered by using a non-uniform fast Fourier transform (Dutt & Rokhlin, 1993; Duijndam & Schonewille, 1999; Feichtinger et al., 1995; Potts et al., 2001; Potter et al., 2017) instead.

Here, an alternative implementation is proposed, based on a series expansion in the time domain and involving only local higher time derivatives of the data. This next section summarizes elementary facts about numerical dispersion and its correction by a Fourier-type approach, using a finite-difference discretization of the one-dimensional wave equation in second-order form as an example. The reader familiar with the subject can skip it. Then, the method is described and tested on a homogeneous problem with an exact solution as well as an inhomogeneous problem with topography, both taken from Mulder (2023). The results are compared to those obtained with the method of Koene et al. (2018) and with higher-order time stepping by the Cauchy–Kovalewski or Lax–Wendroff or Dablain or modified-equation method (Dablain, 1986; Lax & Wendroff, 1960; Shubin & Bell, 1987; von Kowalevsky, 1875). The last section summarizes the main conclusions.

BACKGROUND

Here, elementary facts about errors in the finite-difference method are reviewed. Stork's (2013) dispersion correction and its implementation with Fourier transforms are explained. The method of Koene et al. (2018) is compared to an obvious alternative, which actually might be new in this context.

Consider the one-dimensional scalar wave equation $\partial_{tt}u = c^2\partial_{xx}u$ with solution $u(t,x)$, depending on time t and position x, and a constant wave speed c. The solution is discretized in space on a periodic grid with points $x_j = x_0 + j\Delta x$, $j = 0, 1, \ldots, N_x - 1$. The discretization in time involves time levels $t^n = n\Delta t$, $n = 0, 1, \ldots$, leading to a discrete solution with values $u_j^n = u(t^n, x_j)$. The periodic boundary condition sets $u_{N_x}^n = u_0^n$.

Define a shift operator T_x by $T_x^m u_j^n = u_{j+m}$ for integer m. A central difference discretization in space of order $2M_x$ is given by the operator

$$D_{xx}^{(M_x)} = -\frac{1}{\Delta x^2}\left[w_0 + \sum_{m=1}^{M_x} w_m (T_x^m + T_x^{-m})\right], \qquad (1)$$

with weights (Fornberg, 1987, among other)

$$w_0 = \sum_{k=1}^{M_x} \frac{2}{k^2}, \quad w_m = (-1)^m \sum_{k=m}^{M_x} \frac{2}{k^2}\frac{(k!)^2}{(k-m)!(k+m)!}, \qquad (2)$$

for $1 \le m \le M_x$. If a second-order central scheme is applied in time, the discrete equation becomes

$$\frac{u_j^{n+1} - 2u_j^n + u_j^{n-1}}{\Delta t^2} = c^2 D_{xx}^{(M_x)} u_j^n. \qquad (3)$$

An estimate of the numerical error follows from a Fourier expansion in $\hat{u} = \exp[\mathrm{i}(kx - \omega t)]$ with wavenumber k. In the exact case, $k = \pm\omega/c$. Minus signs are usually ignored because of (anti-)symmetry. Substitution of a Fourier mode in Equation (3) produces

$$-\left[\frac{\sin(\omega\Delta t/2)}{\Delta t/2}\right]^2 \hat{u} = \hat{D}_{xx}^{(M_x)}\hat{u}, \qquad (4)$$

where the Fourier symbol $\hat{D}_{xx}^{(M_x)}$ follows from $D_{xx}^{(M_x)}$ by replacing T_x with $\hat{T} = \mathrm{e}^{\mathrm{i}k\Delta x}$. The right-hand side can be obtained as the truncated series expansion of $[2(c/\Delta x)\arcsin(\sqrt{\eta})]^2$ in η up to the power M_x, followed by substitution of $\eta = \sin^2(k\Delta x/2) = (2 - \hat{T} - \hat{T}^{-1})/4$.

A Taylor series expansion of Equation (4) in Δt and Δx provides

$$-\omega^2\left[1 - \frac{1}{12}(\omega\Delta t)^2\right] = (ck)^2\left[1 - \frac{2\,(M_x!)^2}{[2(M_x+1)]!}(k\Delta x)^{2M_x}\right], \qquad (5)$$

revealing a second-order error in time and an order $2M_x$ in space. The net effect of these errors will appear as numerical dispersion, with the dispersion relation $\omega(k)$ implicitly given by Equation (4) and approximately by Equation (5).

Figure 1a shows an example with $c = 1$ km/s, periodic boundary condition $u(t, x = x_{\max}) = u(t, x = 0)$ with $x_{\max} = 1$ km and discrete initial values $u_j^0 = w_0(x_j)$, $u_j^{-1} = w_0(x_j + c\Delta t)$ for a pulse $w_0(x) = [4\max(0, \xi(1-\xi))]^4$ with $\xi = (x - 400)/100$. The exact solution is $u(t,x) = w_0(x - ct)$, describing a pulse travelling to the right. The time step is set at about half the maximum time step, defined by the Courant–Friedrichs–Lewy (CFL)number: $\nu = c\Delta t/\Delta x \simeq$

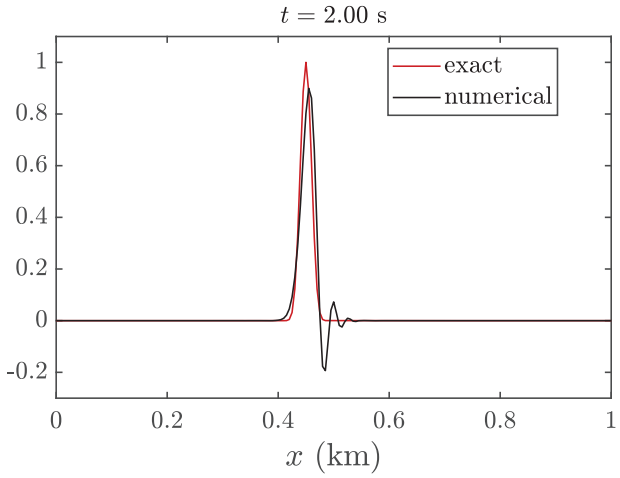

FIGURE 1 A pulse is distorted by numerical dispersion after travelling around the periodic domain twice. The second-order time discretization causes higher frequencies to move faster.

$0.5\,v_{\max}$, where

$$1/v_{\max}^2 = \tfrac{1}{2}\sum_{m=1}^{M_x} 4^m[(m-1)!]^2/(2m)! \tag{6}$$

for a scheme of spatial order $2M_x$. The example has an eighth-order scheme with $M_x = 4$ and $v_{\max} = \sqrt{315/512}$.

The numerical result at time $2x_{\max}/c$, after travelling two times around, is drawn in Figure 1a as a red line and the exact solution as a black line. If the small spatial error is neglected, $\omega \simeq \arcsin(ck\Delta t/2)/(\Delta t/2)$, the phase velocity obeys $c_{\text{phase}}/c \simeq 1 + (\omega\Delta t)^2/24$ and the group velocity $c_{\text{group}}/c = c^{-1}\mathrm{d}\omega/\mathrm{d}k \simeq 1/\sqrt{1-(\omega\Delta t/2)^2} \simeq 1 + (\omega\Delta t)^2/8$, showing that the higher frequencies tend to travel faster, an effect clearly visible in Figure 1a.

Stork (2013) noted that Equation (4) is actually exact in time for a modified frequency $\tilde{\omega} = \omega\,\mathrm{sinc}(\omega\Delta t/2)$ with $\mathrm{sinc}(x) = \sin(x)/x$. To remove the effect of dispersion, he computes the effect of dispersion on a given wavelet as a function of several time intervals and interpolates between the results to obtain a time-variable filter. Other implementations follow the same idea (Anderson et al., 2015; Dai et al., 2014; Li et al., 2016). The approach via the Fourier domain, which is considerably easier to code, was taken by Wang and Xu (2015b) and later by Koene et al. (2018), Mittet (2017), Qin et al. (2017) and Xu et al. (2017). The last corrects a flaw in the approach of Wang and Xu (2015b), but there is an alternative to both, as will be shown below.

Figure 2a sketches the idea. Suppose the angular frequencies on the horizontal axis were obtained after a fast Fourier transform (FFT) on a modelled seismic trace, in this example for equally spaced frequencies from 0 to 50 Hz. Only very few samples are shown on the horizontal axis of Figure 2a for clarity. These angular frequencies should actually be interpreted as the non-uniformly spaced samples of $\tilde{\omega}$ on the vertical axis. If a fast inverse transform is to be used to return to the time domain, the data as a function of $\tilde{\omega}$ should be interpolation to an equidistant grid, from 0 to $(2/\pi)\omega_{\max}$. The obvious approach is to use higher-order interpolation with Lagrange interpolating functions. However, the real and imaginary parts of the spectrum can be highly oscillatory and require a very high order. An alternative is to map the data to amplitude and phase and perform the interpolation in that domain, but phase unwrapping can be difficult. The interpolation operator of the highest order, involving all grid points, amounts to the Fourier interpolation with unevenly spaced points and will be reviewed below. Before doing that, it should be noted that Koene et al. (2018) take an opposite approach, illustrated in Figure 2b. The initial FFT from the time domain to the frequency domain is replaced by a non-uniform Fourier transform that produces angular frequencies ω that have the corresponding $\tilde{\omega}$ on an equidistant grid. The result can then immediately be transformed back to the time domain by an FFT.

An alternative is to first use an FFT and then use the Fourier interpolation to map the exact but non-equidistant angular frequencies $\tilde{\omega}$ to an equidistant grid. Consider the spectrum $g(f)$ with frequencies $f \in [-f_{\max}, f_{\max}]$ and $g(f) = 0$ for $|f| > f_{\max}$. Its Fourier expansion with coefficients a_n is

$$\begin{aligned} g(f) &= \sum_{n=-\infty}^{\infty} a_n \mathrm{e}^{2\pi\mathrm{i}nf/(2f_{\max})}, \\ a_n &= \frac{1}{2f_{\max}}\int_{-f_{\max}}^{f_{\max}} \mathrm{d}f\, g(f)\mathrm{e}^{-2\pi\mathrm{i}nf/(2f_{\max})}. \end{aligned} \tag{7}$$

Assuming that $g(f)$ has been obtained from an FFT on well-sampled periodic time data with constant sampling interval $\Delta t = 1/(2f_{\max})$, we have discrete frequencies f_j, symmetrically arranged relative to f_0 with $f_{-j} = -f_j$. For real-valued time data, $g_j = g(f_j) = g_{-j}^*$, where the asterisk denotes the complex conjugate. If there are n_{f} frequencies with n_{f} odd, take $j = -j_{\max}, \ldots, j_{\max}$, $j_{\max} = (n_{\mathrm{f}}-1)/2$, including $f_0 = 0$. If n_{f} is even, consider the n_{f} indices $j = -j_{\max}+1, \ldots, j_{\max}$ with $j_{\max} = n_{\mathrm{f}}/2$ and again $f_0 = 0$, and let the now real-valued $g_{-j_{\max}} = g_{j_{\max}}$ correspond to the highest frequency, which is proportional to $-1, +1, -1, +1, \ldots$ in the time domain. In this way, we effectively obtain $n_{\mathrm{f}} + 1$ discrete frequencies.

The integral in Equation (7) can be approximated by the trapezoid rule, leading to

$$a_n = \Delta t \sum_{j=-j_{\max}}^{j_{\max}} \tfrac{1}{2}(\tilde{g}_{j-1} + \tilde{g}_j)(f_j - f_{j-1}), \quad \tilde{g}_j = g_j \mathrm{e}^{-2\pi\mathrm{i}n(f_j\Delta t)}, \tag{8}$$

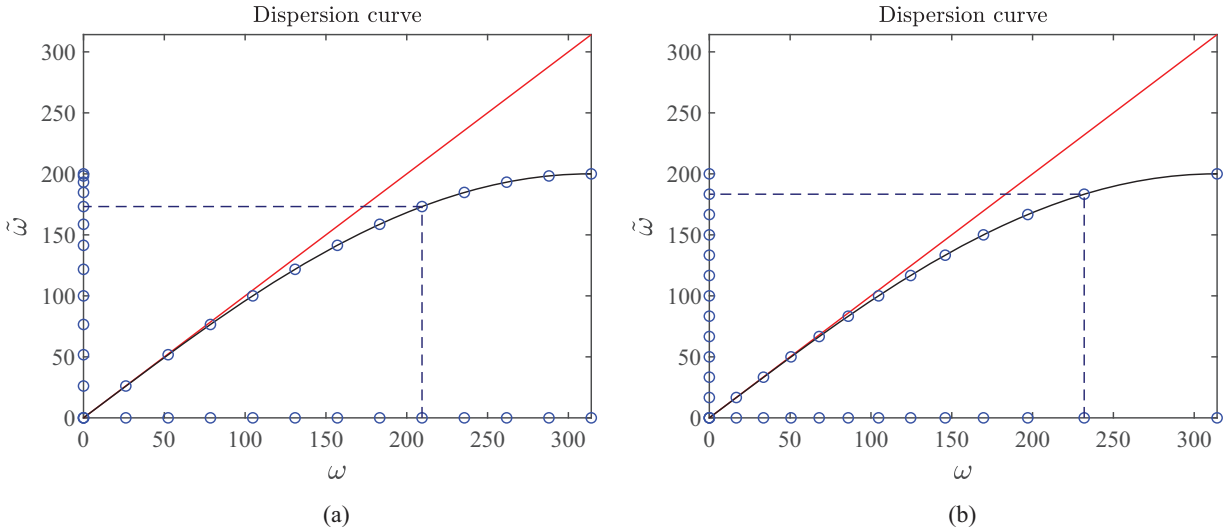

FIGURE 2 Illustration of dispersion correction in the frequency domain. (a) An FFT produces equidistant angular frequencies ω, shown on the horizontal axis. The numerical dispersion curve in black deviates from the true one in red. The vertical axis contains the angular frequencies $\tilde{\omega}$ for which the numerical result is exact. The frequency-domain data can be interpolated to an equidistant grid in $\tilde{\omega}$ followed by an FFT to the time domain or can be transformed directly to the time domain by a non-uniform Fourier transform. (b) A non-uniform Fourier transform can also be applied directly on the time-domain data to obtain values of ω on a non-equidistant grid that produce an equidistant grid for $\tilde{\omega}$, after which an inverse FFT can be applied.

where $f_{\max} = 1/(2\Delta t)$ has been used. In this periodic setting, the trapezoid rule converges exponentially for sufficiently smooth functions, much better than the better-known non-periodic case with only second-order accuracy (Trefethen & Weideman, 2014). With $g_{-j} = g_j^*$ and $\tilde{g}_{-j} = \tilde{g}_j^*$, the expression (8) with $h_{j-1/2} = \Delta t(f_j - f_{j-1})$ reduces to

$$\begin{aligned} a_n &= 2\,\mathrm{Re}\sum_{j=1}^{j_{\max}} \tfrac{1}{2}(\tilde{g}_{j-1} + \tilde{g}_j)h_{j-1/2} \\ &= \mathrm{Re}\left[\tilde{g}_0 h_{1/2} + \tilde{g}_1(h_{1/2} + h_{3/2}) + \cdots \right. \\ &\left. + \tilde{g}_{j_{\max}-1}(h_{j_{\max}-3/2} + h_{j_{\max}-1/2}) + \tilde{g}_{j_{\max}} h_{j_{\max}-1/2}\right], \end{aligned} \tag{9}$$

or

$$a_n = \mathrm{Re}\sum_{j=1}^{j_{\max}} g_j \tilde{h}_j \mathrm{e}^{-2\pi \mathrm{i}\, n(f_j \Delta t)}, \tag{10}$$

where $\tilde{h}_1 = h_{1/2}$, $\tilde{h}_j = h_{j-1/2} + h_{j+/2}$ for $j = 2, \ldots, j_{\max} - 1$, $\tilde{h}_{j_{\max}} = h_{j_{\max}-1/2}$. To obtain the interpolated values of $g(f)$ on an equidistant grid of frequencies, the coefficients have to be substituted into the expansion of $g(f)$ in Equation (7), which can be accomplished by an FFT. That step can be skipped, since we are only interested in the time-domain result, already available as $a_n = a(t^n)$.

Figure 3 presents an example of this procedure on a problem taken from figure 3 in Koene et al. (2018), where dispersion is added to and then removed from an 8-Hz Ricker wavelet that peaks at 0.2 s. The time samples are $t^n = n\Delta t$ with $\Delta t = 0.015$ s and $n = 0, 1, \ldots, 26$. The method described above provides a very similar result to that of Koene et al. (2018), who argue that the approach of Wang and Xu (2015a) is incorrect. However, what the latter refer to as forward time dispersion transform step 3 is, strictly taken, correct if the integral would be approximated by the above approach with the trapezoid rule.

If a seismic trace does not end with zero values, which is usually the case, a smooth taper should be applied at the end. If the data are to be preserved over a given time span, this implies that the modelling time should be extended beyond the desired maximum time to provide sufficiently many extra time samples for the taper (Mulder, 2023).

METHOD

The second-order formulation of the wave equation contains the second time derivative $\partial_{tt} u$, where $u(t, \mathbf{x})$ is the pressure as a function of time t and position $\mathbf{x}$ in the acoustic case or one of the displacement or particle velocity components in the elastic case. A second-order finite-difference approximation is $D_{tt}u^n = (u^{n+1} - 2u^n + u^{n-1})/\Delta t^2$ with time step Δt. The discrete solution at time $t^n = t_0 + n\Delta t$ is denoted by $u^n = u(t^n)$, and it will be assumed that $t_0 = 0$. The discrete approximation is exact for an angular frequency $\tilde{\omega} = \omega\,\mathrm{sinc}(\omega\Delta t/2)$, with $\mathrm{sinc}(x) = \sin(x)/x$. The dispersion correction scheme

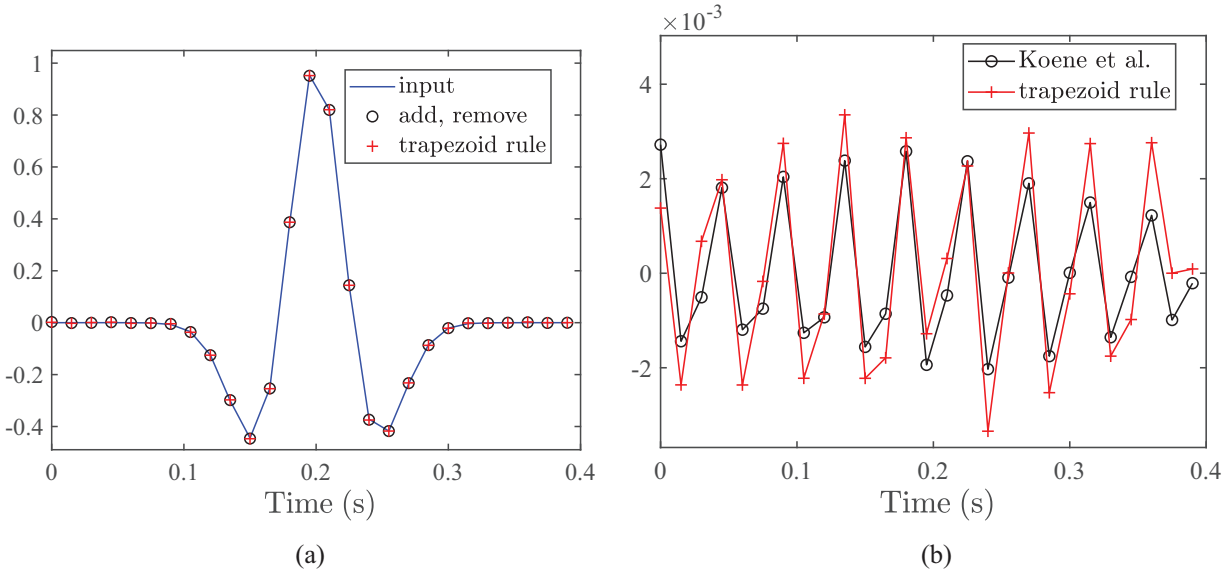

FIGURE 3 Comparison of the approaches outlined in Figure 2 by adding dispersion to a Ricker wavelet and then removing it. (a) The black circles show the result of applying a non-uniform Fourier transform to map the signal from time to frequency and an FFT to go back to time, as proposed by Koene et al. (2018), whereas the red plusses are obtained for the reverse approach, using an FFT to go the frequency domain followed by a non-uniform Fourier transform on non-equidistant frequencies to go back to the time domain. (b) The difference between the output of adding and removing dispersion and the original wavelet for the two methods.

maps the recorded data to the correct angular frequency ω. Since this also changes the wavelet, dispersion has to be added to the original wavelet to obtain the correct spectrum after correcting the data.

Among several later implementations of this idea, the one of Koene et al. (2018) is taken as a starting point. Their forward step, which adds dispersion to the wavelet and is called forward time dispersion transform, is given by

$$\tilde{u}(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \mathrm{d}\omega\, \mathrm{e}^{\mathrm{i}\omega t} \int_{-\infty}^{\infty} \mathrm{d}t'\, \mathrm{e}^{-\mathrm{i}\omega t' \operatorname{sinc}(\omega \Delta t/2)} u(t'). \tag{11}$$

A series expansion in the time step Δt enables the direct evaluation of the integrals and provides

$$\begin{aligned} \tilde{u}(t) &= u(t) + \sum_{k=1}^{\infty} \frac{(\Delta t/2)^{2k}}{(2k+1)!} r_k(t), \\ r_k &= \sum_{\ell=1}^{k} (-1)^{\ell} a_{k,\ell} \partial_t^{2k+\ell} [t^{\ell} u(t)], \end{aligned} \tag{12}$$

as derived in the Appendix. The coefficients $a_{k,\ell}$ in Equation (A.8b) are non-zero for $1 \le \ell \le k$ and are defined such that $a_{k,1} = 1$. The time derivatives are replaced by operators $D^{[j,m]}$, defined as the discrete jth derivative of order m, m even, for a unit grid spacing. The discrete approximation of Equation (12) becomes

$$\begin{aligned} \tilde{u}(t) &\simeq u(t) + \sum_{k=1}^{k_{\max}} \frac{(1/2)^{2k}}{(2k+1)!} \bar{r}_k(t), \\ \bar{r}_k &= \sum_{\ell=1}^{k} a_{k,\ell} (-\Delta t)^{-\ell} D^{[2k+\ell, \max(1, M_t - 2(k-1))]} (t^{\ell} u), \end{aligned} \tag{13}$$

with t and $u(t)$ are given on the same equidistant grid with spacing Δt. The operator $D^{[j,m]}$ is defined as the discrete jth derivative of order m, m even, for a unit grid spacing. If M_x is the order of the error in the spatial discretization, it makes sense to choose the temporal order $M_t > M_x$, leading to $k_{\max} = M_t/2 = \mathrm{ceil}((M_x+1)/2)$, though $k_{\max} = \mathrm{ceil}(M_x/2)$ may already suffice.

The expression for the actual dispersion correction, called inverse time dispersion transform by Koene et al. (2018), is

$$u(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \mathrm{d}\omega\, \mathrm{e}^{\mathrm{i}\omega t} \int_{-\infty}^{\infty} \mathrm{d}t'\, \mathrm{e}^{-\mathrm{i}\omega t' \arcsin(\omega \Delta t/2)/(\omega \Delta t/2)} \tilde{u}(t'), \tag{14}$$

and results in the series expansion

$$\begin{aligned} u(t) &= \tilde{u}(t) + \sum_{k=1}^{\infty} (-1)^k \frac{(\Delta t/4)^{2k} (2k)!}{(2k+1)(k!)^2} s_k(t), \\ s_k &= \sum_{\ell=1}^{k} (-1)^{\ell} b_{k,\ell} \partial_t^{2k+\ell} [t^{\ell} \tilde{u}(t)], \end{aligned} \tag{15}$$

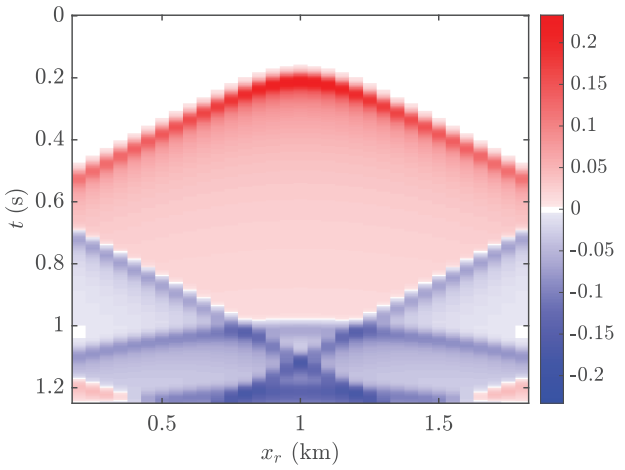

FIGURE 4 Seismogram as function of the horizontal receiver position x_r and time t.

with coefficients $b_{k,\ell}$ given by Equations (A.12b). The discrete counterpart of (15) is

$$u(t) \simeq \tilde{u}(t) + \sum_{k=1}^{k_{\max}} (-1)^k \frac{(1/4)^{2k}(2k)!}{(k!)^2(2k+1)} \bar{s}_k(t),$$

$$\bar{s}_k = \sum_{\ell=1}^{k} b_{k,\ell} (-\Delta t)^{-\ell} K^{-(2k+\ell)} D^{[2k+\ell,\max(1,M_t-2(k-1))]}(t^\ell \tilde{u}). \tag{16}$$

The factor with K accounts for subsampling if the data are recorded at a sampling interval $K\Delta t$ with an integer $K \geq 1$.

RESULTS

Homogeneous model

The method was tested on an earlier homogeneous problem (Mulder, 2023) with degree-4 mass-lumped triangular finite elements (Mulder, 2022, 1996) on an unstructured mesh. The source is placed at the centre of a 2-km wide square domain with reflecting zero boundary conditions, a sound speed of 2 km/s and a density of 2 g/cm^3. The receivers are located 200 m below the source with horizontal offsets from −800 to +800 m at a 50-m interval. The wavelet $w(t) = [4(t/T_w)\{1-(t/T_w)\}]^{16}$ is non-zero for time $t \in (0, T_w)$ and zero otherwise, with $T_w = 0.2$ s. Data, shown in Figure 4, are recorded from $t = 0$ to 1.25 s. The time step was taken at 99% of the maximum time step for stability, determined by the power method (Mises & Pollaczek-Geiringer, 1929; Geevers et al., 2019). Figure 5 displays the errors as a function of $N^{1/2}$, the square root of the number of degrees of freedom N including the zero boundary values and of the measured wall clock time for the time-stepping part only, obtained on a single core and averaged over five runs. Both the relative root-mean-square error (RMS), defined as the RMS error of the error divided by that of the solution, and relative maximum error, defined as the maximum error divided by the maximum solution value, are shown. The grey line in Figure 5a and 5c denotes the expected trend of order $M_x = p+1$ for a finite element of degree $p = 4$. The entries in the legend denote the degree p of the element, the order of the time-stepping scheme if there is a comma and an even integer, and the number of nodes per element, 18 in this case. If there are brackets instead of a comma after the degree p, second-order time stepping with dispersion correction is used. The Fourier approach of Koene et al. (2018) is denoted by '(d)' and the series approach with, for instance, '(s6, 2)'. The first number is $2k_{\max}$, the maximum degree of the series expansion in Δt. The second number refers to half the number of extra points used for the numerical derivatives. For the standard central finite-difference scheme, it is zero. The dashed lines mark the results for second- and higher-order time stepping. The maximum time step for stability increases for higher time orders, listed for instance in Table 4 of Mulder (2013), which partly offsets the higher cost of these schemes. The cost increases roughly by a factor of $M_t/2$ for a time stepping order M_t, relative to second-order time stepping. The sixth-order scheme exceeds the order of the spatial error by one and can serve as a reference for comparing the other time-stepping methods. The Fourier approach of Koene et al. (2018), corresponding to the red drawn line, deteriorates for very small errors – so small, however, that they may never be needed in practice.

The series approach, with the receiver sampling time equal to the time step, is worse with the standard central finite-difference scheme but improves when extra points are added to suppress numerical noise (McDevitt, 2012). In this example, widening the finite-difference stencil with two times 4 points (s6, 4) appears to suffice, both for the RMS and the more erratic maximum error. In terms of efficiency, measured by the elapsed compute time for a given accuracy, the series approach with 2×4 extra points for the numerical derivatives appears to be the most attractive. The time-stepping cost for the Fourier approach is larger because the recording time was extended to accommodate a taper with a length equal to that of the wavelet, 0.2 s. The same was done for the series approach, but by a much smaller amount, with a number of samples equal to half the width of the largest finite-difference stencil. This can be avoided by using one-sided difference operators but then more coding is required.

Figure 6 illustrates the beneficial effect of the enlarged finite-difference stencils on a single trace. The noise in Figure 6a is reduced when extra points are added to the stencil, and three or four extra points on each side are sufficient in this example. The data correspond to the first trace of four

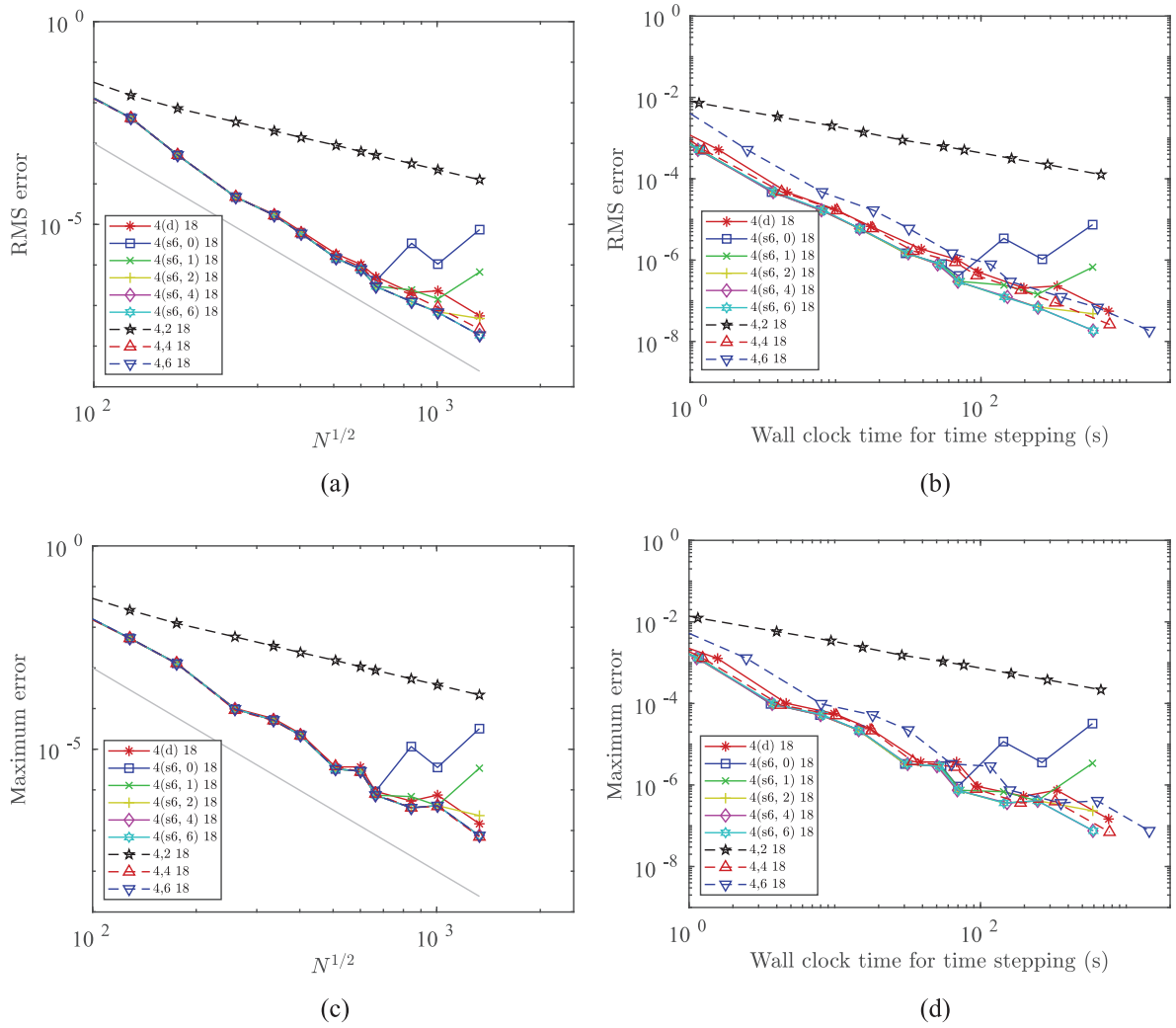

FIGURE 5 Relative RMS (a,b) and maximum errors (c,d) as a function of the square root of the number of degrees of freedom (a,c) and the measured wall clock time for time stepping only (b,d).

runs with a fairly large number of degrees of freedom, $N = 710,765$, for wavelets treated by the series approach with the same number of additional points as used for the data.

Figure 7 shows the effect of subsampling the recorded traces, as commonly done in practice. To avoid additional runs, the same data were used as before but subsampled to the largest time interval that does not exceed 2 ms before applying the correction of Equation (16). The original runs were sampled at the maximum allowable time step. With the subsampling, the standard central high-order finite-difference scheme appears to be sufficient and no additional points are needed for numerical-noise suppression, at least in this example.

Inhomogeneous model with topography

Apart from the time-stepping error, the spatial discretization will play a role. For a homogeneous problem on a rectangular domain with reflecting boundary conditions, this error converges more rapidly the higher the order of the scheme when the mesh is refined. This changes if there are discontinuities in the material properties, with topography as the strongest contrast. In the acoustic case, the pressure is continuous across an interface that describes the positions of the discontinuities, but its derivatives are not. If the model parameters are represented on the finite-difference grid in the most straightforward way, by just sampling, the position of the interface is unknown within a distance of the grid spacing, leading to a first-order term. Therefore, if a finite-difference scheme is of order q, the error will be of the asymptotic form $C_0h^q + C_1h + C_2h^2$ for a grid spacing h. Here, the constants C_k, $k = 0, 1, 2$, depend on the problem. If the spacing h is too large, the error will not follow the asymptotic form. If it is too small, numerical round-off errors will appear, so the estimate is only valid for a range of h-values. With time stepping, the spatial errors will accumulate but in different ways. The first term C_0h^q will affect a wave continuously and grow linearly in time, whereas

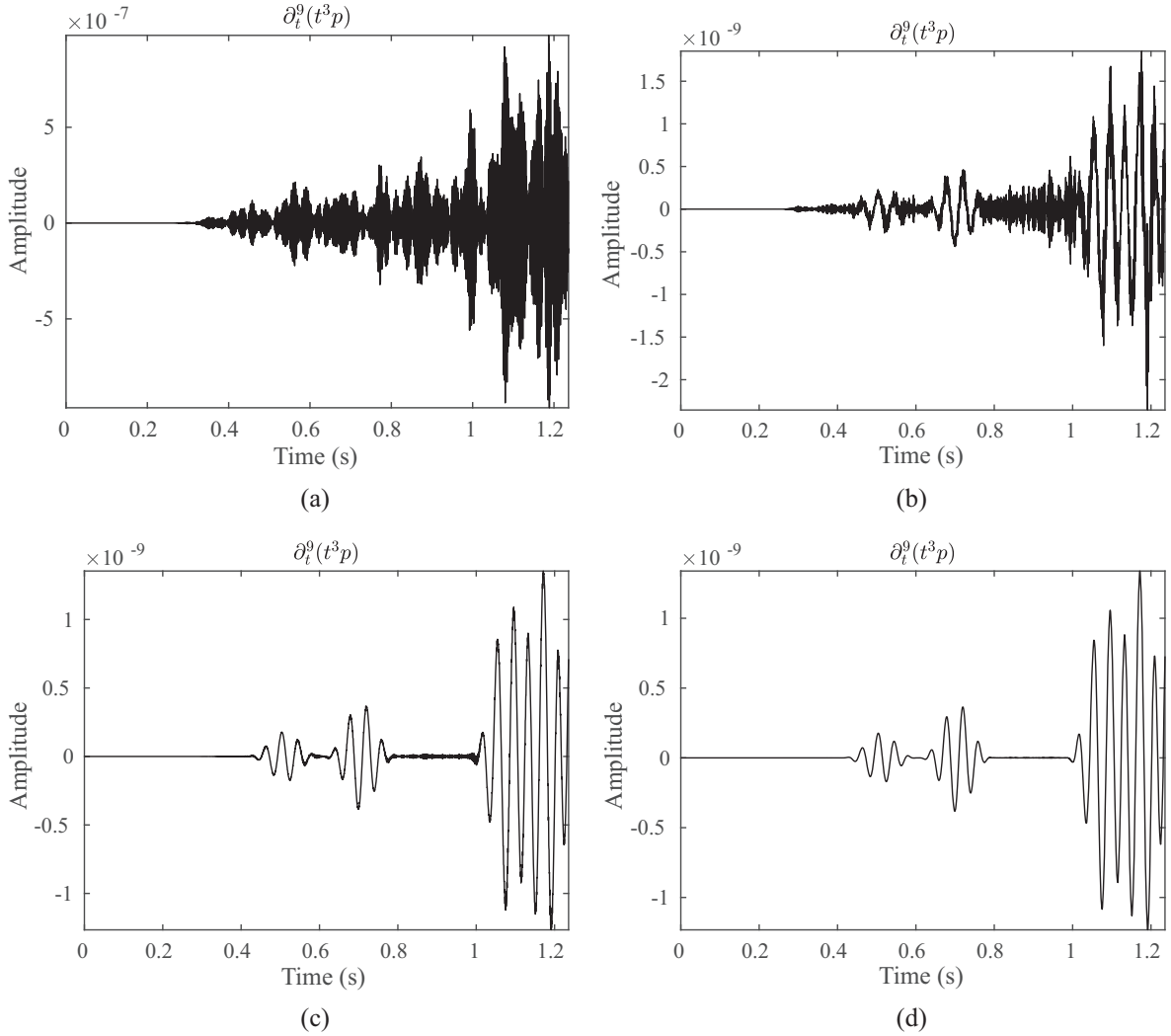

FIGURE 6 Effect of the number of extra stencil points on the removal of numerical noise for the approximation of the term $\partial_t^{2k+\ell}[t^\ell \tilde{u}]$ with $k = 3$ and $\ell = 3$. Half the number of extra points equals 0 (a), 2 (b), 3 (c) or 4 (d).

the other two terms will only contribute when a wave passes an interface. If the model parameters vary smoothly and the number of interfaces is small, a higher-order scheme may still pay off.

The same is true for finite elements. In some cases, however, the first- and second-order errors at interfaces can be suppressed if the mesh precisely follows the interfaces between discontinuous material properties. The error will then behave as $C_f h^{p+1}$ for an element size h and a degree-p polynomial element of the type used here. In general, this cannot be achieved due to, for instance, small-scale details, sharp pinch-outs, curvature, very small distances between neighbouring faults and fractal behaviour of earth properties. Highly distorted meshes will adversely affect the accuracy. The presence of corner singularities, for corners with angles other than 90° in the topography or at interfaces, will degrade the numerical order. For not too complicated models, however, finite elements can perform better than finite differences (e.g., Mulder, 1996; Zhebel et al., 2014).

Figure 8 displays a check-shot setting, taken from Mulder (2023). The seismogram shows traces as a function of time, up to 1.2 s, and receiver depth. Values are clipped at 2% of the maximum amplitude over all traces. The acoustic wave equation is solved. The wavelet is $w(t) = [4(t/T_w)\{1-(t/T_w)\}]^8$ for $t \in (0, T_w)$ and zero otherwise, with $T_w = 0.1$ s. The shot is placed at $x_s = 100$ m and a depth of 6 m depth below the surface, corresponding to $z_s = -825.573$ m. The receiver line at $x_r = 0$ m has 137 receivers at a 20-m interval, starting at $z = -750$ m or 30.62 m below the surface and ending at $z = 1970$ m. The recording time is 1 s. Zero Dirichlet boundary conditions are used on the entire boundary to avoid contamination by less than perfect absorbing boundaries. The error in the seismic data is estimated by comparison to results computed on a very fine mesh.

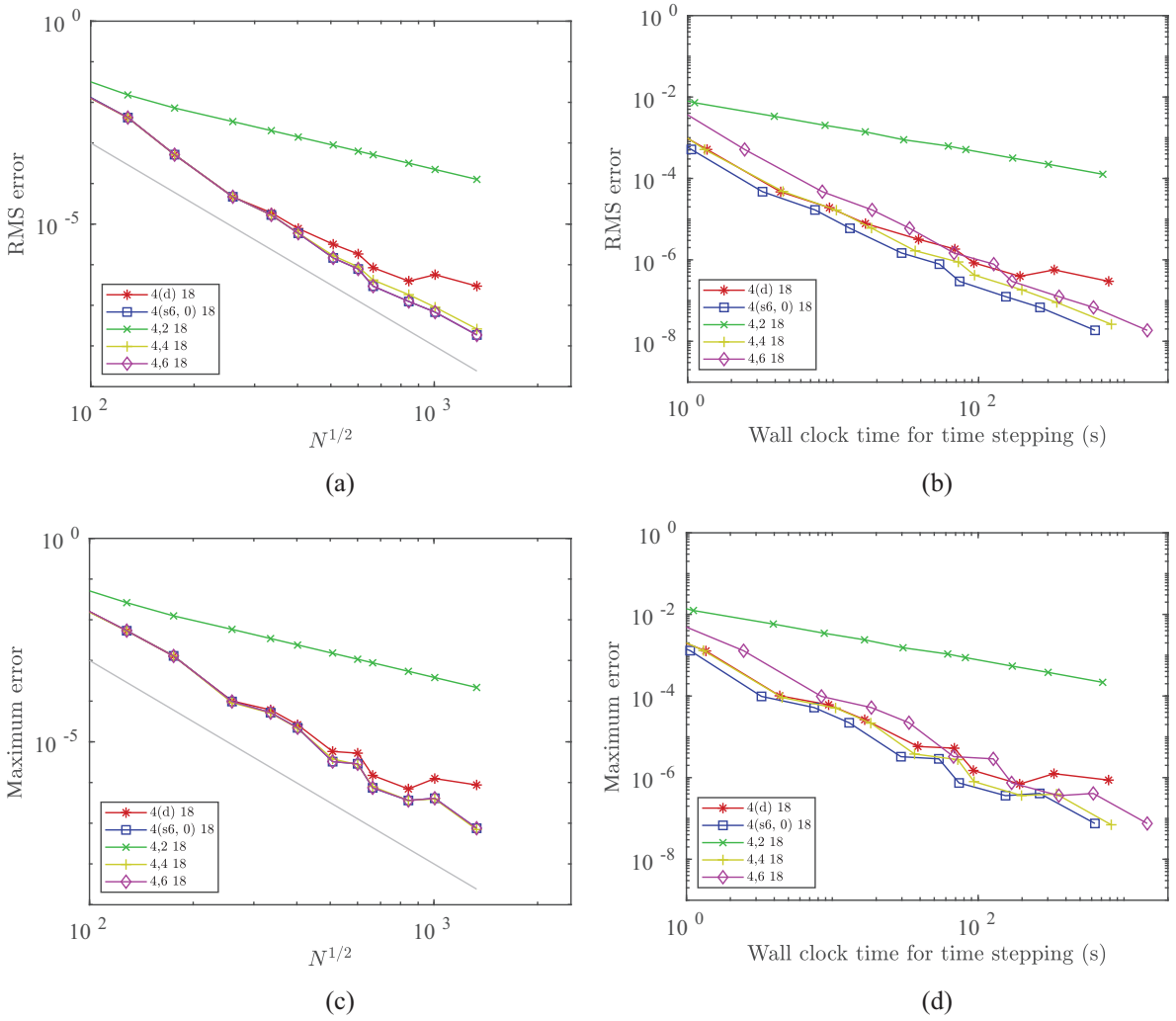

FIGURE 7 As Figure 5 but with subsampling. Apparently, numerical-noise suppression with extra points for the finite-difference stencils is not needed.

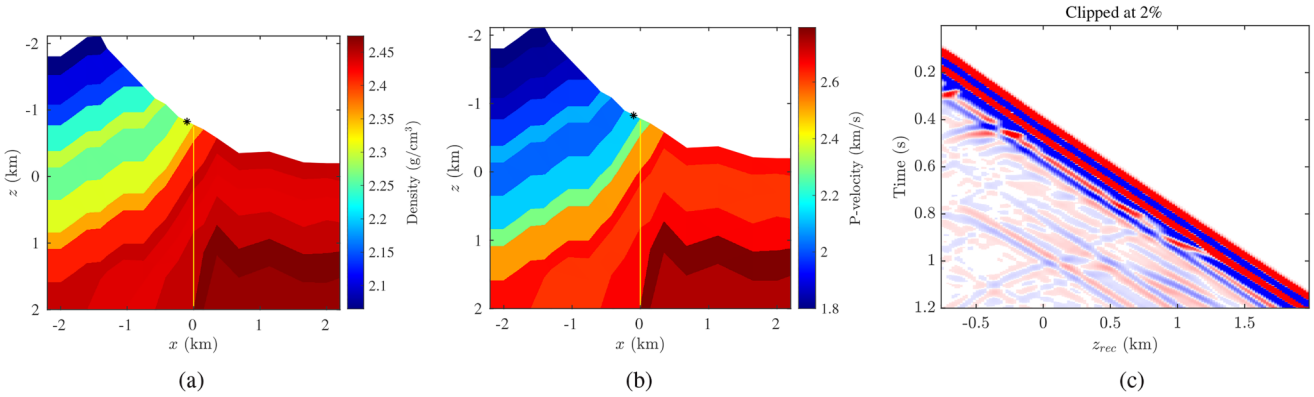

FIGURE 8 Density (a), P-velocity model (b) and seismogram (c) from Mulder (2023). The black star in (a) and (b) marks the source position and the yellow line the receivers.

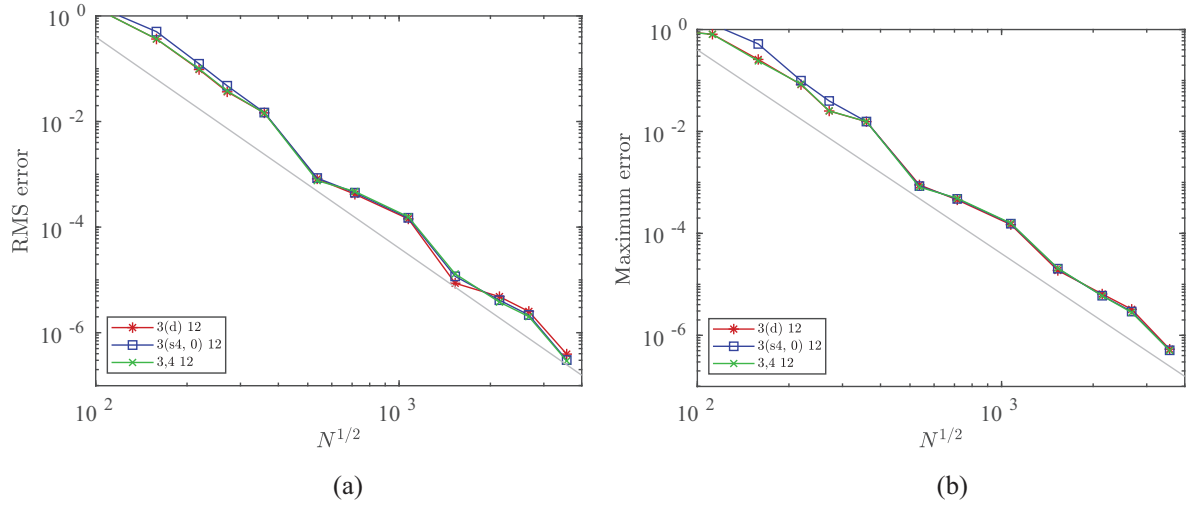

FIGURE 9 Relative RMS (a) and maximum errors (b) as a function of the square root of the number of degrees of freedom for the problem with topography. The grey line corresponds to fourth-order convergence.

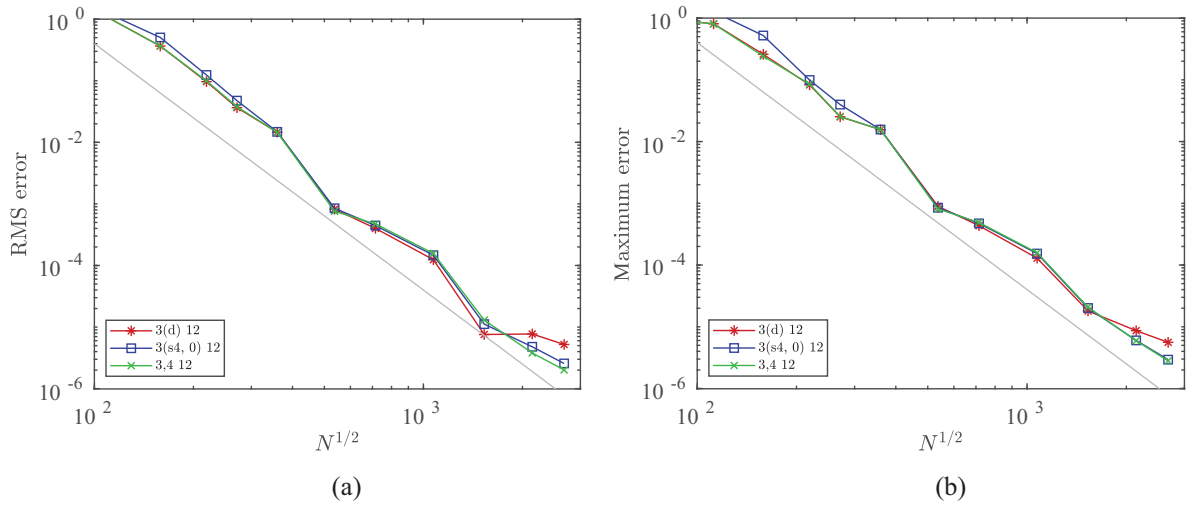

FIGURE 10 As Figure 9, but with subsampling.

Figure 9 shows these estimated errors for a degree-3 element with 12 nodes per element, 3 vertices, 2 interior nodes per edge, and 3 interior nodes instead of 1 to avoid accuracy loss during mass lumping (Crouzeix & Raviart, 1973; Cohen et al., 2001). The fourth-order time stepping scheme serves as reference. The Fourier-based dispersion correction reaches almost the same accuracy with second-order time stepping, as does the series approach with order 4, using $k_{\max} = \text{ceil}(M_x/2)$ rather than $k_{\max} = \text{ceil}((M_x + 1)/2)$ with $M_x = 4$. No extra points appear to be needed for the difference stencil.

Figure 10 displays similar results when subsampling is used. The wavelet was preprocessed without subsampling and only the dispersion correction of the data used a larger sampling time interval, equal to the largest integer multiple of the computational time step not exceeding 2 ms.

The wavelet

The smoothness of the wavelet affects the accuracy of the time stepping scheme. For instance, the wavelet $w(t) = [4(t/T_w)(1 - t/T_w)]^4$ for $t \in [0, T_w]$ and zero otherwise has a discontinuous fourth derivative at $t = 0$ and $t = T_w$ and the fifth derivative does not exist at those points. The time stepping error will not be better than third-order with a high-order time stepping scheme. Dispersion correction with the series method and its high derivatives will break down and perform far worse than the Fourier approach unless the finite-difference stencil is widened with sufficiently many points.

For the series method to be useful, a wavelet should be chosen that is sufficiently many times differentiable, or analytic like a Ricker wavelet. If the wavelet is given by a table,

interpolation with a spline of sufficiently high order, beyond that of the usual cubic B-spline, is recommended.

A hybrid method combines the Fourier approach to add wavelet dispersion with the series approach to remove it from the data traces. The Fourier approach has a higher computational cost but is only applied on a single time series. Test results for the hybrid method were of mixed quality. Although it is quicker to test various parameter setting without rerunning the simulation, the resulting accuracy was not very good, in particular on finer meshes.

CONCLUSIONS

An alternative implementation of temporal dispersion correction for the numerical simulation of wave propagation with second-order explicit time stepping is based on a series expansion and only requires higher time derivatives of the recorded traces to determine the correction terms. High-order finite differencing can be sensitive to numerical round-off errors and may produce noisy results, but this can be suppressed by adding points to the standard finite-difference stencil.

The effectiveness of the method was studied for the acoustic wave equation on a homogeneous problem with an exact solution and an inhomogeneous problem with topography. The results show that the same accuracy can be obtained as with higher-order time stepping, similar to that obtained with Fourier transforms, but a lower compute cost, which is linear instead of quadratic in the number of samples. The cost of the latter, however, can be lowered by using a non-uniform fast Fourier transform.

The series method, though somewhat less robust than the Fourier approach, has the advantage that it can apply the dispersion correction on the fly while doing the time stepping. The range of required time samples can be kept in a buffer. When the subsampling time is reached, the dispersion correction can be applied and the result stored, after which the buffer has to be refilled for the next pass. The Fourier approach requires the whole time series and has to be applied at the end of the computation. In addition, it requires the simulation to be carried out beyond the desired maximum time to allow for a taper at the end of the traces. The series approach was run as well for a longer time to avoid the implementation of one-sided difference operators, but the number of additional samples is smaller, equal to half the finite-difference stencil length.

DATA AVAILABILITY STATEMENT

Data sharing is not applicable to this article as no new data were created or analysed in this study. However, if the computational results shown in the Figures are considered as new data, then the author elects not to share those. Supporting software can be found at https://github.com/Someone789/dispcor.

ORCID


W. A. Mulder https://orcid.org/0000-0001-7020-9297

REFERENCES

Anderson, J.E., Brytik, V. & Ayeni, G. (2015) Numerical temporal dispersion corrections for broadband temporal simulation, RTM and FWI. In: *SEG Technical Program Expanded Abstracts 2015*. Houston, TX: Society of Exploration Geophysicists, pp. 1096–1100.

Bell, E.T. (1934) Exponential polynomials. *Annals of Mathematics*, 35(2), 258–277.

Cohen, G., Joly, P., Roberts, J.E. & Tordjman, N. (2001) Higher order triangular finite elements with mass lumping for the wave equation. *SIAM Journal on Numerical Analysis*, 38(6), 2047–2078.

Comtet, L. (1974) *Advanced combinatorics: The art of finite and infinite expansions*. Dordrecht, the Netherland: Reidel.

Crouzeix, M. & Raviart, P.-A. (1973) Conforming and non conforming finite element methods for solving the stationary Stokes equations. *R.A.I.R.O.* 7(R3), 33–75.

Dablain, M.A. (1986) The application of high-order differencing to the scalar wave equation. *Geophysics*, 51(1), 54–66.

Dai, N., Wu, W. & Liu, H. (2014) Solutions to numerical dispersion error of time FD in RTM. In *SEG Technical Program Expanded Abstracts 2014*. Houston, TX: Society of Exploration Geophysicists, pp. 4027–4031.

Duijndam, A.J.W. & Schonewille, M.A. (1999) Nonuniform fast Fourier transform. *Geophysics*, 64(2), 539–551.

Dutt, A. & Rokhlin, V. (1993) Fast Fourier transforms for nonequispaced data. *SIAM Journal on Scientific Computing*, 14(6), 1368–1393.

Feichtinger, H.G., Gröchenig, K. & Strohmer, T. (1995) Efficient numerical methods in non-uniform sampling theory. *Numerische Mathematik*, 69(4), 423–440.

Fornberg, B. (1987) The pseudospectral method: Comparisons with finite differences for the elastic wave equation. *Geophysics*, 52(4), 483–501.

Geevers, S., Mulder, W.A. & van der Vegt, J.J.W. (2019) Efficient quadrature rules for computing the stiffness matrices of mass-lumped tetrahedral elements for linear wave problems. *SIAM Journal on Scientific Computing*, 41(2), A1041–A1065.

Koene, E.F.M., Robertsson, J.O.A., Broggini, F. & Andersson, F. (2018) Eliminating time dispersion from seismic wave modeling. *Geophysical Journal International*, 213(1), 169–180.

Lax, P. & Wendroff, B. (1960) Systems of conservation laws. *Communications on Pure and Applied Mathematics*, 31(2), 217–237.

Li, Y.E., Wong, M. & Clapp, R. (2016) Equivalent accuracy at a fraction of the cost: overcoming temporal dispersion. *Geophysics*, 81(5), T189–T196.

Matlab (2021) MATLAB version 9.10 (R2021a) Update 4. Natick, MA: The MathWorks Inc.

McDevitt, T.J. (2012) Discrete Lanczos derivatives of noisy data. *International Journal of Computer Mathematics*, 89(7), 916–931.

Mises, R.V. & Pollaczek-Geiringer, H. (1929) Praktische Verfahren der Gleichungsauflösung. *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik*, 9(2), 152–164.

Mittet, R. (2017) On the internal interfaces in finite-difference schemes. *Geophysics*, 82(4), T159–T182.

Mulder, W.A. (1996) A comparison between higher-order finite elements and finite differences for solving the wave equation. In Désidéri, J.-A., LeTallec, P., Oñate, E., Périaux, J. & Stein, E. (Eds.) *Proceedings of the second ECCOMAS conference on numerical methods in engineering*. (pp. 344–350), Chichester, UK: John Wiley & Sons.

Mulder, W.A. (2013) New triangular mass-lumped finite elements of degree six for wave propagation. *Progress in Electromagnetics Research*, 141, 671–692.

Mulder, W.A. (2022) More continuous mass-lumped triangular finite elements. *Journal of Scientific Computing*, 92(38), 1–22.

Mulder, W.A. (2023) Performance of old and new mass-lumped triangular finite elements for wavefield modelling. *Geophysical Prospecting*. early online. https://doi.org/10.1111/1365-2478.13383

Potter, S.F., Gumerov, N.A. & Duraiswami, R. (2017) Fast interpolation of bandlimited functions. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. Piscataway, NJ: IEEE, pp. 4516–4520.

Potts, D., Steidl, G. & Tasche, M. (2001) Fast Fourier transforms for nonequispaced data: a tutorial. In: Benedetto, J.J., Ferreira, P.J.S.G. (Eds.) *Modern sampling theory. Mathematics and Applications. Series; Applied and numerical harmonic analysis*. Boston, MA: Birkhäuser, pp. 247–270. DOI: https://doi.org/10.1007/978-1-4612-0143-4

Qin, Y., Quiring, S. & Nauta, M. (2017) Temporal dispersion correction and prediction by using spectral mapping. In: *79th EAGE Conference & Exhibition, Paris, France, Extended Abstracts*. Houten, the Netherlands: European Association of Geoscientists & Engineers, Th P1 10

Shubin, G.R. & Bell, J.B. (1987) A modified equation approach to constructing fourth order methods for acoustic wave propagation. *SIAM Journal on Scientific and Statistical Computing*, 8(2), 135–151.

Stork, C. (2013) Eliminating nearly all dispersion error from FD modeling and RTM with minimal cost increase. In: *75th EAGE Conference & Exhibition incorporating SPE EUROPEC, Extended Abstract*. Houten, the Netherlands: European Association of Geoscientists & Engineers, cp-348-01021.

Trefethen, L.N. & Weideman, J.A.C. (2014) The exponentially convergent trapezoidal rule. *SIAM Review*, 56(3), 385–458.

von Kowalevsky, S. (1875) Zur Theorie der partiellen Differentialgleichung. *Journal für die reine und angewandte Mathematik*, 80, 1–32.

Wang, M. & Xu, S. (2015a) Finite-difference time dispersion transforms for wave propagation. *Geophysics*, 80(6), WD19–WD25.

Wang, M. & Xu, S. (2015b) Time dispersion transforms in finite difference of wave propagation. In *77th EAGE Conference & Exhibition, Extended Abstract, Volume 2015*. Houten, the Netherlands: European Association of Geoscientists & Engineers, pp. 1–5.

Xu, Z., Jiao, K., Cheng, X., Sun, D., King, R., Nichols, D. & Vigh, D. (2017) Time-dispersion filter for finite-difference modeling and reverse time migration. In *SEG Technical Program Expanded Abstracts 2017*. Houston, TX: Society of Exploration Geophysicists, pp. 4448–4452.

Zhebel, E., Minisini, S., Kononov, A. & Mulder, W.A. (2014) A comparison of continuous mass-lumped finite elements with finite differences for 3-D wave propagation. *Geophysical Prospecting*, 62(5), 1111–1125.

How to cite this article: Mulder, W.A. (2024) Temporal dispersion correction for wave-propagation modelling with a series approach. *Geophysical Prospecting*, 72, 301–314. https://doi.org/10.1111/1365-2478.13411

APPENDIX: DERIVATION

The partial Bell (1934) polynomials $B_{k,\ell}(x_1, \ldots, x_{k+1-\ell})$ are defined by (e.g., Comtet, 1974, section 3.3)

$$\Phi(\xi, u) = \exp\left(u \sum_{j=1}^{\infty} x_j \frac{\xi^j}{j!}\right) = 1 + \sum_{k=1}^{\infty} \frac{\xi^k}{k!} \sum_{\ell=1}^{k} u^\ell B_{k,\ell}(x_1, \ldots, x_{k+1-\ell}). \tag{A.1}$$

They obey the recurrence relation

$$B_{k,\ell} = \sum_{j=1}^{k+1-\ell} \binom{k-1}{j-1} x_j B_{k-j,\ell-1}, \tag{A.2}$$

starting from $B_{0,0} = 1$ as well as $B_{m,0} = 0$ and $B_{0,m} = 0$ for $m \geq 1$. The arguments of $B_{k,\ell}(x_1, \ldots, x_{k+1-\ell})$ have been dropped for brevity. In particular, $B_{k,1} = x_k$ and $B_{k,k} = x_1^k$ for $k \geq 1$.

The problem at hand has $u = -\mathrm{i}\omega t$, $\xi = \omega \Delta t/2$ and $x_j = f^{(j)}(0)$. Applying the expansion (A.1) to (11) and (14), while using $f(0) = 1$ and $\delta(t' - t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \mathrm{d}\omega \exp[\mathrm{i}\omega(t' - t)]$, provides

$$\tilde{u}(t') = \frac{1}{2\pi} \int_{-\infty}^{\infty} \mathrm{d}\omega\, \mathrm{e}^{\mathrm{i}\omega t'} \int_{-\infty}^{\infty} \mathrm{d}t\, \mathrm{e}^{-\mathrm{i}\omega t\, f(\omega \Delta t/2)} u(t) = u(t') + v(t'), \tag{A.3}$$

with

$$v(t') = \frac{1}{2\pi} \int_{-\infty}^{\infty} \mathrm{d}\omega \int_{-\infty}^{\infty} \mathrm{d}t\, \mathrm{e}^{\mathrm{i}\omega(t'-t)} u(t) \sum_{k=1}^{\infty} \frac{(\omega \Delta t/2)^k}{k!} \sum_{\ell=1}^{k} (-\mathrm{i}\omega t)^\ell B_{k,\ell}. \tag{A.4}$$

Now

$$\begin{aligned}\frac{1}{2\pi}\int_{-\infty}^{\infty}\mathrm{d}\omega\int_{-\infty}^{\infty}\mathrm{d}t\,\mathrm{e}^{\mathrm{i}\omega(t'-t)}\omega^k(-\mathrm{i}\omega t)^\ell u(t)\\ =\frac{(-1)^\ell(-\mathrm{i})^k}{2\pi}\int_{-\infty}^{\infty}\mathrm{d}\omega\int_{-\infty}^{\infty}\mathrm{d}t\,t^\ell u(t)\partial_{t'}^{k+\ell}\mathrm{e}^{\mathrm{i}\omega(t'-t)}\\ =(-1)^{k+\ell}\,\mathrm{i}^k\int_{-\infty}^{\infty}\mathrm{d}t\,t^\ell u(t)\,\delta^{(k+\ell)}(t'-t)\\ =\mathrm{i}^k\partial_t^{k+\ell}\left[t^\ell u(t)\right]\Big|_{t=t'}.\end{aligned} \tag{A.5}$$

With that, Equation (A.3) becomes

$$\tilde{u}(t)=u(t)+\sum_{k=1}^{\infty}\frac{(\mathrm{i}\Delta t/2)^k}{k!}\sum_{\ell=1}^{k}B_{k,\ell}\,\partial_t^{k+\ell}\left[t^\ell u(t)\right]. \tag{A.6}$$

In the forward case, which adds dispersion,

$$f(\xi)=\frac{\sin(\xi)}{\xi}=\sum_{j=0}^{\infty}\frac{(-1)^j}{(2j+1)!}\xi^{2j},$$

$f(0)=1$, and

$$x_{2j}=f^{(2j)}(0)=\frac{(-1)^j}{2j+1},\quad x_{2j+1}=0,\quad j\ge 0. \tag{A.7}$$

Equation (A.6) becomes

$$\begin{aligned}\tilde{u}(t)&=u(t)+\sum_{k=1}^{\infty}\frac{(\Delta t/2)^{2k}}{(2k+1)!}r_k(t),\\ r_k&=\sum_{\ell=1}^{k}(-1)^\ell a_{k,\ell}\partial_t^{2k+\ell}[t^\ell u(t)],\end{aligned} \tag{A.8a}$$

where

$$a_{k,\ell}=(-1)^k(2k+1)B_{2k,\ell}(x_1,\ldots,x_{2k+1-\ell}), \tag{A.8b}$$

with x_m from Equation (A.7) and $1\le\ell\le k$. The coefficients $a_{k,\ell}$ are rescaled to obtain $a_{k,1}=1$. Values of $a_{k,\ell}$ up to $k=10$ are listed below. The factor in front of $r_k(t)$ in Equation (A.8a) appears in the series expansion

$$\frac{\sinh(\Delta t/2)}{\Delta t/2}=\sum_{k=1}^{\infty}\frac{(\Delta t/2)^{2k}}{(2k+1)!}. \tag{A.9}$$

In the reverse case, removing dispersion,

$$f(\xi)=\frac{\arcsin(\xi)}{\xi}=\sum_{j=0}^{\infty}\frac{(2j)!}{2^{2j}(2j+1)(j!)^2}\xi^{2j}, \tag{A.10}$$

$f(0)=1$, and

$$x_{2j}=f^{(2j)}(0)=\frac{[\,(2j)!\,]^2}{2^{2j}(2j+1)(j!)^2},\quad x_{2j+1}=0,\quad j\ge 0. \tag{A.11}$$

If Equation (A.6) is written as

$$\begin{aligned}u(t)&=\tilde{u}(t)+\sum_{k=1}^{\infty}(-1)^k\frac{(\Delta t/2)^{2k}(2k)!}{2^{2k}(2k+1)(k!)^2}s_k(t),\\ s_k&=\sum_{\ell=1}^{k}(-1)^\ell b_{k,\ell}\partial_t^{2k+\ell}[t^\ell\tilde{u}(t)],\end{aligned} \tag{A.12a}$$

then

$$b_{k,\ell}=(2k+1)\left[\frac{k!2^k}{(2k)!}\right]^2 B_{2k,\ell}(x_1,\ldots,x_{2k+1-\ell}), \tag{A.12b}$$

with x_m from Equation (A.11), $1\le\ell\le k$ and rescaled such that $b_{k,1}=1$. Note that

$$\frac{\operatorname{arcsinh}(\Delta t/2)}{\Delta t/2}=\sum(-1)^k\frac{(\Delta t/2)^{2k}(2k)!}{2^{2k}(2k+1)(k!)^2}. \tag{A.13}$$

The recurrence relation (A.2) can be simplified when $x_{2j+1}=0$ for $j\ge 0$. First, consider an odd $k=2k'+1, k'\ge 0$ and let $j=2j'$. Then, $B_{2k'+1,1}=x_{2k'+1}=0$, $B_{2k'+1,2k'+1}=x_1^{2k'+1}=0$, and

$$B_{2k'+1,\ell}=\sum_{j'=1}^{\lfloor k'+1-\ell/2\rfloor}\binom{2k'}{2j'-1}x_{2j'}B_{2(k'-j')+1,\ell-1}, \tag{A.14}$$

showing that $B_{2k'+1,\ell}=0$ for $k'\ge 0$ and $0\le\ell\le 2k'+1$. Here, $k=\lfloor a\rfloor$ denotes the largest integer $k\le a$, sometimes expressed as $\mathrm{floor}(a)$, for a real number a.

For the even case, let $k=2k'$, $k'\ge 1$ and $j=2j'$. Then,

$$B_{2k',\ell}\sum_{j'=1}^{\lfloor k'-(\ell-1)/2\rfloor}\binom{2k'-1}{2j'-1}x_{2j'}B_{2(k'-j'),\ell-1}, \tag{A.15}$$

which now only involves $B_{k,\ell}$ with even k.

Equation [3m] in Section 3.3 of Comtet (1974) reads

$$\begin{aligned}B_{k,k-m}(x_1,x_2,\ldots,x_{k+1-(k-m)})=\\ \sum_{j=m+1}^{2m}\binom{k}{j}x_1^{k-j}B_{j,j-m}(0,x_2,\ldots,x_{j+1-(j-m)}).\end{aligned} \tag{A.16}$$

The implied condition $j\le k$ leads to $m\le\lfloor k/2\rfloor$. If $x_1=0$ and $k=2k'$, this shows that $B_{2k',\ell}=0$ for $\ell>k'$. With that, the recurrence relation (A.15) can be rewritten in terms of

$\tilde{B}_{k',\ell} = B_{2k',\ell}$ as

$$\tilde{B}_{k',\ell} = \sum_{j'=1}^{k'+1-\ell} \binom{2k'-1}{2j'-1} x_{2j'}\, \tilde{B}_{k'-j',\ell-1}, \quad 1 \le \ell \le k', \tag{A.17}$$

starting from $\tilde{B}_{0,0} = 1$, $\tilde{B}_{m,0} = \tilde{B}_{0,m} = 0$ for $m \ge 1$, and using $k = 2k'$ and $j = 2j'$ as before.

The coefficients $a_{k,\ell}$ up to $k_{\max} = 10$ or $(\Delta t)^{20}$ are elements of the lower triangular matrix

$$A = \begin{pmatrix}
1 & & & & & & & & & \\
1 & \frac{5}{3} & & & & & & & & \\
1 & 7 & \frac{35}{9} & & & & & & & \\
1 & \frac{123}{5} & 42 & \frac{35}{3} & & & & & & \\
1 & \frac{253}{3} & 341 & \frac{770}{3} & \frac{385}{9} & & & & & \\
1 & \frac{2041}{7} & \frac{38324}{15} & \frac{11869}{3} & \frac{5005}{3} & \frac{5005}{27} & & & & \\
1 & 1023 & 18759 & \frac{484484}{9} & \frac{130130}{3} & \frac{35035}{3} & \frac{25025}{27} & & & \\
1 & \frac{32759}{9} & \frac{413882}{3} & \frac{31384873}{45} & \frac{25967942}{27} & \frac{4254250}{9} & \frac{2382380}{27} & \frac{425425}{81} & & \\
1 & \frac{65531}{5} & \frac{107585809}{105} & \frac{133275614}{15} & \frac{297217817}{15} & \frac{142908766}{9} & \frac{47205158}{9} & \frac{6466460}{9} & \frac{8083075}{243} & \\
1 & \frac{524277}{11} & 7704576 & 112990891 & \frac{9860196129}{25} & 485601753 & \frac{6835048220}{27} & \frac{539949410}{9} & \frac{56581525}{9} & \frac{56581525}{243}
\end{pmatrix}.$$

For $b_{k,\ell}$, we have

$$B = \begin{pmatrix}
1 & & & & & & & & & \\
1 & \frac{5}{27} & & & & & & & & \\
1 & \frac{7}{25} & \frac{7}{405} & & & & & & & \\
1 & \frac{2067}{6125} & \frac{6}{175} & \frac{1}{945} & & & & & & \\
1 & \frac{4477}{11907} & \frac{4829}{99225} & \frac{22}{8505} & \frac{11}{229635} & & & & & \\
1 & \frac{150761}{373527} & \frac{990964}{16372125} & \frac{1261}{297675} & \frac{13}{93555} & \frac{13}{7577955} & & & & \\
1 & \frac{78103}{184041} & \frac{48787}{693693} & \frac{106684}{18243225} & \frac{314}{1216215} & \frac{1}{173745} & \frac{1}{19702683} & & & \\
1 & \frac{164187887}{372683025} & \frac{477888394}{6087156075} & \frac{430651327}{58530346875} & \frac{4836262}{12314176875} & \frac{9826}{820945125} & \frac{68}{351833625} & \frac{17}{13299311025} & & \\
1 & \frac{335021699}{738720125} & \frac{309336853817}{3621857864625} & \frac{1508301434}{172469422125} & \frac{531060317}{995015896875} & \frac{1390078}{69780335625} & \frac{31046}{69780335625} & \frac{76}{13956067125} & \frac{19}{678264862275} & \\
1 & \frac{10892077437}{23467660931} & \frac{169784}{1859715} & \frac{3936101287}{393230282445} & \frac{118566891343}{175549233234375} & \frac{15738043}{540151486875} & \frac{7036}{8741712375} & \frac{3646}{265165275375} & \frac{1}{7576150725} & \frac{1}{1841004626175}
\end{pmatrix}.$$

A Matlab (2021) code `abmatrix.m` is produced as an illustration. The script `abmatrixsym.m` provided the matrices above, using the Symbolic Math Toolbox.