J Eng Math (2007) 58:211–228
DOI 10.1007/s10665-006-9108-4

ORIGINAL PAPER

On the accuracy of finite-difference solutions for nonlinear water waves

Harry B. Bingham · Haiwen Zhang

Received: 6 February 2006 / Accepted: 16 October 2006 / Published online: 24 November 2006

Abstract This paper considers the relative accuracy and efficiency of low- and high-order finite-difference discretisations of the exact potential-flow problem for nonlinear water waves. The method developed is an extension of that employed by Li and Fleming (Coastal Engng 30: 235–238, 1997) to allow arbitrary-order finite-difference schemes and a variable grid spacing. Time-integration is performed using a fourth-order Runge–Kutta scheme. The linear accuracy, stability and convergence properties of the method are analysed and high-order schemes with a stretched vertical grid are found to be advantageous relative to second-order schemes on an even grid. Comparison with highly accurate periodic solutions shows that these conclusions carry over to nonlinear problems and that the advantages of high-order schemes improve with both increasing nonlinearity and increasing accuracy tolerance. The combination of non-uniform grid spacing in the vertical and fourth-order schemes is suggested as optimal for engineering purposes.

Keywords Accuracy · Convergence · Finite-difference methods · Nonlinear waves · Stability

1 Introduction

Marine, coastal and ocean engineers require a means for predicting the propagation of nonlinear water waves and their subsequent interaction with fixed or floating structures. The physics of most such problems is well described by the incompressible Navier–Stokes equations, but direct numerical solution of these equations is limited to very small Reynolds number flows (Re $= O(1000)$.) The introduction of Reynolds-averaging to obtain the Reynolds-Averaged Navier–Stokes Equations (RANSE) and a turbulence closure model allows solutions to be obtained at realistic Reynolds numbers. Although progress has been rapid over the past several decades, the computational effort required by RANSE solvers still imposes severe limitations on domain size and resolution. The next level of approximation is to neglect viscosity and

H. B. Bingham (✉)
Mechanical Engineering, Technical University of Denmark,
DK-2800, Lyngby, Denmark
e-mail: hbb@mek.dtu.dk

H. Zhang
Marine & Foundation Eng., COWI A/S, DK-2800, Lyngby, Denmark
e-mail: hazh@cowi.dk

assume an irrotational flow to obtain a potential-flow problem governed by the Laplace equation (see e.g. [1] for a detailed statement of the problem).

A large number of methods exist for solving the exact Laplace problem for surface gravity waves or some approximation to it. The most widespread method for treating wave–body-interaction problems is the Boundary-Integral-Equation Method (BIEM) which uses Green's theorem to project the three-dimensional (3D) problem onto the boundary surface of the fluid volume. The most mature of these methods are based on a perturbation expansion in wave steepness and the linear free-surface Green function which reduces the computational domain to the structure surface and a small nearby portion of the free surface. Such methods routinely provide second-order solutions which are in widespread use for the design of offshore structures (see e.g. [2,3]). Higher order in nonlinearity can in principle be obtained using the free-space Rankine Green function and larger portions of the free surface (e.g. [4–7]).

Numerous other methods have been devised for projecting the problem onto some portion of the fluid boundary and thereby reducing the computational dimension of the problem by one. Some popular examples are pseudo-spectral methods e.g. [8,9]; and Boussinesq-type methods which assume a polynomial expansion in the vertical e.g. [10,11]. Such methods are relatively efficient, but less flexible in treating wave–body interactions or domains with complicated boundaries.

An obvious alternative to surface projection methods is a direct numerical solution of the 3D Laplace problem, which is perhaps less common, but on the increase. The finite-volume methodology which is widely used by RANSE solvers is readily applied to the simpler Laplace problem, either on its own or as an outer domain surrounding an interior RANSE domain (see e.g. [12]). The finite-element method has also been applied to the problem e.g. in [13], and a finite-difference-based solution was developed in [14].

This paper is motivated by two observations related to the solution of the exact Laplace problem for nonlinear wave–body interaction. First, projection methods exchange degrees of freedom (discretising the internal fluid volume) for complexity (expansions, or Green-function interactions). This exchange is clearly advantageous for perturbation solutions, but the advantage is less obvious for the fully nonlinear problem. In this case so much of the boundary requires discretisation that it is possible for 3D methods to be competitive due to their relative simplicity and inherent ease of obtaining an optimal scaling of the solution effort. The second observation is that most, if not all, existing direct Laplace solvers are second-order accurate (at best), while significant evidence exists for the advantage of using higher-order schemes.

Thirty years ago, Kreiss and Oliger [15] investigated finite-difference solutions to the linear wave equation and showed that fourth-order was optimal in some sense. Experience with finite-difference solutions to Boussinesq-type equations [16] also shows a significant advantage to using fourth-order schemes. Here we extend the method of [14] as follows. Arbitrary-order finite-difference schemes are used to discretise the continuous derivatives; and while we retain a structured grid, we allow for a non-uniform grid spacing to allow clustering of grid points where desired. A fourth-order Runge–Kutta scheme is used for the time integration. The implementation is only in two dimensions (2D) at this point, and here we find that the Generalised Minimal RESidual (GMRES) Krylov subspace iterative method preconditioned by the linearised, second-order version of the matrix leads to optimal scaling of the solution effort (i.e., $O(N)$ where N is the total number of grid points). Reduction of the residual by seven orders of magnitude is achieved in ≈ 10 iterations nearly independently of problem size and the order of the finite-difference schemes used. In 3D we expect that some form of multigrid will be optimal and this is the topic of ongoing research.

After a description of the theory and implementation of the method in Sects. 2 and 3, the linear convergence, accuracy and stability properties are established by standard Fourier analysis in Sect. 4. In Sect. 5 the nonlinear accuracy of the method is quantified by comparison with periodic solutions based on stream-function theory [17]. From this we conclude that high-order schemes and a stretched vertical grid are more efficient than second-order schemes to achieve a target accuracy of solution. This advantage improves with increased nonlinearity and with decreased error tolerance. Fourth-order schemes are suggested as being an optimal balance of accuracy and complexity for engineering purposes and some sample calculations for wave–bottom interaction are presented in Sect. 6. Conclusions are drawn in Sect. 7.

2 Formulation

Consider the irrotational flow of an incompressible inviscid fluid. A Cartesian coordinate system is adopted, with origin on the still water plane and the z-axis pointing vertically upwards, $\mathbf{x} = [x, y]$ is a horizontal vector and t is time. The fluid domain is bounded by the sea bottom at $z = -h(\mathbf{x})$ and the free surface at $z = \eta(\mathbf{x}, t)$. Both η and h are assumed to be single-valued functions of $\mathbf{x}$. Following [18], we express the kinematic and dynamic free-surface boundary conditions in terms of the velocity potential and the vertical component of velocity evaluated directly on the free surface: $\tilde{\phi} = \phi(\mathbf{x}, \eta, t)$, and $\tilde{w} = \left.\frac{\partial \phi}{\partial z}\right|_{z=\eta}$

$$\eta_t = -\nabla\eta\cdot\nabla\tilde{\phi} + \tilde{w}(1 + \nabla\eta\cdot\nabla\eta) \tag{1a}$$

$$\tilde{\phi}_t = -g\,\eta - \frac{1}{2}\nabla\tilde{\phi}\cdot\nabla\tilde{\phi} + \frac{1}{2}\tilde{w}^2(1 + \nabla\eta\cdot\nabla\eta). \tag{1b}$$

Here $\nabla = \left[\partial/\partial x, \partial/\partial y\right]$ is the horizontal gradient operator, g the gravitational acceleration and partial differentiation is indicated when the independent variables appear as subscripts. These relations can be viewed as evolution equations for η and $\tilde{\phi}$ to be integrated forward in time from initial conditions. The horizontal gradients appearing on the right-hand sides can be computed immediately, but obtaining the vertical component of velocity $\tilde{w}$, requires a means of satisfying the Laplace equation throughout the depth of the fluid along with the kinematic bottom boundary condition:

$$\nabla^2\phi + \phi_{zz} = 0, \quad -h < z < \eta \tag{2a}$$

$$\phi_z + \nabla h\cdot\nabla\phi = 0, \quad z = -h. \tag{2b}$$

For a direct solution of this Laplace problem, it is convenient to make a change of variables in the vertical coordinate (the widely used σ-transformation) defined by

$$\sigma(\mathbf{x}, z, t) = \frac{z + h(\mathbf{x})}{\eta(\mathbf{x}, t) + h(\mathbf{x})} = \frac{z + h(\mathbf{x})}{d(\mathbf{x}, t)}, \tag{3}$$

where the total thickness of the fluid layer $d = \eta + h$ has been introduced. This transformation converts the Laplace problem to

$$\nabla^2\phi + \nabla^2\sigma\,\phi_\sigma + 2\nabla\sigma\cdot\nabla\phi_\sigma + \left(\nabla\sigma\cdot\nabla\sigma + \sigma_z^2\right)\phi_{\sigma\sigma} = 0, \quad 0 < \sigma < 1, \tag{4a}$$

$$(\sigma_z + \nabla h\cdot\nabla\sigma)\,\phi_\sigma + \nabla h\cdot\nabla\phi = 0, \quad \sigma = 0, \tag{4b}$$

where the derivatives of σ can be expressed as

$$\nabla\sigma = (1 - \sigma)\frac{\nabla h}{d} - \sigma\frac{\nabla\eta}{d}, \tag{5}$$

$$\nabla^2\sigma = \frac{1 - \sigma}{d}\left(\nabla^2 h - \frac{\nabla h\cdot\nabla h}{d}\right) - \frac{\sigma}{d}\left(\nabla^2\eta - \frac{\nabla\eta\cdot\nabla\eta}{d}\right) - \frac{1 - 2\sigma}{d^2}\nabla h\cdot\nabla\eta - \frac{\nabla\sigma}{d}\cdot(\nabla h + \nabla\eta), \tag{6}$$

$$\sigma_z = \frac{1}{d}, \tag{7}$$

which illustrates the specific derivatives of h and η required for their evaluation. After solving (4) for the potential $\phi(\mathbf{x}, \sigma)$, the vertical component of fluid velocity on the free surface is given by

$$\tilde{w} = \frac{1}{d}\,\phi_\sigma|_{\sigma=1}, \tag{8}$$

which allows (1) to be stepped forward in time, closing the problem. If the internal kinematics of the flow are desired, they can also be computed from $\phi(\mathbf{x}, \sigma)$, for example the fluid velocities are given by

$$u(\mathbf{x}, z) = \phi_x(\mathbf{x}, z) = \phi_x(\mathbf{x}, \sigma) + \sigma_x\phi_\sigma(\mathbf{x}, \sigma), \tag{9}$$

$$v(\mathbf{x}, z) = \phi_y(\mathbf{x}, z) = \phi_y(\mathbf{x}, \sigma) + \sigma_y\phi_\sigma(\mathbf{x}, \sigma), \tag{10}$$

$$w(\mathbf{x}, z) = \phi_z(\mathbf{x}, t) = \frac{1}{d}\phi_\sigma(\mathbf{x}, \sigma). \tag{11}$$

3 Numerical solution

For the time-integration of (1) we employ the classical explicit four-stage, fourth-order Runge–Kutta scheme (see e.g. [19]). For the spatial discretisation, a grid of N_x points is defined along the x-axis at which the time-stepping variables η and $\tilde{\phi}$ are to be evaluated. Periodic or Neumann (specified normal component of velocity) conditions are imposed at the horizontal boundaries. For the solution of (4a) and (4b), N_z points are defined in the vertical under each horizontal grid point, arbitrarily spaced between $\sigma = 0$ and $\sigma = 1$ (the same vertical spacing is, however, chosen under each horizontal grid point.) Choosing r nearby points, allows order $(r-1)$ finite-difference schemes for the first and second derivatives in x and σ to be developed in the standard way by means of Taylor series expansion and the inversion of a small linear system at each grid point.

3.1 Finite-difference discretisations

For the one-dimensional first and second derivatives in x and σ, $r = \alpha + \beta + 1$ points are used where α indicates the number of points to the right (or top) and β the number of points to the left (or bottom) of the point of interest. With periodic conditions all x-derivatives are centrally discretised with $\alpha = \beta$, while for Neumann conditions only points within the computational domain are used leading to off-centred schemes near the boundaries. For σ-derivatives off-centred schemes are used near the bottom and free surface. For the mixed $x\sigma$-derivative, a full square stencil of r^2-points is used and the schemes are thus off-centred near non-periodic boundaries. In this way, all derivatives are formally accurate to $O(\Delta x_*^{r-1})$ where Δx_* is the maximum grid spacing, and this is verified in Sect. 4.

The resultant discretisation of (4a) is then applied at all non-boundary grid points, while the appropriate boundary conditions are imposed at boundary points. This gives a rank $N = N_x N_z$ linear system of equations

$$\mathbf{A}[\phi] = [b], \tag{12}$$

where $\mathbf{A}$ is the matrix of coefficients, $[\phi]$ a vector of the unknown ϕ at each grid point and $[b]$ a vector holding zeros, except at those points corresponding to inhomogeneous boundary conditions.

3.2 Iterative solution of the linear system

The matrix $\mathbf{A}$ in (12) is sparse and contains a maximum of r^2N nonzero elements. A direct solution to this system using sparse-matrix techniques is effective at small values of N but is not competitive (even in 2D) as N increases. This is especially true for higher-order discretisations which give more non-zeros per row and more spreading in the entries, thus leading to extensive fill-in when factoring $\mathbf{A}$, and this quickly becomes prohibitive both in terms of memory use and computational effort. An iterative solution, properly preconditioned, turns out to be much more effective. We employ here the GMRES (Generalised Minimal RESidual) method of [21], preconditioned on the left using the linearised, second-order-accurate version of $\mathbf{A}$. Specifically, we set $\eta = 0$ in (3) which makes σ independent of time and removes all functions of η from the derivatives in (5–7). Second-order finite-difference schemes ($r = 3$) are then used to discretise the system as described above to produce the preconditioning matrix $\mathbf{M}$. The preconditioning step then consists of solving a linear system of the form $\mathbf{M}[q] = [s]$.

Since $\mathbf{M}$ is time-constant, it is built and LU factored only once after which preconditioning requires only a back-substitution step which is generally 10–100 times faster than a factorisation for this problem. As shown in Sect. 5, the iteration count to achieve a relative convergence tolerance of 10^{-7} when using this preconditioner is generally 5–15 and independent of N and kh, although it increases slowly with r

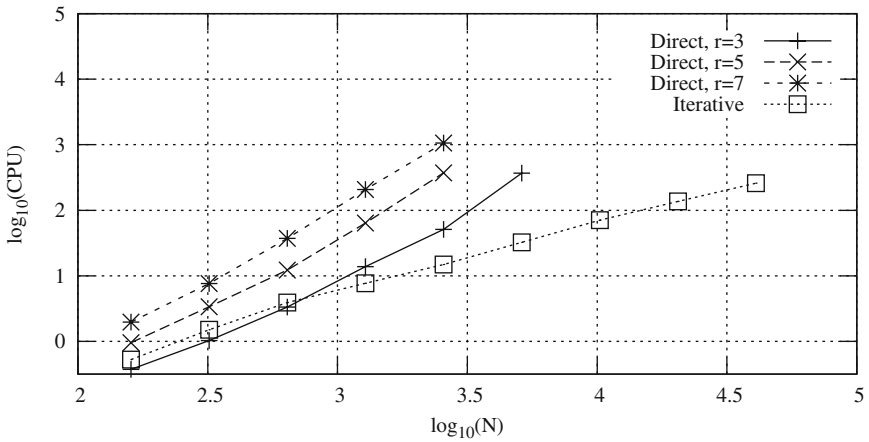

Fig. 1 Scaling of the computational effort for direct and iterative solutions

(approximately a 50% increase for $r = 3$ to $r = 5$). Factorisation and back-substitution is performed using the `MA41` package from the Harwell Subroutine Library. These routines are a potentially parallel sparse multi-frontal variant of Gaussian elimination, which is particularly effective on matrices with a nearly symmetric pattern. The method chooses pivots from the diagonal, using the approximate minimum-degree algorithm of [22]. When solving systems with a single right-hand side, the routine also makes efficient use of level 2 Basic Linear Algebra Subprograms (BLAS), which have been optimised using the Automatically Tuned Linear Algebra Software (ATLAS, see e.g. [23]). For further details on this routine see [20] and references therein. In this work we have only used the serial version of the code.

An example of the scaling of the solution effort for both the direct and the iterative solutions is plotted in Fig. 1, which shows the calculation time to propagate a steady nonlinear wave of approximately 80% of the limiting steepness at $kh = \pi$ for 100 time-steps. The log of the calculation time on a 3.2 GHz Pentium processor with 1 gigabyte of RAM memory is plotted vs. the log of N for the direct solution using second-, fourth-, and sixth-order discretisations as well as the sixth-order discretisation solved iteratively. Using the iterative solution, the curves for different order discretisations are indistinguishable on the scale of this plot. While the direct solutions can be seen to tend towards a super-linear scaling of the effort, the iterative solution scales with precisely N.

While this preconditioning strategy is expected to work equally well in three dimensions as far as iteration counts go, it is possible that the increased complexity of the linearised matrix in that case will result in a less than optimal scaling when compared to a multigrid solver. This topic is under investigation and will be discussed in a follow-up paper.

4 Linear accuracy and stability of the discrete solution

The first step in quantifying the performance of a given discretisation scheme is to consider the linearised version of the problem on a horizontal bottom:

$$\eta_t = \tilde{w}, \tag{13}$$

$$\tilde{\phi}_t = -g\,\eta, \tag{14}$$

$$\nabla^2\phi + \phi_{zz} = 0, \quad -h < z < 0, \tag{15}$$

$$\phi_z = 0, \quad z = -h, \tag{16}$$

where for the linear problem $\tilde{\phi} = \phi(x, 0, t)$ and $\tilde{w} = w(x, 0, t)$. This problem has the well-known travelling-wave solution

$$\eta = \mathfrak{Re}\left\{\frac{H}{2}\,\mathrm{e}^{\mathrm{i}(kx-\omega t)}\right\} \tag{17}$$

$$\phi = \mathfrak{Re}\left\{\frac{-\mathrm{i}\,g}{\omega}\frac{H}{2}\,\frac{\cosh\left[k(z+h)\right]}{\cosh\left(kh\right)}\,\mathrm{e}^{\mathrm{i}(kx-\omega t)}\right\} \tag{18}$$

where H is the wave height, $L = 2\pi/k$ the wave length, $T = 2\pi/\omega$ the wave period, and ω and k are related by the dispersion relation $\omega^2 = gk\tanh\left(kh\right)$.

4.1 Convergence

The essential ingredient of the discrete solution scheme is the evaluation of $\tilde{w}$ from $\tilde{\phi}$. To verify the convergence of the method, we compare the computed value to the exact result $\tilde{w} = k\tanh\left(kh\right)\tilde{\phi}$, for increasingly fine discretisations. To simplify the analysis, we consider a periodic domain in x with a uniform grid spacing Δx. This makes all x-grid points equivalent and allows this part of the problem to be expressed in closed form as a function of the x-resolution (number of grid points per wavelength.) For example, the second-order scheme $\frac{1}{\Delta x^2}(\phi_{i-1} - 2\phi_i + \phi_{i+1})$, where $\phi_i = \phi(x = i\Delta x)$ becomes

$$\frac{\partial^2}{\partial x^2} \to N_x^2\left(-2 + 2\cos\left(\frac{2\pi}{N_x}\right)\right), \tag{19}$$

where we have taken $L = N_x\,\Delta x$ so that N_x represents the number of grid points per wavelength. This is standard von Neumann (Fourier) analysis (e.g. [24]). Choosing a convenient wavelength, $L = 1$, fixes the wavenumber and choosing a value of kh then determines the depth h. Since the discretisation in the vertical is on a (possibly) variable grid and non-periodic, the rest of the problem is solved numerically. Choosing N_z points in the vertical thus produces a reduced version of (12) which is of rank N_z

$$\mathbf{B}\left[\phi\right] = \left[b\right]. \tag{20}$$

Numbering the grid points from the bottom to the free surface, $[b]$ is a vector of zeros except for the last entry which holds the known magnitude of $\tilde{\phi}$. $\mathbf{B}$ is the matrix holding the coefficients for $\partial/\partial z$ in the first row, those for $\partial^2/\partial z^2$ for rows $2 : N_z - 1$ (plus the discrete version of $\partial^2/\partial x^2$ on the diagonal) and finally a 1.0 on the last diagonal. After solving this system for ϕ at each vertical grid point, the vertical first-derivative scheme is applied to get the approximation for $\tilde{w}$ which can be compared to the exact result.

Figure 2 shows the convergence of the calculations for several choices of discretisation scheme at a relative water depth of $kh = 4$. The left-column plots result for a uniform grid using second-, fourth-, and sixth-order schemes ($r = 3, 5$, and 7 points, respectively), while the right-hand column shows the same order schemes applied to a stretched grid which clusters points towards the free surface. The stretched grid used here is defined by $\sigma_j = [\sin\left(\Delta\theta(j-1)\right)]$ where $\Delta\theta = \pi/[2(N_z - 1)]$. Each plot shows the relative error ($|\tilde{w} - \tilde{w}_e|/\tilde{w}_e$) versus N_z on a log-log scale with $\tilde{w}_e$ being the exact result. The different lines represent different values of N_x as shown in the legend on the first plot. An approximate value for the asymptotic slope of the finest x-resolution ($N_x = 1000$) line also appears on each plot, and this has been computed using a least-squares fit to the last four data points on that line. Two points are notable from these plots. First, the estimated asymptotic convergence rate of each method is close to the expected second-, fourth-, or sixth-order rate. Secondly, and perhaps of more practical significance, is the difference in magnitude of the errors for the different order schemes. Of particular interest is the resolution required by each method to achieve a given accuracy of solution, and especially notable are the large gains in accuracy obtained by a fourth-order scheme on a stretched grid relative to the widely used second-order scheme on an even grid. We will return to this point in the next section.

The general trend of the calculations is the same at other values of kh but, as might be expected, the errors get better with smaller kh, and worse with larger kh, reflecting the transition from a linear to an exponential behaviour of $\phi(z)$.

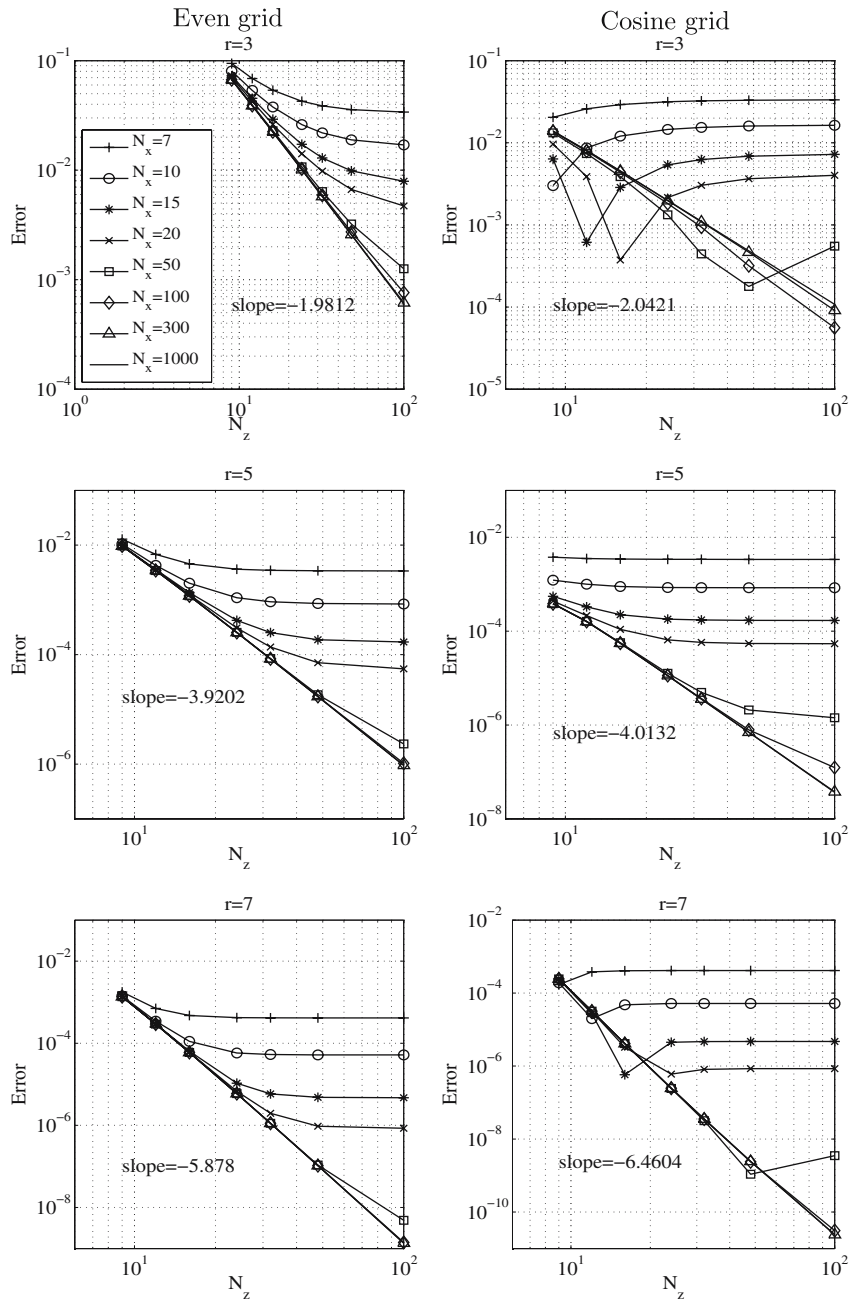

Fig. 2 Convergence of the linear $\tilde{w}$ calculated from $\tilde{\phi}$ at $kh = 4$. The "slope" is the asymptotic slope of the line for $N_x = 1000$. N_z point values are at [9,12,16,24,32,48,100]

4.2 Stability and accuracy

Fructus et al. [9] have shown how the linear part of the time-stepping problem can be integrated exactly by means of a Fourier transform, allowing much larger time steps to be taken for the nonlinear contribution. We have chosen not to do so here because our goal is a model which can treat problems with an arbitrary horizontal boundary and including fixed and floating bodies of complicated form within the computational domain. In this case it is not clear how to apply the required Fourier transform in a robust way.

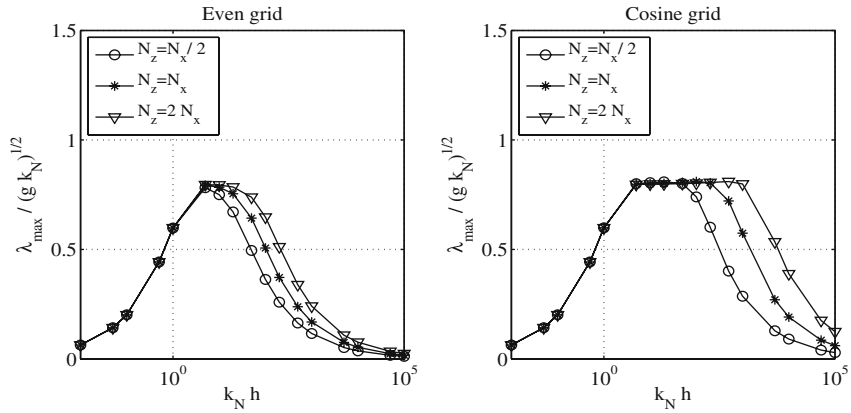

Fig. 3 Maximum eigenvalues ($r = 3$) vs. relative Nyquist depth. Variation with vertical grid refinement shown by the different lines

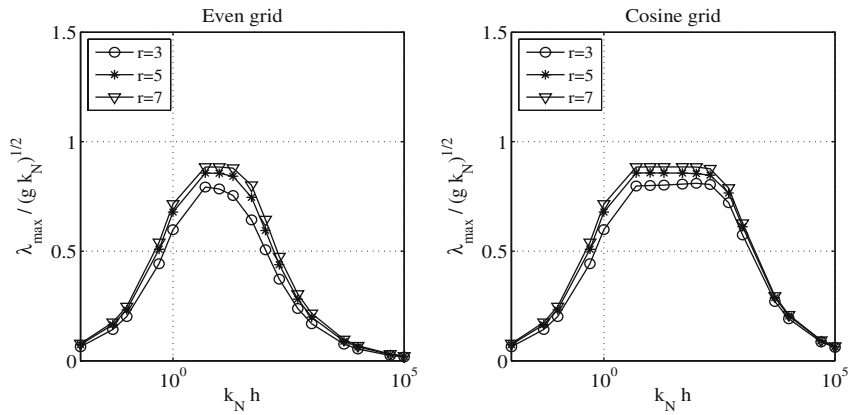

Fig. 4 Maximum eigenvalues using $N_z = N_x$ grid points in the vertical vs. relative Nyquist depth. Variation with increasing order shown by the different lines

The linear accuracy and stability of the method can be quantified by considering the semi-discrete form of the time-stepping equations (13) and (14)

$$\frac{\partial}{\partial t}\begin{bmatrix}\check{\eta}\\ \check{\phi}\end{bmatrix} = \begin{bmatrix}0 & \mathscr{J}_{12}\\ -g & 0\end{bmatrix}\begin{bmatrix}\check{\eta}\\ \check{\phi}\end{bmatrix}, \tag{21}$$

applied to the wave solution of (17). Here $\check{\eta}$ and $\check{\phi}$ are the Fourier amplitudes of η and $\tilde{\phi}$, while the factor $\mathscr{J}_{12}$ is the discrete approximation for $k\tanh(kh)$ discussed in the previous section (i.e. $\tilde{w} = \mathscr{J}_{12}\tilde{\phi}$). This is a method-of-lines approach (see e.g. [19,25]) and is valid as long as the above 2×2 matrix is amenable to an eigenvalue decomposition, in which case stability is governed by the largest of these and the stability region of a particular time-stepping scheme. For the discretisations discussed here, the eigenvalues are purely imaginary, with the largest occurring at the Nyquist mode. Figure 3 plots the magnitude of the maximum eigenvalue (normalised by the Nyquist frequency) as a function of relative depth for the Nyquist wave. These curves are for second-order discretisations ($r = 3$), showing the effect of increasing vertical resolution. A uniform vertical spacing is shown to the left and a stretched grid to the right. Figure 4 plots the same quantities but with a fixed vertical resolution of $N_z = N_x$ and using second-, fourth-, and sixth-order discretisations. These plots can be used to find the stability limit for any desired time-stepping scheme. For a given discretisation, stability is insured by keeping the quantity $\lambda_{\max}\,\Delta t$ within the stability region of the time-stepping scheme of interest (e.g. for fourth-order Runge–Kutta, $|\lambda_{\max}|\,\Delta t \leq 2\sqrt{2}$). Notable

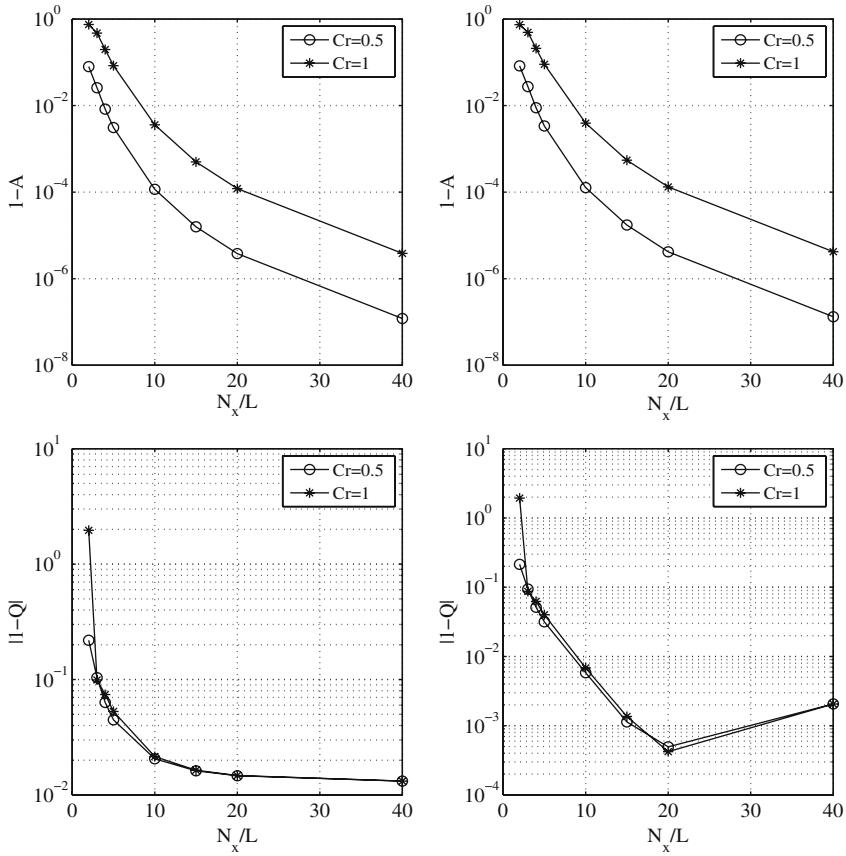

Fig. 5 Relative amplitude and phase errors over one period as a function of resolution for two Courant numbers. Second-order model ($r = 3$) with $N_z = 15$ at $kh = 4$. Even grid to the left, stretched grid to the right

from these plots is the relatively minor role played by both the vertical discretisation and the order of the scheme on the stability limit of the method.

Choosing now to focus on the fourth-order Runge–Kutta method which will be used in practise, we can extend the stability analysis discussed above to determine the overall accuracy of a given scheme. As with the convergence analysis, we consider $L = 1$ so that $N_x = 1/\Delta x$ represents the resolution of the wave (number of grid points per wavelength). For a given relative water depth kh and a vertical resolution N_z, the eigenvalues of the system can then be computed. Applied to a single equation, the fourth-order Runge–Kutta method can be expressed by the complex amplification factor

$$\rho(\Delta t\lambda) = 1 + \Delta t\lambda + \frac{(\Delta t\lambda)^2}{2} + \frac{(\Delta t\lambda)^3}{6} + \frac{(\Delta t\lambda)^4}{24} \tag{22}$$

where λ denotes an eigenvalue of the system. Equation 22 expresses the evolution of the solution in the eigenvector basis for a single time step, including both a relative amplification given by $|\rho(\Delta t\lambda)|$, and a phase shift of $\delta = \arg(\rho(\Delta t\lambda))$. Clearly, for a stable solution we must have $|\rho(\Delta t\lambda)| \leq 1$. As measures of overall accuracy we define:

$$A \equiv |\rho(\Delta t\lambda)|^{N_t}, \quad N_t = \frac{T}{\Delta t} \tag{23}$$

$$Q \equiv \frac{c_{\text{num}}}{c} = \frac{\delta}{\theta\, C_r}, \quad \theta = \frac{2\pi}{N_x} \tag{24}$$

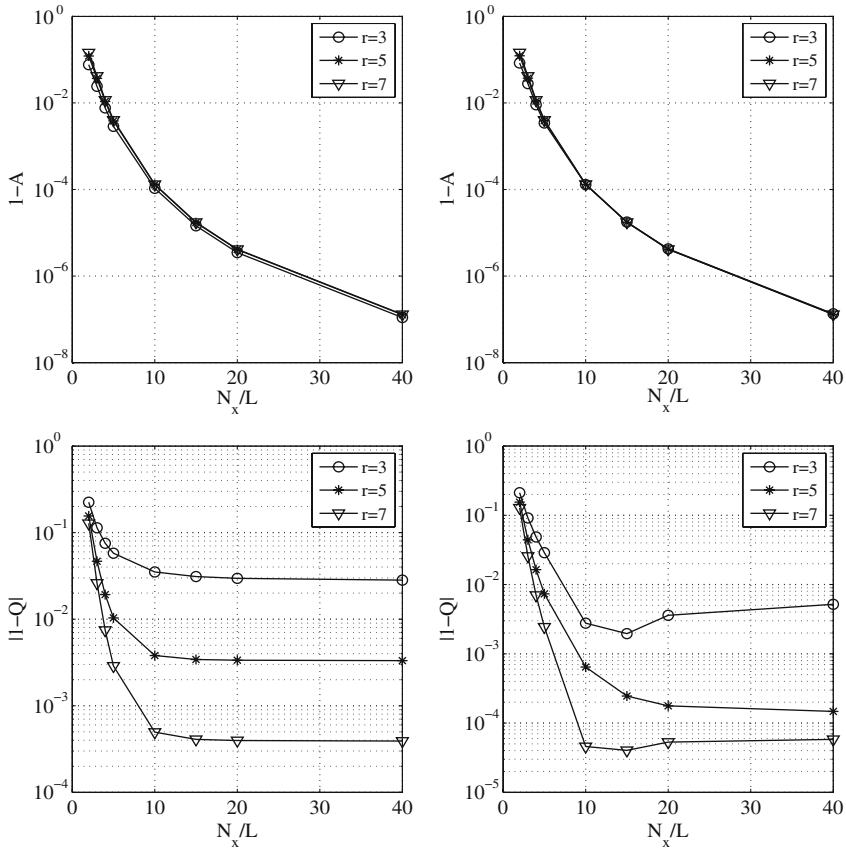

Fig. 6 Relative amplitude and phase errors over one period as a function of horizontal resolution with $N_z = 10$ using second- through sixth-order discretisations. For $C_r = 0.5$ and $kh = 4$. Even grid to the left, stretched grid to the right

i.e., the relative amplitude and phase error over one complete wave period. Here the hyperbolic Courant number is $C_r = c\Delta t/\Delta x$ with c the exact phase speed and the expression for Q comes from the numerical wave period $T_{\text{num}} = 2\pi\,\Delta t/\delta$, which leads to a discrete phase speed of $c_{\text{num}} = L/T_{\text{num}} = \Delta x\,\delta/(\Delta t\,\theta)$.

Figure 5 plots the amplitude and phase errors of the second-order model with a vertical resolution of $N_z = 10$, and at the two Courant numbers 1 and 0.5. These plots are for a relative water depth of $kh = 4$; results for an evenly spaced grid appear to the left and for a variable grid to the right. These plots illustrate two general features of the error. First, dispersion (phase) errors dominate, and are typically several orders of magnitude larger than diffusion (amplitude) errors; second, reducing the Courant number improves the diffusion error but not the dispersion error. Figure 6 plots the errors at $C_r = 0.5$ with a vertical resolution of $N_z = 10$ showing the effect of increasing the order of method from second to sixth. This plot makes it clear that increasing the order of the spatial derivatives has no significant effect on the diffusion errors but dramatically improves the dispersion error. Figure 7 plots dispersion errors for the same conditions but with $N_z = 15$ and $N_z = 20$.

Finally, Fig. 8 collects the dispersion errors for second-, fourth- and sixth-order discretisations on both a uniform and a stretched vertical grid as a function of grid resolution. These plots are for $C_r = 1$ and $kh = 4$, and, as found during the convergence calculations, the trend is very similar at other relative water depths but the errors increase with larger kh and decrease with smaller kh for the same vertical resolution.

Table 1 collects the approximate resolution required by each method to obtain relative phase errors of 10^{-3}, 10^{-4}, and 10^{-5}.

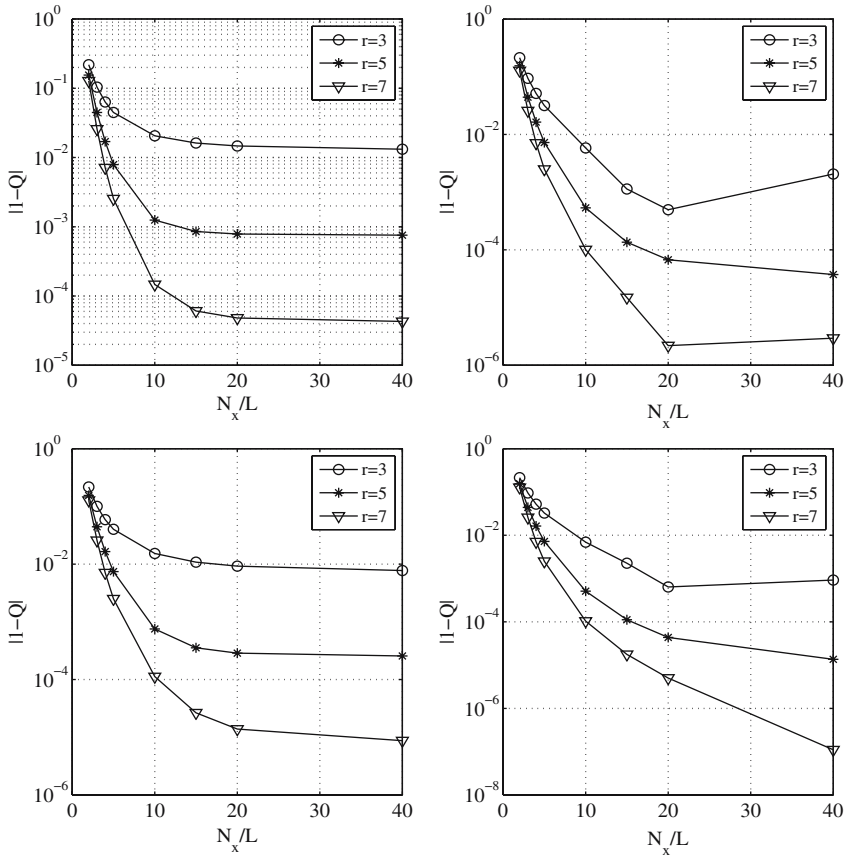

Fig. 7 Relative phase errors over one period as a function of horizontal resolution with two different vertical resolutions using second- through sixth-order discretisations. For $C_r = 0.5$, and $kh = 4$. Even grid to the left, stretched grid to the right; top line $N_z = 15$, bottom line $N_z = 20$

5 Nonlinear accuracy and efficiency

To quantify the accuracy and efficiency of the schemes discussed above on nonlinear problems, we consider a series of highly accurate periodic solutions computed using the stream-function-theory method of [17]. Each wave is characterised by a height H, a length L, the water depth h, and a mean Eulerian velocity u_E which is set to zero to model waves in an infinite domain. To cover the full range of interest, we have chosen the conditions shown in Table 2 for testing the model. Three kh-values appear corresponding to shallow, intermediate and deep water conditions, and for each of these there are two levels of nonlinearity at approximately 10% and 90% of the theoretical limiting steepness as found by [26]. (See also [27] where a convenient rational fit to the data is given.)

For each of the conditions of Table 2 the four models defined in Table 3 have been run. Each test was run for a total of five periods after which the relative error per wave period was computed from

$$\text{Error} = \frac{\|\eta_5 - \eta_e\|_2}{5\|\eta_e\|_2}, \quad \|\cdot\|_2 \equiv \text{two norm}, \tag{25}$$

where η_5 is the computed surface elevation after exactly five wave periods and η_e is the target result from stream-function theory. A Courant number of $C_r = 1$ was used for all cases. The results are collected in Fig. 9 with the mildly nonlinear case to the left and the strongly nonlinear case to the right,

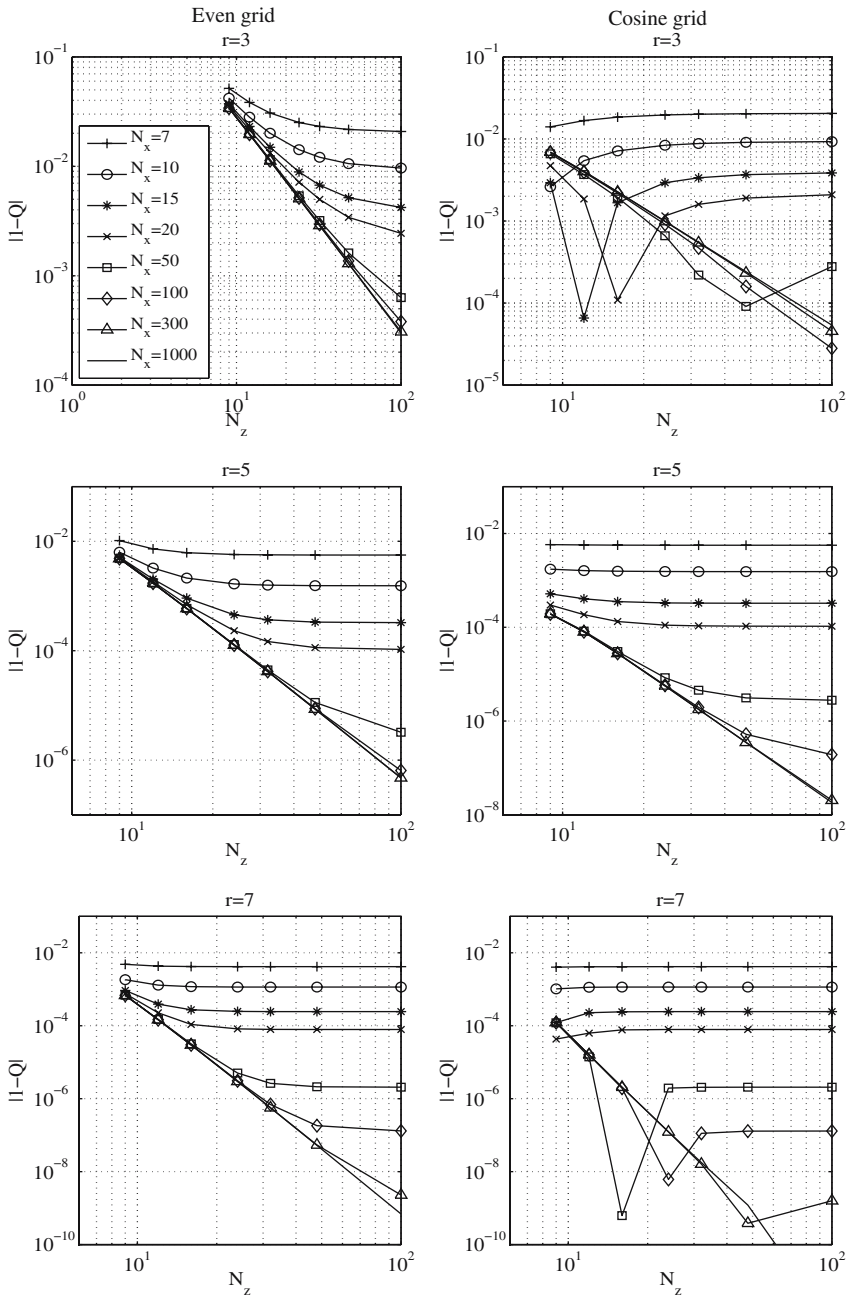

Fig. 8 Linear dispersion errors for second-, fourth- and sixth-order discretisations as a function of resolution. The Courant number $C_r = 1$, and $kh = 4$

kh-values increase from top to bottom. The point values correspond to the grid resolutions: $(N_x, N_z) = [(16, 9), (32, 9), (64, 12), (128, 24)]$.

The behaviour of the errors for the 10% case are consistent with the predictions made in Sect. 4. A relative residual tolerance of 10^{-7} was used in the iterative scheme for these calculations, so relative errors of less than 10^{-5} to 10^{-6} can not be expected. For the highly-nonlinear cases the errors are typically larger, but follow a similar trend. The case at $kh = 0.5$ resembles a very high solitary wave (very sharp and narrow peak) and required the application of a filter to obtain stable results. The filter used was a simple

Table 1 Approximate resolution required to obtain a given accuracy in the linear $\tilde{w}$ at $kh = 4$

Model	Approximate $N_x \times N_z$ for an error of		
	10^{-3}	10^{-4}	10^{-5}
$r = 3$, even	100×80	n.a.	n.a.
$r = 3$, cosine	50×32	100×80	n.a.
$r = 5$, even	16×16	32×32	50×60
$r = 5$, cosine	12×10	20×16	50×25
$r = 7$, even	10×10	10×16	15×24
$r = 7$, cosine	7×9	10×10	15×15

"n.a." indicates that the values are off the scale of Fig. 8.

Table 2 Periodic nonlinear wave conditions used to test the model

kh	0.5 (shallow)	2 (intermediate)	2π (deep)
$H/L \approx 10\%\ (H/L)_{\max}$	0.0059	0.011	0.0135
$H/L \approx 90\%\ (H/L)_{\max}$	0.053	0.10	0.12

Table 3 The four models tested

r	Order of accuracy	Vertical grid
3	2nd	even
3	2nd	cosine
5	4th	cosine
7	6th	cosine

truncation of the Fourier space at a cut-off frequency of half the Nyquist frequency (i.e. FFT, truncate, FFT back). Apart from this case, no smoothing or filtering was used in these calculations.

Figure 10 plots the iterations required by each model vs. N for the 90% steepness case in deep and shallow water. The 10% case required approximately half as many iterations.

These results are consistent with the linear analysis and demonstrate that, if results at a given accuracy are desired, a significant improvement in efficiency can be obtained by moving from second-order to fourth-order in the spatial-differencing scheme and stretching the vertical grid. It is also clear that decreasing accuracy tolerance and increasing nonlinearity favour higher-order methods.

6 Sample applications to shoaling wave problems

The analysis described above suggests that a fourth-order model with a stretched vertical grid of around ten points and a horizontal resolution of 15–20 points per wavelength will be suitable for general applications of the method. We include here calculations using such a model for two wave–bottom interaction test cases: linear shoaling up a beach; nonlinear harmonic generation over a submerged bar.

6.1 Linear shoaling

The derivatives of η appearing in the sigma-transformed Laplace problem were validated by the nonlinear periodic test cases described above, but the terms involving derivatives of h were not. To validate these terms, and to test the variable horizontal grid-spacing features of the code, we consider the problem of a linear wave shoaling up a beach. We use a bathymetry defined by

$$h(x) = h_0 - \frac{h_0 - h_1}{2}\left[1 + \tanh\left(\frac{\sin\left(\frac{\pi x}{l}\right)}{1 - \left(\frac{2x}{l}\right)^2}\right)\right], \quad -\frac{l}{2} \le x \le \frac{l}{2} \tag{26}$$

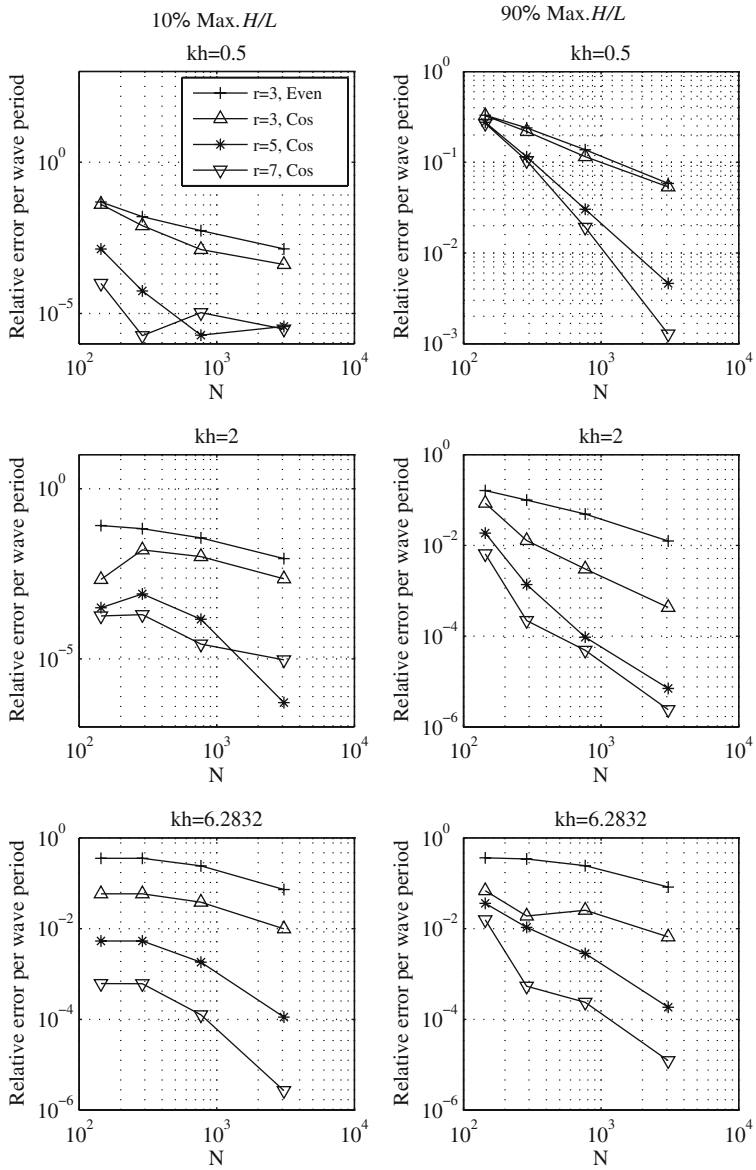

Fig. 9 Nonlinear errors (25) for the four versions of the model listed in Table 3. All plots follow the legend shown

with $h_0 = 0.5$, $h_1 = 0.025$, and $l = 12$ (see Fig. 11). This gives a mean slope of $h_x \approx 0.05$. A linear wave of period $T = 0.8014$s and deep-water length $L_0 = 1$m is generated at the left end of the domain and allowed to propagate until a steady state has been reached everywhere. This produces relative conditions from deep to shallow water, corresponding to $0.4 \leq kh \leq \pi$.

Wave generation and absorption is achieved by relaxing the numerical solution towards a specified analytic solution over regions near the boundaries of the domain. This is an idea which is often applied in ocean circulation modelling (see e.g. [28] who modified the idea proposed in [29]). Relaxation zones for simultaneous generation and absorption of waves are created by defining a relaxation coefficient $0 \leq c_r(\mathbf{x}) \leq 1$, and an exact desired solution $(\eta_e, \tilde{\phi}_e)$. After each stage of the time integration, the solution within a relaxation zone is then redefined to be

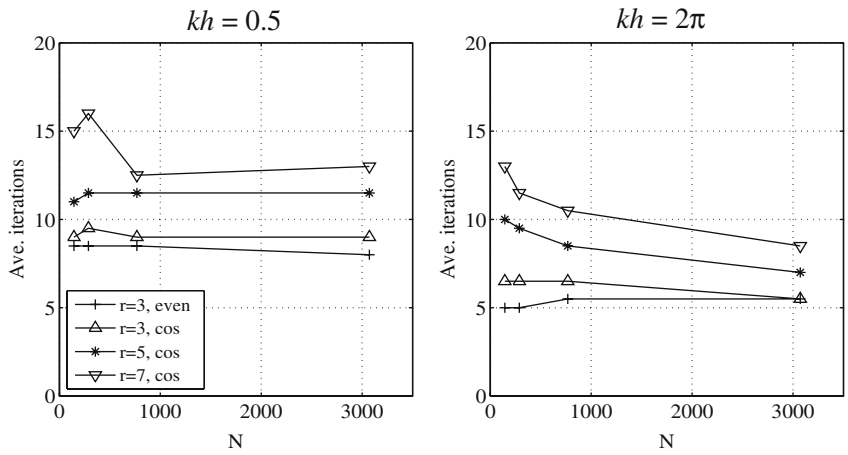

Fig. 10 Average iteration counts for the 90% steepness case

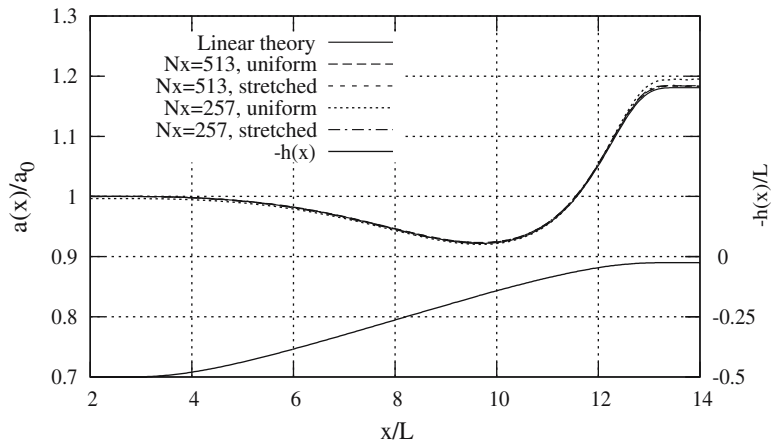

Fig. 11 Harmonic analysis of a linear wave shoaling up a beach

$$\eta(\mathbf{x},t) = [1 - c_r(\mathbf{x})]\,\eta(\mathbf{x},t) + c_r(\mathbf{x})\,\eta_e(\mathbf{x},t),$$
$$\tilde{\phi}(\mathbf{x},t) = [1 - c_r(\mathbf{x})]\,\tilde{\phi}(\mathbf{x},t) + c_r(\mathbf{x})\,\tilde{\phi}_e(\mathbf{x},t). \tag{27}$$

To avoid reflections, c_r should be smooth near the junction between the relaxation zone and the rest of the computational domain. For pure absorption, the exact solution is simply taken to be zero. Zones of two wave lengths near both ends have been used here, which generally results in overall reflected wave heights of less than 0.1%.

Figure 11 plots the amplitude envelope of the converged steady-state time signals for several choices of horizontal discretisation, along with the exact result predicted by energy conservation:

$$\frac{a(x)}{a_0} = \left(\frac{k(x)\left(1 + \frac{2k_0h_0}{\sinh(2k_0h_0)}\right)}{k_0\left(1 + \frac{2k(x)h(x)}{\sinh(2k(x)h(x))}\right)} \right)^{\frac{1}{2}}. \tag{28}$$

For all four discretisations, $N_z = 9$ points in the vertical have been used and the time step is chosen to give a deep-water $C_r = 1$. Results are shown using a coarse and a fine grid ($N_x = 257$ and 513 points) and with both a uniform and a stretched horizontal grid spacing. The uniform grids provide resolutions of 16 and 32 points per wavelength, respectively, at the deep end of the domain but at the shallow end, where

the wavelength has been reduced to ≈ 0.4 m, the resolution is much coarser. The stretched grids keep the resolution approximately constant along the length of the domain by defining $x_i = l \tanh(C s_i)/\tanh(C)$, $s_i = (i-1)\mathrm{d}s$, $\mathrm{d}s = 1/(N_x - 1)$, with l the length of the domain and C a constant which controls the stretching. For this case, $l = 16$ and $C = 1.5$ were chosen. While the three other results can be seen to have converged to the exact solution, the uniformly spaced coarse grid results show significant errors due to the under-resolution towards the shallow end of the domain.

6.2 Harmonic generation over a submerged bar

As a final example for testing all the features of the model, we consider the transformation of a mildly nonlinear, monochromatic wave as it travels up and over a submerged bar. As the wave shoals up the front face of the bar, it steepens dramatically, accumulating higher harmonics which are released on the down-slope producing an irregular pattern behind the bar. This phenomenon is widely used as a strenuous test case for nonlinear models of surface wave propagation on a variable bottom. Details of the experiment can be found in [30], which is usually scaled by a factor of two, to make it equivalent to [31].

The top plot in Fig. 12 shows the bathymetry of the experiment and a snapshot of the steady-state surface elevation scaled by a factor of three to make it more visible. The lower plot in this figure shows a harmonic analysis of the time series after a steady state has been reached over the entire domain for the long-wave case ($T = 2.01$ s). The calculations were made using $N_x = 513$, $N_z = 9$ and $\Delta t = .025$ s (deep water $C_r \approx 0.75$.) Relaxation zones of length $2L$ at both ends of the domain were used to generate and absorb the waves. Stream-function theory was used for the incident wave. The harmonic analysis was performed by making a least-squares fit of a sum of nine harmonics to the time series (measured or computed) at a number of points along the tank length. This gave a residual of at least four orders of magnitude smaller

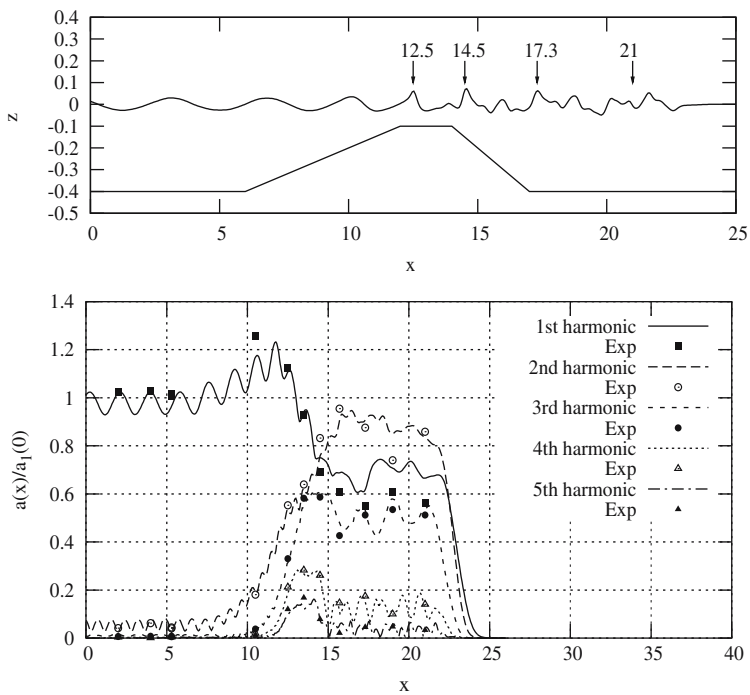

Fig. 12 Top: bar test geometry with a snapshot of the surface elevation (scaled by a factor 3) along with the measurement locations. Bottom: harmonic analysis of the time series at steady-state

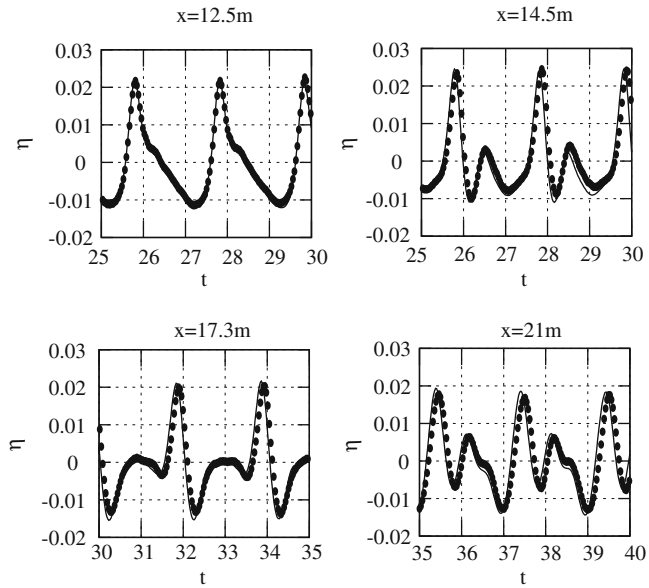

Fig. 13 Time series of wave elevations over the bar. Solid line—Calculations; dots—Measurements

than the first harmonic, indicating that level of accuracy. Time series of the elevations at the points indicated in Fig. 12 are shown in Fig. 13. The comparison with the measured results is quite good.

7 Conclusions

The accuracy and efficiency of a finite-difference-based solution to the exact potential-flow problem for nonlinear waves has been established in 2D. The model is flexible in terms of the order of the numerical schemes and the grid spacing; with a computational effort which scales with N, the total number of grid points used. A fourth-order-accurate model with a stretched vertical grid of approximately ten points, a resolution of 15–20 points per wavelength and a Courant number of one has been shown to be adequate for general purpose applications. This model is readily extended to 3D and the inclusion of fixed and floating bodies is conceptually straightforward, which are the topics of ongoing development.

Acknowledgements Support from the Danish Centre for Scientific Computing is acknowledged.

References

1. Newman JN (1977) Marine hydrodynamics. The MIT Press, Cambridge, Massachusetts
2. Newman JN (2005) Efficient hydrodynamic analysis of very large floating structures. Marine Struct 18:169–180
3. Newman JN, Lee C-H (2002) Boundary-element methods in offshore structure analysis. J Offshore Mech Arctic Engng 124:81–89
4. Sclavounos PD, Borgen H (2004) Seakeeping analysis of a high-speed monohull with a motion-control bow hydrofoil. J Ship Res 48:77–117
5. Bertram V, Yasukawa H, Thiart G (2005) Evaluation of local pressures in ships using potential flow models. In: Oceans 2005 – Europe (IEEE Cat. No. 05EX1042), vol 1. IEEE, pp 113–117
6. Büchmann B, Ferrant P, Skourup J (2000) Run-up on a body in waves and current. fully nonlinear and finite-order calculations. Appl Ocean Res 22:349–360
7. Grilli ST, Guyenne P, Dias F (2001) A fully non-linear model for three-dimensional overturning waves over an arbitrary bottom. Int J Num Methods Fluids 35:829–867

8. Dommermuth DG, Yue DKP (1987) A high-order spectral method for the study of nonlinear gravity waves. J Fluid Mech 184:267–288
9. Fructus D, Clamond D, Grue J, Kristiansen Ø (2005) An efficient model for three-dimensional surface wave simulations. part i: Free space problems. J Comput Phys 205:665–685
10. Fuhrman DR, Bingham HB, Madsen PA (2005) Nonlinear wave-structure interaction with a high-order Boussinesq model. Coastal Engng 52:655–672
11. Shi F, Kirby JT, Dalrymple RA, Chen Q (2003) Wave simulations in ponce de leon inlet using Boussinesq model. J Waterway Port Coastal Ocean Engng 129:124–135
12. Chen H-C, Lee S-K (1999) Rans/laplace calculations of nolinear waves induced by surface-piercing bodies. J Engng Mech 125(11):1231–1242
13. Wu GX, Ma QW, Eatock-Taylor R (1998) Numerical simulation of sloshing waves in a 3d tank based on a finite element method. Appl Ocean Res 20:337–355
14. Li B, Fleming CA (1997) A three-dimensional multigrid model for fully nonlinear water waves. Coastal Engng 30:235–258
15. Kreiss HO, Oliger J (1972) Comparison of accurate methods for the integration of hyperboloic equations. Tellus Series A 24(3):199–215
16. Fuhrman DR, Bingham HB (2004) Numerical solutions of fully nonlinear and highly dispersive Boussinesq equations. Int J Num Methods Fluids 44(3):231–255
17. Fenton JD (1988) The numerical solution of steady water wave problems. Comput Geosci 14(3):357–68
18. Zakharov VE (1968) Stability of periodic waves of finite amplitude on the surface of a deep fluid. J Appl Mech Tech Phys 9:190–194
19. Iserles A (1996) A first course in the numerical analysis of differential equations. Cambridge University Press
20. Duff IS (1989) Multiprocessing a sparse matrix code on the alliant fx/8. J Comput Appl Math 27:229–239
21. Saad Y, Schultz MH (1986) GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. SIAM J Sci Stat Comput 7:856–869
22. Amestoy PR, Davis TA, Duff IS (1996) An approximate minimum degree ordering algorithm. SIAM J Mat Anal Appl 17:886–905
23. Whaley RC, Petitet A, Dongarra JJ (2000) Automated empirical optimization of software and the atlas project, 2000. Available: http://math-atlas.sourceforge.net/
24. Hirsch C (1988) Numerical computation of internal and external flows. Volume 1: Fundamentals of numerical discretizaton. John Wiley & Sons, New York
25. Trefethen LN (1996) Finite difference and spectral methods for ordinary and partial differential equations. Available: http://web.comlab.ox.ac.uk/oucl/work/nick.trefethen/pdetext.html, Unpublished text
26. Williams JM (1981) Limiting gravity waves in water of finite depth. Phil Trans R Soy London A 302:139–188
27. Fenton JD (1990) Nonlinear wave theories. In: Le Mehaute B, Hanes DM (eds) The sea. John Wiley & Sons, pp 3–25
28. Jensen TG (1998) Open boundary conditions in stratified ocean models. J Marine Syst 16:297–322
29. Larsen J, Dancy H (1983) Open boundaries in short wave simulations – a new approach. Coastal Engng 7:285–297
30. Luth HR, Klopman B, Kitou N (1994) Projects 13 g: kinematics of waves breaking partially on an offshore bar; ldv measurements for waves with and without a net onshore current. Technical Report H1573, Delft Hydraulics, Delft, The Netherlands
31. Beji S, Battjes JA (1993) Experimental investigation of wave propagation over a bar. Coastal Engng 19:151–162