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

Bingham, Harry B.; Zhang, Haiwen

Published in:
Journal of Engineering Mathematics

Link to article, DOI:
10.1007/s10665-006-9108-4

Publication date:
2007

On the accuracy of finite difference solutions for nonlinear water waves

Harry B. Bingham * (hbb@mek.dtu.dk)
Mechanical Engineering, Technical University of Denmark, DK-2800 Lyngby, Denmark

Haiwen Zhang (zhw@dhi.dk)
DHI Water & Environment, DK-2970, Hørsholm, Denmark

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 [1] 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 are suggested as optimal for engineering purposes.

Keywords: Nonlinear waves, finite difference methods, accuracy, 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 assume an irrotational flow to obtain a potential flow problem governed by the Laplace equation (see *e.g.* [2] for a detailed statement of the problem).

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

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 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.* [3, 4]). 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.* [5, 6, 7, 8]).

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.* [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.

Compared to these surface projection methods, direct numerical solution of the 3D Laplace problem is 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 it's 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.* by [13], and a finite difference based solution was developed by [1].

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 & Oliger [14] 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 [15] also shows a significant advantage

to using fourth-order schemes. Here we extend the method of [1] 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 it is likely 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 §2 and §3, the linear convergence, accuracy and stability properties are established by standard Fourier analysis in §4. In §5 the nonlinear accuracy of the method is quantified by comparison with periodic solutions based on stream function theory [16]. From this we conclude that high-order schemes and a stretched vertical grid are more efficient then 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.

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 [17], 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 $\tilde{w} = \left.\frac{\partial \phi}{\partial z}\right|_{z=\eta}$

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

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

Here $\nabla = [\partial/\partial x, \partial/\partial y]$ 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 immediately computed, 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:

$$\nabla^2\phi + \phi_{zz} = 0, \quad -h < z < \eta \tag{3}$$

$$\phi_z + \nabla h \cdot \nabla\phi = 0, \quad z = -h. \tag{4}$$

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

$$\sigma(\mathbf{x}, z, t) = \frac{z + h(\mathbf{x})}{\eta(\mathbf{x}, t) + h(\mathbf{x})} = \frac{z + h(\mathbf{x})}{d(\mathbf{x}, t)}, \tag{5}$$

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

$$\nabla^2\phi + \nabla^2\sigma\,\phi_\sigma + 2\nabla\sigma\cdot\nabla\phi_\sigma + \left(\nabla\sigma\cdot\nabla\sigma + \sigma_z^2\right)\phi_{\sigma\sigma} = 0, \;\; 0 < \sigma < 1 \tag{6}$$

$$\left(\sigma_z + \nabla h\cdot\nabla\sigma\right)\phi_\sigma + \nabla h\cdot\nabla\phi = 0 \;\; \sigma = 0, \tag{7}$$

where the derivatives of σ can be expressed as

$$\nabla\sigma = (1-\sigma)\frac{\nabla h}{d} - \sigma\frac{\nabla\eta}{d} \tag{8}$$

$$\nabla^2\sigma = \frac{1-\sigma}{d}\left(\nabla^2 h - \frac{\nabla h\cdot\nabla h}{d}\right) - \frac{\sigma}{d}\left(\nabla^2\eta - \frac{\nabla\eta\cdot\nabla\eta}{d}\right) - \frac{1-2\sigma}{d^2}\nabla h\cdot\nabla\eta - \frac{\nabla\sigma}{d}\cdot(\nabla h + \nabla\eta) \tag{9}$$

$$\sigma_z = \frac{1}{d} \tag{10}$$

which illustrates the specific derivatives of h and η required for their evaluation. After solving (6) & (7) for the potential $\phi(\mathbf{x}, \sigma)$, the vertical component of fluid velocity on the free surface is given by

$$\tilde{w} = \frac{1}{d}\,\phi_\sigma|_{\sigma=1} \tag{11}$$

which allows (1) & (2) 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

$$u(\mathbf{x},z) = \phi_x(\mathbf{x},z) = \phi_x(\mathbf{x},\sigma) + \sigma_x\phi_\sigma(\mathbf{x},\sigma) \quad (12)$$

$$v(\mathbf{x},z) = \phi_y(\mathbf{x},z) = \phi_y(\mathbf{x},\sigma) + \sigma_y\phi_\sigma(\mathbf{x},\sigma), \quad (13)$$

$$w(\mathbf{x},z) = \phi_z(\mathbf{x},t) = \frac{1}{d}\phi_\sigma(\mathbf{x},\sigma). \quad (14)$$

3. Numerical solution

As the goal of this paper is to investigate the relative performance of different spatial finite-difference discretisaton schemes for solving the above stated Laplace problem, we restrict our attention to one horizontal dimension, periodic and of uniform depth. For the time-integration of (1) & (2) we employ the classical explicit four-stage, fourth-order Runge-Kutta scheme (see *e.g.* [18]). 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 evolved. For the analysis of this section a uniform grid spacing Δx is assumed, although in the numerical code the spacing is arbitrary. For the solution of (6) & (7), N_z points are defined in the vertical, arbitrarily spaced between $\sigma = 0$ and $\sigma = 1$. 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.

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. As the domain is periodic in x, all x-derivatives are centrally discretised with $\alpha = \beta$, while for the σ-derivatives only points within the computational domain are used leading to off-centred schemes near the bottom and free-surface. For the mixed $x\sigma$-derivative, a full square stencil of r^2-points is used and the evaluation point is always centred in x but will be off-centred in σ near the free-surface and bottom. In this way, all derivatives are formally accurate to $O(\Delta x_*^{r-1})$ where Δx_* is the maximum grid spacing. The order of accuracy of the schemes is verified in §4.

The resultant discretisation of (6) is then applied at all grid points in x and the internal grid points in σ (*i.e.* not including those along $\sigma = 0$ and $\sigma = 1$). The boundary conditions are imposed by applying the discrete version of (7) to those grid points lying along $\sigma = 0$, while

the condition $\phi(\mathbf{x}, 1) = \tilde{\phi}(\mathbf{x})$ is applied to the points along $\sigma = 1$ by placing $\tilde{\phi}$ on the right hand side and zeroing all coefficients of that equation except the diagonal which is set to one. This gives a rank $N = N_x\, N_z$ linear system of equations

$$\mathbf{A}[\phi] = [b] \tag{15}$$

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 $\sigma = 1$ where it contains the known value of $\tilde{\phi}$.

3.2. Iterative solution of the linear system

The matrix $\mathbf{A}$ in (15) is sparse and contains approximately $r^2 N$ 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 [19], preconditioned on the left using the linearised, second-order accurate version of $\mathbf{A}$. Specifically, we set $\eta = 0$ in (5) which makes σ independent of time and removes all functions of η from the derivatives in (8-10). 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 $\mathbf{M}[q] = [b]$.

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 §5, the iteration count to achieve a relative convergence tolerance of 10^{-7} when using this preconditioner is generally $O(10)$ and very nearly independent of N and r and kh. 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 [20]. 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.* [21])). For

further details on this routine see [22] 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 Figure 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

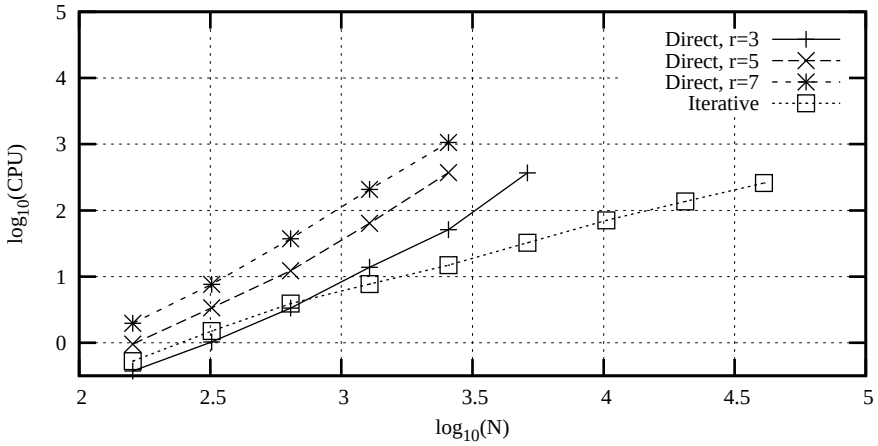

Figure 1. Scaling of the computational effort for direct and iterative solutions.

the calculation time on a 3.2 GHz Pentium processor with 1 giga-byte 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:

$$\eta_t \;=\; \tilde{w} \tag{16}$$

$$\tilde{\phi}_t = -g\,\eta \tag{17}$$
$$\nabla^2\phi + \phi_{zz} = 0, \qquad -h < z < 0 \tag{18}$$
$$\phi_z = 0, \qquad z = -h \tag{19}$$

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

$$\eta = \Re\left\{\frac{H}{2}\, e^{\mathrm{i}(kx-\omega t)}\right\} \tag{20}$$
$$\phi = \Re\left\{\frac{-\mathrm{i}\,g}{\omega}\frac{H}{2}\,\frac{\cosh\left[k(z+h)\right]}{\cosh\left(kh\right)}\, e^{\mathrm{i}(kx-\omega t)}\right\} \tag{21}$$

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 check the convergence of a given method, we compare the computed value to the exact result $\tilde{w} = \tanh\left(kh\right)\tilde{\phi}$, for increasingly fine discretisations. Since the problem is periodic in the x-direction and centrally discretised on a uniform grid, all x-grid points are equivalent and we can apply the second x-derivative scheme to the wave solution (21) to express it as a simple 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

$$\frac{\partial^2}{\partial x^2} \to N_x^2\left(-2 + 2\cos\left(\frac{2\pi}{N_x}\right)\right) \tag{22}$$

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.* [23]). 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 (15) which is of order N_z

$$\mathbf{B}\left[\phi\right] = \left[b\right]. \tag{23}$$

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 $\tilde{\phi}$. **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 results for a uniform grid using second-, fourth-, and sixth-order schemes ($r = 3, 5,$ & 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 $\sigma_j = h[-1 + \sin(\Delta\theta(j-1))]$ where $\Delta\theta = \pi/[2(N_z - 1)]$. Each plot shows the relative error $\log_{10}(|\tilde{w} - \tilde{w}_e|/\tilde{w}_e)$ versus $\log_{10}(N_z)$, with $\tilde{w}_e$ 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 profile.

4.2. Stability and accuracy

The linear accuracy and stability of the method can be quantified by considering the semi-discrete form of the time-stepping equations (16) & (17)

$$\frac{\partial}{\partial t}\begin{bmatrix} \check{\eta} \\ \check{\phi}_0 \end{bmatrix} = \begin{bmatrix} 0 & \mathcal{J}_{12} \\ -g & 0 \end{bmatrix}\begin{bmatrix} \check{\eta} \\ \check{\phi}_0 \end{bmatrix}, \tag{24}$$

applied to the wave solution of (20). Here $\check{\eta}$ and $\check{\phi}_0$ are the Fourier amplitudes of η and $\tilde{\phi}$, while the factor $\mathcal{J}_{12}$ is the discrete approximation for $\tanh(kh)$ discussed in the previous section (*i.e.* $\tilde{w} = \mathcal{J}_{12}\,\tilde{\phi}$). This is

a method of lines approach (see *e.g.* [18, 24]) 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 any 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 from these plots is the relatively minor role played by both the vertical discretisaton 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, the stability analysis discussed above can be extended 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

$$\rho(\Delta t\lambda) = 1 + \Delta t\lambda + \frac{(\Delta t\lambda)^2}{2} + \frac{(\Delta t\lambda)^3}{6} + \frac{(\Delta t\lambda)^4}{24} \tag{25}$$

where λ denotes an eigenvalue of the system. Equation (25) expresses the evolution 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:

$$A \equiv |\rho(\Delta t\lambda)|^{N_t}, \quad N_t = \frac{T}{\Delta t} \tag{26}$$

$$Q \equiv \frac{c_{\text{num}}}{c} = \frac{\delta}{\theta\, C_r}, \quad \theta = \frac{2\pi}{N_x} \tag{27}$$

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, and second; reducing the Courant number improves the diffusion error but not the dispersion error. Figure 6 plots the errors at $Cr = 0.5$ 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, Figure 8 collects the dispersion errors for second-, fourth- and sixth-order discretisations on both uniform and a stretched vertical grid as a function of grid resolution. These plots are for a Courant number of one 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 I collects the approximate resolution required by each method to obtain relative phase errors of 10^{-3}, 10^{-4}, and 10^{-5}.

Table I. Approximate resolution required to obtain a given accuracy in the linear $\tilde{w}$ at $kh = 4$. "n.a." indicates that the values are off the scale of Figure 2.

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

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 [16]. 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 (see Figure 5). To cover the full range of interest, we have chosen the conditions shown in Table II 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 [25]. (See also [26] where a convenient rational fit to the data is given.)

Table II. Periodic nonlinear wave conditions used to test the model.

kh	0.5 (shallow)	2 (intermediate)	2π (deep)
$H/L \approx 10\%\,(H/L)_{\max}$	.0059	.011	.0135
$H/L \approx 90\%\,(H/L)_{\max}$	.053	.10	.12

For each of the conditions of Table II the four models defined in Table III have been run. Each test was run for a total of five periods

Table III. The four models tested.

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

after which the relative error per wave period was computed from

$$\text{Error} = \frac{\|\eta_5 - \eta_e\|_2}{5\|\eta_e\|_2} \tag{28}$$

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 Figure 5 with the mildly nonlinear case to the left and the strongly nonlinear case to the right, kh values increase from top to bottom.

The behaviour of the errors for the 10% case are consistent with the predictions made in §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 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 5 plots the iterations required by each model vs. N for the 90% steepness case in deep and shallow water. The 10% required approximately half as many iterations.

These results 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. It is also clear that decreasing accuracy tolerance and increasing nonlinearity favour higher-order methods.

References

1. B. Li and C.A. Fleming. A three-dimensional multigrid model for fully nonlinear water waves. *Coastal Engineering*, 30:235–258, 1997.
2. J. N. Newman. *Marine Hydrodynamics*. The MIT Press, Cambridge, Massachusetts, 1977.
3. J. N. Newman. Efficient hydrodynamic analysis of very large floating structures. *Marine Structures*, 18:169–180, 2005.
4. J. N. Newman and C.-H. Lee. Boundary-element methods in offshore structure analysis. *Journal of Offshore Mechanics & Arctic Engineering*, 124:81–89, 2002.
5. P. D. Sclavounos and H. Borgen. Seakeeping analysis of a high-speed monohull with a motion-control bow hydrofoil. *J. Ship Res.*, 48:77–117, 1987.
6. V. Bertram, H. Yasukawa, and G. Thiart. Evaluation of local pressures in ships using potential flow models. In *Oceans 2005 - Europe (IEEE Cat. No. 05EX1042)*, pages 113–17 Vol.1. IEEE, 2005.
7. B. Büchmann, P. Ferrant, and J. Skourup. Run-up on a body in waves and current. fully nonlinear and finite-order calculations. *Applied Ocean Res.*, 22:349–360, 2000.
8. S.T. Grilli, P. Guyenne, and F. Dias. A fully non-linear model for three-dimensional overturning waves over an arbitrary bottom. *Int. J. Num. Methods in Fluids*, 35:829–867, 2001.
9. D. G. Dommermuth and D. K. P. Yue. A high-order spectral method for the study of nonlinear gravity waves. *J. Fluid Mech.*, 184:267–288, 1987.
10. D.R. Fuhrman, H.B. Bingham, and P.A. Madsen. Nonlinear wave-structure interaction with a high-order Boussinesq model. *Coastal Engineering*, 52:655–672, 2005.

11. F. Shi, J.T. Kirby, R.A. Dalrymple, and Q Chen. Wave simulations in ponce de leon inlet using Boussinesq model. *J. of Waterway Port Coastal and Ocean Eng.*, 129:124–135, 2003.
12. H.-C. Chen and S.-K. Lee. Rans/laplace calculations of nolinear waves induced by surface-piercing bodies. *J. Engineering Mech.*, 125(11):1231–1242, 1999.
13. G.X. Wu, Q.W. Ma, and R. Eatock-Taylor. Numerical simulation of sloshing waves in a 3d tank based on a finite element method. *Applied Ocean Res.*, 20:337–355, 1998.
14. H. O. Kreiss and J. Oliger. Comparison of accurate methods for the integration of hyperboloic equations. *Tellus Series A*, 24(3):199–215, 1972.
15. D.R. Fuhrman and H.B. Bingham. Numerical solutions of fully nonlinear and highly dispersive Boussinesq equations. *Int. J. Num. Methods in Fluids*, 44(3):231–255, 2004.
16. J.D. Fenton. The numerical solution of steady water wave problems. *Comput. Geosci.*, 14(3):357–68, 1988.
17. V. E. Zakharov. Stability of periodic waves of finite amplitude on the surface of a deep fluid. *J. Appl. Mech. Tech. Phys.*, 9:190–194, 1968.
18. A. Iserles. *A first course in the numerical analysis of differential equations.* Cambridge University Press, 1996.
19. Y. Saad and M. H. Schultz. GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.*, 7:856–869, 1986.
20. P.R. Amestoy, T.A. Davis, and I.S. Duff. An approximate minimum degree ordering algorithm. *SIAM J. on Mat. Anal. Appl.*, 17:886–905, 1996.
21. R.C. Whaley, A. Petitet, and J.J. Dongarra. Automated empirical optimization of software and the atlas project, 2000. *Available: http://math-atlas.sourceforge.net/*, 2000.
22. I.S. Duff. Multiprocessing a sparse matrix code on the alliant fx/8. *J. Comput. Appl. Math.*, 27:229–239, 1989.
23. Hirsch. *Numerical computation of internal and external flows. Volume 1: Fundamentals of numerical discretizaton.* John Wiley & Sons, New York, 1988.
24. L.N. Trefethen. *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, 1996.
25. J. M. Williams. Limiting gravity waves in water of finite depth. *Phil. Trans. Roy. Soy. Lond. A*, 302:139–188, 1981.
26. J.D. Fenton. Nonlinear wave theories. In D.M. Le Mehaute, B. & Hanes, editor, *The Sea*, pages 3–25. John Wiley & Sons., 1990.

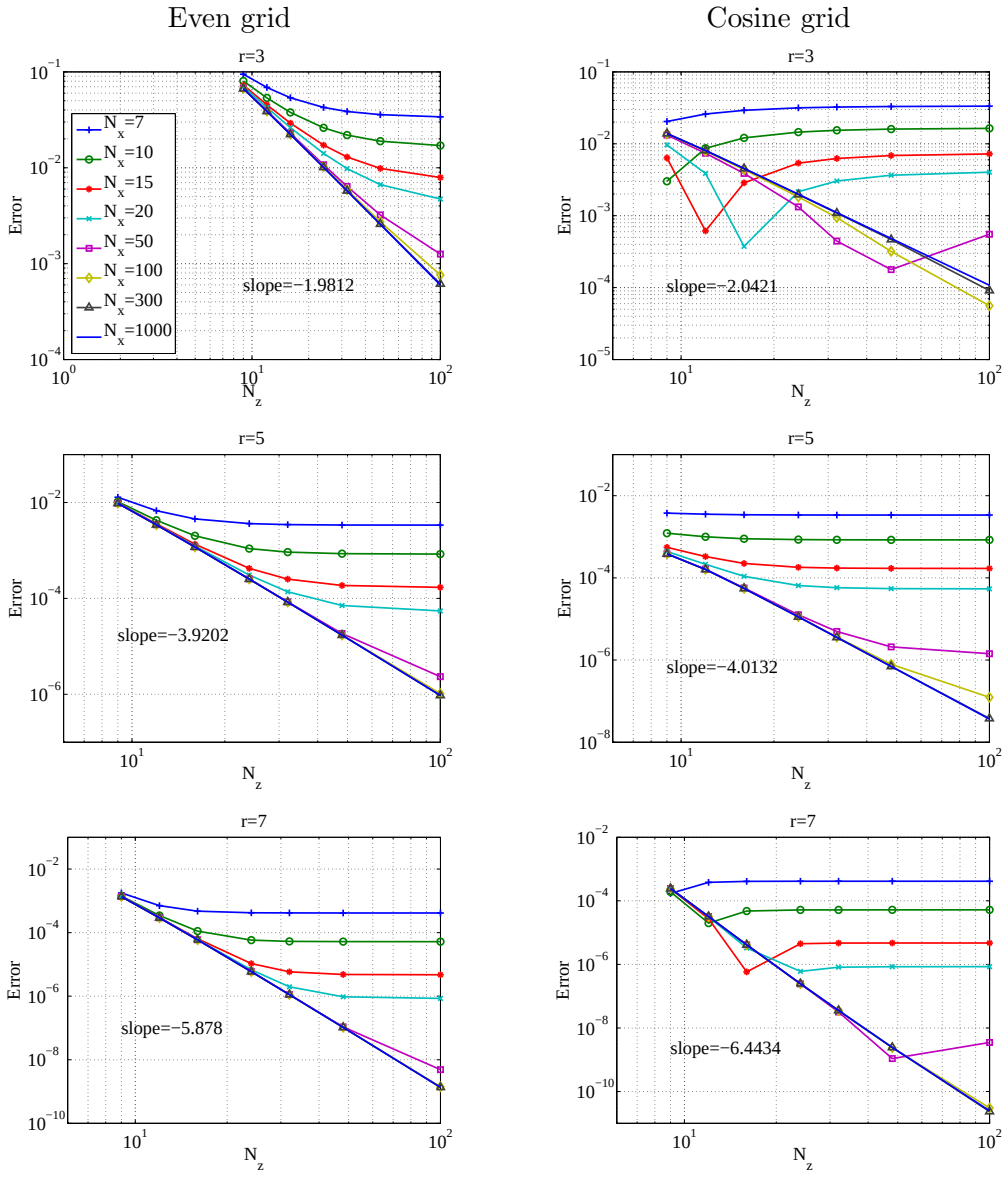

Figure 2. Convergence of the linear $\tilde{w}$ calculated from $\tilde{\phi}$ at $kh = 4$. The "slope" is the asymptotic slope of the line for $N_x = 1000$. N_z point values are at [9,12,16,24,32,48,100].

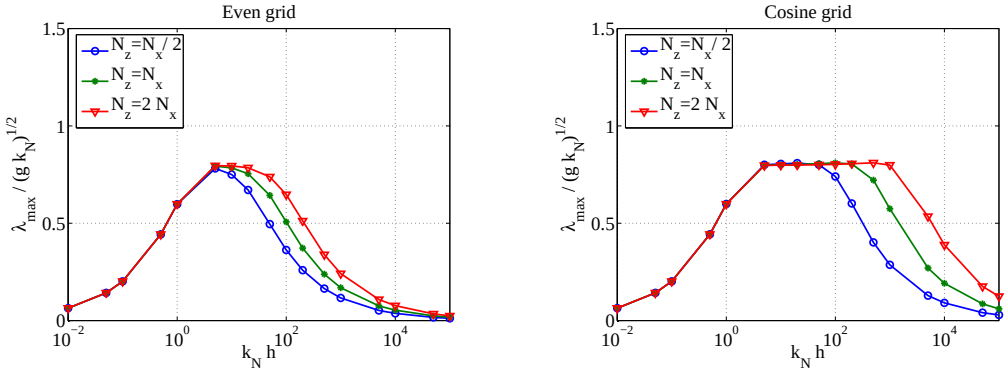

Figure 3. Maximum eigenvalues ($r = 3$) vs. relative Nyquist depth. Variation with vertical grid refinement shown by the different lines.

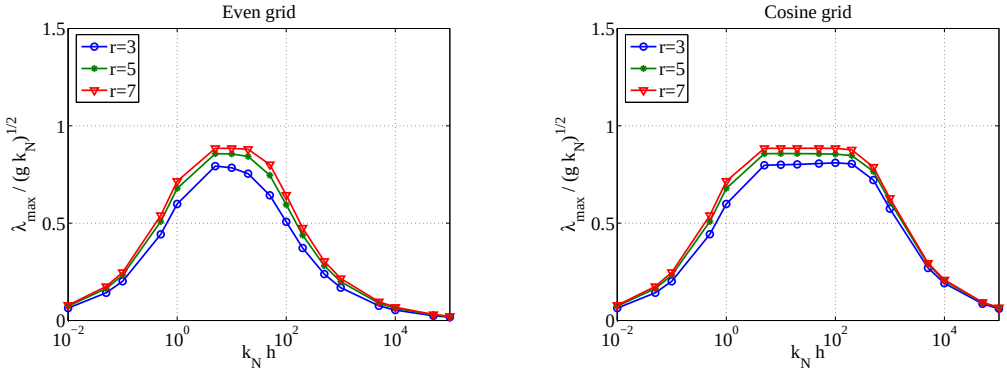

Figure 4. Maximum eigenvalues using $N_z = N_x$ grid points in the vertical vs. relative Nyquist depth. Variation with increasing order shown by the different lines.

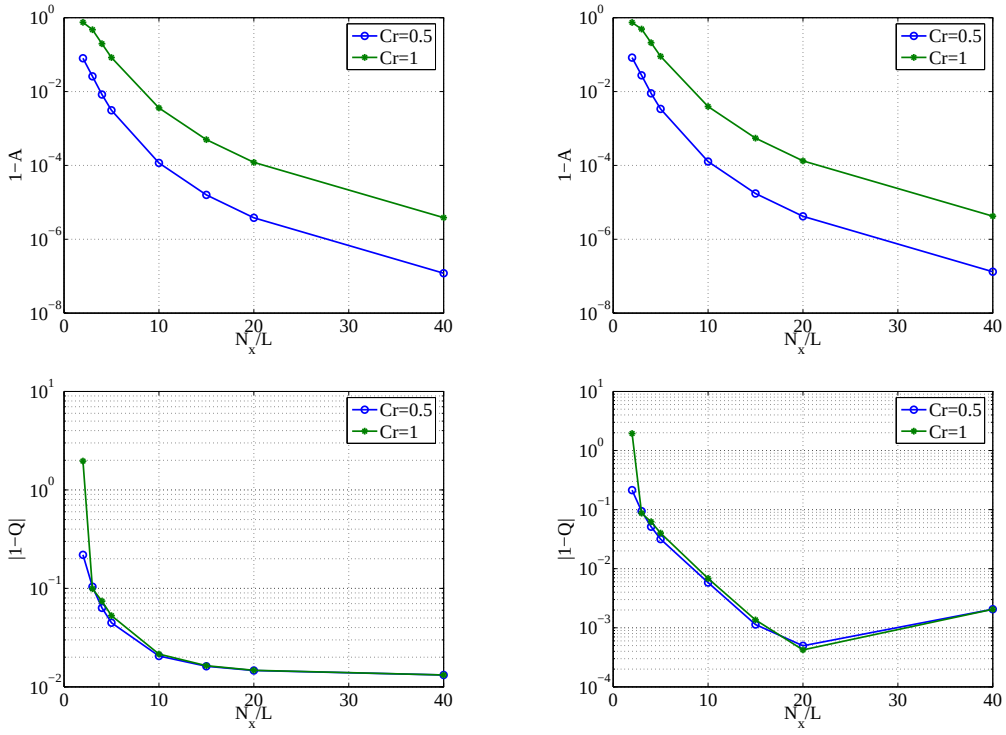

Figure 5. Relative amplitude and phase errors over one period as a function of resolution for two Courant numbers. 2nd-order model ($r = 3$) with $N_z = 15$ at $kh = 4$.

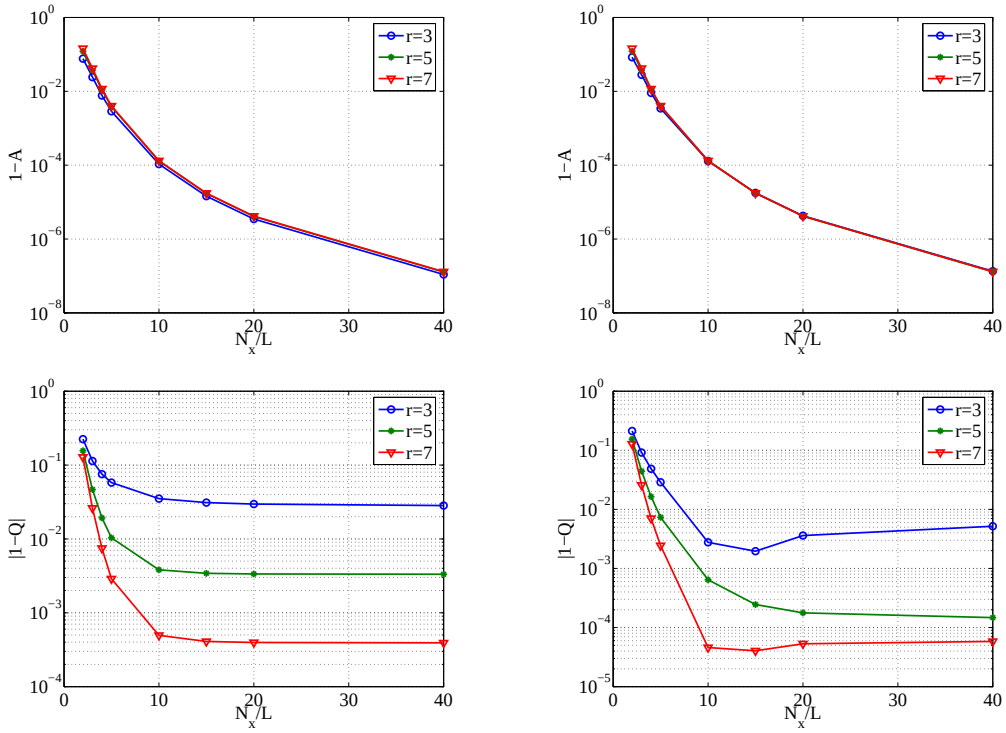

Figure 6. Relative amplitude and phase errors over one period as a function of horizontal resolution with $N_z = 10$ using 2nd- through 6th-order discretisations. For $Cr = 0.5$, $N_z = 10$, and $kh = 4$.

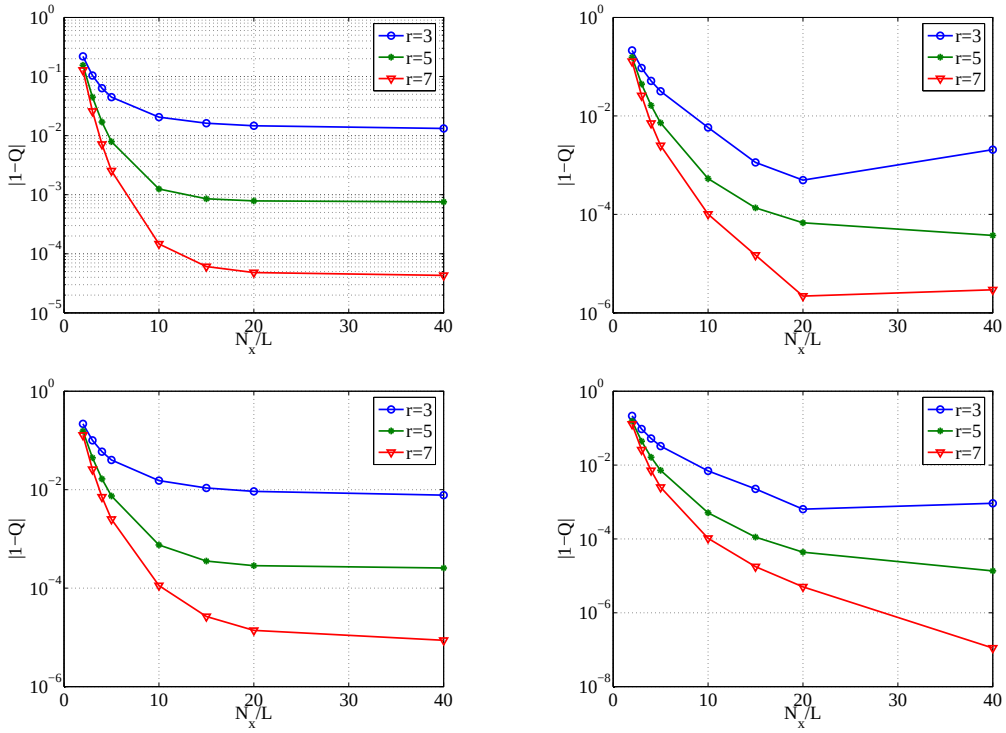

Figure 7. Relative phase errors over one period as a function of horizontal resolution with $N_z = 15$ & 20 using 2nd- through 6th-order discretisations. For $Cr = 0.5$, $N_z = 10$, and $kh = 4$.

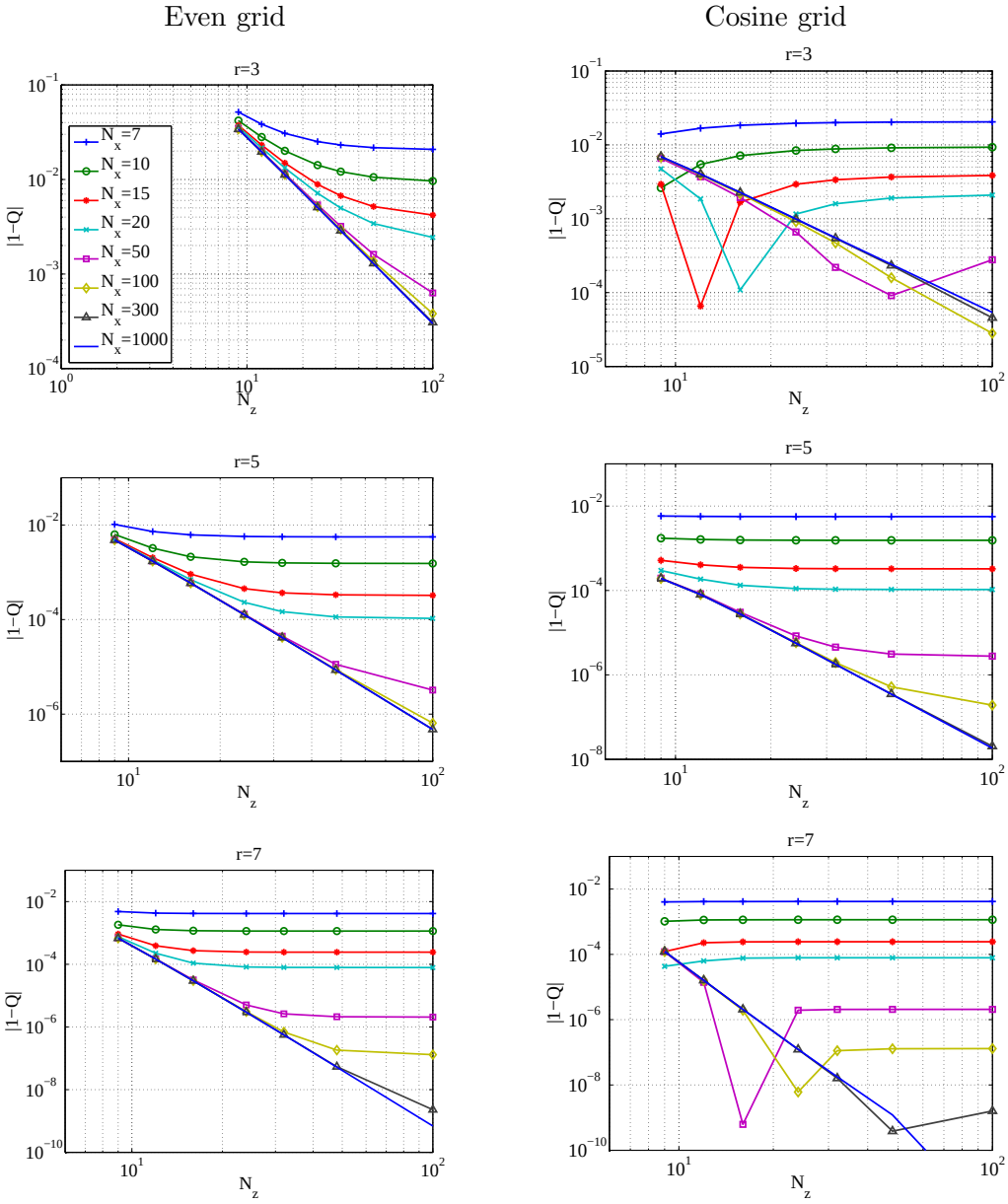

Figure 8. Linear dispersion errors for 2nd-, 4th- & 6th-order discretisatons as a function of resolution. The Courant number $Cr = 1$, and $kh = 4$.

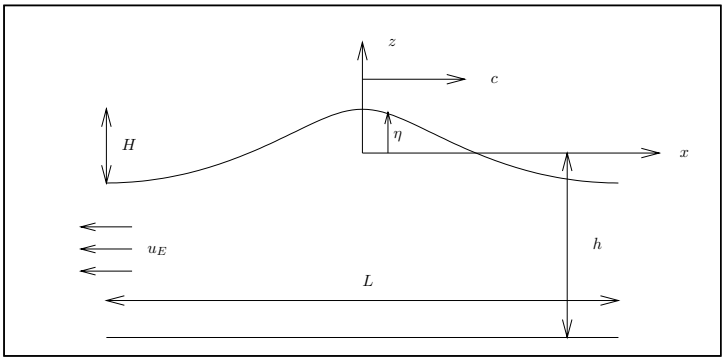

Figure 9. Definition sketch for the steady wave solution.

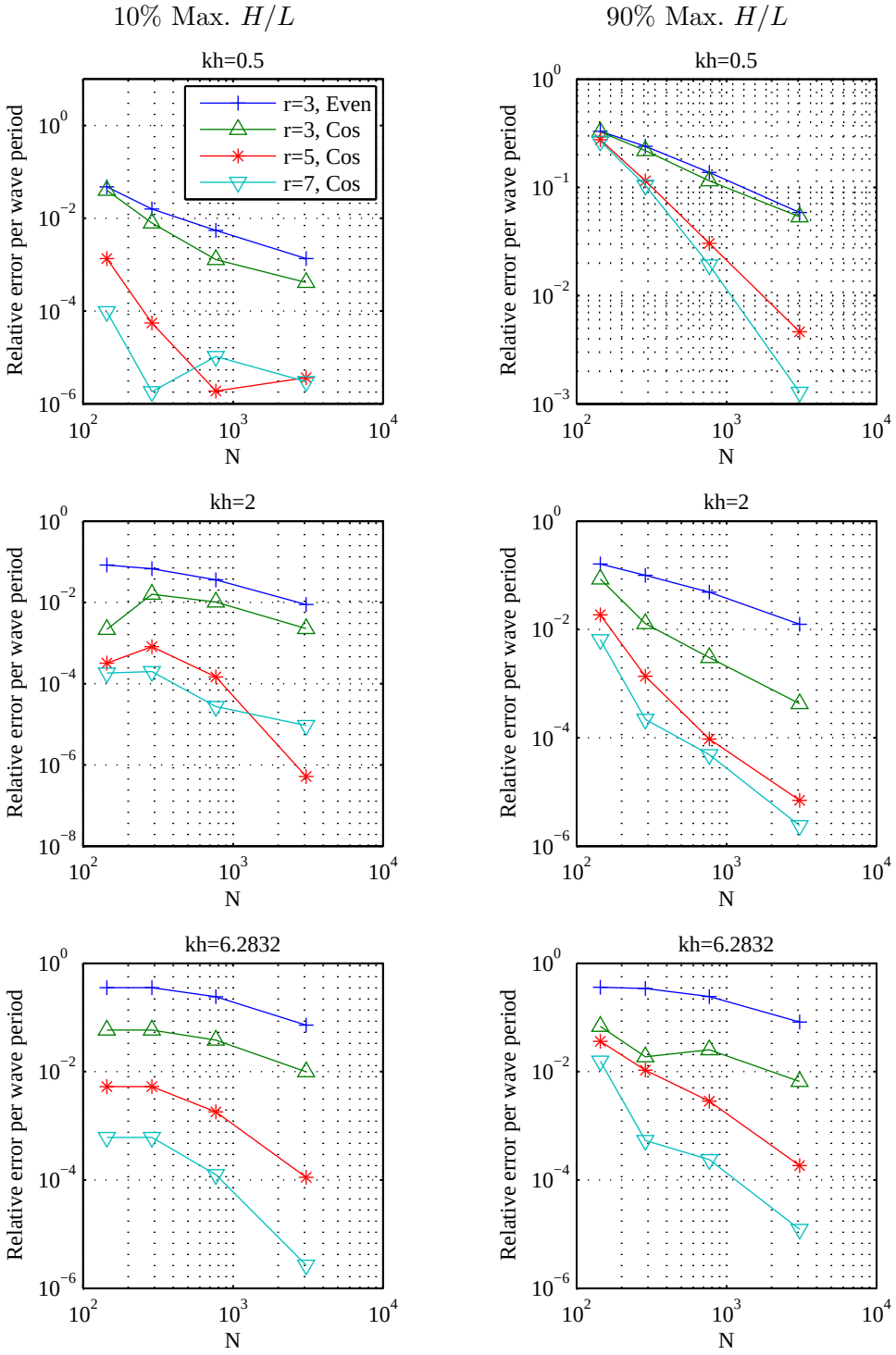

Figure 10. Nonlinear errors (28) for the four versions of the model listed in Table III. All plots follow the legend shown.

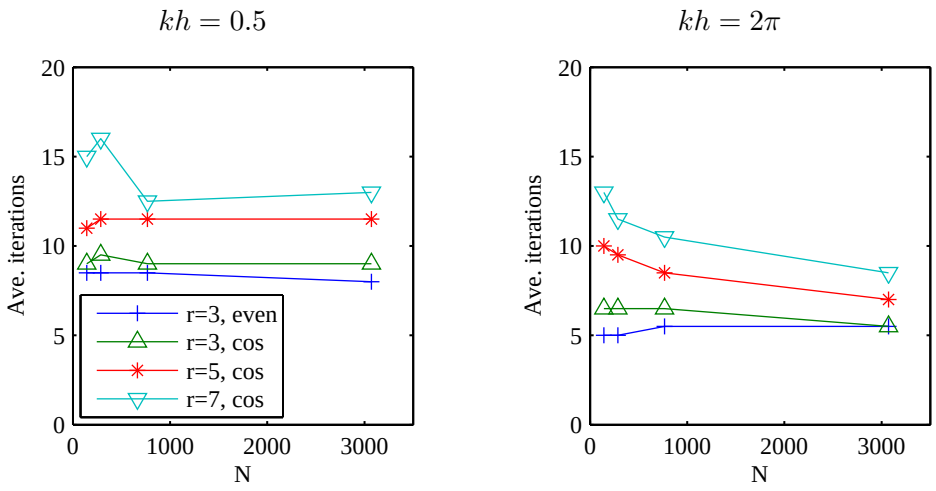

Figure 11. Average iteration counts for the 90% steepness case.