Finite difference solutions for nonlinear water waves using an immersed boundary method

Xu, Yan; Bingham, Harry B.; Shao, Yanlin

Published in:
International Journal for Numerical Methods in Fluids

Link to article, DOI:
[10.1002/fld.4922](https://doi.org/10.1002/fld.4922)

Publication date:
2021

Document Version
Peer reviewed version

[Link back to DTU Orbit](#)

Citation (APA):
Xu, Y., Bingham, H. B., & Shao, Y. (2021). Finite difference solutions for nonlinear water waves using an immersed boundary method. *International Journal for Numerical Methods in Fluids*, 93(4), 1143-1162. <https://doi.org/10.1002/fld.4922>

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Finite difference solutions for nonlinear water waves using an immersed boundary method

Yan Xu, Harry B. Bingham, Yanlin Shao

Department of Mechanical Engineering, Technical University of Denmark

*yanxucn@mek.dtu.dk, hbb@mek.dtu.dk, ylsao@mek.dtu.dk

Abstract

In this paper, high-order finite difference solutions of the exact potential-flow problem for nonlinear water waves are developed which treat both the free surface and the uneven sea bottom boundary conditions using an Immersed Boundary Method (IBM).

The convergence, accuracy and stability of this approach is first established for the linear problem, using various orders of scheme and different grid discretization strategies. The nonlinear wave problem on a flat bottom is then considered, and the solutions are compared with the highly accurate stream function theory solution. The convergence performance of the numerical solution for this nonlinear wave problem is established. Finally, linear wave shoaling and nonlinear wave propagation over a submerged bar are also tested. A preconditioned iterative solution strategy is also developed and shown to provide optimal scaling of the solution effort with increasing number of unknowns.

Compared to existing free surface tracking methods, the IBM retains the same level of accuracy and stability when solving the linear problem. For the nonlinear problem, the IBM behaves reasonably in terms of accuracy, but generally with slightly higher computational effort. When it comes to the wave-body interaction problem, it is expected that the IBM will be advantageous compared to the σ -transform method, since it removes the necessity of constructing an artificial continuous free-surface inside the body.

This article has been accepted for publication and undergone full peer review but has not been through the copyediting, typesetting, pagination and proofreading process which may lead to differences between this version and the Version of Record. Please cite this article as doi: 10.1002/fld.4922

1. Introduction

Nonlinear water wave propagation, wave-wave and wave-structure interaction are critical issues for marine engineers and designers. The Navier-Stokes (N-S) equations give an accurate physical description of such problems, but it is still very demanding to compute a direct numerical solution, even for relatively small Reynolds numbers ($Re = O(1000)$). For larger Reynolds numbers of 10^6 to 10^{10} , which are typical in maritime engineering, Reynolds-averaging and large-eddy simulation methods are applied in order to reduce the resolution requirements, but practical solutions are still quite time consuming. For these problems, where viscous effects are largely confined to very thin boundary layers, potential flow methods can provide an efficient alternative.

A number of numerical methods have been developed for solving the Laplace equation (potential flow). The most widely used method in the maritime industry is the Boundary Element Method (BEM) based on Green's theorem. The biggest advantage of the BEM is the projection of the three-dimensional (3D) problem onto the boundary surface of the fluid volume, so that the meshing effort and total number of unknowns are reduced dramatically. However, the matrix to be solved by the traditional BEM is full, leading to a solution effort and memory requirement which both scale with $O(N^2)$, where N is number of unknowns on the boundary surface. Sparsification techniques exist however, which can reduce the scaling to $O(N \log N)$, or even $O(N)$. Grilli et al. [1] applied the pre-corrected fast Fourier transform method to the BEM, reducing the operational counts and memory requirement to $O(N \log N)$. Y.J. Liu & N. Nishimura [2] further reduced the needs to $O(N)$ by implementing the fast multipole method. The conventional BEM method can robustly provide second-order numerical accuracy, and third-order convergence has also been demonstrated, for example by Yan & Liu [3].

A great many volume-discretization methods also exist including: finite difference methods, finite element methods, finite volume methods and spectral element methods. These methods can choose the order of accuracy flexibly, and are generally able to achieve high-order accuracy. Wu & Hu [4] is an example of a finite element method solution for fully non-linear wave-structure interaction. In their work, the grid was fully re-meshed following the wave and body motion, while the information was transformed from the old grid to the new grid by interpolation. R. Cienfuegos et al. [5] developed a fourth-order compact finite volume scheme for fully nonlinear and weakly dispersive Boussinesq-type equations. Engsig-Karup et al. [6] presented a robust spectral element method for fully nonlinear water wave propagation, which

balances high accuracy with the use of an unstructured mesh that can be adapted to arbitrary geometry.

The Harmonic Polynomial Cell (HPC) method is a relatively new discretization method, developed by Shao & Faltinsen [7]. The general idea is to discretize the fluid domain by overlapping cells, and interpolate the solution by a set of harmonic polynomials on each cell. These interpolating polynomials are elementary solutions to the Laplace equation, with the most convenient choice reaching 3rd to 4th order accuracy. To solve the moving body in an infinite fluid problem, Hansen et al. [8] applied the Immersed Boundary Method (IBM) to HPC and determined the accuracy of this approach by testing a fixed cylinder in an oscillatory flow. More recently, the same approach was also adapted to track the free surface by Hansen et al. [9] and furtherly developed into an Immersed Boundary Overlapping Grid (IBOG) strategy by Ma et al. [10].

Bingham & Zhang [11] and Engsig-Karup et al. [12] developed a nonlinear wave solver based on the high-order finite difference method and applied a σ -transform in the vertical direction to obtain a time-constant computational domain. They also developed a multigrid preconditioned GMRES iterative solver that makes the solution competitive with BEM methods in terms of computational efficiency to achieve a given accuracy. Based on these works, Kontos et al. [13] proposed a simplified version of the Weighted Essentially Non-Oscillatory (WENO) finite difference scheme to solve the nonlinear wave propagation problem in a moving frame of reference that mimics a forward speed seakeeping problem. This work was also extended to 2D wave-structure interaction by Hicks [14] who introduced a weighted least-squares based IBM to capture the moving structure. The work presented in this paper is motivated by two observations related to the nonlinear wave-structure interaction solutions described above. First, the combination of an IBM for the body-boundary and a σ -transform for the free-surface and bottom boundary, as adopted by [15], requires the construction of an artificial C^2 continuous free-surface inside the body. While this can be done with some success in 2D, it is likely to be much more problematic in 3D. Secondly, the work of [8] shows quite promising results for the application of an IBM technique for all domain boundaries in the context of the HPC method. Thus, the goal of the present paper is to investigate the accuracy and efficiency of a finite difference method solution where all of the fluid domain boundaries are introduced using the IBM method. We establish the accuracy and convergence of the method for linear and nonlinear wave propagation on both constant- and variable-depth fluids, demonstrating that the accuracy is comparable to that achieved by the σ -transform approach. A preconditioned GMRES iterative solution strategy is also developed and shown to give optimal scaling of the solution effort in 2D. The introduction of a floating body and extension to 3D is ongoing work and will be presented in follow-up papers.

This paper is organized as follows. Section 2 reviews the formulation of the potential flow problem. Section 3 introduces the numerical method, highlighting the layout of the immersed boundary grid around the free surface and a nonplanar seafloor. In section 4, the accuracy and stability of the solution for the linear problem is confirmed. In section 5, nonlinear wave propagating problems are solved and the results are compared with stream function solutions to investigate the convergence of the numerical scheme for nonlinear water-wave modelling. The corresponding computational effort based on the GMRES solver are also presented. In Section 6 and 7, a linear shoaling wave test and a nonlinear submerged bar test are carried out, respectively, followed by conclusions in Section 8.

2. Formulation

As shown in Fig. 1, a Cartesian coordinate system is chosen with an origin O , a vertical and upwards z -axis and an x - y plane corresponding to the mean water level. In this paper only the two-dimensional (2D) problem is considered, thus the y axis will be ignored in the following. The still water depth is $h(x)$, and the free surface position is given by $z = \eta(x, t)$. Considering the irrotational flow of an incompressible inviscid fluid, a velocity potential $\phi(x, z, t)$ can be found throughout the fluid domain, and the partial derivatives of ϕ with respect to the two space variables result in the corresponding velocity in each direction, $[u, w] = (\partial_x \phi, \partial_z \phi)$.

The governing equation in the fluid domain is the Laplace equation (1b). With all known boundary conditions, the potential value throughout the fluid domain can be obtained by solving the governing equation. When the boundary value problem is solved in the time domain, the boundary conditions include the potential value on the free surface $\tilde{\phi} = \phi(x, \eta, t)$ and the kinematic bottom boundary condition, as written in equation (1a) and (1c) respectively. At the horizontal truncation boundaries, we will either apply Neumann or periodic conditions, depending on the specific case that we are going to solve.

$$\phi = \tilde{\phi}, \quad z = \eta \quad (1a)$$

$$\partial_{xx}\phi + \partial_{zz}\phi = 0, \quad -h \leq z < \eta \quad (1b)$$

$$\partial_z \phi + \partial_x h \cdot \partial_x \phi = 0, \quad z = -h \quad (1c)$$

The kinematic and dynamic boundary conditions that govern the free surface evolution can be expressed as:

$$\eta_t = -\partial_x \eta \cdot \partial_x \tilde{\phi} + \tilde{w}(1 + \partial_x \eta \cdot \partial_x \eta) \quad (2a)$$

$$\partial_t \tilde{\phi} = -g\eta - \frac{1}{2}(\partial_x \tilde{\phi} \cdot \partial_x \tilde{\phi} - \tilde{w}^2(1 + \partial_x \eta \cdot \partial_x \eta)) \quad (2b)$$

where g is the gravitational acceleration, and $\tilde{w} = w(x, \eta, t)$ is the vertical velocity on the free surface.

3. Numerical method

The above boundary value problem is solved by a *method of lines* approach. The spatial derivatives in the partial differential equations are discretized first, leaving the time variable continuous. Specifically, the Laplace problem is first solved by a finite difference method and the potential ϕ in the fluid domain is obtained. Then the spatial derivative terms in the two free surface equations can be calculated by interpolation of ϕ , and the time integration is finally conducted by the classical explicit four-stage fourth-order Runge-Kutta scheme. A gentle 10th order Savitzky-Golay type smoothing function [11] is applied to prevent instability in strongly nonlinear situations.

3.1 Finite difference discretization

We consider a 2D computational domain, as shown in Fig. 2, which is extended in the vertical direction to enclose the entire physical fluid domain and meet the needs of the immersed boundary method. The free surface boundary, the bottom boundary and two vertical truncation boundaries form the complete boundary of the fluid domain.

The computational domain is discretized into a fixed background structured grid, with $N = N_x \cdot N_z$ points, where N_x and N_z are the number of points along the x and z axes respectively. The intersections between the free surface and vertical grid lines define the free surface points, which are shown as the blue crosses in Fig. 2. Following the same principle, a series of bottom points, marked by black crosses, can be found on the bottom boundary. The points in the fluid domain are defined as fluid points. All the remaining points are called dummy points. It is worth noting in Fig. 2 that the free surface points usually lie somewhere between two adjacent grid points, while it is rare for them to fall exactly onto the background grid.

The dummy points are not involved in any calculations, but in order to maintain a convenient solution logic for the time stepping process, these dummy points are kept in the system, and the potential values are simply set to zero. The potential values on each fluid point are the unknowns, and together with the potential values on dummy points, they can be collected into an $N \times 1$ column vector ϕ .

The basic idea of the finite difference method is to develop N finite difference interpolation equations by means of Taylor series expansion and the inversion of a small linear system for all the N unknown values. The result is expressed by an $N \times N$ coefficient matrix $\mathbf{A}$ and an $N \times 1$ vector $\mathbf{d}$ as follows:

$$\mathbf{A} \phi = \mathbf{d}$$

To build an r -order finite difference scheme, an $r + 1$ point stencil is needed. Thus for the fluid points, the second-derivative operator in both the x and the z directions are built, each of which has a stencil size of $r + 1$, and the Laplace operator is the sum. For the interior fluid points, centered schemes are preferred, while off-centered schemes are applied when the fluid points are near the boundaries.

3.2 Detailed discussion of the boundary conditions

Both Dirichlet-type and Neumann-type boundary conditions appear in the problem. For both of them, a robust ghost point method similar to that developed in [11] is implemented. The main idea is to add fictitious computational points outside the fluid domain. These points are called ghost points, and their values are found from the boundary conditions. This ensures that both the boundary conditions and the Laplace equation are satisfied at all boundary points.

Considering the free surface Dirichlet condition Eq. (1a) first, one layer of ghost points is created from the dummy points just above the free surface, as indicated in Fig. 3 (a) by the red square points. Using a second-order scheme as an example, the three points located in the blue dotted box can be used to interpolate the value of the potential at the indicated free surface point which is also located in the blue dotted box and marked by the blue cross. Setting this interpolation equal to the known value then represents the equation imposed at this ghost point location.

The free surface Neumann condition (1c) is more complex. The $\partial_z \phi$ value can be interpolated as described above, using the points in the blue dotted box shown in Fig. 3(b). To interpolate $\partial_x \phi$ however, the 9-point, 2D stencil in the red dotted box is adopted.

3.3 Detailed discussion of the immersed boundary method

The ghost point method described in the previous section already refers to a boundary that is immersed in the background grid. This section will explain the details of how the interpolation equations for the fluid points near the boundary are developed. Without loss of generality, the free surface boundary is selected as the study object in this section.

Fig. 4 shows a local grid map near the free surface. The red and blue dotted boxes represent the second order interpolation stencils for $\partial_{xx} \phi$ and $\partial_{zz} \phi$ at the central fluid point respectively.

However, if a higher-order scheme was used, there would not be enough points in the horizontal direction to build the stencil without involving dummy points, when the wave is steep or the resolution is too coarse. To address this problem, auxiliary points

are introduced as needed. Fig. 5 shows fourth-order interpolation stencils for $\partial_{xx}\phi$ and $\partial_{zz}\phi$ at the same fluid point as above. The green points are the auxiliary points and are forced to satisfy the Laplace equation.

By assigning the Laplace equation to the auxiliary points, the fluid is de facto extended into the air. To ensure that all fluid points have viable neighbors included in their original centered stencils, all dummy points which appear in these stencils are converted to auxiliary points. New, possibly off-centered stencils are then created for the auxiliary points themselves. In the program, the central difference stencils of the horizontal derivatives at each fluid point are first assembled, then the intersection between those stencils' points and the dummy points are identified and converted to auxiliary points.

Considering again the situation shown in Fig. 5, we have found that the solution can be improved if the ghost point under the auxiliary points exchanges roles with the top auxiliary point in the same column. This strategy leads to a more continuous extension of the potential solution into the air.

It is noticeable that all these point indexes can be changed with time stepping. A fluid point at the current time step can become a ghost point, an auxiliary, or a dummy point in the next step. The consequences of this will be discussed in the following sections.

3.4 Grid stretching strategy

The resolution in the vertical direction can be enhanced around the immersed boundary area, which can improve the accuracy without sacrificing efficiency. Considering the cases that will be tested in this paper, the bottom boundary has less impact on accuracy than the free surface. Thus the stretching strategy in this paper only refines the grid near the free surface. Fig. 6 shows an example of a uniform grid and a stretched grid.

3.5 Iterative solution of the linear system

The coefficient matrix A is a sparse matrix with a main diagonal bandwidth of about $2N_z$. The number of nonzero elements in A depends on the number of dummy points in the computational domain, but it will not exceed a maximum of $2rN$. The robust Generalized Minimal Residual (GMRES) method is employed here to solve this linear system.

Furthermore, the linear system should be first preconditioned on the left to minimize the required number of iterations. Learning from the experience of the σ transform method, a linearized second-order version of A is used as the basis of the preconditioning matrix A_{Pre} . But compared to the σ transform method, the immersed boundary method requires the modification on the basis at every time step

due to the changing point distribution in the field. Namely, every time the system is to be solved, the dummy point information should be first erased from the basis by setting the main diagonal element to one and all the rest of the elements in the same row and column to zero. The free surface ghost points in matrix A are initially fluid points and satisfy the Laplace equation in the basis, but experience shows that they should instead be treated as dummy points in matrix A_{Pre} in order to improve the efficiency of GMRES. By using dots to represent the nonzero elements, Fig. 7 shows the structures of the A_{Pre} basis, the corresponding A_{Pre} and the second order scheme A matrix for a small 5×6 periodic wave problem. For a second order scheme, the difference between A_{Pre} and A only involves the free surface ghost points. There are unit coefficients in the A_{Pre} , but in A matrix there are interpolation coefficients for imposing the free surface Dirichlet condition.

Since the current study is limited to 2D problems, the solution of the preconditioning step is conducted using a direct sparse-matrix solver in Matlab. For a 3-dimensional problem, one should consider improving this process by implementing a multigrid strategy. This topic will be discussed in a follow-up paper.

4. Linear accuracy and stability

First, the linearized version of the problem is solved to verify our approach and quantify its performance. The main goal here is to confirm that the immersed boundary scheme retains the same accuracy, convergence and stability properties as the σ -transform scheme adopted in [11].

The linearized boundary value problem in a fluid domain with a flat bottom is expressed as:

$$\phi = \tilde{\phi}, \quad z = 0 \quad (3a)$$

$$\nabla^2 \phi + \partial_{zz} \phi = 0, \quad -h \leq z < 0 \quad (3b)$$

$$\partial_z \phi = 0, \quad z = -h \quad (3c)$$

$$\eta_t = \tilde{w} \quad (3d)$$

$$\partial_t \tilde{\phi} = -g\eta, \quad (3e)$$

where $\tilde{\phi} = \phi(x, 0, t)$ and $\tilde{w} = w(x, 0, t)$.

The analytical solution for a traveling wave is given by:

$$\eta = Re \left\{ \frac{H}{2} e^{i(kx - \omega t)} \right\} \quad (4a)$$

$$\phi = \operatorname{Re} \left\{ -\frac{igH}{\omega} \frac{\cosh[k(z+h)]}{2 \cosh(kh)} e^{i(kx-\omega t)} \right\} \quad (4b)$$

where H is the wave height, the wavelength $L = 2\pi/k$, the wave period $T = 2\pi/\omega$, and ω and k are the wave frequency and wave number respectively. See for example [16] for a review of linear wave theory.

4.1 Convergence

The accuracy of the implemented Laplace equation solver is evaluated by comparing the numerical solution of $\tilde{w}$ with the exact solution $\tilde{w} = k \tanh(kh)\tilde{\phi}$.

More specifically, a relatively coarse resolution grid is first employed to solve the Laplace equation (3b) and get the ϕ solution in the fluid domain for a given time instant. Then $\tilde{w}$ at each free surface point can be interpolated from ϕ . Finally the largest relative error ($|\tilde{w} - \tilde{w}_e|/\tilde{w}_e$) along a whole wave length can be found to represent the accuracy level of this $\tilde{w}$ solution. After that, the resolution of the grid is increased and the exercise is repeated.

Fig. 8 shows the results of a case with wavelength $L = 1$, water depth $h = 2/\pi$ and $kh = 4$. The total length of the computational domain $L_x = 1$, i.e. one wavelength. The total height of the computational domain $L_z = 1.2$, ensuring that the free surface and free surface ghost points can be contained in the domain while their exact positions are not predetermined. The plots have the same structure as Fig. 2 of [11], to allow for a direct comparison with the corresponding results from the σ transform method. Each plot in this figure shows the convergence curves for an r -point, $(r-1)$ -order scheme, with increasing vertical resolution N_z , while the different lines show increasing values of horizontal resolution N_x . It is notable that the N_z here is not the total number of vertical points, but excludes the dummy points. Using the least squares method, an approximate value for the asymptotic slope of the finest resolution is also shown in each plot, and all of them satisfy the expected convergence rate.

Firstly, it can be seen that large oscillations appear in all the convergence curves for uniform grids. The changing position of the mean water level in the background grid gives rise to the change of interpolation stencils that relate to the free surface equations. Take the second order scheme case ($r = 3$) for instance, where there are only three points in the interpolation stencil for $\tilde{w}$. The location of the free surface between the three points will lead to different factors in the truncation error as the point moves from very close to the middle point (a centered stencil) out to very close to the top point where it becomes fully one-sided. Therefore, the first plot shows the strongest oscillations, while they gradually reduce for the higher order schemes, since the relative change in the truncation errors due to this effect reduces with the order of

the scheme. For most of the curves, the frequency of oscillation increases with increasing N_z . Since the vertical grid spacing becomes smaller at high resolution, the relative change of position of the free surface within the enclosing horizontal grid lines becomes larger. The stretched grids generally show much smaller oscillations compared to the uniform grids.

In terms of the magnitude of the errors, another notable point is that the upper bound of the error oscillation for the even grid gives a similar solution to the corresponding stretched grid solution, but the lower points actually can produce some more accurate solutions than the stretched grid. Compared with the σ transform method [11], the IBM generally shows a same level of accuracy, while in some situations the IBM can be more accurate.

To further study the accuracy of the linear problem, the dispersion errors as a function of relative water depth kh are also studied. The dispersion relation of a linear wave can be written as:

$$\frac{\tilde{\omega}}{k\tilde{\phi}} = \tanh(kh) \quad (5)$$

The dispersion error is measured by the accuracy of the left hand side. In order to change the value of kh , only the value of h is changed, while the wave number is kept constant at one. The results are shown in Fig. 9.

For each plot, the order of the scheme, the grid discretization strategy and the number of horizontal points per wavelength N_x are fixed. Each curve represents the dispersion errors under one vertical resolution. Several different N_z values are plotted to demonstrate the influence of mesh resolution in the vertical direction. Each point represents the maximum relative error along the free surface in each case. The length of the computation domain is set to be one wavelength, and the height is set such that the linearized free surface is located exactly in the middle between the top two horizontal grid lines. Thus, there is no oscillation phenomenon in this case.

The two plots on the top show the dispersion errors using 2nd order schemes on a uniform and a vertically stretched grid respectively, both using 75 grid points in the horizontal direction. The grid stretching strategy is the same as that used in the previous study, with a hyperbolic tangent stretching function around the free surface. It can be seen that the dispersion errors decrease with mesh refinement, but increase with increasing relative water depth. The first plot shows that the dispersion error on a uniform grid can barely be held within 10^{-3} for short waves ($kh > 30$) using a quite high resolution ($N_z=150$), while the second plot shows the resolution requirement to reach same level of accuracy is only about half ($N_z=75$) after stretching the grid.

Similarly, the lower two plots show the results based on 4th and 6th order schemes on stretched grids respectively. These two plots show that the dispersion errors have the

same trend, but compared with the second order results it is clear that applying high order schemes can significantly improve the accuracy. For relatively low resolution ($N_x = 9, N_z = 23$), the error of the 6th order scheme still remains below 10^{-3} in the short wave situation.

These results will be considered in the following nonlinear propagating wave simulations, where a similar level of accuracy should first be expected under similar vertical resolution and relative water depth.

4.2 Stability

Since a method of lines approach is applied, linear stability can be quantified by investigating the semi-discrete form of the time stepping equations:

$$\frac{\delta}{\delta t} \begin{bmatrix} \check{\eta} \\ \check{\phi} \end{bmatrix} = \begin{bmatrix} 0 & J_{12} \\ -g & 0 \end{bmatrix} \begin{bmatrix} \check{\eta} \\ \check{\phi} \end{bmatrix} \quad (6)$$

where $\check{\eta}$ and $\check{\phi}$ are the Fourier amplitudes of η and ϕ . Here J_{12} is an $N_x \times N_x$ matrix which satisfies $\tilde{w} = J_{12} \tilde{\phi}$ and can be constructed from the system matrix A and a free surface vertical velocity interpolation matrix D_z , as shown for example by [11]. The eigenvalues λ of the above $2N \times 2N$ block matrix, where $N = N_x N_z$, can be calculated under different orders of finite difference schemes. To ensure stability, the maximum magnitude of the eigenvalue of the above block matrix, $|\lambda|_{max}$, multiplied by the time step size, $|\lambda|_{max} \Delta t$, must be within the stability region of the specific time-stepping scheme. For the explicit fourth-order Runge-Kutta method used in our case, $|\lambda|_{max} \Delta t \leq 2\sqrt{2}$.

The Nyquist wave number is $k_N = \pi/L_x * (N_x - 1)$, and the Nyquist wave frequency is $\omega_N = \sqrt{gk_N \tanh(k_N h)} \cong \sqrt{gk_N}$. Fig. 10 plots $|\lambda|_{max}/\omega_N$ for a 2nd order scheme. Here the computational domain remains the same as in the previous section, with fixed $N_x = 24$, while N_z varies. The water depth is constant, but the location of the free surface within the grid lines is not predetermined. The horizontal length of the computational domain increases giving rise to a change in the Nyquist wave number k_N . Notably, there is no oscillation since the water depth and N_z are both fixed for each curve.

It can be seen from the plots that each curve starts from zero when the water depth is small, then goes up with $k_N h$ and reaches the maximum value just below 1.0, remaining at a constant for a while and reducing again as the water depth increases further until $|\lambda|_{max}/\omega_N$ drops back to the zero. The change of N_z only causes some offsets in the deeper water depth region, while the eigenvalue starts decreasing earlier when there are fewer grid points in the vertical direction. The overall trend of the

eigenvalues is very similar to that which was found for the σ transform method in Sec. 4.2 of [11].

Fig. 11 plots the $|\lambda|_{max}/\omega_N$ for different orders of scheme. Here the vertical grid number $N_z = N_x$, and all the results are normalized by the Nyquist frequency. It can be seen that the eigenvalue curves are in accordance with Fig. 10, while the eigenvalue increases with the order of the scheme. Compared with the σ transform method [11], the IBM using fourth and sixth order finite difference schemes generally produces larger eigenvalues at all the relative water depths, thus posing a more stricter on the time step size. However, since practical calculations are always limited by accuracy rather than stability requirements, this difference is not of practical significance. Given the similarity shown here between the linear accuracy and stability behavior of the current method and the sigma-transform method of [11], we conclude that adopting a hyperbolic Courant number $C_r = \frac{U\Delta t}{\Delta x}$, where U is the phase speed of the wave, of between 0.5 and 1 will generally give the expected overall accuracy.

In practice, we can use the accuracy analysis presented above to choose our grid resolution as follows. Considering the shortest linear wave component that we would like to resolve, and a desired accuracy level, we consult Fig. 8 to choose a horizontal grid spacing. Then by consulting Fig. 9, using the largest corresponding kh value for this wave, we choose the required resolution in the vertical.

5. Nonlinear accuracy and efficiency

The stream function theory method of [17] is applied to provide initial conditions and comparison benchmarks for the following nonlinear study. Given a wave height H , wave length L , water depth h and a mean Eulerian velocity u_E , a stream function solution can produce the wave elevation η and velocity potential ϕ function for a specific wave. It should be noted that the determination of wave height H is related to wave length L and water depth h in the above parameters, by considering the wave breaking limitation given by [18]. And u_E is set to zero in order to implement the periodic conditions on the two vertical boundaries.

To cover as much as possible the range of interest, the conditions in Table 1 are chosen for testing. Three different relative water depths kh are chosen, representing shallow, intermediate and deep water respectively. For each water depth, both 10% and 90% of maximum steepness cases are considered.

Using the same error evaluation method as in the linear convergence study, Fig. 12 shows the nonlinear convergence curves on a uniform grid for both the IBM and the σ -transform methods. Specifically, we specify the exact $\tilde{\phi}$ on the exact free surface

computed from stream function theory, and solve for the corresponding vertical velocity $\tilde{w}$. The error is then found by comparing to the exact value from stream function theory. For this plot, N_z is fixed while N_x is changing for each curve. The two methods can generally achieve the expected order of convergence before leveling off at an accuracy limit. For the 10% steepness waves, the IBM shows nearly the same level of accuracy as the σ -transform method, except in deep water where some oscillations occur in the IBM curves. For the 90% steepness waves, the IBM basically performs better than the σ -transform method, especially in the shallow water cases. This may be explained by the fact that the IBM applies slightly more centered interpolation stencils for the free surface vertical velocity than are applied by the σ -transform method.

Next, we introduce the time stepping. The four schemes shown in Table 2 are considered for each wave condition. In order to reduce time-stepping errors and focus mainly on the spatial errors, the Courant number is chosen to be $Cr=0.35$. To prevent instability for the extremely nonlinear case, a 10th order Savitzky-Golay low-pass filter is applied to both the wave elevation and the velocity potential on the free surface at each time step. For each test case, the accuracy of our solution is evaluated by computing the relative error per wave length as:

$$Error = \frac{\|w_3 - w_e\|_2}{3\|w_e\|_2}, \quad \|\cdot\|_2 = \left(\sum |\cdot|^2\right)^{1/2} \quad (7)$$

Where w_3 is the vertical velocity on the free surface after the case has been run for 3 periods, and w_e is the corresponding solution calculated by the stream function method. The results are shown in Fig. 13, in which the right column shows the strongly nonlinear situation and the left column shows the weakly nonlinear situation. The four points in each curve represent the four grid resolutions:

$$(N_x, N_z) = [(19,39), (39,39), (69,39), (109,39)].$$

Firstly, it can be seen from the left column that the relative error of a weakly nonlinear wave can reach a similar level to that achieved in the linear case, while the strongly nonlinear waves show a much larger accuracy loss of around two orders of magnitude. Nevertheless, the basic tendency in those plots are consistent with linear cases, and the higher order schemes are always more accurate than 2nd order schemes. Considering the effectiveness of the grid stretching strategy, only for intermediate and deep water waves the stretched grid show better accuracy than the even grid under high resolution. While for shallow water wave or low resolution situations the stretched grid fails to show much advantage.

From the perspective of energy conservation, Fig. 14 shows the evolution of the relative error in the numerical solution over ten wave periods. The total wave energy is calculated from [11],

$$E = \frac{\rho}{2} \int_{S_0} (\tilde{\phi} \eta_t + g \eta^2) dx \quad (8)$$

where S_0 is the still water free plane, and the two integrand terms represent the kinematic and potential energy respectively. The test water depth is set to $kh = 2$, the numerical resolution is set to $N_x = N_z = 39$, the Courant number is 0.35 and a vertically stretched grid is used. The two plots in Fig. 14 show the error results for small and large wave steepness respectively, each plot contains three error curves that represent the results of three different order of schemes. Considering the small steepness plot, the three error curves both increase quickly in the first five periods, then the error growth slows down and eventually reaches a level around 10^{-4} . Although the second order scheme shows more oscillation than the other two schemes, the order of scheme seems to have no decisive contribution to the energy conservation accuracy in the weakly nonlinear situation. However, the advantage of the high order scheme becomes visible in the large steepness case, especially during the first five periods. Besides that, in the large steepness situation the oscillation becomes stronger, and the error value itself becomes larger, eventually reaching nearly 10^{-2} .

As the nonlinear wave height increases, solving the problem becomes more time-consuming. Fig. 15 shows the iteration requirements for the 90% steepness wave, using the same schemes shown in Table 2. The relative residual tolerance for GMRES is set to be 10^{-7} , and the initial guess is a linear extrapolation from the previous two Runge-Kutta stages. It can be seen that the iteration counts are generally below 15 for all the second and fourth order schemes, and between 20 and 30 steps for the sixth order scheme. Solving the shallow water wave looks slightly more expensive than solving a deep water wave, especially in the high resolution sixth order scheme case. Nevertheless, for all the schemes the iteration count is essentially constant with resolution, which guaranties an optimal scaling of the solution effort.

6. Sample application to shoaling waves

The immersed boundary method implemented above has been validated for waves on a constant depth, but not yet for the effects of a non-uniform depth. To solve such problems, an $r \times r$ -point, 2D stencil around each bottom ghost point is assembled to interpolate the horizontal derivative of ϕ on the bottom boundary, using the numerical method described in Section 3.2.

The problem of a linear wave shoaling up a beach is considered, where the bathymetry is defined as:

$$h(x) = h_0 - \left\{ h_1 - \frac{h_1 - h_2}{2} \left[1 + \tanh \left(\frac{\sin\left(\frac{\pi x}{l}\right)}{1 - \left(\frac{2x}{l}\right)^2} \right) \right] \right\}, \quad -\frac{l}{2} \leq x \leq \frac{l}{2} \quad (8)$$

where h_0 is the mean water height in the computational domain, h_1 and h_2 are the bottom height at the two horizontal ends of computational domain respectively, and l is the total horizontal length of the domain. With $h_0 = 0.5$, $h_1 = 0$, $h_2 = 0.05$ and $l = 15$, a slowly changing bottom boundary with a mean slope of $h_x \approx 0.03$ can be defined.

A linear wave with $L = 1\text{m}$ and $T = 0.8014\text{s}$ is generated at the left vertical boundary of computational domain, using a relaxation zone method where at each stage of the time-integration, the surface elevation and potential are redefined by:

$$\eta(x, t) = C_r(x)\eta_e(x, t) + [1 - C_r(x)]\eta(x, t) \quad (9a)$$

$$\tilde{\phi}(x, t) = C_r(x)\tilde{\phi}_e(x, t) + [1 - C_r(x)]\tilde{\phi}(x, t) \quad (9b)$$

where the relaxation coefficient function C_r is defined as:

$$C_r(x) = \frac{\exp(x_R^{3.5}) - 1}{\exp(1) - 1}, \quad 0 \leq x_R \leq 1 \quad (10)$$

Here x_R is the local coordinate within the relaxation zone, after scaling the real length into a unit length and setting $x_R = 0$ at the start position of the wave generating zone. The length of the relaxation zone is two wavelengths.

For absorption, artificial damping terms are added to the two free surface boundary equations, using the method proposed by Clamond et al [19]

$$\eta_t = -\nabla\eta \cdot \nabla\tilde{\phi} + \tilde{w}(1 + \nabla\eta \cdot \nabla\eta) - \gamma\eta \quad (11a)$$

$$\partial_t\tilde{\phi} = -g\eta - \frac{1}{2}(\nabla\tilde{\phi} \cdot \nabla\tilde{\phi} - \nabla\tilde{w}^2(1 + \nabla\eta \cdot \nabla\eta)) - \nabla^{-1} \cdot (\gamma\nabla\tilde{\phi}). \quad (11b)$$

Here the operator ∇^{-1} is the inverse of the horizontal derivative operator coefficient matrix and the damping coefficient γ is defined as:

$$\gamma(x) = \omega \cdot \frac{\exp(x_R^2) - 1}{\exp(1) - 1}, \quad 0 \leq x_R \leq 1 \quad (12)$$

where x_R is the local coordinate within the damping zone, and ω is the inlet wave frequency. The length of the absorption zone is also set to two wavelengths.

After the wave achieves a steady state, the wave elevation time series at each horizontal position is recorded for 4 periods. By least-squares fitting to a sinusoid at the incident wave frequency, a harmonic analysis is performed over the wave

elevation time series at each horizontal discrete point (x) to determine the local wave amplitude. Fig. 16 compares the numerical results with the analytical solution (13), see for example [16]. In the plot, the curves represent the ratio of local amplitude to the incident wave amplitude during the shoaling process, which is given by

$$\frac{a(x)}{a_0} = \left(\frac{k(x) \left(1 + \frac{2k_0 h_0}{\sin(2k_0 h_0)} \right)}{k_0 \left(1 + \frac{2k(x) h(x)}{\sin(2k(x) h(x))} \right)} \right)^{\frac{1}{2}} \quad (13)$$

In all these calculations, the mesh in the vertical direction is stretched and the resolution is set to $N_z = 43$. The Courant number based on the deep water resolution is set to $Cr = 0.5$. The N_x in the legend represent the number of points in horizontal direction per wave length.

For all numerical solutions, the accuracy is high at the deep water end, while differences appear in the shallow water zone. Comparing the three fourth order results, the solutions clearly converge to the analytical one when the resolution is refined. By comparing the three results where $N_x = 18$, the accuracy is also improved by using higher order schemes. Also, by comparing the second order $N_x = 18$ curve with the fourth order $N_x = 13$ curve, or by comparing the fourth order $N_x = 23$ curve with the sixth order $N_x = 18$ curve, the results also indicate that the benefit from higher order schemes is more significant than resolution refinement in terms of accuracy.

7. Harmonic generation over a submerged bar

Finally a nonlinear wave propagation test over a submerged bar is conducted to validate the present numerical scheme for solving general nonlinear water wave propagation problems. The same wave generation and absorption methods are utilized as in the previous shoaling wave case. The nonlinear incident wave is calculated using stream function theory.

The experimental data from [20] is adopted for a comparison. In that experiment, the mean water level is 0.4m, the incident monochromatic nonlinear wave has a 0.02m wave height and wave period of 2.02s. The submerged bar is 0.3m high, the flat top of the bar is 2m long, the up-slope is 0.05 and the down-slope is 0.1. The bar starts at the 6m position after wave generating zone and there is an 8m long flat bottom after the bar. Details of the experiment can be found in [21]. The numerical computational domain follows the experimental bathymetry but adds a wave generating zone and wave absorption zone at the both ends of the domain, both of them are 2 wavelengths long. The top plot in Fig. 17 shows the bottom layout and a snapshot of the steady state surface elevation.

The horizontal grid points number $N_x = 674$, the vertical grid points number $N_z = 54$, the Courant number $Cr=0.35$ at the deep water zone. The grid is clustered around the free surface. A 10th order Savitzky-Golay smoothing filter is applied at each time step to keep the wave propagation stable. A harmonic analysis is performed by making a least-squares fit as described above, except here we include the first five multiples of the primary harmonic frequency.

The lower plot of Fig. 17 shows the harmonic analysis results, the curves represent the ratio of local amplitude to the incident wave amplitude. Time series of the elevations at four horizontal positions are shown in Fig. 18. The figures show that the wave steepness first increases dramatically as it shoals up the bar, at the same time the higher order harmonics start to accumulate while the first order wave starts decline after it reaches the top of bar. Then the wave continues propagating through the down-slope and back to the flat bottom, where higher harmonics are released and an irregular wave pattern after the bar is formed. The bottom plot of Fig. 17 shows that the numerical curves generally seem consistent with the experimental data in both amplitude and phase. The largest errors occur in the first and second order results however, which are most obvious from the highest point of the bar and during the downhill process. Fig. 18 gives supplementary information in the time domain, so the accuracy of the computation is further confirmed. But the lower two plots also display a little unexpected high frequency components, and the two positions are all around the top of the submerged bar. Nevertheless, the plots in both frequency and time domain indicate a good agreement between numerical and measured results.

Conclusion

The accuracy and stability of a finite difference based solution to the exact potential-flow problem for nonlinear waves with an IBM treatment of both the free surface and bottom boundaries has been established in 2D. The IBM is found to be of comparable accuracy and stability to the more commonly used σ -transform method. In the next step, the IBM approach is expected to be more easily extended to wave-structure interaction problems, since the IBM removes the need to construct an artificial C^2 continuous free-surface inside the body. Preliminary results have also been presented towards an optimal iterative solution strategy for this method, and this is a topic of further study.

Acknowledgement

This work is jointly funded by the Department of Mechanical Engineering, Technical University Denmark and the China Scholarship Council (CSC) under project number 201706680065.

Reference

1. Grilli, S. T., Skourup, J., & Svendsen, I. A. An efficient boundary element method for nonlinear water waves. *Engineering Analysis with Boundary Elements* 1989, 6(2), 97-107.
2. Liu, Y. J., & Nishimura, N. The fast multipole boundary element method for potential problems: a tutorial. *Engineering Analysis with Boundary Elements* 2006, 30(5), 371-381.
3. Yan H, Liu Y. An efficient high-order boundary element method for nonlinear wave-wave and wave-body interactions[J]. *Journal of Computational Physics*, 2011, 230(2): 402-424.
4. Wu G X, Hu Z Z. Simulation of nonlinear interactions between waves and floating bodies through a finite-element-based numerical tank[J]. *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 2004, 460(2050): 2797-2817.
5. Cienfuegos R, Barthelémy E, Bonneton P. A fourth - order compact finite volume scheme for fully nonlinear and weakly dispersive Boussinesq - type equations. Part II: boundary conditions and validation[J]. *International Journal for Numerical Methods in Fluids*, 2007, 53(9): 1423-1455.
6. Engsig-Karup A P, Eskilsson C, Bigoni D. A stabilised nodal spectral element method for fully nonlinear water waves[J]. *Journal of Computational Physics*, 2016, 318: 1-21.
7. Shao Y L, Faltinsen O M. A harmonic polynomial cell (HPC) method for 3D Laplace equation with application in marine hydrodynamics[J]. *Journal of Computational Physics*, 2014, 274: 312-332.
8. Hanssen F C W, Greco M, Shao Y. The harmonic polynomial cell method for moving bodies immersed in a Cartesian background grid[C]//ASME 2015 34th International Conference on Ocean, Offshore and Arctic Engineering. American Society of Mechanical Engineers Digital Collection, 2015.
9. Hanssen F C W, Bardazzi A, Lugni C, et al. Free - surface tracking in 2D with the harmonic polynomial cell method: Two alternative strategies[J]. *International Journal for Numerical Methods in Engineering*, 2018, 113(2): 311-351.
10. Ma S, Hanssen F C W, Siddiqui M A, et al. Local and global properties of the harmonic polynomial cell method: In - depth analysis in two dimensions[J]. *International Journal for Numerical Methods in Engineering*, 2018, 113(4): 681-718.

11. Bingham H B, Zhang H. On the accuracy of finite-difference solutions for nonlinear water waves[J]. *Journal of Engineering Mathematics*, 2007, 58(1-4): 211-228.
12. Engsig-Karup A P, Bingham H B, Lindberg O. An efficient flexible-order model for 3D nonlinear water waves[J]. *Journal of computational physics*, 2009, 228(6): 2100-2118.
13. Kontos S, Bingham H B, Lindberg O, et al. A robust WENO scheme for nonlinear waves in a moving reference frame[J]. *Journal of Hydrodynamics*, Ser. B, 2016, 28(3): 482-488.
14. Hicks J B H, Bingham H B, Read R. Control Signal Optimization for Non-Linear Wave Generation[C]//ASME 2018 37th International Conference on Ocean, Offshore and Arctic Engineering. American Society of Mechanical Engineers Digital Collection, 2018.
15. Kontos S, Bingham H B, Engsig-Karup A P, et al. Robust Numerical Methods for Nonlinear Wave-Structure Interaction in a Moving Frame of Reference[M]. Technical University of Denmark (DTU), 2016.
16. Mei, Chiang C. *The applied dynamics of ocean surface waves*. Vol. 1. World scientific, 1989.
17. Fenton J D. Numerical methods for nonlinear waves[M]//Advances in coastal and ocean engineering. 1999: 241-324.
18. Fenton J D. Nonlinear wave theories[J]. *The Sea-Ocean Engineering Science*, 1990, 9: 3-25.
19. Clamond D, Fructus D, Grue J, et al. An efficient model for three-dimensional surface wave simulations. Part II: Generation and absorption[J]. *Journal of computational physics*, 2005, 205(2): 686-705.
20. Luth H R, Klopman G, Kitou N. Kinematics of waves breaking partially on an offshore bar; LDV measurements of waves with and without a net onshore current[J]. Report H-1573, Delft Hydraulics, 1994, 40.
21. Beji S, Battjes J A. Experimental investigation of wave propagation over a bar[J]. *Coastal Engineering*, 1993, 19(1-2): 151-162.

TABLES

Table 1 Initial wave conditions used for nonlinear wave tests

kh	0.5	2	2π
----	-----	---	--------

wave height (10% of breaking limit)	0.0059	0.011	0.0135
wave height (90% of breaking limit)	0.0525	0.1206	0.1273

Table 2 The finite difference scheme for nonlinear wave tests

r	Order of accuracy	Vertical grid
3	2	even
3	2	stretched
5	4	stretched
7	6	stretched

Fig. 1 Coordinate system and boundary diagram

122x56mm (600 x 600 DPI)

Fig. 2 Finite difference discretization

130x98mm (600 x 600 DPI)

Fig. 3 Implementation of the ghost point method

181x69mm (600 x 600 DPI)

Fig. 4 Finite difference grid near free surface

131x66mm (600 x 600 DPI)

Fig. 5 Finite difference grid near free surface

129x65mm (600 x 600 DPI)

Fig. 6. Uniform grid and stretched grid

161x71mm (600 x 600 DPI)

Fig. 7 Matrix structure of the preconditioner and the A matrix

154x51mm (600 x 600 DPI)

Fig. 8. Convergence of the linear vertical velocity on the free surface at $kh=4$
174x236mm (300 x 300 DPI)

Fig. 9 Dispersion errors with relative water depth for different resolutions and schemes.

145x144mm (300 x 300 DPI)

Fig. 10 Maximum stability eigenvalue vs. the relative Nyquist depth under different vertical grid refinements.

199x99mm (300 x 300 DPI)

Fig. 11 Maximum stability eigenvalues vs. relative Nyquist depth for different FD schemes.

198x99mm (300 x 300 DPI)

Fig. 12. Convergence of the nonlinear vertical velocity on the free surface compared to the σ -transform method.

242x167mm (600 x 600 DPI)

Fig. 13. Nonlinear errors for the six test cases listed in Table 2

171x240mm (300 x 300 DPI)

Fig. 14. Nonlinear errors for wave energy conservation.

218x86mm (300 x 300 DPI)

Fig. 15 Average iteration counts for the 90% steepness wave.

165x88mm (600 x 600 DPI)

Fig. 16. Harmonic analysis of a linear shoaling wave.

239x82mm (300 x 300 DPI)

Fig. 17 Top) submerged bar geometry layout and a snapshot of the wave surface. Bottom) harmonic analysis of the wave elevation time series.

202x115mm (600 x 600 DPI)

Fig. 18 Time series of wave elevations over the bar. Solid lines – Calculations; Circles – Measurement.

181x191mm (600 x 600 DPI)