Free-Surface Tracking with the Harmonic Polynomial Cell Method: Two Alternative Strategies

F-C. W. Hanssen[a*], A. Bardazzi[b], C. Lugni[a,b], M. Greco[a,b]

[a]*Centre for Autonomous Marine Operations and Systems (AMOS) , Department of Marine Technology, NTNU, Trondheim, Norway*
[b]*CNR-INSEAN, National Research Council-Marine Technology Institute, Via di Vallerano 139, 00128 Roma, Italy*

Abstract

Several cases of nonlinear-wave propagation are studied numerically in two dimensions within the framework of potential flow. The Laplace equation is solved with the Harmonic Polynomial cell (HPC) method, which is a field method with high-order accuracy. In the HPC method, the computational domain is divided into overlapping cells. Within each cell, the velocity potential is represented by a sum of harmonic polynomials. Two different methods denoted as Immersed Boundary (IB) and Multi-grid (MG) are used to track the free surface. The former treats the free surface as an immersed boundary in a fixed, Cartesian background grid, while the latter uses a free-surface fitted grid that overlaps with a Cartesian background grid. The simulated cases include several nonlinear wave mechanisms, such as high steepness and shallow-water effects. For one of the cases, a numerical scheme to suppress local wave breaking is introduced. Such scheme can serve as a practical mean to ensure numerical stability in simulations where local breaking is not significant for the result. For all the considered cases, both the IB and MG method generally give satisfactory agreement with known reference results. Although the two free-surface tracking methods mostly have similar performance, some differences between them are pointed out. These include aspects related to modelling of particular physical problems as well as their computational efficiency when combined with the HPC method.

Keywords

Harmonic polynomial cell method
Immersed boundary method
Multi-grid method
Nonlinear waves
Overlapping grid
Potential flow

Abbreviations

BEM	Boundary element method	HPC	Harmonic polynomial cell
CFL	Courant-Friedrichs-Lewy	IB	Immersed boundary
FDM	Finite difference method	MG	Multi-grid

* Corresponding author. E-mail address: finn-christian.hanssen@ntnu.no. Address: NTNU Centre for Autonomous Marine Operations and Systems, Marine Technology Centre, NO-7491 Trondheim, NORWAY

FVM	Finite volume method	RK4	Fourth-order Runge-Kutta

1 Introduction

For several decades, linear or second order perturbation analysis based on potential flow has served as basis for the analysis tools in the offshore and coastal engineering industries. Such methods assume that the surface elevation is a linear or weakly nonlinear process, see e.g. Chapter 3 in [1]. However, in many cases these assumptions are questionable. As examples, violent sloshing waves inside a LNG tank are highly nonlinear, and steep ocean waves in storms certainly exhibit significant nonlinear behavior. To describe these nonlinear features accurately, we must invoke higher-order theories. Examples of such theories are the Stokes wave theory [2] and the Fourier representation by Rienecker and Fenton [3]. The latter builds on the stream-function theory originally proposed by Dean [4]. Such higher-order theories generally represent the elevation and kinematics of nonlinear waves with good accuracy. However, they only apply for periodic waves, and are thus not able to describe irregular phenomena. Alternatively, nonlinear modal theories to describe resonant wave motion inside a closed tank have been developed by Faltinsen and Timokha [5]. These are able to describe many relevant phenomena well, such as nonlinear dispersion, but have an upper limitation with respect to wave steepness. In general, one finds that there exists several wave theories that work well when certain assumptions related to e.g. the water depth or the wave steepness are met. To develop a solution that applies for a wide range of water depths and wave steepness, the immediate alternative is to develop the flow in time by solving the nonlinear hydrodynamic boundary value problem in a numerical wave tank. Although more computationally demanding, this framework enables us to simulate unsteady waves as well as nonlinear wave-body interactions.

The boundary element method (BEM) has traditionally been the most widely used method to solve potential-flow problems. Examples where BEM is used to simulate nonlinear water waves are found in e.g. Grilli et al. [6] and Wang et al. [7]. As an alternative, Wu & Eatock Taylor [8] presented a finite element method (FEM) implementation that they found to be significantly more efficient than the traditional potential-flow BEMs. Although less common, it is also possible to solve potential-flow problems with the finite volume method (FVM) or the finite difference method (FDM). An overview of the different numerical methods is given e.g. in [5]. In the present paper, we study fully nonlinear water waves in a potential-flow numerical wave tank where we instead use the Harmonic Polynomial Cell (HPC) method to solve the governing Laplace equation. The HPC method, recently proposed by Shao and Faltinsen [9], [10], is a higher-order field method where we discretize the computational domain by overlapping cells. Within each cell, a sum of harmonic polynomials represents the velocity potential in a continuous way, and the overlapping-cell structure provides connectivity between adjacent grid nodes. Liang et al. [11] studied a range of problems relevant in marine hydrodynamics by means of the HPC method, including some expansions to deal with singular flows at sharp corners and pressure jumps across lifting surfaces. Fredriksen [12] used the HPC method as a potential-flow solver in a hybrid potential-viscous flow method to study the gap resonance in a moonpool. If we consider the HPC method as a measure to discretize a general field equation, its applicability is not restricted to potential flows. Bardazzi et al. [13] demonstrated how accurately a generalized version of the HPC method is able to solve the Poisson equation in a viscous-flow solver even with singular features in the forcing term. The method is therefore useful within a projection method [14] for a high-order accurate solution of the pressure field in incompressible flows governed by the Navier-Stokes equations.

The present paper incorporates two fundamentally different free-surface tracking schemes: In the first approach, the free surface is strictly an immersed boundary in a Cartesian background grid with markers that are restricted to move purely along one axis to track the free surface. We refer to this as the immersed boundary (IB) approach. In the second approach, which we refer to as the multi-grid (MG) approach, the free surface is fitted with an additional grid that is overlapping with a fixed Cartesian background grid. The deformation of the free-surface fitted grid in time is governed by the evolution of the free surface itself. Both methods use markers that move in a semi-Lagrangian way to enforce the free-surface boundary conditions, and none of them can thus model overturning waves. Transition to a fully-Lagrangian approach to model overturning waves can be achieved in both methods, but is not pursued in the present paper. Nevertheless, with the semi-Lagrangian description both methods demonstrate the ability to simulate waves up to the point of overturning with good accuracy. The main advantages of the IB approach is that it is straightforward to implement, and that the associated CPU time is limited because the grid remains fixed throughout the simulation. As will be demonstrated later, the MG method can also achieve similar or even better CPU efficiency in cases where the fixed background grid can be made considerably coarser than the free-surface fitted grid. We apply the numerical wave tank with both the IB and MG methods to simulate a range of cases with either analytical, numerical or experimental reference solutions. These cases involve waves with various nonlinear mechanisms including very steep waves and shallow-water waves.

The organization of the remainder of the paper is as follows: Section 2 presents the numerical formulation and discusses distinct features of the two different free-surface treatments. Section 3 outlines the selected simulation cases, for which section 4 presents results. In section 5, main conclusions are drawn.

2 Numerical Formulation

2.1 The Harmonic Polynomial Cell Method

Shao and Faltinsen proposed the HPC method as an accurate and efficient numerical method to solve the Laplace equation in two or three dimensions [9], [10]. In this paper, we restrict ourselves to study two-dimensional problems. A brief summary of the HPC method in its "classical" formulation in two dimensions is given in the following, assuming a Cartesian coordinate system with axes x and z.

For an incompressible and inviscid fluid, the fluid motion at a given time is described by the velocity potential $\varphi(x,z)$. The governing equation for the velocity potential is the Laplace equation:

$$\nabla^2\varphi(x,z) = 0. \tag{1}$$

Quadrilateral, overlapping cells are used to discretize the fluid volume. As illustrated in Figure 1, each cell consists of eight nodes along the boundaries (with local node indices $1-8$), and an interior node with index 9. We introduce a local (ζ,ξ) coordinate system in each cell with origin in the interior node, i.e.

$$\begin{aligned}\zeta &= x - x_9\\ \xi &= z - z_9,\end{aligned} \tag{2}$$

where (x,z) are global coordinates.

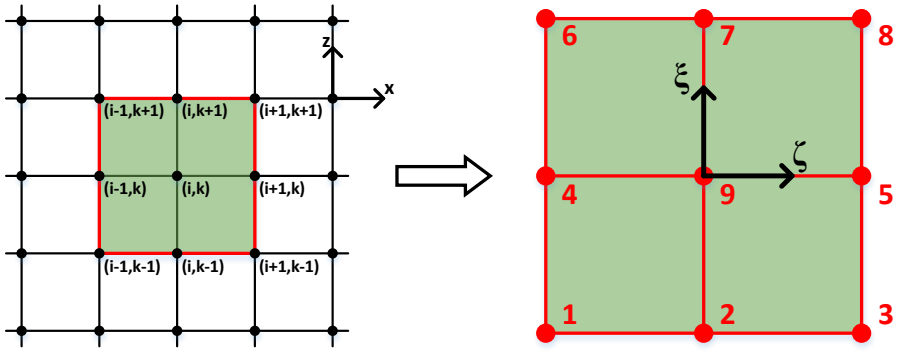


Figure 1 Left: Global grid numbering. Right: Local coordinate system and node numbering within one cell.

A sum of harmonic polynomials with undetermined coefficients represents the velocity potential within each cell. The set of harmonic polynomials must be complete up to a certain order, i.e. for a N^{th} order representation, all polynomials of order lower than or equal to $N-1$ must be included. Because the number of unknowns must equal the number of available equations, it is not always possible to include all polynomials of order N. The choice of which polynomials to include in this case is examined by Ma et al. [15]. Since each of the harmonic polynomials by definition satisfies the Laplace equation, the velocity potential within a cell also automatically satisfies this. As in the original formulation of the HPC method [9], all polynomials up to third order and one fourth-order polynomial are included in the present implementation. Upon expressing the unknown coefficients associated with each polynomial as a function of the velocity potential in the nodes along the cell's boundaries, the velocity potential at an arbitrary point within the cell is

$$\varphi(\zeta,\xi) = \sum_{i=1}^{8}\left[\sum_{j=1}^{8} c_{j,i} f_j(\zeta,\xi)\right]\varphi_i, \tag{3}$$

and its spatial derivative

$$\nabla\varphi(\zeta,\xi) = \sum_{i=1}^{8}\left[\sum_{j=1}^{8} c_{j,i} \nabla f_j(\zeta,\xi)\right]\varphi_i. \tag{4}$$

In (3), f_j is either the real or imaginary part of a harmonic polynomial of a given order, and $c_{j,i}$ is an element of the inverse of a matrix $[D]$ with elements $d_{i,j} = f_j(\zeta_i,\xi_i)$. A consequence of selecting the interior node as the origin in each cell is that only the first harmonic polynomial is non-zero at this location:

$$f_i(\zeta_9,\xi_9) = f_i(0,0) = \begin{cases} 1 & i = 1 \\ 0 & i > 1 \end{cases}. \tag{5}$$

This gives the connectivity equation, stating that the velocity potential in each node is a linear combination of the potential in the eight surrounding nodes:

$$\varphi_9 = \varphi(\zeta_9, \xi_9) = \sum_{i=1}^{8} c_{1,i}\varphi_i. \qquad (6)$$

Equation (6) applies for every node in the computational domain that is not along its edges, since all of them will be the center node of a particular cell. For any grid node, it is straightforward to impose a Dirichlet boundary condition directly, $\varphi = \varphi_D$. To impose a Dirichlet condition in a non-nodal point, equation (3) is applied with the appropriate local coordinate $(\zeta, \xi) = (\zeta_p, \xi_p)$ in the selected cell. For nodes along a Neumann boundary, the boundary condition follows by taking the normal derivative of equation (3):

$$\frac{\partial \varphi}{\partial n}(\zeta, \xi) = \nabla\varphi(\zeta, \xi) \cdot \vec{n}(x, z) = \sum_{i=1}^{8} \left[\sum_{j=1}^{8} c_{j,i} \nabla f_j(\zeta, \xi) \cdot \vec{n}(x, z) \right] \varphi_i. \qquad (7)$$

Equation (6) together with boundary conditions are then combined to build a global linear, algebraic matrix system for the velocity potential in the entire computational domain on the form $[A]\{\varphi\} = \{b\}$. The coefficient matrix $[A]$ is sparse, and generally dominated by entries of the connectivity equation (6). With a proper preconditioner obtained beforehand, iterative sparse solvers such as GMRES or BiCGStab provide efficient solutions to this matrix equation.

Since equation (3) includes an incomplete set of polynomials up to fourth order, we can expect the velocity potential to have between third- and fourth order accuracy in the spatial coordinates. Ma et al. [15] present a comprehensive study of different aspects influencing the accuracy in the HPC method. In brief, use of regular grids with square cells is recommended, for which the convergence of the velocity potential can be even higher than fourth order inside the fluid domain. Nevertheless, the accuracy related to Neumann boundaries generally governs the overall accuracy in practical problems. The aforementioned reference presents several treatments to overcome this restriction. None of these are included in the present implementation of the HPC method, and as a result the accuracy associated with Neumann boundaries herein can generally never be higher than fourth order. Furthermore, it is found that the accuracy of the solution may be significantly worse in severely stretched or distorted cells compared to their square counterpart. In cases where we do not apply a uniform grid, we should therefore monitor the degree of stretching and distortion.

2.2 Free Surface Treated as an Immersed Boundary (IB)

Hanssen et al. [16] made a first attempt to combine the HPC method with an immersed boundary (IB) method for imposing body-boundary conditions on a solid body in an exterior flow. This study comprised an additional interpolation scheme in the direction of the local normal vector of the boundary to represent the flow at the intersection point. In the present work we use a similar approach, but avoid the additional interpolation scheme by directly using equation (3). In the following, we describe the implementation of the IB method used in connection with Dirichlet conditions for the markers on a free surface.

We represent the initial state of the free surface by distributing a set of markers with even spacing in horizontal direction. In case the wave propagation is initiated by a physical wavemaker, the free surface is assumed to be at rest at $t = 0$. As illustrated in Figure 2, the horizontal grid spacing is so that there is one

marker located exactly at every vertical grid line in the structured grid. Cells used to account for each marker are shaded with grey in the figure. Within the IB strategy, the nodes in such cell that are above the free surface become ghost nodes. The boundary condition for a ghost node is not set explicitly. Instead, we use the row of a particular ghost node in the global matrix system to set a Dirichlet condition for the velocity potential in the related free-surface marker. Specifically, this means that equation (3) is evaluated at the local marker's position (ζ_m, ξ_m) in the cell-fixed coordinate system. The primary cells associated with each marker is always selected based on the criterion that the marker is in the upper half plane of the cell, and the marker's horizontal position coincides with the vertical centerline of the cell through local cell nodes 2-9-7 (Figure 1). In cases where there are two ghost nodes in a grid column above the free surface, the marker along the same vertical grid line is used for the boundary condition associated with the upper ghost node. This means that the boundary condition for the same marker can be set twice, but never in the same cell. Thus, this does not lead to a singular global coefficient matrix.

At any given time step, nodes that are neither fluid nor ghost nodes are disregarded in the global matrix system. The meaning of different nodes in the grid, as well as the size of the total computational domain, thus changes as the free surface deforms. This means that parts of the coefficient matrix $[A]$ has to be rebuilt every time step. This comes with some additional CPU cost. On the other hand, since the grid remains fixed, the elements $c_{i,j}$ of the inverted matrix $[D]$ are pre-calculated once at the beginning of the simulation. This is advantageous from a computational standpoint, since the inversion of the local $[D]$-matrices at every time step consumes a significant portion of the overall CPU time in a body-fitted grid approach.

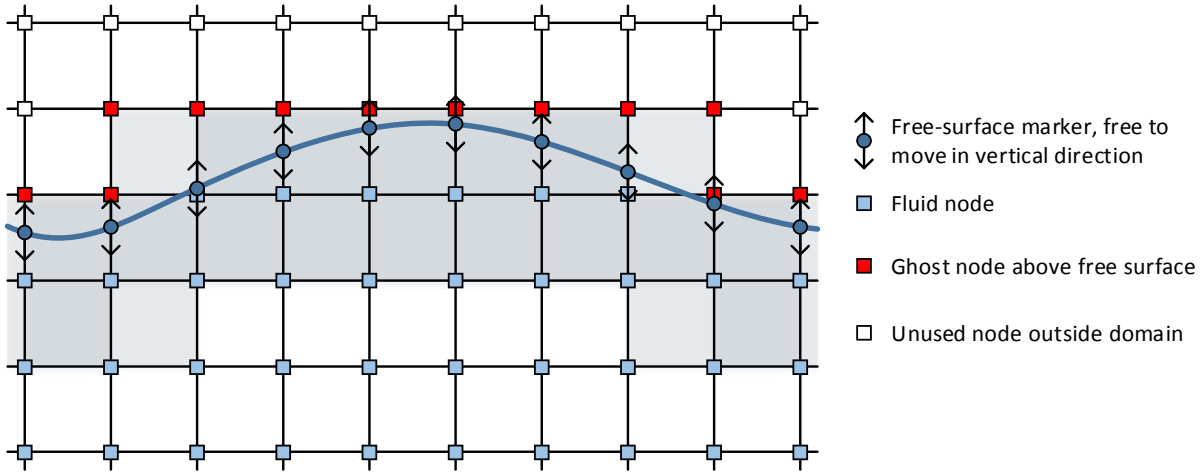


Figure 2 IB free-surface method: The free surface is an immersed boundary in a structured Cartesian grid, and is tracked by markers moving only in vertical direction (as indicated by the arrows associated with each marker). The cells used to impose free-surface conditions are grey shaded.

The semi-Lagrangian kinematic and dynamic free-surface conditions for a vertically moving marker with global coordinates (x_m, z_m) are as follows:

$$\frac{Dx_m}{Dt} = 0, \frac{Dz_m}{Dt} = \frac{\partial \varphi}{\partial z} - \frac{\partial \varphi}{\partial x}\frac{\partial \eta}{\partial x} - \nu(x^*)\eta \qquad (8)$$

$$\frac{D\varphi_m}{Dt} = -\frac{1}{2}\left|\left(\frac{\partial\varphi}{\partial x}\right)^2 + \left(\frac{\partial\varphi}{\partial z}\right)^2\right| - g\eta + \frac{\partial\eta}{\partial t}\frac{\partial\varphi}{\partial z} - \nu(x^*)\varphi. \quad (9)$$

The term $\nu(x^*)$ is a damping coefficient associated with a numerical beach used to avoid wave reflections from the tank wall on the opposite side of the wavemaker. $\nu(x^*)$ is therefore different from zero only in a limited zone close to the downstream boundary of the computational domain, as described later in the text.

In equations (8) and (9) we have made use of the semi-Lagrangian time derivative $D/Dt = \partial/\partial t + \vec{v}\cdot\nabla$, with $\vec{v} = (Dx_m/Dt, Dz_m/Dt) = (0, Dz_m/Dt)$. All spatial derivatives are evaluated at the instantaneous position (x_m, z_m) of the marker, and the wave slope $\partial\eta/\partial x$ is calculated from a higher-order finite difference scheme. A standard fourth-order explicit Runge-Kutta scheme (RK4) integrates the position and velocity potential of the markers in time.

Since the velocity potential is represented as a continuous function within each cell through equation (3), the HPC method is found particularly convenient within an IB scheme. As seen, we can set the Dirichlet condition at the free-surface markers directly, even though the markers generally do not coincide with any of the grid nodes. Consequently, the IB treatment does not diminished the quality of the solution. On the contrary, Ma et al. [15] shows that the solution within a square cell generally has higher accuracy close to the center of a cell than along its boundaries, which means that the IB method in some scenarios can lead to a higher accuracy than a boundary-fitted grid.

2.2.1 Suppression of Breaking Waves

In the semi-Lagrangian formulation used in both the IB and MG methods, the free surface has to be a single-valued function of the horizontal coordinate. As mentioned, this makes it impossible to simulate breaking or overturning waves. In cases where tendencies of wave breaking are present, but the actual breaking is either localized or not of primary interest, it can however be useful to suppress wave breaking in order to ensure numerical robustness. The scheme presented in the following applies for both the IB and MG formulation, but is only implemented for the IB method.

The physical process of wave breaking is associated with dissipation of energy. Various numerical schemes exist to model this energy dissipation. Subramani [17] outlines a method with an additional pressure term in the dynamic free-surface condition as well as an alternative approach with a spatial polynomial representation to replace the breaking wave with a smoothed free-surface profile. A similar method is deployed by Wang et al. [7], applying a 'peeling' technique to remove the top water layer from the crest of the wave in order to avoid undesirable breaking. Paulsen et al. [18] use a Savitsky-Golay filter locally on the free surface where breaking occurs. All the above methods effectively remove energy from the wave crest to account for dissipation due to breaking. The method of adding a dissipative term to the dynamic free-surface condition is preferred in the present work, hence the dynamic free-surface condition given in equation (9) is modified into

$$\frac{D\varphi_m}{Dt} = -\frac{1}{2}\left|\left(\frac{\partial\varphi}{\partial x}\right)^2 + \left(\frac{\partial\varphi}{\partial z}\right)^2\right| - g\eta + \frac{Dz_m}{Dt}\frac{\partial\varphi}{\partial z} - \nu(x^*)\varphi - \mu_b. \quad (10)$$

The additional dissipative term μ_b is defined as

$$\mu_b(x, x_0) = \mu_{max}\theta(x)|\nabla\varphi|^2 \cdot sign\left(\frac{\partial\varphi}{\partial z}\right), \quad (11)$$

where $sign(\partial\varphi/\partial z) = 1$ if $\partial\varphi/\partial z \geq 0$ and $sign(\partial\varphi/\partial z) = -1$ if $\partial\varphi/\partial z < 0$. μ_{max} is an empirical dissipation coefficient and $\theta(x)$ is an envelope function on the form

$$\theta(x, x_0) = \frac{1}{2}\left(1.0 + cos\left(\frac{\pi(x - x_0)}{L_0}\right)\right). \quad (12)$$

x_0 is here the x-coordinate of the point where breaking is identified and L_0 is the half length of the envelope. We typically set L_0 equal to the grid spacing Δx multiplied with some factor n, where n is a sufficiently small number to only remove energy from a localized area close to the instable region. If $|x - x_0| > L_0$, $\theta(x)$ is set to zero. Due to the envelope function $\theta(x)$, the local dissipation at position x_0 will thus also affect some of the adjacent markers. In some cases, a marker will accumulate contributions from several such envelopes. For a marker at position x that is within the envelope of N nearby markers, the total dissipation is then the sum of the individual contributions from markers $i = 1, \dots, N$ at positions x_i:

$$\mu_b(x) = \sum_{i=1}^{N} \mu_{b,i}(x; x_0 = x_i). \quad (13)$$

A principal illustration of the total dissipation accumulated from five local dissipation envelopes is given in Figure 3.

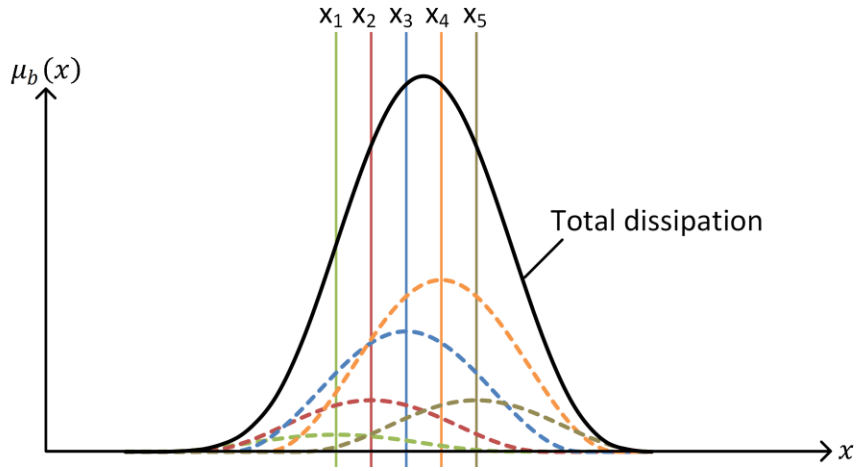


Figure 3 Accumulation of free-surface dissipation along the horizontal axis from local dissipation-envelopes centered at $x_1, \dots, x_5$.

In order to know when the dissipative term in equation (10) shall be invoked, a criterion to identify breaking has to be established. As stated by Barthelemy et al. [19], we may consider wave breaking as a threshold process. Several suggested criteria, typically classified as geometric, kinematic or dynamic, exist to describe this threshold. Subramani [17] uses a geometric criterion based on the local curvature of the free surface in his study. Paulsen et al. [18] use a kinematic criterion based on the vertical acceleration of the fluid particles at the free surface, activating their free-surface filter wherever $\varphi_{zt} < -\gamma g$ with $\gamma \in$

$[0.5,1.0]$ and g the acceleration of gravity. Tulin and Li [20] suggested a dynamic criterion to identify wave breaking based on energy fluxes, arguing that the breaking process originates from intra-wave instabilities. Recently Barthelemy et al. [19] have developed a criterion based on the ratio between the horizontal fluid particle velocity at the surface and the local phase velocity:

$$B_x(x) = \frac{\partial \varphi}{\partial x} / c_x. \qquad (14)$$

Equation (14) can be evaluated along the entire free surface at every time step with limited additional CPU cost, since the fluid particle velocity is already evaluated in the free-surface conditions. The additional effort required comes from evaluating the local phase velocity c_x. To do this, we here follow the method suggested by Kurina and van Groesen [21], which is also used by Seiffert and Ducrozet [22]. The local phase velocity is here defined as

$$c_x(x,t) = \sqrt{\frac{g \cdot tanh[k(x,t)h]}{k(x,t)}}, \qquad (15)$$

where g is the acceleration of gravity, h the water depth and $k(x,t)$ the local wave number. $k(x,t)$ is found by calculating the Hilbert transform $\mathrm{H}(\eta)$ of the free surface η with respect to x at time t:

$$k(x,t) = \frac{1}{\eta^2 + \mathrm{H}(\eta)^2}\left(\eta \frac{\partial \mathrm{H}(\eta)}{\partial x} - \mathrm{H}(\eta)\frac{\partial \eta}{\partial x}\right). \qquad (16)$$

The spatial derivative of $\mathrm{H}(\eta)$ in equation (16) is determined from a numerically fitted B-spline representation of the Hilbert transform through all the free-surface markers.

Barthelemy et al. [19] found equation (15) to successfully predict wave breaking using a threshold value between 0.85 and 0.86 for $B_x(x)$. Seiffert and Ducrozet [22] confirm that B_x in the range $0.84 - 0.86$ identifies the onset of breaking accurately for various tested wave packets.

2.3 Free-Surface Modelled with a Multi Grid (MG) Method

Generally we must apply a viscous solver to model the flow also after the breaking wave implodes on the free surface or on a structure. In some cases, however, it can be of interest to know the wave profile and associated wave kinematics in an overturning wave just before it reattaches to the free surface or to a structure. As an example, the potential-flow solution of an overturning wave can be useful when the impact loads in a sloshing tank due to the so-called flip-through phenomenon are studied ([23], [24], [25]). Although the IB method outlined above in principle also can facilitate a fully Lagrangian free-surface modelling to account for overturning waves, it is expected that this will require the grid to be significantly refined locally near the overturning. A local refinement is not directly achievable with a fixed grid, which motivates the development of the multi-grid (MG) method.

The general idea behind the MG approach consists in the overlap among several structured grids that exchange information about the velocity potential in certain connection nodes. Here only two grids are considered: a background grid and a free-surface fitted grid that follows the free-surface deformation in time in accordance with the dynamic and kinematic free-surface boundary conditions. Figure 4 shows a

typical configuration for the two grids: blue dots are nodes in the background grid, which generally extends above the physical domain (the area below the red curve). Nodes in the background grid can move in and out of the fluid domain during the free-surface evolution. Red dots indicate markers that are used to track the free-surface deformation, while magenta dots are nodes in the free-surface fitted grid. It is worth noting that the free-surface markers in the refined grid can move in a semi- or fully Lagrangian manner depending on the features of the problem studied. In this work, the semi Lagrangian formulation is assumed for all the test cases considered except for the ones with a physical wavemaker. Then, a fully Lagrangian formulation is applied locally near the wavemaker to take the moving boundary properly into account, and gradually transitioned into a semi-Lagrangian formulation further away.

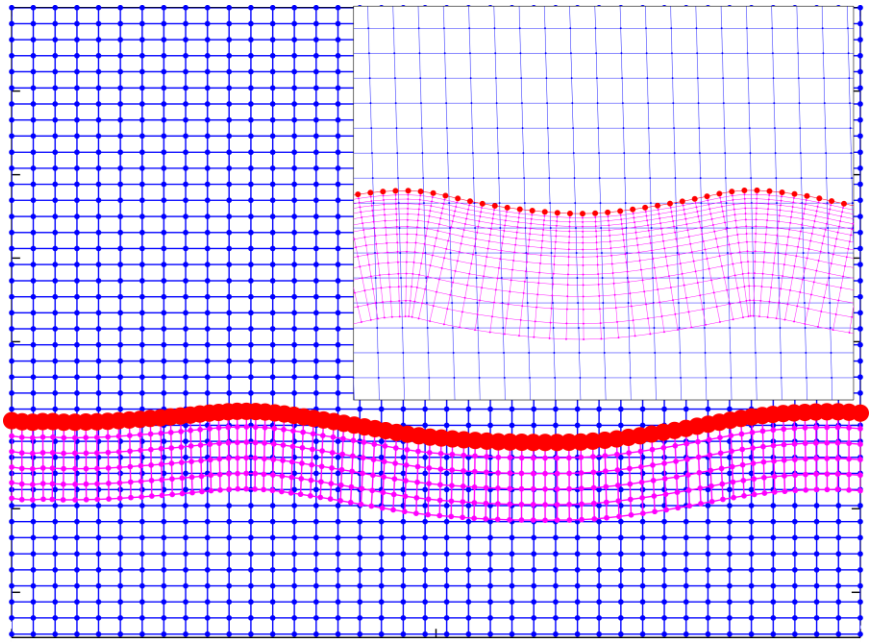


Figure 4 Example of discretization in the MG method. The upper right frame shows an enlarged part of the free-surface fitted grid.

The free-surface fitted grid initially has a uniform spacing in horizontal direction along the free surface both in case of an initial condition different from rest and in the case of calm water. At the successive time instants, it is constructed starting from the Lagrangian nodes on the free surface and going below the local free surface in the normal direction (see Figure 5). The distribution of nodes in the normal direction can be either linear, quadratic or cubic. Because the free-surface fitted grid generally is built in a direction normal to the local free surface, particular care must be paid to the nodes close to solid sidewall boundaries to avoid a situation with nodes located outside the physical domain. Such scenario is illustrated with red lines in Figure 5. To avoid this, a ramp function that rotates the normal vector used to generate the free-surface fitted grid is used near the boundary, resulting in the grid illustrated with black lines in Figure 5. This means that near the solid boundary, the generated grid is not orthogonal to the free surface.

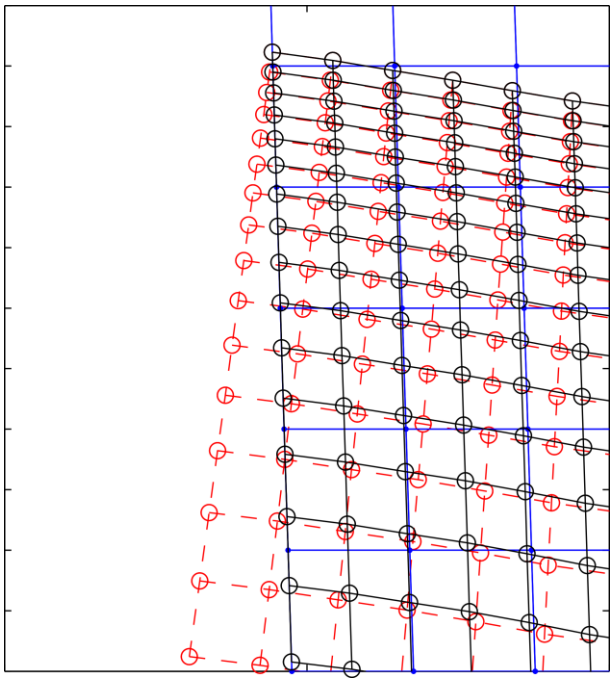


Figure 5 Effect of employing a ramp function for the normal direction in the construction of the free-surface fitted grid close to a vertical solid domain boundary. In this figure, a linear ramp function applies.

Figure 6 can be used to describe how boundary conditions are assigned to establish a boundary value problem for the velocity potential. On the vertical and bottom boundaries of the background grid, the impermeability boundary condition is enforced. In the free-surface fitted grid, the Dirichlet boundary condition for the velocity potential is imposed on the free surface with an impermeability Neumann boundary condition along the two side boundaries. Consequently, the upper nodes in the background grid and lower nodes in the free-surface fitted grid lack boundary conditions. They are enforced as coupled boundary conditions at the connection points between the two grids, marked as diamonds in Figure 6. Here the yellow diamonds belong to the background grid while the green ones belong to the free-surface fitted moving grid. These nodes are assigned with coupling-boundary conditions from the other grid derived directly from equation (3) in the form

$$\varphi_{grid1}(x,z) = \sum_{i=1}^{8}\left[\sum_{j=1}^{8} c_{j,i} f_j(\zeta,\xi)\right]\varphi_i\Big|_{grid2}, \qquad (17)$$

where 'grid1' and 'grid2' refer to the free-surface fitted grid and the background grid or vice versa. (ζ,ξ) is the local coordinates of the point (x,z) inside the cell used in 'grid2'. If the connection point is located along a boundary of the physical domain, the spatial derivative of equation (17) is taken in a similar form as equation (7). The cell associated with an arbitrary connection point is chosen from the background grid if the connection point belongs to the free-surface fitted grid and vice-versa in a way so that the distance between the point and the cell's central node is minimized. For a node in the background grid to be labelled as a connection point, it must belong to a continuous series of at least three nodes along the same grid row. This means that every connection point must be a boundary node for at least one cell in the same grid. Disregarding this condition may lead to spatial oscillations in the velocity potential. Because

such nodes are typically located just below the free surface, it may hence lead to instability in the temporal evolution of the free surface due to inaccuracies in the fluid particle velocities at the free surface. An additional source of oscillations in the velocity potential comes from the connection points of the free-surface fitted grid that also belong to either the wavemaker or the opposite vertical wall, where an impermeability Neumann boundary condition is applied. To overcome the problem, the solution of an additional "local" boundary value problem only for the free-surface fitted grid is required. In particular, once the velocity potential is known from the "global" boundary value problem, the "local" problem is established as follows: On the solid sidewalls and on the free surface, the boundary conditions remain unchanged. On the bottom of the free-surface fitted grid, the velocity potential from the global solution is given as Dirichlet condition. The new solution is then used to compute the fluid particle velocities on the free-surface as spatial derivative according to equation (4).

As mentioned, grid stretching can be useful in order to reduce the computational cost. Moreover, it is difficult to avoid in case of a very steep wave, an overturning wave or a water jet as in the flip-through case. Here, the (local) geometric features of the free surface force the surface-fitted grid to be very fine. Then, stretching of the background grid is required to ensure proper overlapping between the two grids. In order to avoid two-way communication within the same cell in the overlapping between grids, the following rule applies: A node in the background grid that gives the potential to the free-surface fitted grid, cannot at the same time belong to a cell in the background grid that also receives the potential from a node in the free-surface fitted grid (and vice versa).

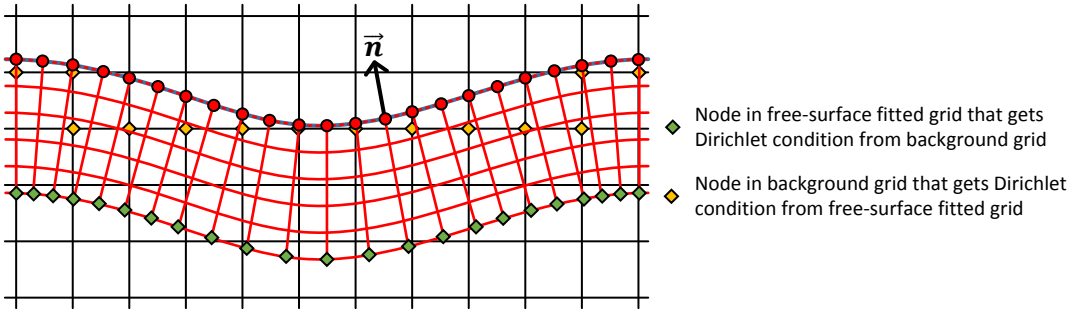


Figure 6 MG free-surface treatment: Semi-Lagrangian markers track the free surface deformation, and a structured, deforming grid is fitted to the free surface. The free-surface fitted grid is overlapping with a Cartesian background grid, and communication between the two grids occur at the highlighted nodes in each grid.

2.4 Time Evolution, Wave Generation and Wave Absorption

2.4.1 Time Marching

Both the IB and MG methods use a standard RK4 method to update the elevation and velocity potential of the free surface in time. This means that to prolong the solution from t to $t + \Delta t$, we need the solution at t and at three auxiliary time instants. Thus, for each time step, four solutions of the full boundary value problem for the velocity potential in the entire fluid domain are required.

2.4.2 Initiation of Waves Generated by a Wavemaker

A sudden start of a wavemaker at one end of the numerical wave tank can possibly lead to numerical instabilities and eventual breakdown of the simulation. Peregrine [26] explained this as a logarithmic temporal singularity in the associated initial value problem. To avoid this, we apply a ramp function $r(t)$ to the wavemaker signal $\bar{s}(t)$ so that the modified signal becomes $s(t) = \bar{s}(t) \cdot r(t)$. The ramp function is defined as

$$r(t) = \begin{cases} \frac{1}{2}\left(cos\left(\pi \frac{t}{T_{ramp}}\right)\right) & t < T_{ramp} \\ 1 & t \geq T_{ramp} \end{cases}, \qquad (18)$$

where T_{ramp} is the duration of the ramp. Usually, $T_{ramp} \in [2T, 5T]$, where T is the oscillation period of the wavemaker. The minimum required value of T_{ramp} depends on several factors, e.g. the wave steepness.

2.4.3 Modeling of a Physical Wavemaker in the IB Method

In the IB method, we model a physical flap (or piston) wavemaker by introducing an additional structured body-fitted and body-fixed grid on the flap (or piston). As evidenced by Figure 7, this grid slides on top of the background grid, and possesses its own set of free-surface markers. At any given time step, properties of the markers in the background grid covered by the overlapping grid are interpolated from the markers in this grid through a B-spline representation. The rationale behind this is that we assume that the solution close to the wavemaker has the most accurate representation in the wavemaker-fixed grid.

It is appropriate to point out that the principal difference between the wavemaker-fixed grid used in the IB method and the free-surface fitted grid in the MG method, is that the free surface is immersed in the wavemaker-fixed grid. This also implies that the grid will not be refined in areas with large local curvature of the free surface, i.e. its local geometry remains fixed throughout.

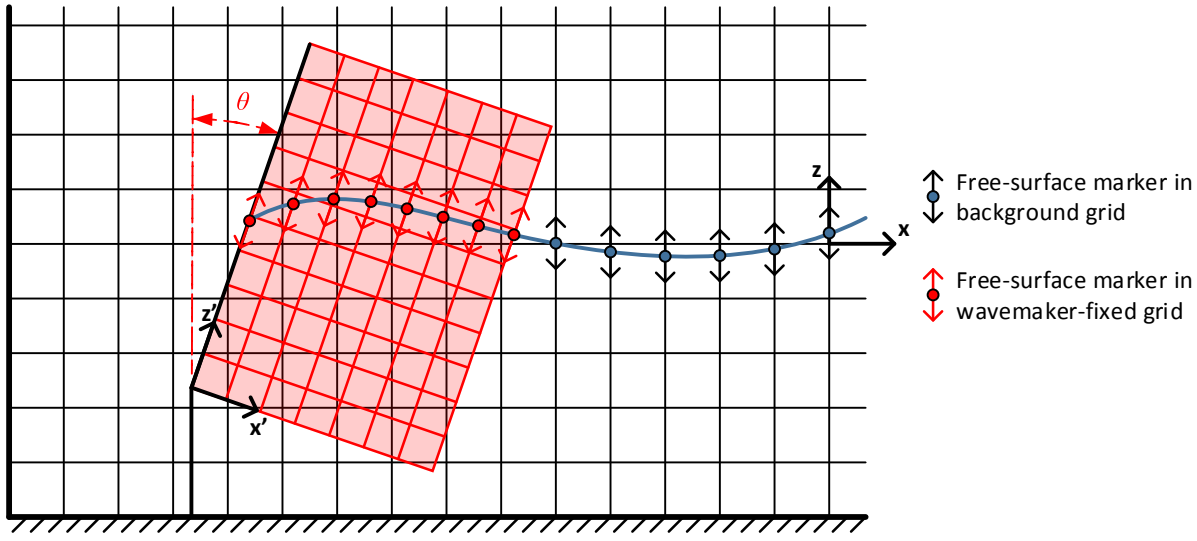


Figure 7 Modeling of physical wavemaker in IB method. Separate sets of markers are distributed in the Cartesian background grid and in the wavemaker-fixed grid. The markers are only allowed to move along their respective vertical coordinate axes, as indicated by the arrows associated with each marker.

Two important features must be addressed concerning the wavemaker-fixed grid: As described by the free-surface conditions in equations (8) - (9), the markers in the background grid are restricted to move along the global vertical axis. In the wavemaker-fixed grid, the markers are instead restricted to move along the local z' axis, where $'$ refers to the wavemaker-fixed reference frame. The semi-Lagrangian kinematic and dynamic free-surface conditions for a marker in this reference frame are:

$$\frac{Dx'_m}{Dt} = 0, \frac{Dz'_m}{Dt} = \frac{\partial \varphi}{\partial z'} + \left(u'_{grid} - \frac{\partial \varphi}{\partial x'}\right)\frac{\partial \eta'}{\partial x'} \qquad (19)$$

$$\frac{D\varphi_m}{Dt} = -\frac{1}{2}\left|\left(\frac{\partial \varphi}{\partial x'}\right)^2 + \left(\frac{\partial \varphi}{\partial z'}\right)^2\right| - U_g + u'_{grid}\frac{\partial \varphi}{\partial x'} + \left(w'_{grid} + \frac{\partial \eta'}{\partial t}\right)\frac{\partial \varphi}{\partial z'}. \qquad (20)$$

u'_{grid} and w'_{grid} are the horizontal and vertical velocity components of the grid point at a position (x'_m, z'_m) due to the wavemaker motion and $U_g = g\eta$ is the gravity potential. All quantities in the above equations marked with ‘ relate to an Earth-fixed (inertial) reference frame, but are resolved along the coordinate axes of the $x'z'$ system. Details on the derivation of equation (19) and (20) can be found in Appendix A.

Secondly, the two grids exchange information regarding the velocity potential through a two-way communication similar to that used in the MG method. With the aid of Figure 8, this can be explained as follows: In the wavemaker-fixed grid, the cells that at a given time step are outside the fluid domain and that do not contain any free-surface markers, are flagged as inactive. The remainder of the domain is active (the region bounded by the red lines in the figure). For nodes in the active part of the domain that are located along the right or lower grid boundaries, as indicated by green diamonds in the figure, equation (3) interpolates the velocity potential from the background grid:

$$\varphi_{WMG}(x', z') = \sum_{i=1}^{8}\left[\sum_{j=1}^{8} c_{j,i} f_j(\zeta, \xi)\right]\varphi_i\Big|_{BGG}. \qquad (21)$$

The right-hand side of equation (21) is evaluated from a cell in the background grid that contains the node in the wavemaker-fixed grid with coordinates (x', z'). Several cells generally fulfill this requirement. In this case, we select the cell where the interpolation point is closer to the cell center. Similarly, active nodes in the background grid that are inside the overlapping region get the velocity potential from interpolation in the wavemaker-fixed grid:

$$\varphi_{BGG}(x, z) = \sum_{i=1}^{8}\left[\sum_{j=1}^{8} c_{j,i} f_j(\zeta', \xi')\right]\varphi_i\Big|_{WMG}. \qquad (22)$$

These nodes correspond to the black diamonds in Figure 8. The cell used to evaluate the right-hand side of equation (22) is found by following the same strategy as for equation (21).

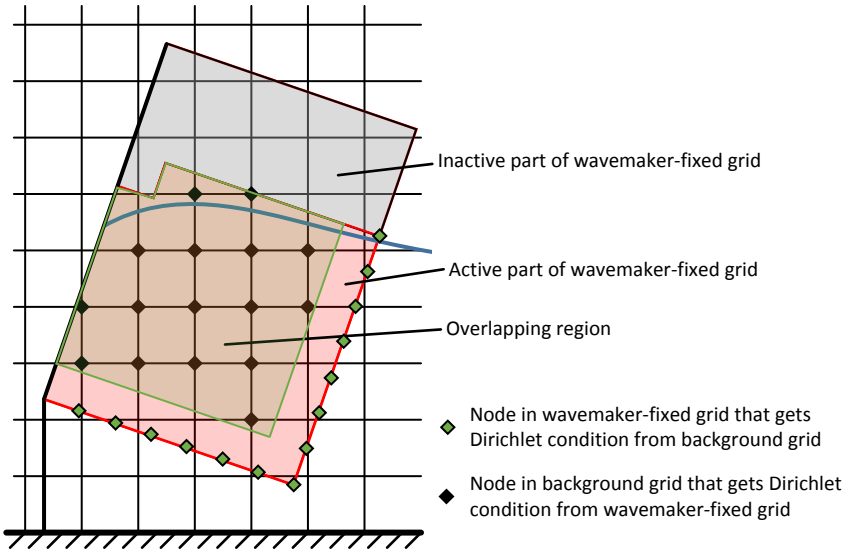


Figure 8 Coupling scheme between nodes in background grid and wavemaker-fixed grid in the IB method.

2.4.4 Modeling of a Physical Wavemaker in the MG Method

As already mentioned, the free-surface markers generally move only in vertical direction using a semi-Lagrangian formulation for the kinematic and dynamic free-surface conditions. However, in case of a physical wavemaker some adjustment must be made in order to account for the motion of the boundary. In this case, the background grid is rebuilt at every time step taking into account the instantaneous position of the wavemaker. For the free-surface markers close to the wavemaker, a fully-Lagrangian formulation for the free-surface boundary conditions is assumed in order to remain in the physical domain without having to resort to interpolation (or re-meshing). In particular, two ramp functions are applied both to the horizontal and vertical marker velocity components to transform the free-surface boundary conditions from the semi-Lagrangian version for the markers far away to the fully-Lagrangian version for the markers on the wavemaker. The corresponding kinematic and dynamic free-surface conditions are written as

$$\frac{Dx_p}{Dt} = \frac{\partial \varphi}{\partial x}(1 - R(x))$$
$$\frac{Dz_p}{Dt} = \frac{\partial \varphi}{\partial z} - \frac{\partial \varphi}{\partial x}\frac{\partial \eta}{\partial x}R_1(x) - \nu(x^*)\eta \qquad (23)$$

$$\frac{D\varphi_p}{Dt} = -\frac{1}{2}\left|\left(\frac{\partial \varphi}{\partial x}\right)^2 + \left(\frac{\partial \varphi}{\partial z}\right)^2\right| - g\eta - \left(\frac{\partial \varphi}{\partial x}\right)^2 R(x) - \frac{\partial \varphi}{\partial x}\frac{\partial \varphi}{\partial z}\frac{\partial \eta}{\partial x}R_1(x) - \nu(x^*)\varphi, \qquad (24)$$

where the ramp function for the free-surface fitted grid's normal direction is defined as

$$R(x) = \begin{cases} 5\left(1 - cos\left(\frac{\pi x}{L_r}\right)\right) & x < L_r \\ 1 & x \geq L_r \end{cases} \qquad (25)$$
$$R_1(x) = \frac{R^2(x)}{R^2(x) - R(x) + 1}.$$

In equation (25), L_r is the length of the ramp zone. Typically L_r can be chosen so that the ramp is only applied in a localized region near the wavemaker, but this will depend on the stroke of the wavemaker.

2.4.5 Absorption of Waves with a Numerical Beach

As explained earlier, damping terms associated with a numerical beach are added to the kinematic and dynamic free-surface conditions to avoid wave reflections (see equations (8) and (9)). A cubic function used by e.g. Greco [27] expresses the damping coefficient $\nu(x^*)$:

$$\nu(x^*) = \begin{cases} 0 & x^* < 0 \\ \nu_{max}(-2\zeta^3 + 3\zeta^2) & 0 \leq x^* < l_{beach} \end{cases}. \qquad (26)$$

l_{beach} is the length of the numerical beach, and the auxiliary coordinate x^* is defined as $x^* = x - L_x/2 + l_{beach}$ with the origin of the x axis in the center of a tank with length L_x. In equation (26), $\zeta = x^*/l_{beach}$ and ν_{max} is an empirical damping coefficient. $\nu_{max} \in [1,3]$ combined with a long beach is generally found more appropriate than a higher ν_{max} combined with a very steep beach, since the latter may lead to reflections from the beach itself. With "long" we here mean that l_{beach} is at least equal to two wavelengths of the longest relevant wave component.

3 Overview of Case Studies

Several case studies are conducted with the numerical wave tank outlined in section 2 with both the IB and MG free-surface tracking methods. A brief overview of the cases and the main reason why they are chosen is given in Table 1. A more detailed description of each case, along with their sources for analytical, experimental or numerical reference results follows thereafter. Section 4 presents results for all the cases described below.

Table 1 Brief description of studied cases. Key features of each case are described, referring to the particular physical aspects involved in each case.

Case no.	Description	Key aspects/reason for selecting case
1	Initial-value problem with periodic boundary conditions.	• Wave propagation in an infinitely long wave tank without a physical wavemaker. • Convergence of results for various values of wave steepness.
2	Solitary wave in shallow water.	• Ability to propagate single waveform over a distance.
3	Focused wave packet.	• Propagation of wave train with different wavenumber components. • Very steep/breaking wave.
4	Periodic waves in shallow water.	• Nonlinear dispersion, transfer of energy between harmonic modes.
5	Periodic waves in deep water.	• Propagation of regular waves with different steepness in a long wave tank.

3.1 Case 1: Initial Value Problem with Periodic Boundary Conditions

We first consider a numerical wave tank with periodic boundary conditions. In this particular case, specifying the initial values of the wave elevation and free-surface potential from a known solution initiates the wave propagation. For subsequent time steps, the RK4 scheme evolves the free surface in time. Figure 9 illustrates the associated boundary value problem with periodic boundary conditions. The initial state of the free surface comes from the higher-order solution given by Rienecker and Fenton [3], which also serves as an accurate approximation of the analytical solution at subsequent time steps. The Rienecker and Fenton (RF) solution builds on Dean's stream-function theory, representing steadily progressing periodic waves in a potential flow over a flat seabed as a finite Fourier series. We are free to decide the number of Fourier components, but too few components will generally make the solution less accurate. The more nonlinear features the wave possesses, the larger the number of Fourier components required. However, using an excessive number of Fourier components makes the analytical solution computationally more expensive, since each component is associated with solving a nonlinear equation.

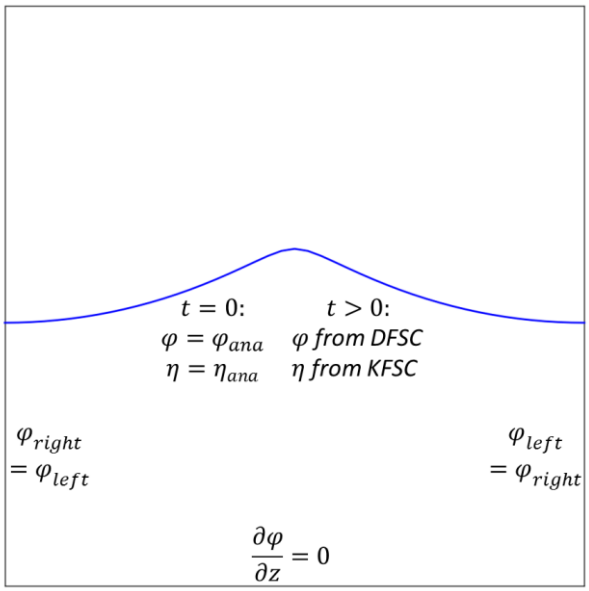


Figure 9 Principle of numerical wave tank with periodic boundary conditions. DFSC refers to the dynamic free-surface condition while KFSC refers to the kinematic free-surface condition.

The periodic boundary conditions effectively model an infinitely long wave tank. Since the numerical solution initiates from the theoretical one, the wave generation comes from an initial-value problem without a numerical wavemaker that could influence the accuracy of the simulated waves. Thus, we consider this as an ideal case for studying the wave-propagating abilities of the two grid approaches. In addition, since we have verification results in form of an analytical expression, it is a convenient case for doing convergence studies.

Cases with four different values of the wave steepness kA are studied: $kA = 0.1$ represents a wave with a relatively moderate slope, whereas $kA = 0.4$ is a very steep wave. To illustrate how steep this wave is, let us take as a kinematic criterion for wave breaking when the horizontal crest-particle velocity exceeds the wave's phase speed. Then, by using a Stokes wave solution, the maximum achievable wave height before breaking in finite water depth according to Mei et al. [28] is

$$\left(\frac{H}{\lambda}\right)_{max} = 0.14\, tanh(kh), \qquad (27)$$

where H is the wave height, λ is the wave length and k is the wave number. Here we study a case with water depth $h = 0.5\lambda$, which gives a maximum steepness of $kA_{max} \approx 0.438$ from equation (27). This means that in the case $kA = 0.4$, we try to propagate over a long time a wave that is close to breaking. This represents a challenging scenario with respect to stability of the wave, both experimentally and not least numerically.

For each wave steepness, we consider four grid densities to evaluate the grid convergence. Only square grids are used throughout, although the free-surface fitted grid in the MG simulations become distorted due to the free-surface deformation. The time step is fixed at $\Delta t = 0.01s$ in all the simulations. The wave period corresponding to the given wave length is approximately $2.5s$ (although it decreases slightly with

increasing steepness), meaning that we use approximately 250 time steps per wave period in the numerical simulation. This temporal discretization is based on the convergence study presented in Appendix B, so to study the grid convergence with confidence that the results are converged in time. Table 2 describes the spatial discretization in all analysis cases for the different values of kA. For all cases, the length of the tank is set equal to the wavelength $L = \lambda = 10m$, and the water depth is $h = 5m$.

Table 2 Analysis cases for wave tank with periodic boundary conditions. $(n_{x,IB}, n_{z,IB})$ are the horizontal and vertical number of grid points in the IB simulations, $(n_{x,MGb}, n_{z,MGb})$ are the horizontal and vertical number of grid points in background grid in the MG simulations and $(n_{x,MGf}, n_{z,MGf})$ are the tangential and normal number of grid points in in the corresponding free-surface fitted MG grid.

$kA\ (-)$	$n_{x,IB}\ (-)$	$n_{z,IB}\ (-)$	$n_{x,MGb}\ (-)$	$Nn_{z,MGb}\ (-)$	$n_{x,MGf}\ (-)$	$Nn_{z,MGf}\ (-)$
0.1	15	15	15	15	15	8
	30	30	30	30	30	8
	45	45	45	45	45	8
	60	60	60	60	60	8
0.2	30	30	30	30	30	8
	45	45	45	45	45	8
	60	60	60	60	60	8
	75	75	75	75	75	8
0.3	30	30	30	30	30	8
	45	45	45	45	45	8
	60	60	60	60	60	8
	75	75	75	75	75	8
0.4	45	45	45	45	45	8
	60	60	60	60	60	8
	75	75	75	75	75	8
	90	90	90	90	90	8

In the IB method, the free-surface elevation and velocity potential at the free surface are filtered at every time step by applying a 12th order Savitsky-Golay filter. This is found necessary to simulate the steepest waves over a long time period without any instability developing. If free-surface filtering is not applied, sawtooth instability eventually occurs for waves over a certain steepness. The onset of instability occurs earlier for waves with high steepness than it does for waves with more moderate steepness. Longuet-Higgins and Cokelet [29] were the first to suggest the use of such a spatial filter to avoid these types of instabilities. In the MG method, the use of a free-surface filter is not found to be necessary.

3.2 Case 2: Solitary Wave in Shallow Water

Russell [30] in 1844 first reported about the existence of a solitary wave, characterized by a single wave crest that can travel over a large distance with a constant wave profile. The years following saw the development of several theories to describe solitary waves. In the present paper, we concentrate on the ninth-order solution by Fenton [31]. The solution is an approximate one, with the free-surface elevation $\eta(x,t)$ expanded in a series with parameter $\epsilon = A/h$, where A is the amplitude of the solitary wave and h is the water depth. In an analogue way, a series expansion truncated at ninth order also expresses the

phase speed c. Fenton concludes that this method can give accurate solutions for ϵ up to 0.75. However, the practical limit seems to be somewhat lower ($\epsilon \approx 0.60$).

To study the propagation of a solitary wave numerically, we can pursue one of the two following alternative ways to generate the wave: Either, we can specify the initial state of the free surface, or we can generate the wave with a piston-type wavemaker. The former alternative represents an initial value problem well suited for a periodic computational domain similar to that in Figure 9. Here we instead choose to generate the solitary wave with an appropriately moved piston wavemaker. Following the approach of Wu et al. [32], by assuming the piston velocity to be equal to the average horizontal water particle velocity under the crest of the target wave at the position of the piston, we have that

$$\frac{dx_{wm}(t)}{dt} = \frac{c\eta}{h+\eta}\bigg|_{x=x_{wm}} . \qquad (28)$$

The phase speed c and surface elevation η follow from the ninth-order solution described above (hereafter referred to as the analytical solution). We use equation (28) to generate a wavemaker signal, characterized by its displacement $x_{wm}(t)$ and velocity $dx_{wm}(t)/dt$, by assuming that the analytical wave crest initially is located a certain distance before the wavemaker. We then let the analytical wave propagate with its analytical phase speed in the direction of the wavemaker, and at each time step evaluate the velocity of the wavemaker at its instantaneous position and integrate its displacement forward in time. The scenario is illustrated in the left part of Figure 10, while the right part shows the obtained wavemaker signal for $\epsilon = 0.40$ and $h = 0.4m$. Here the crest of the analytical solution is initially located $6.5m$ left of the wavemaker, which is found to be sufficient for all simulated cases. With sufficient, we here mean that the initial position of the analytical crest is so that it gives a negligible perturbation of the free surface at the wavemaker position. Wu et al. [32] concludes that equation (28) in combination with the ninth-order analytical solution enables accurate numerical simulations up to $\epsilon = 0.40$.

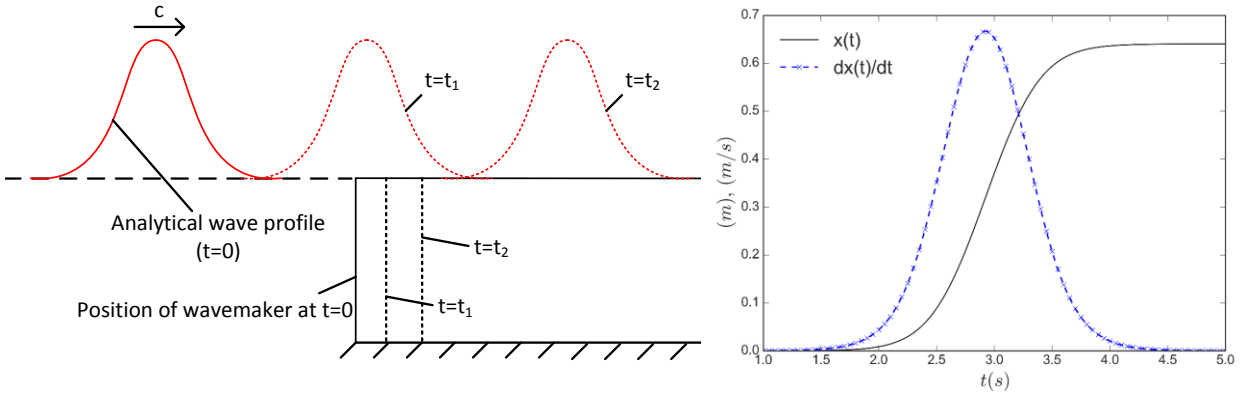


Figure 10 Left: Principle for generation of wavemaker signal. The analytical wave profile and associated piston position is depicted for three time instants $t = 0$ and $t_2 > t_1 > 0$. Right: example of piston displacement and velocity for $\epsilon = 0.40$ and $h = 0.4m$.

To determine an appropriate grid density for the numerical simulations, we need an estimate of the wavelength λ associated with the solitary wave. Here, we use the wavenumber as suggested by Rayleigh [32],

$$k = \sqrt{\frac{3A}{4h^2(h+A)}}, \quad (29)$$

so that $\lambda = 2\pi/k$. Furthermore, we can use the phase speed from the analytical solution to determine an associated time step Δt that fulfills the Courant-Friedrichs-Lewy (CFL) requirement $CFL = c\Delta t/\Delta x \leq 1$, where Δx is the horizontal grid spacing. In all cases, the length of the tank was $L = 30.0m$ and the depth $h = 0.40m$. The time step is fixed at $\Delta t = 0.015s$ in all the simulations, with the rest of the parameters as presented in Table 3. One must be aware that the same discretization is used for simulations with different values of ϵ. With this choice, the number of grid points per wavelength varies from approximately 142 (for $\epsilon = 0.20$) to 95 (for $\epsilon = 0.60$). This is deemed sufficient to give an accurate resolution of the wave profile for all values of ϵ, which justifies the choice to keep a constant discretization. The time step is set so that all the cases fulfill the CFL requirement. The numerical setup does not include a numerical beach, and the post-processing therefore only considers the part of the time series before the solitary wave reaches the downstream tank wall.

Table 3 Analysis cases for solitary wave. $(\Delta x_{IB}, \Delta z_{IB})$ is the grid spacing in the IB simulations, $(\Delta x_{MGb}, \Delta z_{MGb})$ is the grid spacing in the background grid in the MG simulations and $(\Delta x_{MGf}, \Delta z_{MGf})$ is the average grid spacing in the corresponding free-surface fitted grid. The amplitude, wavelength, phase speed and CFL number based on the analytical solution are presented for all values of ϵ.

$\Delta x_{IB}\ (m)$	$\Delta z_{IB}\ (m)$	$\Delta x_{MGb}\ (m)$	$\Delta z_{MGb}\ (m)$	$\Delta x_{MGf}\ (m)$	$\Delta z_{MGf}\ (m)$
0.05	0.05	0.05	0.05	0.05	0.017

Parameters from analytical solution					
ϵ	0.20	0.30	0.40	0.50	0.60
$A\ (m)$	0.08	0.12	0.16	0.20	0.24
$\lambda\ (m)$	7.11	6.04	5.43	5.03	4.74
$c\ (m/s)$	2.17	2.25	2.33	2.41	2.48
$CFL\ (-)$	0.65	0.68	0.70	0.72	0.74

3.3 Case 3: Focused Wave Packet

Dommermuth et al. [33] produced a very steep and eventually plunging wave experimentally by generating a wave packet with a programmed piston-type wavemaker. The involved wave components were generated starting from the shortest and then increasing progressively towards the longest wavelength to generate an amplified wave crest at a certain location and time. The experiments took place in a $25m$ long, $0.7m$ wide and $0.6m$ deep-water channel. The tank was fitted with a damping beach starting at a distance $19.5m$ from the wavemaker. Nine wave probes were installed along the tank at the following non-dimensional distances from the wavemaker:

$$x/h = \{3.17, 5.00, 6.67, 8.33, 9.17, 10.00, 10.83, 11.83, 12.17\},$$

where h is the water depth. The breaking wave occurs in the focusing point at $x/h \approx 12.08$ after $t\sqrt{g/h} \approx 52.15$. Simulations were carried out with the three grid densities described in Table 4. For all simulations, a time step $\Delta\bar{t} = \Delta t\sqrt{g/h} = 0.05$ is used. The simulations include a numerical damping zone as described by equation (26) with parameters $l_{beach}/h = 8.0$ and $\nu_{max} = 2.0$. This choice can be justified by the fact that the frequency component of the wavemaker signal with largest velocity ($\bar{\omega} = \omega\sqrt{g/h} = 1.139$ and

non-dimensional velocity amplitude $\bar{U} = U/\sqrt{gh} = 0.0115$) gives CFL numbers ($CFL = \frac{\bar{U}\Delta\bar{t}}{\Delta x/h}$) well below unity with all the grid densities applied. The length of the numerical beach is approximately equal to the length of the beach used in the experiments ($5.5m/h \approx 9.2$).

Table 4 Grid refinements for focused wave packet by Dommermuth et al. [33]. $(\Delta x_{IB}, \Delta z_{IB})$ is the grid spacing in the IB simulations, $(\Delta x_{MGb}, \Delta z_{MGb})$ is the grid spacing in the background grid in the MG simulations and $(\Delta x_{MGf}, \Delta z_{MGf})$ is the average grid spacing in the corresponding free-surface fitted grid.

Case	$\Delta x_{IB}/h\ (-)$	$\Delta z_{IB}/h\ (m)$	$\Delta x_{MGb}/h\ (-)$	$\Delta z_{MGb}/h\ (-)$	$\Delta x_{MGf}/h\ (-)$	$\Delta z_{MGf}/h\ (-)$
Coarse	0.08	0.08	0.11	0.08	0.11	0.07
Medium	0.06	0.06	0.07	0.06	0.07	0.05
Fine	0.04	0.04	0.05	0.04	0.05	0.04

The focused wave packet also serves as an evaluation case for the previously described scheme to suppress wave-breaking tendencies in the IB method. Actually, the wave in this case is a plunging breaker, which represents a considerably more energetic phenomenon than a spilling breaker. Thus, we can expect that this type of breaking is rather challenging to suppress numerically without introducing significant artificial energy dissipation. For this reason, the selected case can give us valuable information about how the suppression scheme operates when pushed to its limit. Though not done here, this strategy could also be applied to the MG method, as well as to other methods using free-surface conditions in the same form.

3.4 Case 4: Periodic Waves in Shallow Water

Chapalain et al. [34] studied the transformation of long waves of moderate steepness in shallow water due to nonlinear transfer of energy between harmonic modes. In a $35.54m$ long and $0.55m$ wide wave channel, a harmonically oscillating piston wavemaker generated waves at one end of the tank. We concentrate on the experiments conducted with water depth $h = 0.4m$ for two different cases. In the experiments, the harmonic motion of the wavemaker is described by

$$x(t) = -s \cdot cos(\omega t), \qquad (30)$$

where s is the stroke and ω is the oscillation frequency as given in Table 5. This table also describes the grid density used in the corresponding numerical simulations.

Table 5 Parameters for wavemaker signal described by equation (30) for analyzed cases from [34]. The grid parameters for the numerical simulations are also given, where $(\Delta x_{IB}, \Delta z_{IB})$ is the grid spacing in the IB simulations, $(\Delta x_{MGb}, \Delta z_{MGb})$ is the grid spacing in the background grid in the MG simulations and $(\Delta x_{MGf}, \Delta z_{MGf})$ is the average grid spacing in the corresponding free-surface fitted grid.

Case	$s\ (m)$	$\omega\ (s^{-1})$	$\Delta x_{IB}\ (m)$	$\Delta z_{IB}\ (m)$	$\Delta x_{MGb}\ (m)$	$\Delta z_{MGb}\ (m)$	$\Delta x_{MGf}\ (m)$	$\Delta z_{MGf}\ (m)$
A	0.078	2.5133	0.08	0.08	0.16	0.08	0.08	0.03
C	0.113	1.7952						

By using the linear dispersion relation to estimate the associated wavelengths in Table 5, the number of grid points per wavelength is approximately 60 for case A and 85 for case C. The experimental setup did not include a damping beach, which according to [34] was to avoid parasitic reflections. Thus, in order to

omit reflected waves, the post-processing only included analysis of waves over the first $25m$ of the channel. More specifically, the analysis consisted in computing the amplitudes of the first four harmonic modes of the wave elevation from a Fourier analysis.

In the numerical simulations, we use a $45m$ long domain. A time step of $\Delta t = 0.08s$ is used throughout, which gives more than 30 time steps per oscillation period for both case A and C. The ramp function from equation (18) is deployed with $T_{ramp} = 2 \cdot 2\pi/\omega$ to initiate the wave generation. No details regarding the ramp function in the experiments are available. However, since the objective of the study is to compute the Fourier amplitudes over the first $25m$ of the tank, the only concern in our numerical setup is to provide a steady-state wave-elevation time series along all relevant stations of the tank. The latter must then be long enough to establish the Fourier components properly. In principle, it is sufficient to record one complete steady-state wave period in each wave probe to achieve this. With "steady state", we here mean the part of the time series after transient effects have disappeared.

3.5 Case 5: Periodic Waves in Deep Water

Lugni [35] investigated the propagation of deep-water periodic waves with different steepness both experimentally and numerically. The experiments took place in a very long towing tank with a water depth of $3.6m$ and with a flap wavemaker hinged at half the water depth. The wave elevation was recorded in a wave probe located $12m$ away from the wavemaker. Wave reflections were not of concern in the experiments due to the length of the tank. We select the experimental cases with wave steepness $kA = \{0.10, 0.15, 0.20\}$ for the numerical study, using the exact time signal of the wavemaker from the experiments as input. In the simulations, we restrict the length of the wave tank to $25m$ which is a little more than eight wave lengths ($\lambda \approx 2.8m$). The frequency of the wavemaker motion is $\omega = 4.71s^{-1}$ for all wave amplitudes. A numerical beach is included towards the end of the tank with parameters $l_{beach} = 8m$ (i.e. $l_{beach} \approx 3\lambda$) and $\nu_{max} = 2.0$. A time step $\Delta t = 0.033s$, giving approximately 40 time steps per wave period, and grid sizes as given by Table 6 are used throughout. The grid sizes are chosen in such way that they give at least 28 grid points per wavelength on the free surface.

Table 6 Numerical analysis cases for periodic wave in deep water. $(\Delta x_{IB}, \Delta z_{IB})$ is the grid spacing in the IB simulations, $(\Delta x_{MGb}, \Delta z_{MGb})$ is the grid spacing in the background grid in the MG simulations and $(\Delta x_{MGf}, \Delta z_{MGf})$ is the average grid spacing in the corresponding free-surface fitted grid.

$\Delta x_{IB}\ (m)$	$\Delta z_{IB}\ (m)$	$\Delta x_{MGb}\ (m)$	$\Delta z_{MGb}\ (m)$	$\Delta x_{MGf}\ (m)$	$\Delta z_{MGf}\ (m)$
0.10	0.10	0.20	0.20	0.10	0.07

4 Results from Case Studies

In this section, we discuss results from the simulation cases described in section 3.

4.1 Case 1: Initial Value Problem with Periodic Boundary Conditions

As mentioned previously, we consider the approximate solution by Rienecker and Fenton [3] as a semi-analytical solution for the initial value problem with periodic boundary conditions. Strictly speaking, this solution is approximate and approaches the exact solution in the limit $N_{RF} \to \infty$, where N_{RF} is the number

of components included in the Fourier series. Here we generally use $N_{RF} = 20$, since the contribution from higher-order terms is negligible and anyway difficult to resolve numerically in a reasonable way (since the associated wavelengths of these components are very small).

A general quantity evaluated from the analytical solution is denoted χ_{RF}, and can represent either the surface elevation or the velocity potential in an arbitrary point in the fluid or along the free surface. The N^{th} norm error of the corresponding numerical solution χ_{num} at a given time t is computed as

$$\left|L_{N,\chi}(t)\right| = \left(\frac{\sum_{i=1}^{M}|\chi_{num}(\vec{x}_i,t) - \chi_{RF}(\vec{x}_i,t)|^N}{\sum_{i=1}^{M}|\chi_{RF}(\vec{x}_i,t)|^N}\right)^{\frac{1}{N}}, \quad (31)$$

where $i = 1, \dots, M$ represents a set of spatial points. In the following, we concentrate on the $|L_2|$ norms of the free-surface elevation ($\left|L_{2\eta}\right|$) and the free-surface velocity potential ($\left|L_{2\varphi,FS}\right|$) computed at every time step of the simulation. In addition, we evaluate the conservation of fluid volume in the tank by integrating the surface elevation over the length of the domain and normalizing it against the fluid volume V_0 in the domain at rest:

$$\Delta V(t) = \frac{\int_{-0.5L_x}^{0.5L_x}\eta(x,t)dx}{V_0}. \quad (32)$$

The change in potential energy in the tank follows by integrating the difference between the wave elevation and the mean water level $\bar{\eta}(t)$ over the length of the domain,

$$PE(t) = \frac{E_p(t)}{\rho g} = \int_{-0.5L_x}^{0.5L_x}[\eta(x,t) - \bar{\eta}(t)]dx, \quad (33)$$

while conservation of kinetic energy associated with the free-surface process is estimated from

$$KE(t) = \frac{E_k(t)}{\rho} = \frac{1}{2}\int_{-0.5L_x}^{0.5L_x}|\nabla\varphi(x,\eta(x,t),t)|^2dx. \quad (34)$$

The numerical integration in equation (32) - (34) is carried out by either Simpson's rule or the trapezoidal rule, which give indistinguishable results in the analysis. $E_p(t)$ and $E_k(t)$ here represent the fully dimensional potential and kinetic energy per unit length in y direction.

Figure 11 - Figure 14 show how the quantities described above develop over time in the simulation for the four different values of kA. In the figures, $PE(t)$ is omitted if $\Delta V(t)$ is displayed and vice versa, since they exhibit similar behavior. The figures also include analytical values for $\Delta V(t)$, $PE(t)$ and $KE(t)$. These are obtained by applying equation (32) - (34) directly on the analytical free-surface elevation. The IB and MG free-surface treatments generally give $|L_2|$ norms of similar magnitude for both the free-surface elevation and velocity potential. In some cases, the MG errors are smaller, while in other they are somewhat larger than the IB errors. This probably relates to the modelling approach in the MG method, where the condition of the free-surface grid and the overlapping with the background grid may change as a function of the grid parameters. Specifically, since there is an interpolation between the grids, the accuracy of the potential at the interpolation points may vary slightly with the grid configuration.

Moreover, the amount of stretching and distortion in the free-surface fitted grid in the MG method may also vary. On the other hand, in the IB method the accuracy associated with the free surface depends on the position of the free-surface marker within each free-surface cell (see Figure 2). Thus, if we consider a single marker, the accuracy of the potential can oscillate in a stochastic-like manner in time. We expect this behavior to cancel out globally when we consider the entire free surface, although there is strictly no guarantee that this will always be the case. Even though the $|L_2|$ norms indicate differences between the IB and MG methods, the results for $\Delta V(t)$, $PE(t)$ and $KE(t)$ appear visually in perfect agreement with a few notable exceptions:

- For the coarsest IB grid in Figure 11 ($n_x = n_z = 15$), there is a small drift in $\Delta V(t)$ so that it decreases in time. The magnitude of the drift is so that it is hard to observe visually from the figure.
- For the coarsest MG grid in Figure 14 ($n_x = n_z = 45$), inconsistent behavior of $|L_{2\eta}|$ and $|L_{2\varphi,FS}|$ is observed after $t/T \approx 10$.
- For the three finest IB grids in Figure 14 ($60 \leq n_x = n_z \leq 90$), numerical instability occurs toward the end of the simulation. This is manifested through oscillatory behavior and eventual divergence of $|L_{2\eta}|$ and $|L_{2\varphi,FS}|$.
- For the IB simulation in Figure 14 with $n_x = 60$, $|L_{2\eta}|$ is lower than for $n_x = 75$. Although the difference is minor (logarithmic scale), this is unexpected and was not observed for $kA < 0.4$. For an immersed boundary method, we can always anticipate some degree of oscillatory spatial convergence, which may contribute to explain this observation. In this respect, whether n_z is an even or odd number will also have some influence in the IB scheme due to the way we select cells to be associated with each free-surface marker.

For all the values of kA, the estimates of fluid volume, potential energy and kinetic energy converge against the analytical value when the grid density increases. For $kA < 0.4$, the results are always visually converged for the finest grid density in each case.

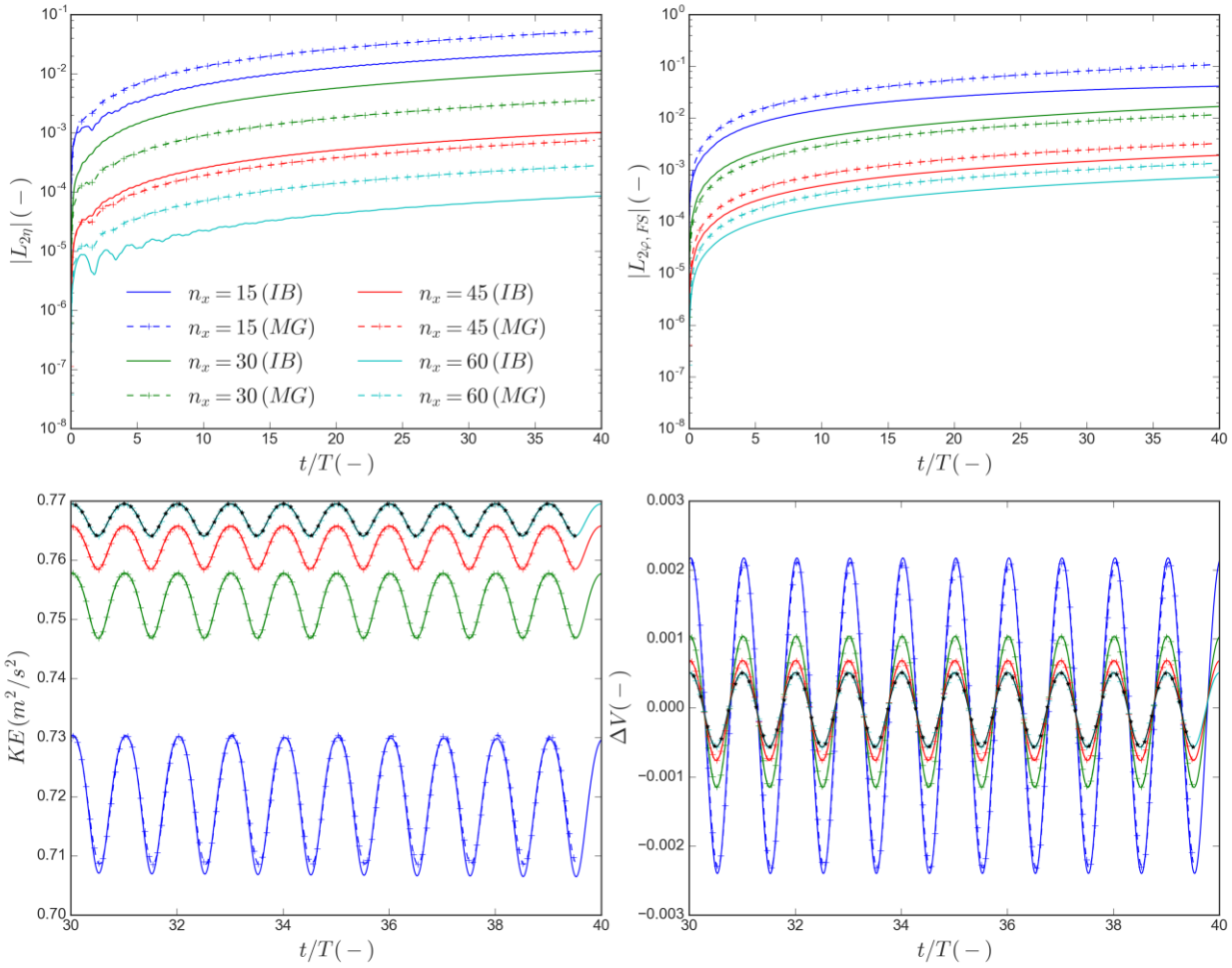


Figure 11 L_2 errors of free-surface elevation ($|L_{2\eta}|$) and free-surface potential ($|L_{2\varphi,FS}|$), kinetic-energy and integrated fluid volume estimates plotted as a function of time normalized by wave period for steepness $kA = 0.1$. Black lines with filled markers show analytical values. The following notation applies: $n_x = n_{x,IB} = n_{x,MGb} = n_{x,MGf}$, $n_z = n_{z,IB} = n_{z,MGb}$. $n_{z,MGf} = 8$ for all cases.

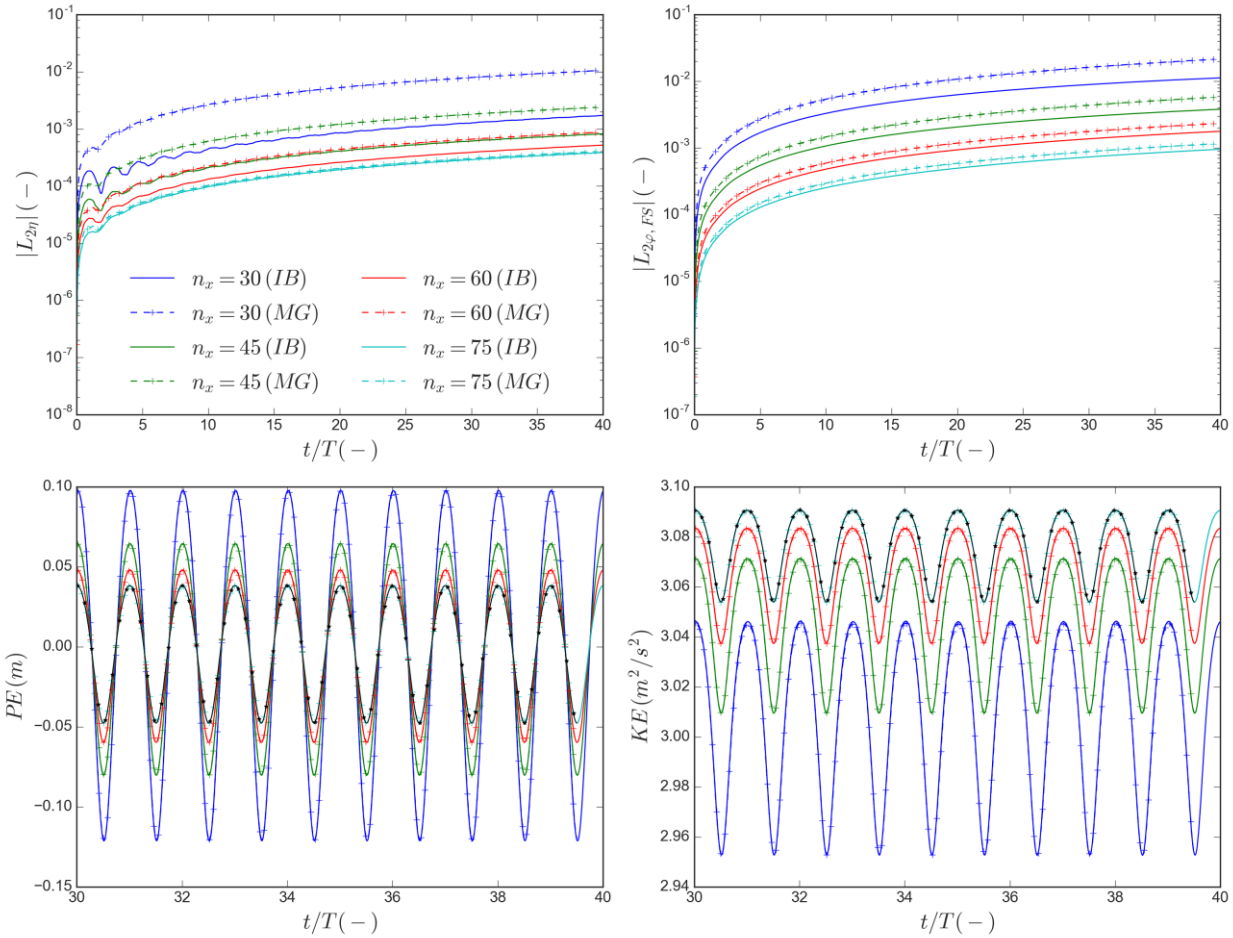


Figure 12 L_2 errors of free-surface elevation ($|L_{2\eta}|$) and free-surface potential ($|L_{2\varphi,FS}|$), potential energy and kinetic-energy estimates plotted as a function of time normalized by wave period for steepness $kA = 0.2$. Black lines with filled markers show analytical values. The following notation applies: $n_x = n_{x,IB} = n_{x,MGb} = n_{x,MGf}$, $n_z = n_{z,IB} = n_{z,MGb}$. $n_{z,MGf} = 8$ for all cases.

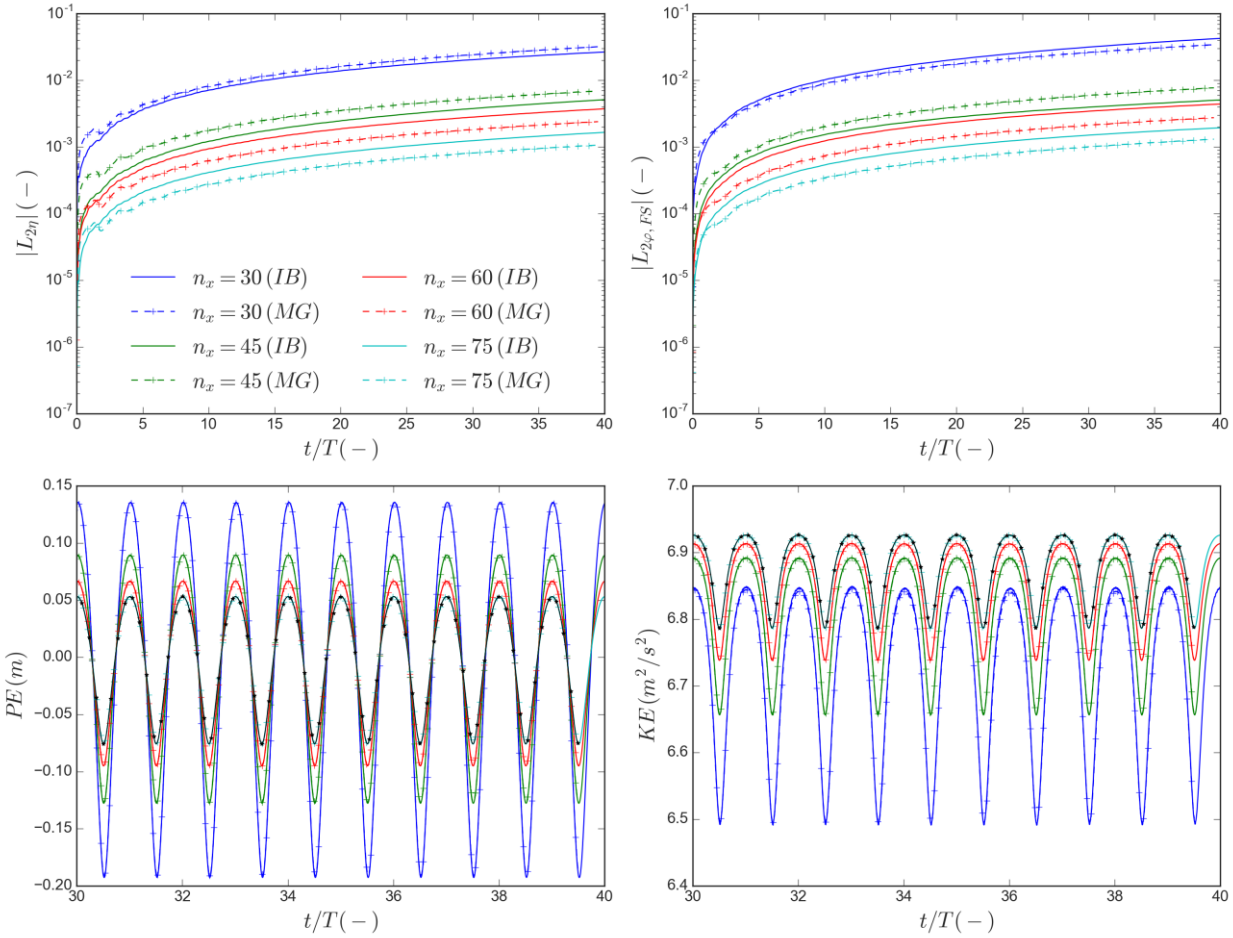


Figure 13 L_2 errors of free-surface elevation ($|L_{2\eta}|$) and free-surface potential ($|L_{2\varphi,FS}|$), potential energy and kinetic-energy estimates plotted as a function of time normalized by wave period for steepness $kA = 0.3$. Black lines with filled markers show analytical values. The following notation applies: $n_x = n_{x,IB} = n_{x,MGb} = n_{x,MGf}$, $n_z = n_{z,IB} = n_{z,MGb}$. $n_{z,MGf} = 8$ for all cases.

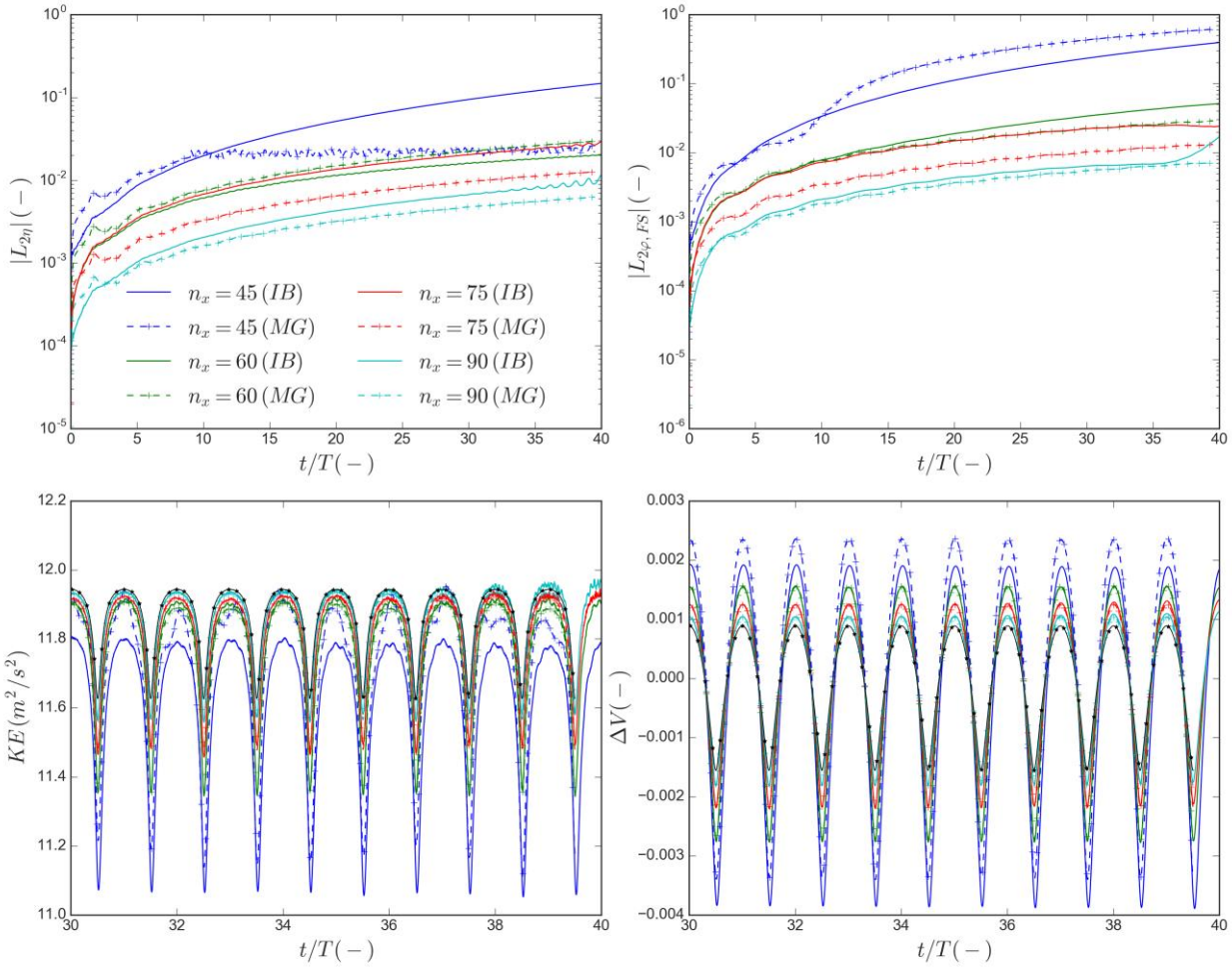


Figure 14 L_2 errors of free-surface elevation ($|L_{2\eta}|$) and free-surface potential ($|L_{2\varphi,FS}|$), kinetic-energy and integrated fluid volume estimates plotted as a function of time normalized by wave period for steepness $kA = 0.4$. Black lines with filled markers show analytical values. The following notation applies: $n_x = n_{x,IB} = n_{x,MGb} = n_{x,MGf}$, $n_z = n_{z,IB} = n_{z,MGb}$. $n_{z,MGf} = 8$ for all cases.

One should stress that the results in Figure 14 are for a very steep wave ($kA = 0.4$) challenging to simulate over a long time with a constant form. Figure 15 shows snapshots of the wave elevation at the end of the IB simulation after 40 wave periods for the lowest and highest wave steepness, each with two different levels of spatial discretization. The wave with $kA = 0.1$ has only moderate crest-to-trough asymmetry, whereas the wave with $kA = 0.4$ has a crest that is nearly double as large as the absolute value of the trough. For both values of wave steepness, the coarsest grids show a mild phase difference between the simulated and analytical wave at the end of the simulation. This phase difference is no longer visually observable with the finer grids. In addition, the lower right snapshot in the figure shows the nature of the aforementioned numerical instability in the IB results for $n_x = n_z = 90$ towards the end of the simulation. This instability manifests itself in the area of the wave crest, and has the appearance of a beginning sawtooth-like behavior.

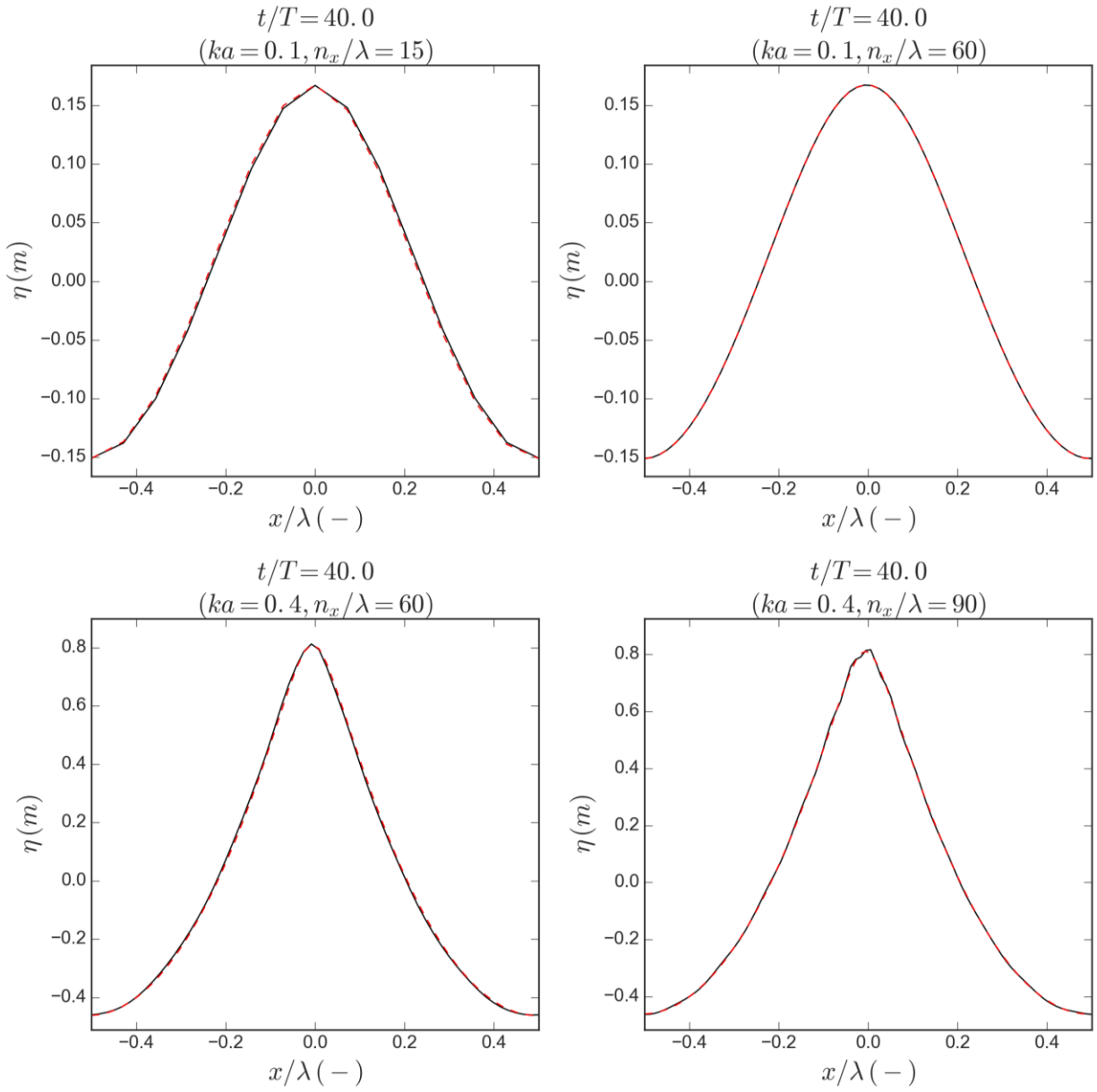


Figure 15 Snapshots of wave elevation in the tank after 40 wave periods for wave steepness $kA = 0.1$ (upper row) and $kA = 0.4$ (lower row), with a coarse (left column) and fine (right column) spatial discretization. Numerical results for the IB method are plotted with a black line (-) and analytical results at the same horizontal grid points are plotted with a red dashed line (--).

It is interesting to note that while the finest grid leads to numerical instability in the IB method for $kA = 0.4$ even when a 12th order Savitsky-Golay free-surface filter is applied, this is not observed in the MG method. It is thus likely that the configuration of the free surface with respect to the grid gives birth to the instability. However, since the onset of the instability is gradual, it is not possible to pinpoint exactly the time step where it emerges. To understand better the nature of the instability in the IB method, we compute the relative absolute error of the numerical wave elevation as

$$\left|e_\eta(x,t)\right| = \frac{\left|\eta_{num}(x,t) - \eta_{RF}(x,t)\right|}{\eta_{RF,max}}, \qquad (35)$$

where $\eta_{RF,max}$ is the maximum wave elevation from the analytical solution. $\left|e_\eta(x,t)\right|$ is plotted for eight different time instances for the IB method in Figure 16. To the left for the case with $kA = 0.3$, $\left(n_{x,IB}, n_{z,IB}\right) = (75,75)$ and to the right for the case with $kA = 0.4$, $\left(n_{x,IB}, n_{z,IB}\right) = (90,90)$. The same scale of the ordinate axis is used in all plots: $0.0 \leq \left|e_\eta(x,t)\right| < 0.012$. With $kA = 0.3$, the error does not increase much with time, and generally has its minimum at the positions of the wave crest and wave through. Although larger, $\left|e_\eta(x,t)\right|$ displays similar behavior for the first time instance shown for $kA = 0.4$. Thereafter, a sawtooth-type of behavior develops around the wave crest. The error pattern develops rather symmetrically about the wave crest, which propagates in positive x-direction (i.e. towards the right in the figures). The last time instance shown in the figure coincide with the time when oscillations can be seen in $\left|L_{2\eta}\right|$ from Figure 14.

In simulations where a free-surface filter is not applied, the instable behavior shown to the right in Figure 16 quickly escalates and the simulation fails within a limited number of time steps after its onset. The free-surface filter is able to slow down this escalation significantly so that the simulation survives for several wave periods. The filter is however not able to prevent the onset of instability in the first place. It is possible that another type of spatial filter, or the applied Savitsky-Golay filter with other settings, would be more successful in this respect. This is not pursued further in the present paper, since the purpose mainly has been to identify possible limitations in terms of wave steepness for the two methods.

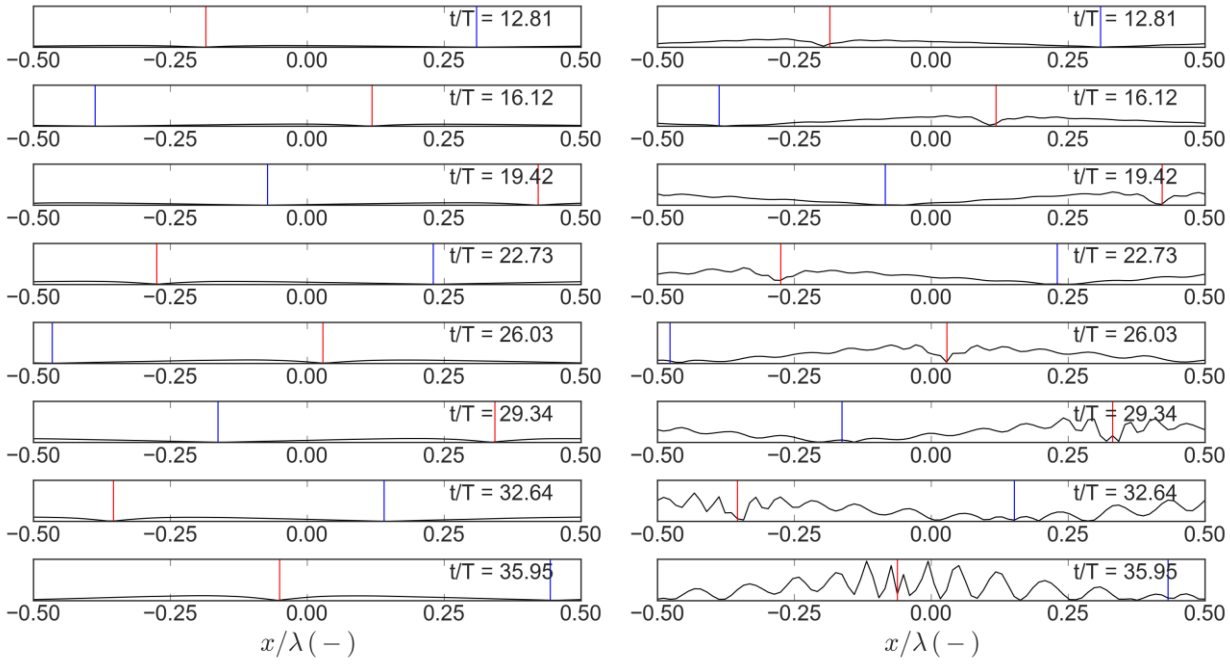


Figure 16 Error of the wave elevation in the IB method relative to the analytical wave elevation at eight different time instances. Left: $kA = 0.3$ and $n_{x,IB} = n_{z,IB} = 75$, right: $kA = 0.4$ and $n_{x,IB} = n_{z,IB} = 90$. All the figures share the same y-axis defined as $0.0 \leq \left|e_\eta\right| < 0.012$. At a given time instance, the blue and red vertical lines indicate the positions of the wave trough and crest, respectively.

4.1.1 CPU Time

Figure 17 shows the average CPU time per time step for all the cases in Table 2. In the left plot, where this variable is shown as a function of the number of grid points n_x in horizontal direction, it appears that the CPU time generally is lower for the IB method. This is somehow misleading, because the MG method gives a larger number of unknowns for a specific value of n_x due to the additional free-surface fitted grid. Thus, when the CPU time is plotted as a function of the number of unknowns in the matrix system, the methods give very comparable CPU time. It should be noted that a large part of the CPU time is spent building the global matrix system at each time step, and this part of the algorithms can probably be further optimized. Especially for the IB method, where the grid geometry remains constant throughout, further gains are likely to be achievable.

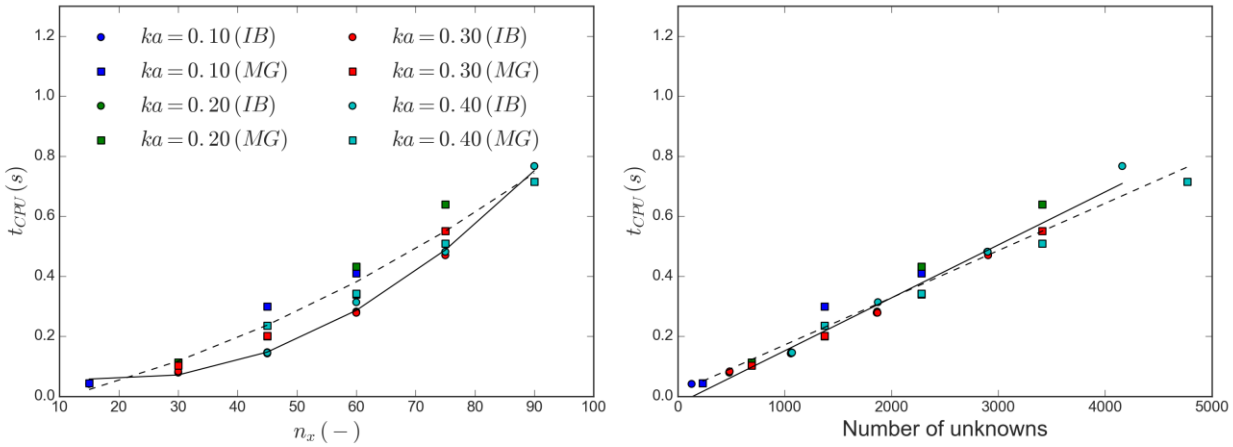


Figure 17 Average CPU time per time step for all cases in the numerical wave tank with periodic boundary conditions. The full black lines are fitted to the IB results, while the dashed black lines are fitted to the MG results. In the left plot, the CPU time is plotted against the number of nodes per wavelength in horizontal direction and the curve-fit is quadratic, whereas in the right plot, the CPU time is plotted against the average number of unknowns and the curve-fit is linear.

Figure 17 does not give full justice to the CPU efficiency achievable in the MG method, since the density of the background grid is set equal to the density of the corresponding grid in the IB method in the presented cases. One of the significant advantages of the MG approach is the possibility to achieve a considerable reduction of computational cost, both in term of time and memory usage, with a proper discretization of the physical domain without loss of accuracy. In our case, dealing with a free-surface potential flow problem at the limit of deep-water conditions, the higher gradients of the velocity potential are located close to the free surface. Hence, a finer discretization is required near the free surface compared to the bottom of the domain. In the test cases shown in Figure 11 - Figure 14 the ratio between the two grids can be considered fixed, i.e. the number of points used for the discretization of the background grid is equal as the number of Lagrangian marker used for the free-surface.

In the following, we use the case with $kA = 0.3$ as an example. Figure 18 shows a comparison of the temporal evolution of the L_2 norms for the free-surface elevation and velocity potential identified with continuous and dashed lines respectively. Two background grids are considered: The blue lines refer to the $n_x = 60$-case analyzed beforehand in Figure 13. Here the ratio between grid density of the background grid and the free-surface fitted grid, defined as the ratio between the number of points used for the discretization in horizontal/tangential direction, is equal to 1. The green line refers to a more optimized

configuration, characterized by the following parameter: $N_{x,MGb} = N_{z,MGb} = 20$ and $(N_{x,MGf}, N_{z,MGf}) = (60,12)$, giving a grid point ratio of 1:3. $(N_{x,MGb}, N_{z,MGb})$ are the number of grid points in horizontal and vertical directions in the background grid, while $(N_{x,MGf}, N_{z,MGf})$ are the number of grid points in tangential and normal directions in the free-surface fitted grid. An indicative number of unknowns, at each time step, for the two configurations is $N = 2280$ and $N_{opt} = 920$.

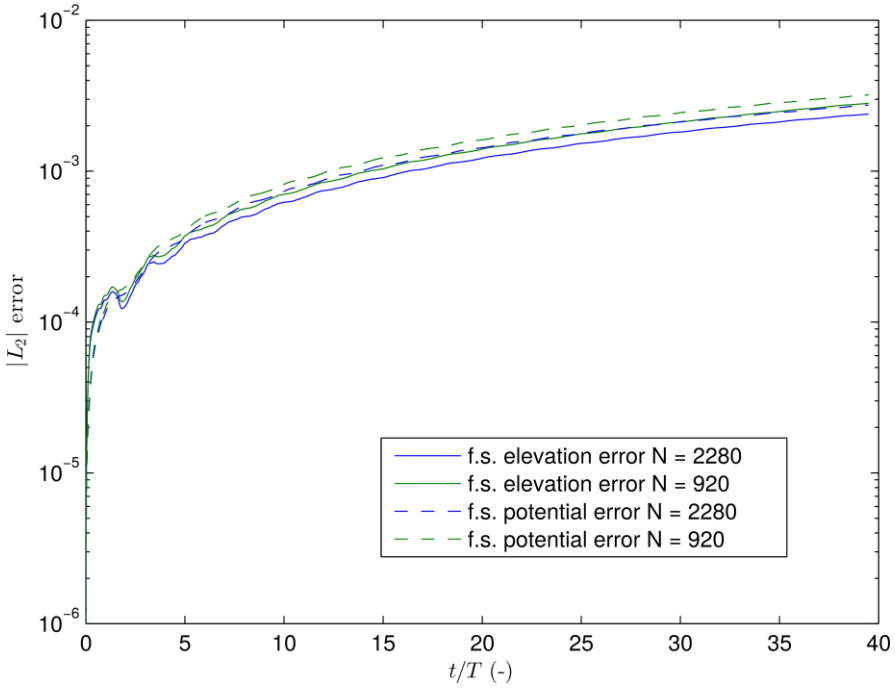


Figure 18 Comparison of the L_2 norm for the free-surface elevation (continuous line) and velocity potential (dashed line) for the MG method. The blue curve refers to the simulation with the original discretization, while the green refers to the optimized configuration.

Figure 19 shows the temporal evolution of the error of the mass, kinetic energy and potential energy for the two configurations with respect to the analytical values. With a time step of $\Delta t = 0.01s$, both grid configurations result in a similar level of accuracy for all the plotted quantities. However, the optimized configuration ($N = 920$) performs approximately 2.36 times faster than the original one and requires 59% less memory. These results suggest that the MG approach can be very useful for problems with complex geometry where a local refinement of the computational grid is required and when the largest flow gradients are localized in vicinity of the free surface. In such cases, the MG method can perform significantly faster than the IB method.

The red curve plotted together with the mass error is proportional to Δct, where Δc is the shift in phase velocity computed from the spatial distance between the zero-crossing of the numerical and analytical solution. This indicates that the increasing trend of the error in mass (and potential energy) mainly is due to an error in the phase velocity, which causes a linear growth with time. Such error in phase velocity is usually caused by the so-called dispersion error.

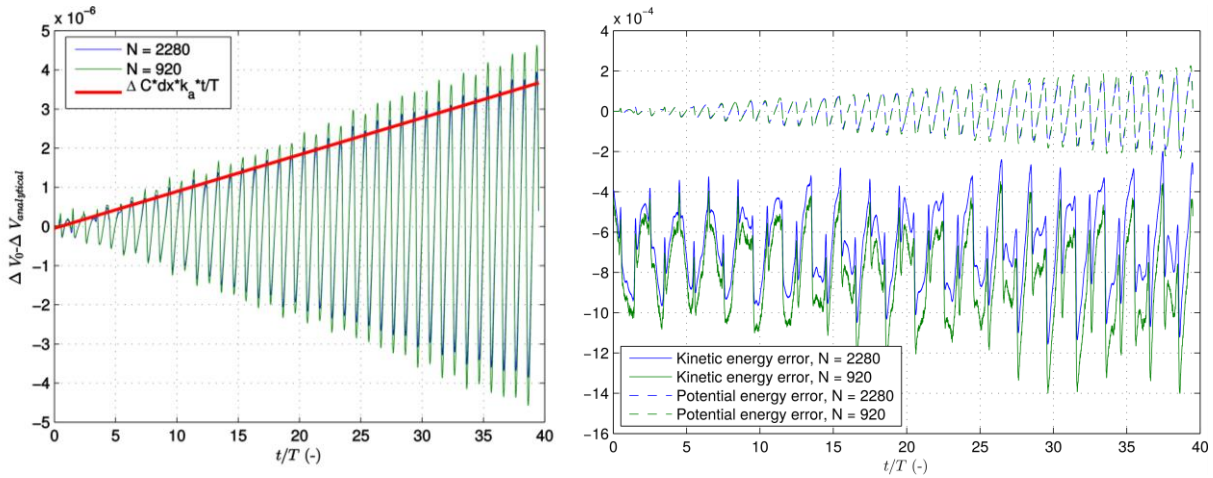


Figure 19 Comparison of the error between the 1:1 grid points rate (blue line) and the optimized one (1:3) (green line) for the MG method. Left: Conservation of fluid volume. The red curve represents the estimated phase velocity error for the $N = 2280$ case, where Δc is the error in phase velocity, dx is the horizontal grid spacing and k_a is the wavenumber. Right: Kinetic (continuous line) and potential (dashed line) energy.

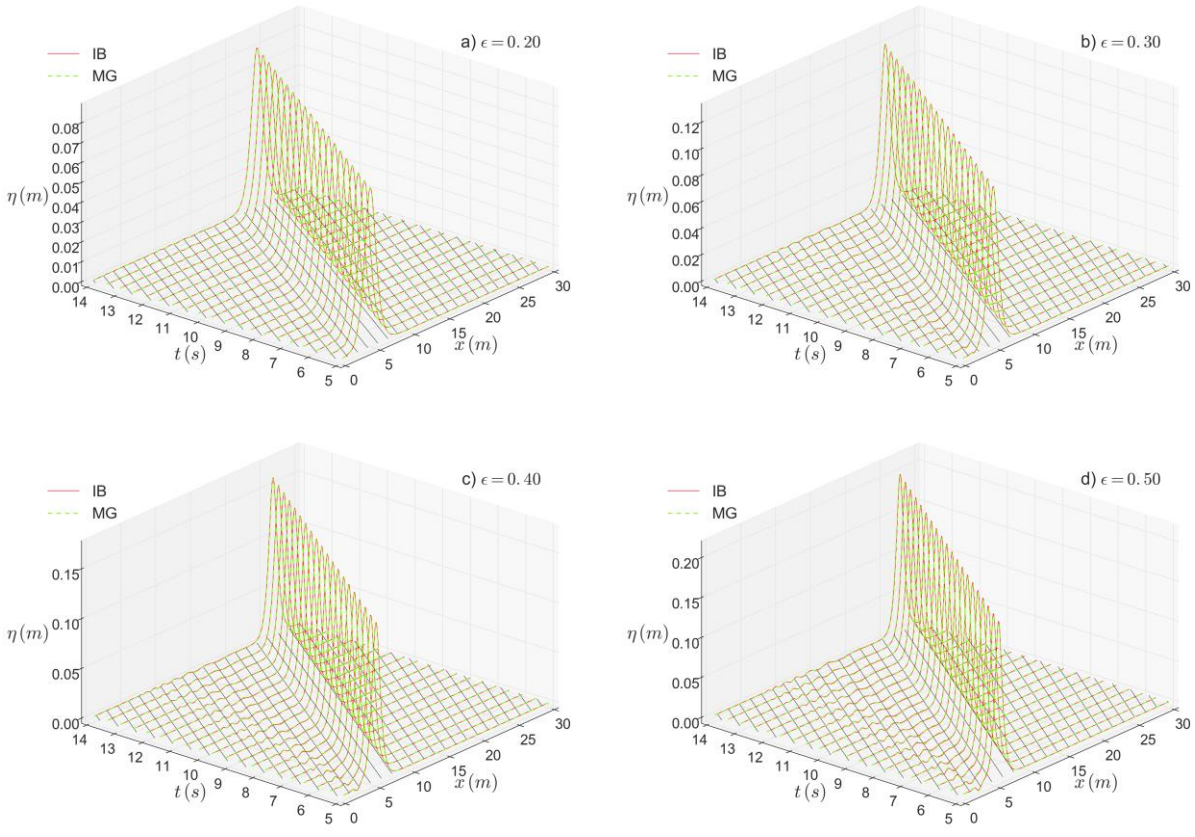
4.2 Case 2: Solitary Wave in Shallow Water

For the solitary wave cases in Table 3, interesting quantities include the temporal evolution of the maximum wave elevation, the preservation of the wave shape and phase speed, and the amount of trailing waves behind the propagating wave. Figure 20 shows time-space plots for the wave elevation from the numerical simulations for all values of $\epsilon = A/h$ as well as maximum elevation in the tank as a function of time. The time-space plots indicate a visually satisfactory agreement between the free-surface elevation from the simulations using the IB and MG methods, both in terms of wave shape and phase speed. Also the shorter trailing waves generated behind the solitary wave, generally more profound for higher values of ϵ, show good agreement between the methods. The diagonal black lines in the $x - t$ plane represent iso-lines with constant phase according to the analytical phase speed, i.e. $x = ct$. Carefully comparing the surface elevation with these indicates that the phase speed in the numerical simulations differs slightly from the analytical one for the two highest values of ϵ. Figure 21 confirms this by comparing snapshots of the numerical wave elevation with the analytical solution for all ϵ. All snapshots correspond to the time instant when the analytical wave crest is at $20m$ from the wavemaker. For $\epsilon \geq 0.40$, a visible phase lag is seen in the numerical solution, i.e. the numerical phase speed is lower than the analytical one. The lower right figure shows the maximum wave elevation in the tank normalized by the water depth h as a function of time. For ϵ larger than 0.40 it is clearly seen that the maximum amplitude is too low in the numerical simulations compared to the specified value. A similar divergence with respect to the analytical solution was also observed by Wu et al. [32] for ϵ above 0.40.

The deviation between the numerical results and the analytical solution, and the fact that it increases as ϵ increases, is likely to be related to the way the waves are generated in the simulations. The wavemaker signal produced by equation (28) assumes that the horizontal water particle velocity is uniform from the wave crest to the seabed under the solitary wave. This assumption is reasonable for lower ϵ, but is violated for higher ϵ. As ϵ increases, higher-order terms in the solution becomes more important. Equation (28) neglects such higher-order terms, which means that the initial condition for the wave contains some higher-order error terms compared to the analytical solution. These will lead to dispersion errors that

cause both the observed trailing waves as well as the difference in phase velocity. This tendency is supported by the findings of Zhou et al. [36]. With a third-order accurate initial condition for the solitary wave in their BEM simulation, they found a good agreement with an analytical solution for $\epsilon = 0.2$ but not for $\epsilon = 0.5$. The nature of this error was similar as in the present paper, i.e. manifested through generation of trailing waves and a phase-velocity difference. By instead using the higher-order solution by Clamond and Dutykh [37] as initial condition, they found numerical results that were in good agreement with the analytical solution also for $\epsilon = 0.5$. This strongly indicates that the reason for the difference between the numerical and analytical solution for $\epsilon > 0.4$ in the present paper is indeed related to the wave-generation mechanism rather than to the numerical methods themselves. By accounting for higher-order terms in the wavemaker signal, or alternatively by generating the wave as an initial-value problem, we thus expect an improved agreement with the analytical solution also for higher values of ϵ. These alternatives are not further pursued in the present work.

For $\epsilon > 0.4$, the tendency is that the maximum amplitude is slightly lower with the MG method than in the IB method, and the trailing waves have a slightly higher amplitude. This indicates that the MG method introduces a slightly higher dispersion error for large values of ϵ. It is possible that this is due to the grid configuration used in the particular simulations: From Table 3 it is clear that the free-surface fitted grid used in the MG method contains significantly stretched cells towards the free surface.



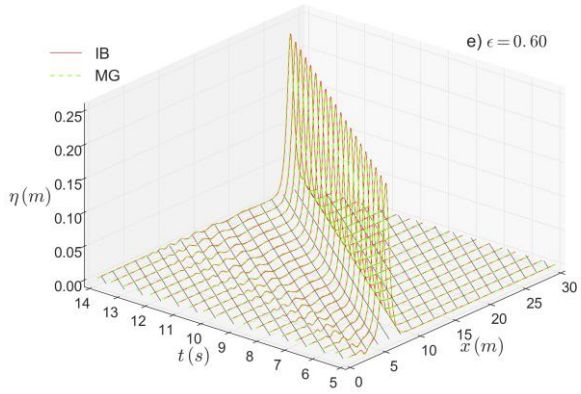
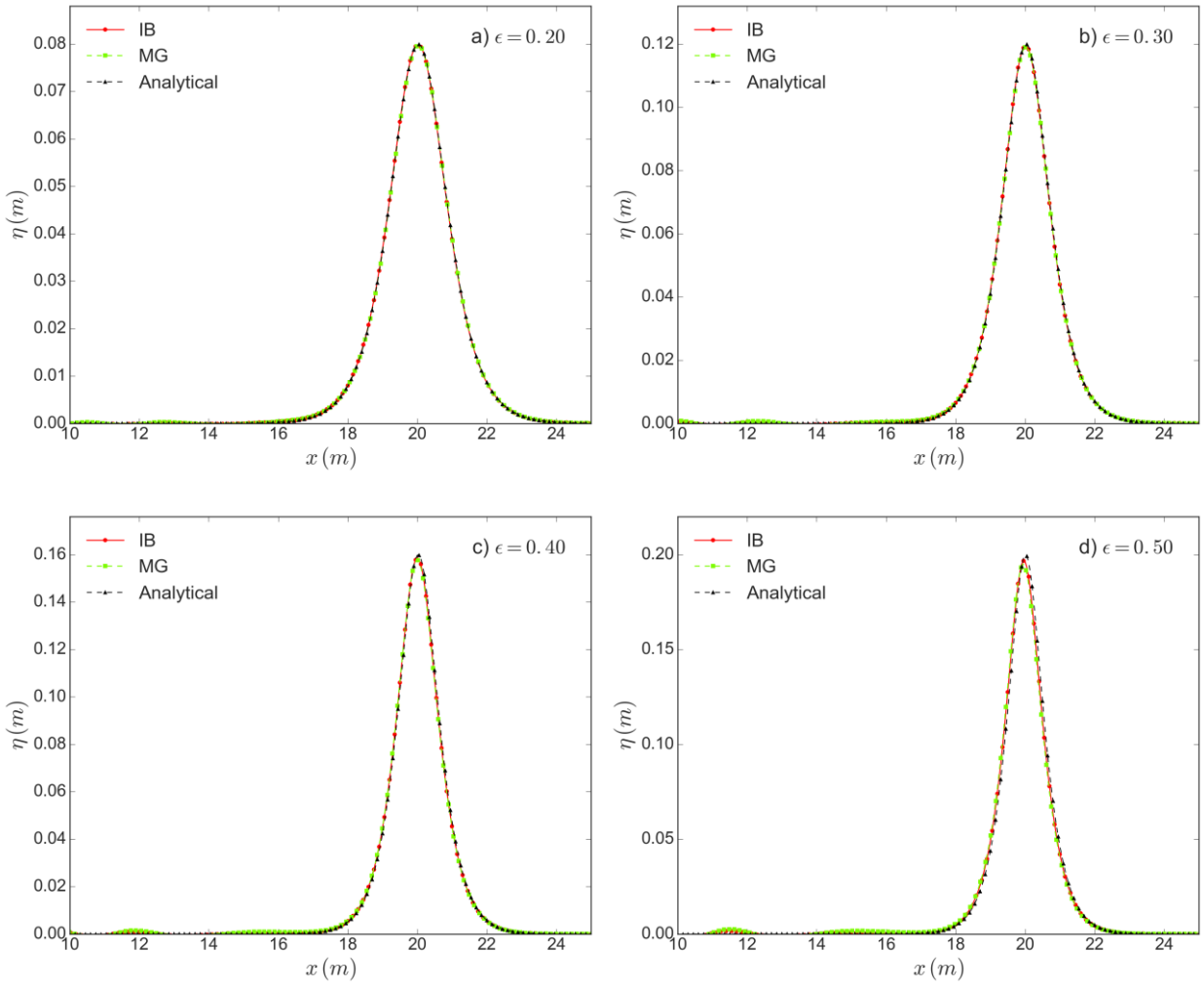


Figure 20 Space-time plots of the surface elevation from the numerical simulations with the IB (-) and MG (--) free-surface treatments. The black diagonal lines in the $x - t$ plane corresponds to lines with constant phase with the analytical phase speed, i.e. $x = ct$.



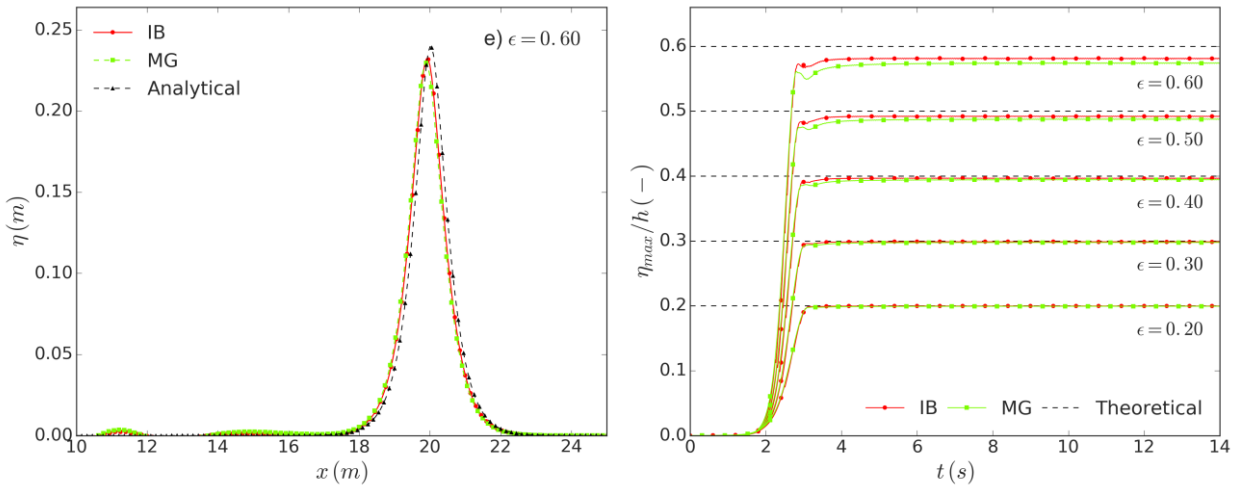


Figure 21 The first five figures show snapshots of the numerical wave elevations compared with the analytical solution for different values of ϵ at the time instant when the analytical position of the crest is $20m$ from the wavemaker. The lower right figure shows the maximum numerical wave elevation in the tank as a function of time for the different values of ϵ.

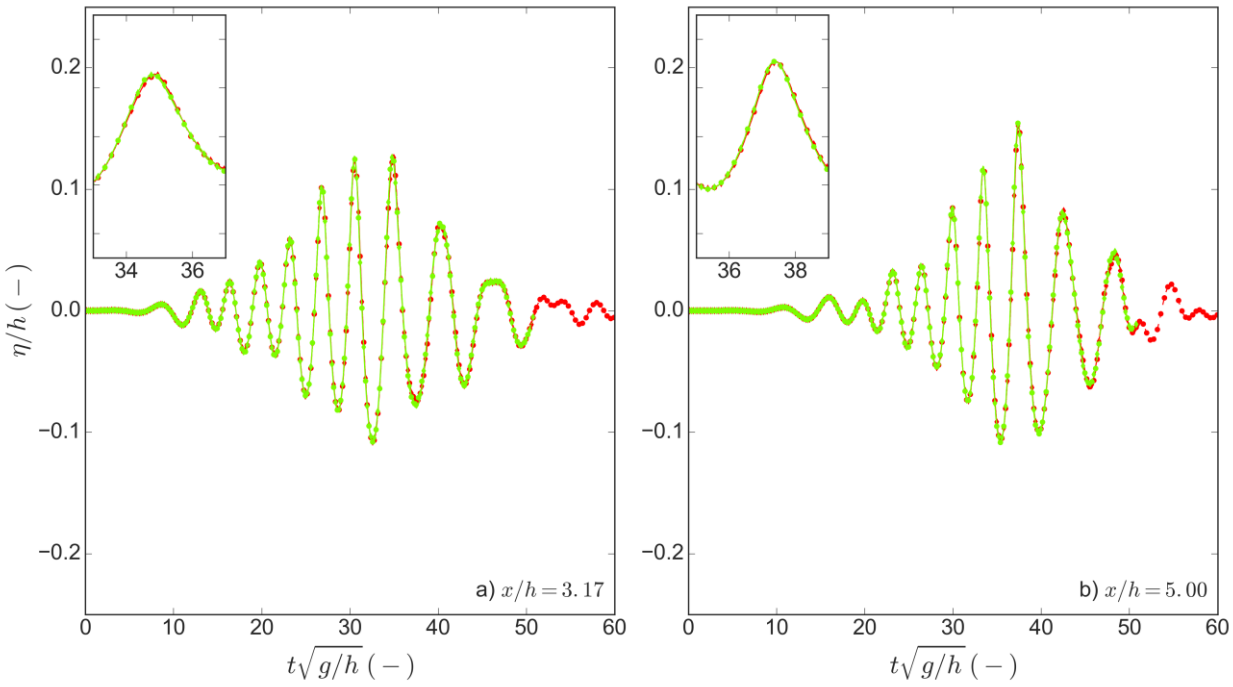
4.3 Case 3: Focused Wave Packet

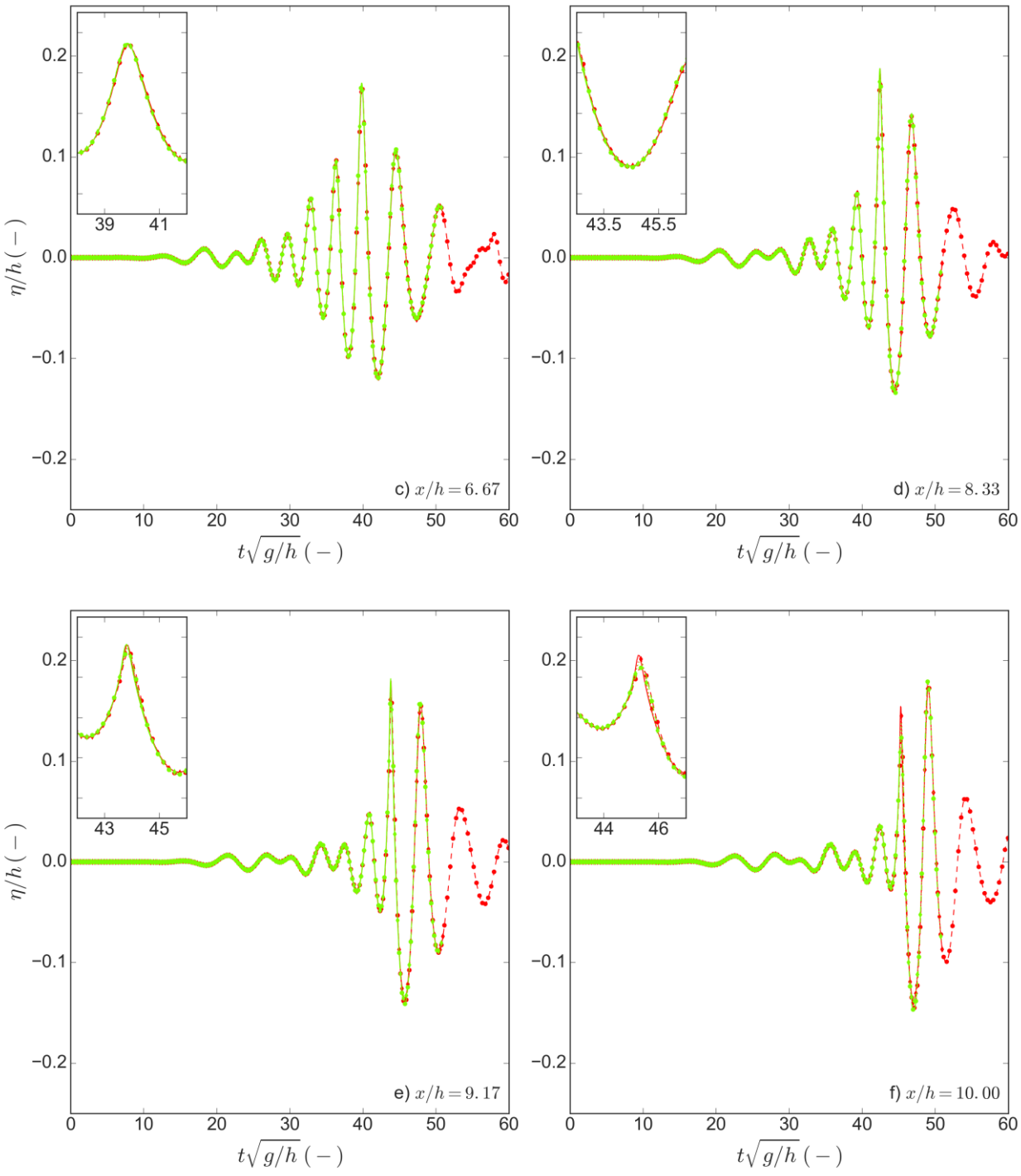
To assess the numerical results for the focused wave packet defined by Dommermuth et al. [33], we concentrate on the wave elevation in the previously defined wave probes along the tank. Figure 22 compares the simulated wave elevation for the IB and MG methods with each other for the grid-spacing cases defined in Table 4. Figure 23 compares the IB and MG results with the "fine" grid in Table 4 with the experimental results from Dommermuth et al. [33] as well as with the BEM results from Lugni [35]. For the IB method, the results with the breaking-suppression scheme described in section 2.2.1 are also included in this plot. A few things stand out:

- Up to the point where breaking occurs (approximately after $t\sqrt{g/h} \approx 52$ at $x/h \approx 12$), the IB and MG results are in good agreement with each other and show little difference between the different grid densities. The results are generally in good agreement with the experimental and BEM results.
- All MG and IB simulations without breaking suppression, except for the coarse-grid IB case, break down around the time when breaking occurred in the experiments. Up until this point, the results show little sensitivity towards the grid density.
- It can be speculated that the reason why the IB simulation with coarsest grid, even without a breaking suppression scheme, survives is that the coarse grid filters away some of the instability that leads up to breaking.
- The breaking suppression scheme used together with the IB free-surface treatment seems to perform well, in the sense that it enables the simulations to survive, and the results remain in good agreement with the experiments after the time of breaking. A dedicated discussion about the breaking scheme's performance follows later.

It is emphasized that we expect the MG method to perform equally well as the IB method with a breaking-suppression scheme. I.e. the observations made in this section with respect to suppression of wave breaking does not imply that the IB method is better than the MG method to preserve numerical robustness. The main point to be made is that the suppression scheme works according to its intention in the selected case.

Figure 24 offers an enlarged time-series plot for the medium-grid case at a wave probe just before the focusing point. It is clear that the IB and MG results are in good agreement with each other, even though some difference in amplitude appears close to $t\sqrt{g/h} \approx 47$. The HPC results are in fair agreement with the BEM results, and deemed within the error margin of the experimental results. The breaking suppression scheme reduces the wave amplitude slightly in the IB method also before the time of wave breaking. This is due to a "secondary" focusing point appearing closer to the wavemaker at an earlier time, which generated a wave crest that was steep enough to exceed the $B_x(x)$ criterion in equation (14).




0.2
0.1
0.0
−0.1
−0.2
$\eta/h\,(-)$
0
10
20
30
40
50
60
$t\sqrt{g/h}\,(-)$
39
41
c) $x/h = 6.67$
43.5
45.5
d) $x/h = 8.33$
43
45
e) $x/h = 9.17$
44
46
f) $x/h = 10.00$

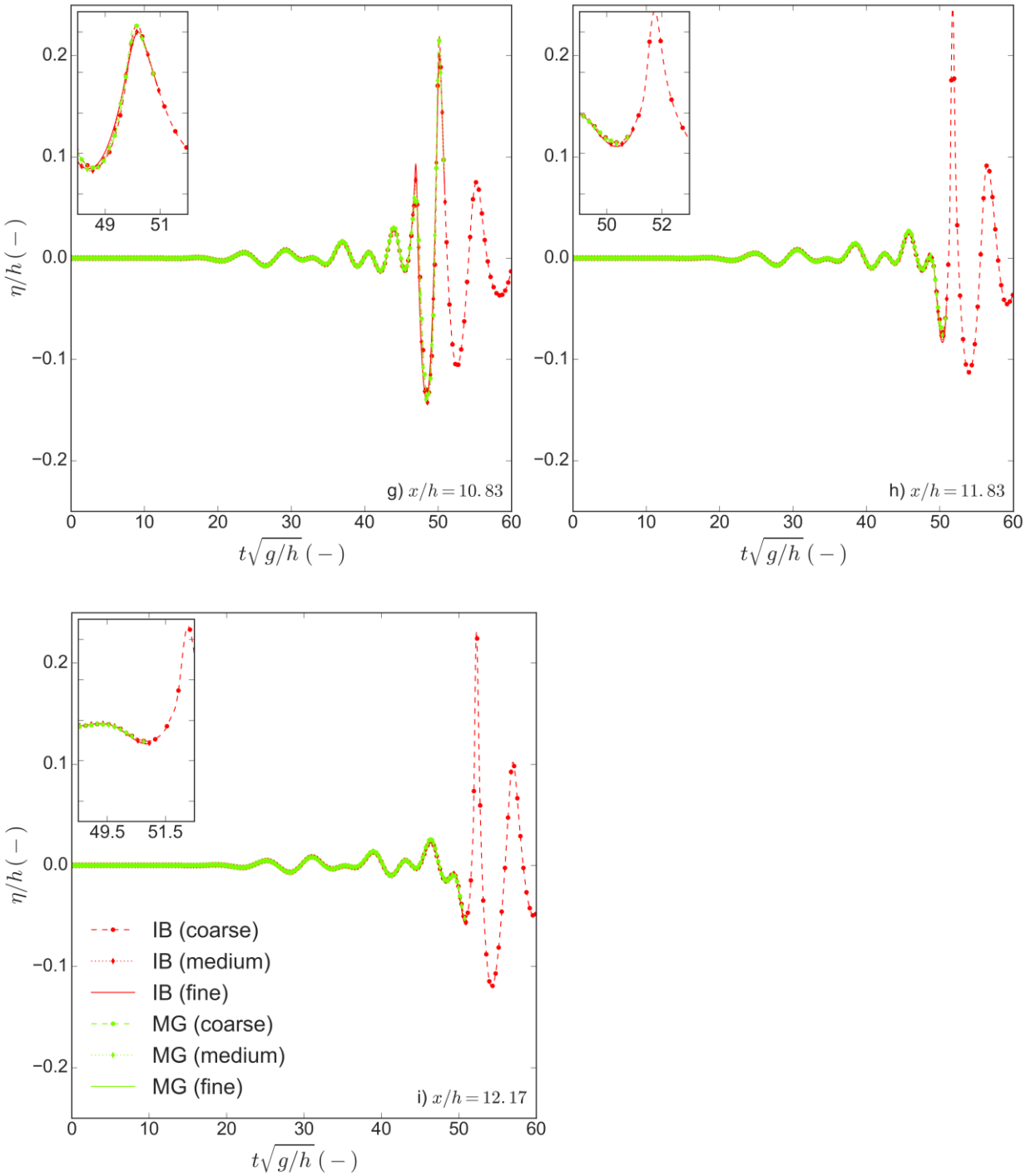
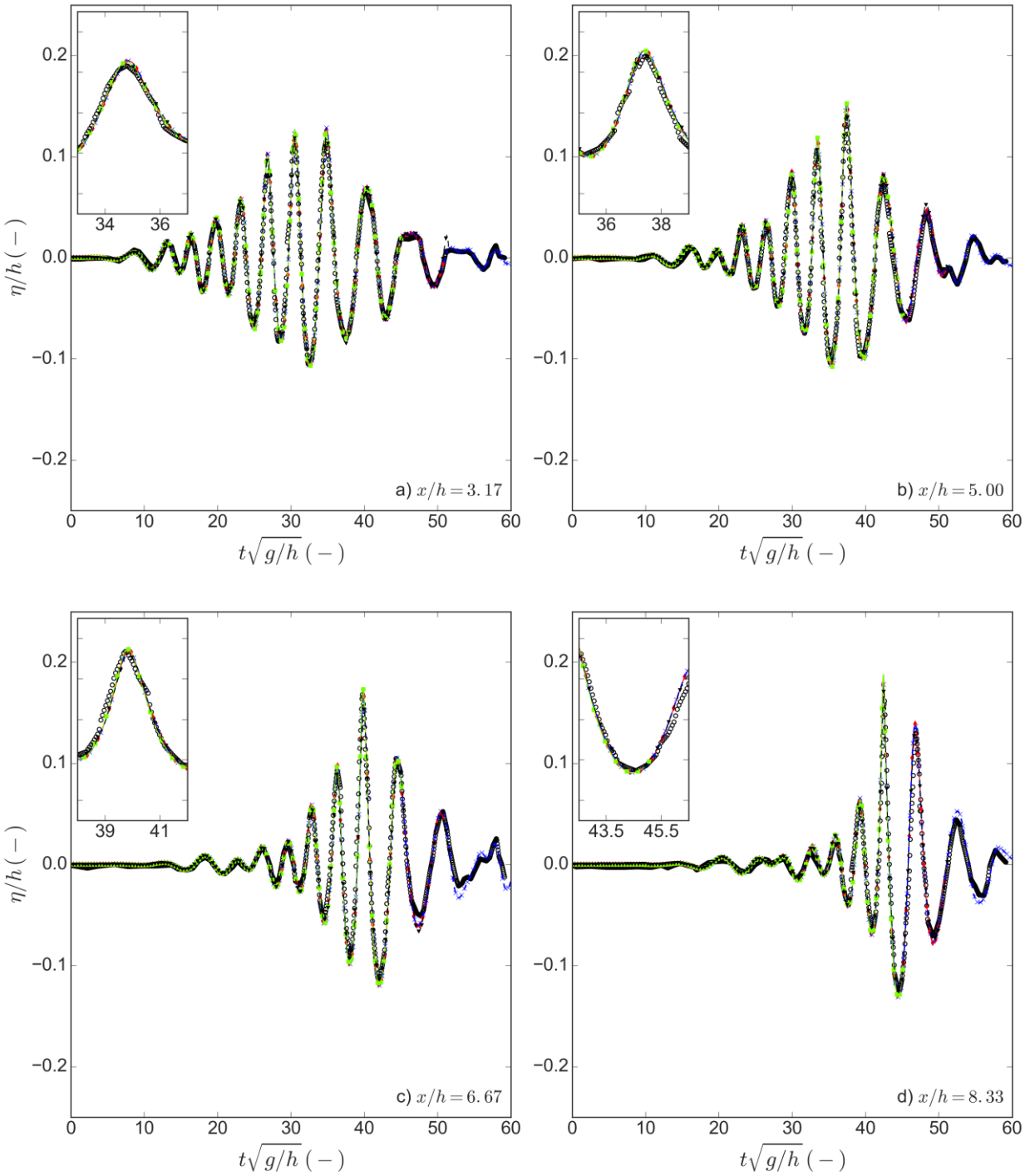
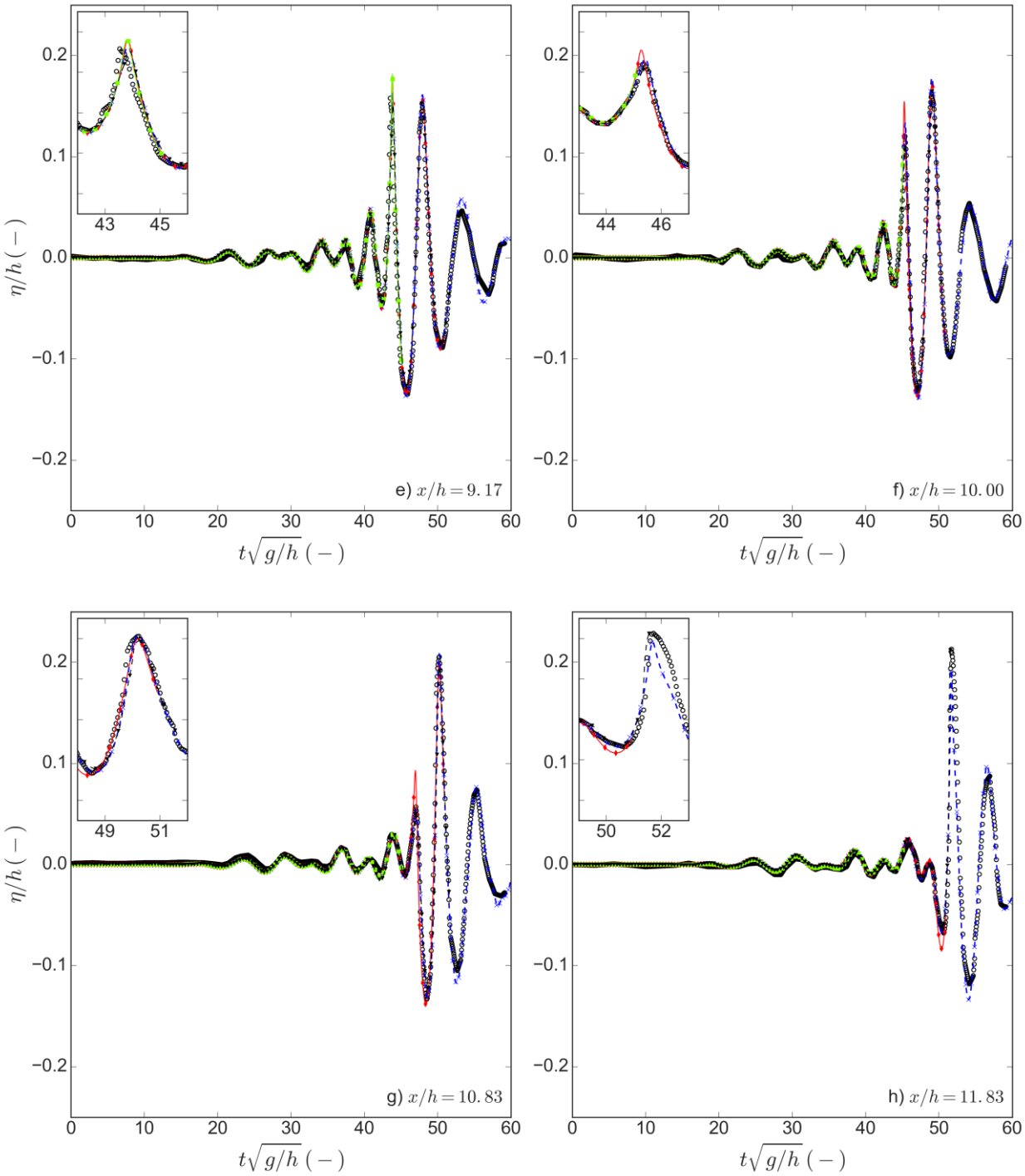


Figure 22 Comparison of wave-elevation time series at nine wave probes along the length of the tank for the focused wave packet with the "coarse", "medium" and "fine" discretization from Table 4. IB refers to the HPC results with the IB approach while MG refers to the HPC results with the MG approach. The upper left plot inside each plot shows the most significant peak in each time series.


a) $x/h = 3.17$
b) $x/h = 5.00$
c) $x/h = 6.67$
d) $x/h = 8.33$
$\eta/h\,(-)$
$t\sqrt{g/h}\,(-)$


e) $x/h = 9.17$
f) $x/h = 10.00$
g) $x/h = 10.83$
h) $x/h = 11.83$
$\eta/h\,(-)$
$t\sqrt{g/h}\,(-)$

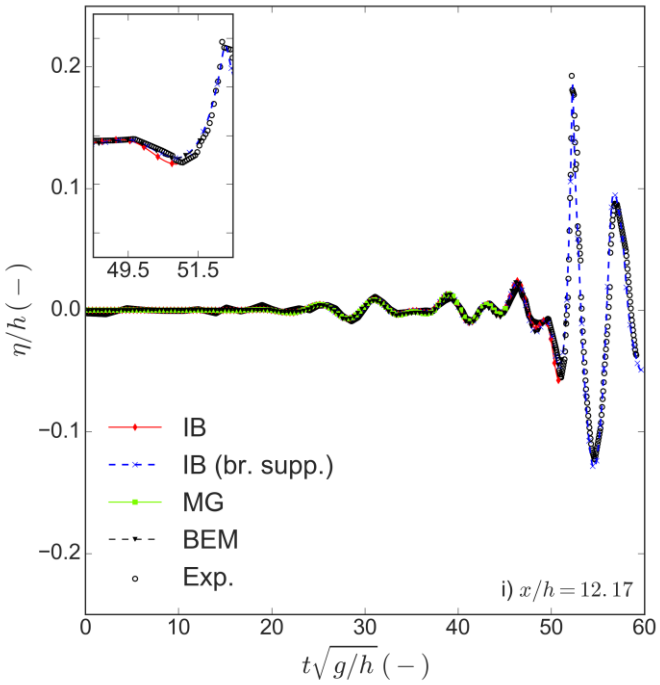


Figure 23 Comparison of wave-elevation time series at nine wave probes along the length of the tank for the focused wave packet with the "fine" discretization from Table 4. IB refers to the HPC results with the IB approach, IB (br. supp.) refers to the IB results with breaking suppression, MG refers to the HPC results with the MG approach, BEM refers to the numerical results from Lugni [35] and Exp. refers to the experimental results from Dommermuth et al. [33]. The upper left plot inside each plot shows the most significant peak in each time series.

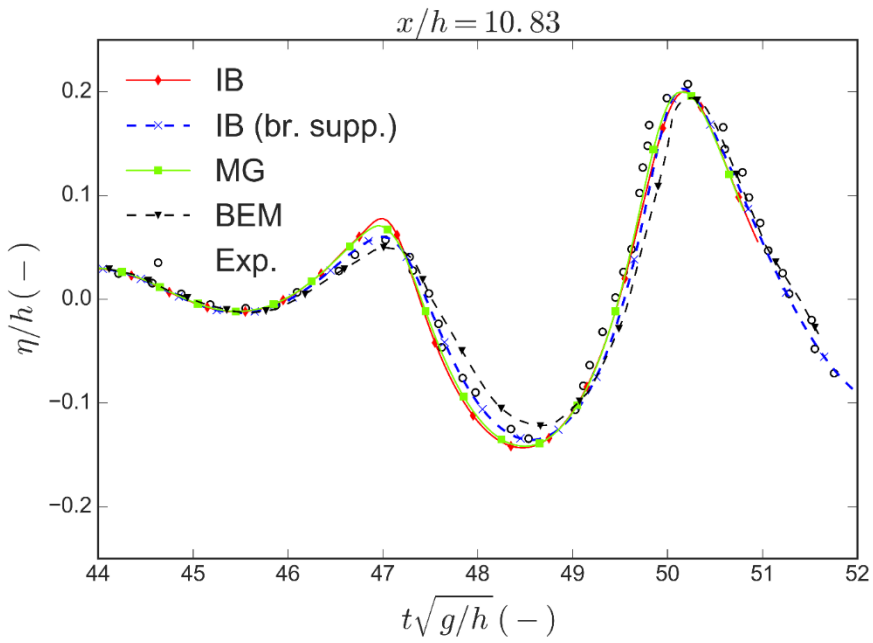


Figure 24 Zoomed plot for the time series of wave probe at $x/h = 10.83$. The IB and MG results are with the "medium" discretization from Table 4.

4.3.1 Suppression of Wave Breaking

The capability of the wave-breaking suppression scheme, that identifies the onset of wave breaking and adds a dissipation term locally in the dynamic free-surface condition, is demonstrated in Figure 25 by using the medium-grid case from Table 4 as an example. The left column shows the time series of the wave elevation close to the breaking point, the potential energy and the kinetic-energy estimate from equations (33) and (34) with and without the suppression scheme. In the right column, the upper plot shows snapshots of the wave elevation at different time instants up to the point where the simulation without suppression scheme breaks down. The remaining plots show the relative horizontal particle velocities from the simulation with breaking suppression and the corresponding dissipation term μ_b at the same time instances. When $B_x = \varphi_x / c_0 > 0.86$, a smooth local distribution of free-surface dissipation μ_b emerges over adjacent markers.

As was already mentioned in section 3, the studied case represents a plunging breaker in the experiments, which is a much more energetic phenomenon than a spilling breaker. Thus, we cannot expect the breaking-suppression scheme to model this phenomenon fully in the present framework, which applies a semi-Lagrangian formulation for the free-surface markers. A more accurate modelling of the overturning, plunging breaker would require a fully Lagrangian approach. Moreover, close to the impact of the plunging wave with the underlying liquid, more general modelling is needed. This is because the air flow below the breaker can affect the water evolution and, after the impact, the water flow becomes rotational. However, as explained in section 3, the goal here is not to provide a fully accurate modelling of the breaking phenomenon but rather to ensure numerical stability when breaking can occur locally. The motivation behind using the challenging plunging-breaker scenario to test the scheme is to see how it performs when tested to the limit, i.e. how much artificial dissipation it introduces. This will give some information regarding how robust the scheme functions, keeping in mind that we in reality intend to use it in scenarios where less violent type of wave breaking is expected.

The first thing noticeable from Figure 25, is that the suppression scheme enables the simulation to continue beyond the point of breaking. We also see that the potential energy time series remains unchanged, while there is a small reduction in kinetic energy close to the time of breaking. It should be noted that the first time the suppression scheme is invoked briefly in this simulation is after $t\sqrt{g/h} \approx 40$, which explains why the kinetic-energy estimate differs slightly with and without dissipation suppression after this time. The upper left plot shows that the comparison between the experimental and numerical wave-elevation time series remains good also beyond the time of breaking. The snapshots of wave elevation at different time instances highlight that the suppression scheme makes the wave surface maintain a steep yet relatively smooth profile up to the time where the original simulation terminates.

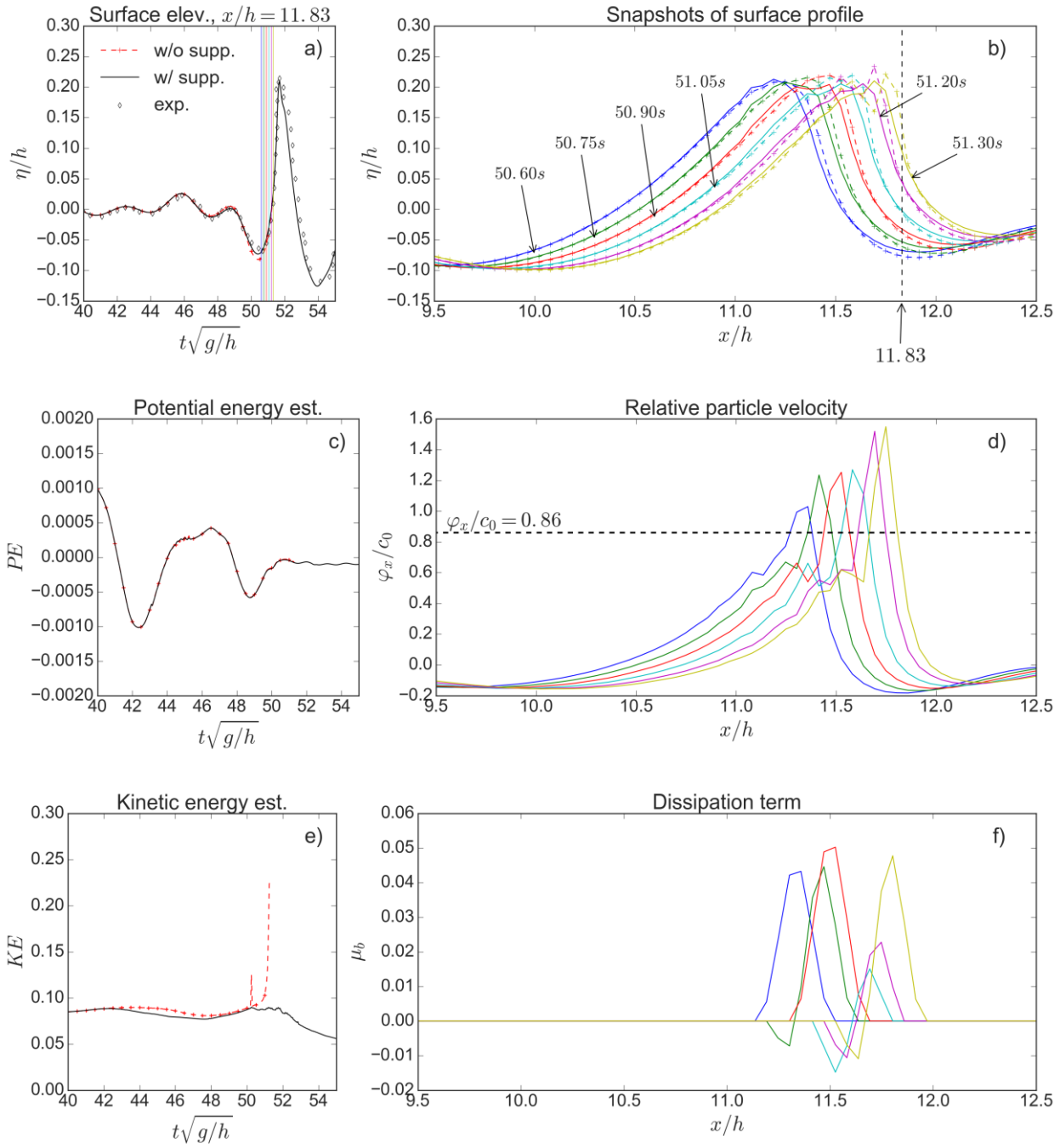


Figure 25 Influence of the breaking-suppression scheme. a) Time series of non-dimensional wave elevation in wave probes close to the focusing point. b) Snapshot of wave elevation at different time instances with (full lines) and without breaking suppression (dotted lines with crosses). c) Time series of potential energy. d) Ratio between fluid particle velocity in x-direction and local phase velocity from simulation with breaking suppression for the different time instances. e) Time series of kinetic-energy estimate. f) Dissipation term applied in dynamic free-surface condition for different time instances.

The present study shows that the implemented scheme to prevent wave breaking indeed has the potential to provide stability in numerical simulations with moderate breaking, since it even shows acceptable

performance for a plunging breaker. In cases where the details of overturning waves are important to capture for a particular problem, it is anyway necessary to apply a fully Lagrangian approach. In such scenarios, it may even be necessary to replace the potential-flow model with a Navier-Stokes solver to account for rotational-flow effects after the wave hits the free surface below, as well as development of viscous effects and turbulence in a proper manner.

Even though Figure 25 shows that the implemented scheme successfully identifies and suppresses wave breaking in the simulation and thereby makes it more robust, a legitimate criticism of the method is the empirical nature of the parameters going into the dissipation term in equation (10). Presently, these come from a tuning process so to avoid numerical instability, and the obtained parameters may be inadequate for other cases. A method for estimating these parameters directly from the physical problem would therefore be desirable, enabling the parameters to be defined dynamically during the simulation. Such an approach would also make the routine more automatic by eliminating the need for manual input to the routine, and might deserve attention in further studies deploying the scheme.

4.4 Case 4: Periodic Waves in Shallow Water

The nonlinear feature of particular interest for the shallow-water case with periodic wavemaker motion is the transfer of energy between harmonic modes due to shallow-water effects. To study this we do a Fourier analysis of the surface-elevation time series in numerical wave probes distributed along the first $25m$ of the wave tank and compare with corresponding experimental results from Chapalain et al. [34]. We also compare with numerical HPC results from Shao and Faltinsen [9]. In their case, a (global) free-surface fitted grid is adopted leading to vertical stretching of the cells due to surface deformation.

Figure 26 shows the harmonic amplitudes $A_n, n = 1, \dots, 4$ as a function of distance from the wavemaker for both case A and C as defined in Table 5. For case A, the IB and MG results are in close agreement with each other, and except for an apparent phase shift for the second harmonic, are also in good agreement with the experiments. For case C, the IB and MG results are still in good agreement with each other. However, the comparison with experimental results show some larger deviations. This can possibly be related to uncertainties concerning viscous dissipation in the experiments [34]. Shao and Faltinsen added an artificial dissipation term in the free-surface condition to account for this, which explains why the difference with respect to their results increases away from the wavemaker. There exist some sources of uncertainty for the experimental results, in particular related to the limited dimension of the wave tank used. This may lead to viscous effects from the sidewalls, and severely restricts the length of the measured time series (no beach was installed). No details exist regarding the ramp function used in the experiments. These aspects may have influenced the Fourier analysis in [34], since at least one complete wave cycle without influence from transient effects or wave reflections is required to give accurate harmonic amplitudes.

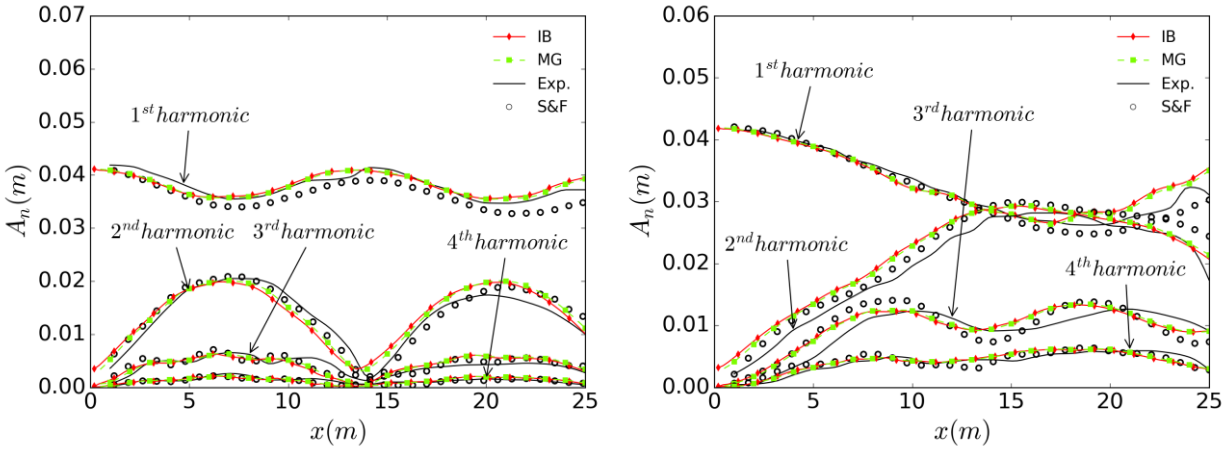


Figure 26 Amplitudes of first four harmonic wave components along the wave tank for case A (left) and case C (right). x is the distance from the wavemaker. Exp. refers to the experimental results from [34], S&F refers to the HPC results from [9], IB refers to the HPC results with the IB approach and MG refers to the HPC results with the MG approach.

4.5 Case 5: Periodic Waves in Deep Water

Figure 27 shows a comparison of the wave-elevation time series in a probe $12m$ downstream of the wavemaker for the deep-water periodic-wave case studied numerically and experimentally by Lugni [35]. The numerical simulations and the experiments share the same flap signal to describe the wavemaker motion. As a result, the numerical results are in good agreement with the experimental ones, including the amplified wave-to-trough amplitude of the transient leading waves in the wave train. After steady-state oscillations are established, the numerical simulations remain in good agreement with the experiments for all values of kA. No decrease in wave amplitude due to numerical dissipation or phase difference due to dispersion error are visible. The IB and MG results remain in close agreement throughout for all values of the wave steepness. Towards the end of the simulation, it appears that the present results are in slightly better agreement with the experiments than the BEM results.

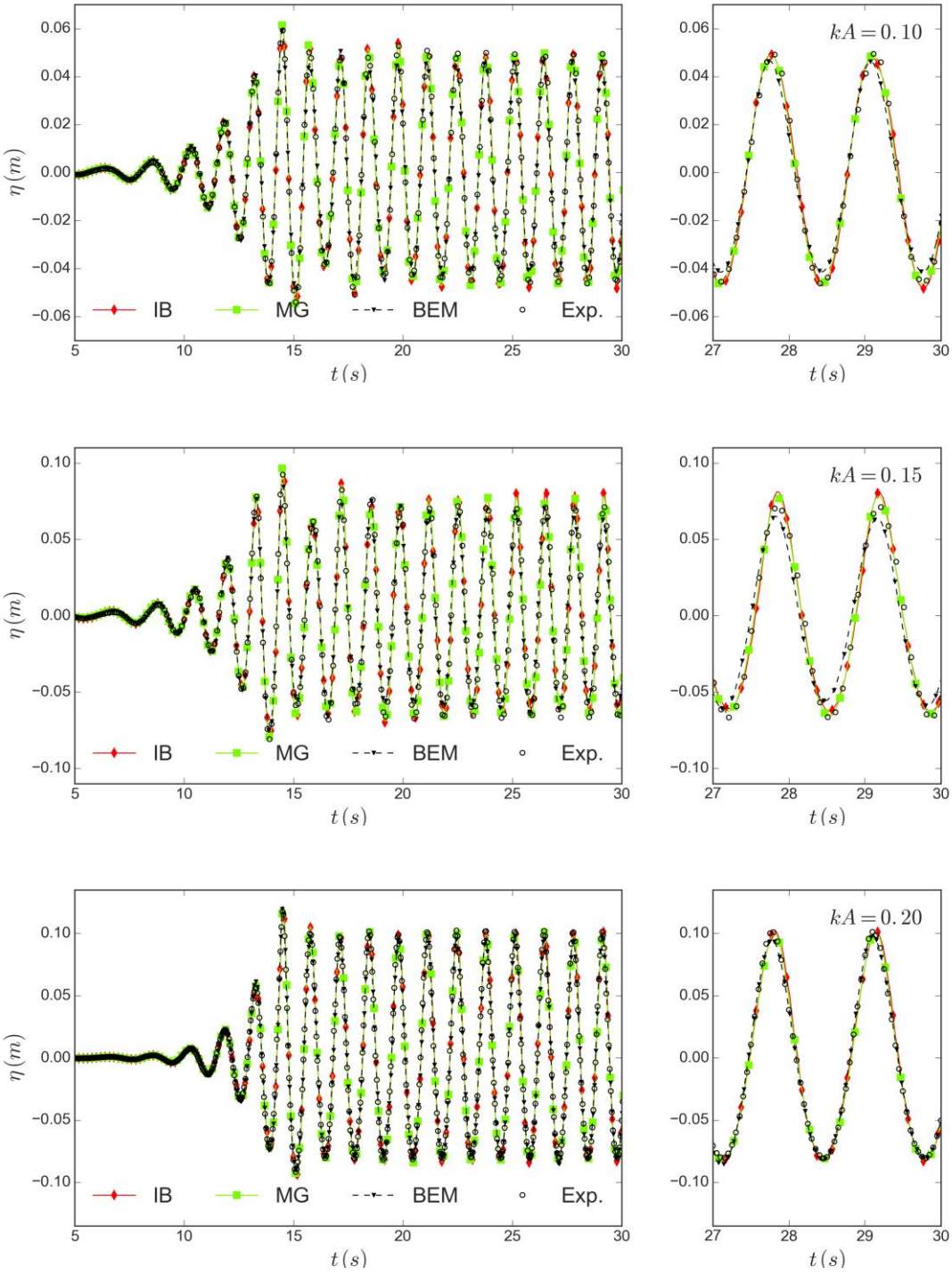


Figure 27 Comparison of wave-elevation time series at wave probe $12m$ from the wavemaker for the deep-water periodic wave described in Table 6. IB refers to the HPC results with the IB approach, MG refers to the HPC results with the MG approach, while BEM and Exp. refer to the numerical and experimental results from Lugni [35].

4.6 Evaluation of the Two Methods

The main motivation for using both the IB and MG method throughout this paper is to determine their relative differences, strengths and weaknesses. Such information can be helpful in selecting free-surface modelling for a particular problem. A summary of the main findings in this respect is given in Table 7.

Table 7 Properties of the present implementations of the MG and IB method with respect to different aspects.

	Immersed Boundary (IB)	Multi-grid (MG)
Grid generation	• Structured squared grids, easy to implement.	• Free-surface grid must be updated each time step. • Special care required close to the boundaries of the physical domain.
Free-surface tracking	• Free surface condition generally imposed in the interior of a cell, which can be beneficial for accuracy of velocity potential.	• Seems to be more robust than IB method in handling very steep waves without resulting in instability.
	A fully-Lagrangian formulation to account for overturning waves is possible to implement in both methods. However, the present study indicates that the MG method is more suited in this case because it can more easily refine the grid locally when the free surface has a large curvature.	
CPU time	• Can be made efficient because the grid geometry remains fixed in time. • Requires (partial) re-building of the global coefficient matrix at every time step because the actual computational domain may change with time depending on the position of the free surface and the wavemaker.	• Can in many cases be made equally or more efficient than the IB method by applying a coarser background grid without loss of accuracy, as was demonstrated in Figure 18 and Figure 19.

5 Conclusions

This paper has presented a nonlinear potential-flow numerical wave tank based on the harmonic polynomial cell (HPC) method. The free surface is dealt with by two different methods: One is an immersed boundary (IB) method, where the free surface is considered submerged in a fixed background grid and tracked by following marker particles in a semi-Lagrangian way. The other is a multi-grid (MG) method, where a structured free-surface fitted grid deforms with the free surface and overlaps with a structured background grid. The version of the MG method in this paper uses semi-Lagrangian markers track the free surface.

Long-time propagation of waves with increasing steepness in a periodic wave tank served as the first case study. With a few exceptions, the achieved accuracy in the IB and MG methods are in good agreement when compared with the analytical solution given by Rienecker and Fenton. Both methods were able to simulate a close-to-breaking wave over 40 wave periods. However, for the finest grids and the highest wave steepness, the IB method appears more susceptible to developing sawtooth-like instabilities. As the grid size increases, the IB method requires slightly lower CPU time than the MG method for the defined

cases. However, it was demonstrated that the MG method could be made more than 2.5 times faster by using a coarser background grid without any loss of accuracy. Thus, the MG method can perform faster than the IB method to reach a certain level of accuracy, in particular for cases where the largest flow gradients are concentrated near the free surface. The majority of the CPU time in the IB method is spent building the global coefficient matrix at each time step, and the efficiency of this operation can be improved compared to the present implementation.

Subsequently, problems with various distinct nonlinear wave features were considered. These include the propagation of a solitary wave over a long distance, a plunging wave packet, periodic waves generated by a piston wavemaker in shallow water and periodic waves generated by a flap wavemaker in deep water. Generally, the results by the IB and MG methods are in close agreement with each other, and in good agreement with the reference results for each of the studied cases. The only case with some small differences between the IB and MG results was the plunging wave packet. Here, waves with very high steepness emerge locally in the tank, which is challenging to capture numerically. At the time of plunging in the experiments, the numerical simulations fail due to the semi-Lagrangian free-surface description, which means that they cannot model overturning waves. For the IB method, a scheme to suppress local wave breaking was implemented in order to enable the simulation to continue beyond the point of breaking. With the scheme activated, the simulation was able to continue beyond the point of plunging, seemingly with results still in good agreement with the experiments. This shows that such scheme can make a numerical simulation more robust in cases where local wave breaking occurs but has little significance for the studied problem. Admittedly, a weakness of the scheme is that the involved parameters that go into it are of empirical nature. Although not done in the present study, such breaking-suppression scheme is directly applicable also in the MG method, and is expected to perform similarly as in the IB implementation.

In the present paper, only two-dimensional cases are considered. However, both the IB and MG methods can be extended to three dimensions without significant difficulties. The ways of modeling a physical wavemaker in the present paper are directly applicable for dealing with wave-structure interactions in general, e.g. to refine the solution near a floating body, an additional body-fixed grid can be introduced surrounding the body.

Independently of which free-surface approach used, the HPC method proves to be a robust and versatile numerical framework. Due to its continuous representation of the velocity potential within each cell, it allows for a straightforward and accurate coupling between overlapping grids. When modelling a physical wavemaker in the IB method, this feature provides a robust and accurate coupling between different grids. It also provides a straightforward way to impose free-surface conditions away from grid nodes in the IB method. In the MG method, the continuous representation of the velocity potential ensures a smooth coupling between the free-surface fitted grid and background grid. Another significant feature from a user perspective is that the HPC method enables us to compute pressure and velocity in arbitrary points in the fluid without constructing ad-hoc interpolation schemes. Instead, we can directly use the harmonic-polynomial representation without any loss of accuracy.

Acknowledgements

The authors would like to acknowledge Shaojun Ma for useful discussions related to the studies involving the wave tank with periodic boundary conditions.

This work has been carried out at the Centre for Autonomous Marine Operations and Systems (AMOS). The Research Council of Norway is acknowledged as the main sponsor of AMOS. This work was partly supported by the Research Council of Norway through the Centre of Excellence funding scheme, Project number 223254-AMOS. This research activity has also been partially supported by the Flagship Project RITMARE - The Italian Research for Sea - coordinated by the Italian National Research Council and funded by the Italian Ministry of Education, University and Research within the National Research Program 2011-2013 and by the Italian National Research Council under the Project Short-term mobility (STM) 2016.

References

[1] Dean, R. G., Dalrymple, R. A. (1991). Water wave mechanics for engineers and scientists. Singapore: World Scientific Publishing.

[2] Fenton, J. D. (1985). A fifth-order Stokes theory for steady waves. Journal of waterway, port, coastal, and ocean engineering, 111(2), 216-234.

[3] Rienecker, M. M., Fenton, J. D. (1981). A Fourier approximation method for steady water waves. Journal of Fluid Mechanics, 104, 119-37.

[4] Dean, R. G. (1965). Stream function representation of nonlinear ocean waves. Journal of Geophysical Research, 70(18), 4561-4572.

[5] Faltinsen, O. M., Timokha, A. N. (2009). Sloshing. Cambridge, NY: Cambridge University Press.

[6] Grilli, S. T., Skourup, J., Svendsen, I. A. (1989). An efficient boundary element method for nonlinear water waves. Engineering Analysis with Boundary Elements, 6(2), 97-107.

[7] Wang, P., Yao, Y., Tulin, M. P. (1995). An efficient numerical tank for non-linear water waves, based on the multi-subdomain approach with BEM. International Journal for Numerical Methods in Fluids, 20(12), 1315-1336.

[8] Wu, G. X., Taylor, R. E. (1994). Finite element analysis of two-dimensional non-linear transient water waves. Applied Ocean Research, 16(6), 363-372.

[9] Shao, Y. L., Faltinsen, O. M. (2012). Towards efficient fully-nonlinear potential-flow solvers in marine hydrodynamics. In ASME 2012 31st International Conference on Ocean, Offshore and Arctic Engineering (pp. 369-380). American Society of Mechanical Engineers.

[10] Shao, Y. L., Faltinsen, O. M. (2014). A harmonic polynomial cell (HPC) method for 3D Laplace equation with application in marine hydrodynamics. Journal of Computational Physics, 274, 312-332.

[11] Liang, H., Faltinsen, O. M., Shao, Y. L. (2015). Application of a 2D harmonic polynomial cell (HPC) method to singular flows and lifting problems. Applied Ocean Research, 53, 75-90.

[12] Fredriksen, A. G. (2015). A numerical and experimental study of a two-dimensional body with moonpool in waves and current. PhD thesis, Norwegian University of Science and Technology, Trondheim, Norway.

[13] Bardazzi, A., Lugni, C., Antuono, M., Graziani, G., Faltinsen, O. M. (2015). Generalized HPC method for the Poisson equation. Journal of Computational Physics, 299, 630-648.

[14] Chorin, A. J. (1968). Numerical solution of the Navier-Stokes equations. Mathematics of computation, 22(104), 745-762.

[15]Ma, S., Hanssen, F-C. W., Siddiqui, M. A., Greco, M., Faltinsen, O. M. (2016). Local and Global Properties of the Harmonic Polynomial Cell (HPC) Method: In-depth Analysis in Two Dimensions. Submitted to Journal of Computational Physics.

[16]Hanssen, F-C. W., Greco, M., Shao, Y. (2015). The Harmonic Polynomial Cell Method for Moving Bodies Immersed in a Cartesian Background Grid. In ASME 2015 34th International Conference on Ocean, Offshore and Arctic Engineering (pp. V011T12A019-V011T12A019). American Society of Mechanical Engineers.

[17] Subramani, A. K. (2000). Computations of Highly Nonlinear Free-Surface Flows, with Applications to Arbitrary and Complex Hull Forms. PhD thesis, The University of Michigan, Ann Arbor, US.

[18]Paulsen, B. T., Bredmose, H., Bingham, H. B. (2014). An efficient domain decomposition strategy for wave loads on surface piercing circular cylinders. Coastal Engineering, 86, 57-76.

[19]Barthelemy, X., Banner, M. L., Peirson, W. L., Fedele, F., Allis, M., Dias, F. (2015). On the local properties of highly nonlinear unsteady gravity water waves. Part 2. Dynamics and onset of breaking. Submitted. URL http://arxiv.org/abs/1508.06002.

[20]Tulin, M. P., Li, J. J. (1992). On the breaking of energetic waves. International Journal of Offshore and Polar Engineering, 2(01).

[21]Kurnia, R., van Groesen, E. (2014). High order Hamiltonian water wave models with wave-breaking mechanism. Coastal Engineering, 93, 55-70.

[22]Seiffert, B. R., Ducrozet, G. (2015). Deep water wave-breaking in a High-Order Spectral model. 30th International Workshop on Water Waves and Floating Bodies, Bristol, UK.

[23]Peregrine, D. H. (2003). Water-wave impact on walls. Annual review of fluid mechanics, 35(1), 23-43.

[24]Lugni, C., Brocchini, M., Faltinsen, O. M. (2006). Wave impact loads: The role of the flip-through. Physics of Fluids (1994-present), 18(12), 122101.

[25]Lugni, C., Miozzi, M., Brocchini, M., Faltinsen, O. M. (2010). Evolution of the air cavity during a depressurized wave impact. I. The kinematic flow field. Physics of Fluids (1994-present), 22(5), 056101.

[26]Peregrine, D. H. (1972). Flow due to a vertical plate moving in a channel. Unpublished note.

[27]Greco, M. (2001). A two-dimensional study of green-water loading. PhD thesis, Norwegian University of Science and Technology, Trondheim, Norway.

[28]Mei, C. C., Stiassnie, M., Yue, D. K. P. (2005). Theory and applications of ocean surface waves: nonlinear aspects (Vol. 23). Singapore: World Scientific Publishing.

[29]Longuet-Higgins, M. S., Cokelet, E. D. (1976). The deformation of steep surface waves on water. I. A numerical method of computation. In Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences (Vol. 350, No. 1660, pp. 1-26). The Royal Society.

[30]Russell, J. S. (1844). Report on waves. British Association Report.

[31]Fenton, J. (1972). A ninth-order solution for the solitary wave. Journal of Fluid Mechanics, 53, 257-271.

[32]Wu, N. J., Tsay, T. K., Chen, Y. Y. (2014). Generation of stable solitary waves by a piston-type wave maker. Wave motion, 51(2), 2040-255.

[33]Dommermuth, D. G., Yue, D. K., Lin, W. M., Rapp, R. J., Chan, E. S., Melville, W. K. (1988). Deep-water plunging breakers: a comparison between potential theory and experiments. Journal of Fluid Mechanics, 189, 423-442.

[34]Chapalain, G., Cointe, R., Temperville, A. (1992). Observed and modeled resonantly interacting progressive water-waves. Coastal engineering, 16(3), 267-300.

[35]Lugni, C. (1999). An investigation on the interaction between free-surface waves and freely-floating bodies. PhD thesis, University of Rome “La Sapienza”, Rome, Italy.

[36]Zhou, B. Z., Wu, G. X., Meng, Q. C. (2016). Interactions of fully nonlinear solitary wave with a freely floating vertical cylinder. Engineering Analysis with Boundary Elements, 69, 119-131.
[37]Clamond, D., Dutykh, D. (2013). Fast accurate computation of the fully nonlinear solitary surface gravity waves. Computers & fluids, 84, 35-38.

Appendix A – Derivation of Free-Surface Conditions in Overlapping Grids

Here we show how to derive the kinematic and dynamic free surface conditions for free-surface markers in the wavemaker-fixed reference frame. The wavemaker fixed grid is a structured grid, and the markers move along the vertical grid lines in the local reference frame in a semi-Lagrangian manner (see Figure 7). For simplicity, we avoid terms associated with a numerical beach in the following derivation.

We start out from the fully nonlinear Eulerian free-surface conditions in the Earth-fixed reference frame with horizontal axis x and vertical axis z pointing upwards:

$$\begin{aligned} \frac{\partial \eta(x,t)}{\partial t} &= \frac{\partial \varphi}{\partial z} - \nabla\varphi \cdot \nabla\eta \\ \frac{\partial \varphi(x,z,t)}{\partial t} &= -\frac{1}{2}|\nabla\varphi|^2 - U_g. \end{aligned} \tag{A 1}$$

U_g is the gravity potential that in the Earth-fixed reference frame equals $g\eta$. Two steps are required to express the semi-Lagrangian free-surface conditions in the moving reference frame:

1. Redefine the Eulerian conditions in the non-inertial wavemaker-fixed reference frame with local horizontal axis x' and local vertical axis z'.
2. Reformulate the Eulerian conditions in the $x'z'$ reference frame in a semi-Lagrangian form.

The $x'z'$ reference frame has origin in the lower left corner of the wavemaker-fixed grid, coinciding with the hinge location in case of a flap-type wavemaker. When the wavemaker is at rest, the $x'z'$ reference frame is parallel to the Earth-fixed xz reference frame.

When the wavemaker is moving, this gives an arbitrary grid point in the wavemaker-fixed grid a velocity $\vec{v}'_{grid} = \vec{v}'_0 + \vec{\omega} \times \vec{r}'$. $\vec{v}'_0$ is here the translatory velocity of the wavemaker (only nonzero in case of a piston-type wavemaker), $\vec{\omega}$ is the angular velocity about the hinge (only nonzero in case of a flap-type wavemaker) and $\vec{r}'$ is the vector between the grid point and the origin of the reference frame. $\vec{v}'_{grid}$ expresses the velocity of the grid point relative to the Earth-fixed reference frame, however, decomposed along the axes of the $x'z'$ reference frame.

The first step follows from eq. 2.59 in [5]:

$$\begin{aligned} \frac{\partial \eta(x,t)}{\partial t} &= \frac{\partial \varphi}{\partial z} - \nabla\varphi \cdot \nabla\eta \\ \frac{\partial \varphi(x,z,t)}{\partial t} &= -\frac{1}{2}|\nabla\varphi|^2 - U_g. \end{aligned} \tag{A 2}$$

With

$$\left.\frac{\partial}{\partial t}\right|_{wavemaker-fixed} = \left.\frac{\partial}{\partial t}\right|_{Earth-fixed} + \vec{v}'_{grid} \cdot \nabla' \tag{A 3}$$

this gives the Eulerian free-surface conditions in the wavemaker-fixed reference frame as follows:

$$\begin{aligned}\left.\frac{\partial \eta'(x',t)}{\partial t}\right|_{Wavemaker-fixed} &\equiv \frac{\partial \eta'(x',t)}{\partial t} = \frac{\partial \varphi}{\partial z'} - \nabla'\varphi \cdot \nabla'\eta' + \vec{v}'_{grid} \cdot \nabla'\eta' \\ \left.\frac{\partial \varphi(x',z',t)}{\partial t}\right|_{Wavemaker-fixed} &\equiv \frac{\partial \varphi(x',z',t)}{\partial t} = -\frac{1}{2}|\nabla'\varphi|^2 - U_g + \vec{v}'_{grid} \cdot \nabla'\varphi.\end{aligned} \tag{A 4}$$

We have here used the fact that the scalar product between two vectors remains the same in the xz and $x'z'$ reference frames. The free-surface elevation does no longer refer to z but rather to z'.

To go from an Eulerian to a semi-Lagrangian reference frame, we involve the material derivative to follow marker particles as they move in z' direction:

$$\begin{aligned}\frac{Dz'_p}{Dt} &= \left.\frac{\partial z'_p}{\partial t}\right|_{Wavemaker-fixed} + \vec{v}' \cdot \nabla' z'_p \\ \frac{D\varphi_p}{Dt} &= \left.\frac{\partial \varphi_p}{\partial t}\right|_{Wavemaker-fixed} + \vec{v}' \cdot \nabla' \varphi_p.\end{aligned} \tag{A 5}$$

where z'_p is the vertical position of the marker in the $x'z'$ reference. $\vec{v}'$ is the semi-Lagrangian velocity of the marker:

$$\vec{v}' = \left(0, \frac{\partial \eta'}{\partial t}\right). \tag{A 6}$$

By substitution, and by using $\partial z'_p/\partial t = \partial \eta'(x',t)/\partial t$, we get directly that

$$\begin{aligned}\frac{Dz'_p}{Dt} &= \frac{\partial \varphi}{\partial z'} - \nabla'\varphi \cdot \nabla'\eta' + \vec{v}'_{grid} \cdot \nabla'\eta' + \vec{v}' \cdot \nabla' z'_p \\ \frac{D\varphi_p}{Dt} &= -\frac{1}{2}|\nabla'\varphi|^2 - U_g + \vec{v}'_{grid} \cdot \nabla'\varphi + \vec{v}' \cdot \nabla' \varphi'_p.\end{aligned} \tag{A 7}$$

Writing out the terms and using $\vec{v}'_{grid} = \left(u'_{grid}, w'_{grid}\right)$, we arrive at the final semi-Lagrangian free-surface conditions in the wavemaker-fixed reference frame:

$$\begin{aligned}\frac{Dz'_p}{Dt} &= \frac{\partial \varphi}{\partial z'} - \frac{\partial \varphi}{\partial x'}\frac{\partial \eta'}{\partial x'} + u'_{grid}\frac{\partial \eta'}{\partial x'} = \frac{\partial \varphi}{\partial z'} + \left(u'_{grid} - \frac{\partial \varphi}{\partial x'}\right)\frac{\partial \eta'}{\partial x'} \\ \frac{D\varphi_p}{Dt} &= -\frac{1}{2}\left|\left(\frac{\partial \varphi}{\partial x'}\right)^2 + \left(\frac{\partial \varphi}{\partial z'}\right)^2\right| - U_g + u'_{grid}\frac{\partial \varphi}{\partial x'} + w'_{grid}\frac{\partial \varphi}{\partial z'} + \frac{\partial \eta'}{\partial t}\frac{\partial \varphi}{\partial z'} \\ &= -\frac{1}{2}\left|\left(\frac{\partial \varphi}{\partial x'}\right)^2 + \left(\frac{\partial \varphi}{\partial z'}\right)^2\right| - U_g + u'_{grid}\frac{\partial \varphi}{\partial x'} + \left(w'_{grid} + \frac{\partial \eta'}{\partial t}\right)\frac{\partial \varphi}{\partial z'}.\end{aligned} \tag{A 8}$$

Since we only allow the markers to move in z' direction, it follows directly that $Dx'_p/Dt = 0$. In the above derivation it should be noted that the gravity potential for practical purposes is expressed in the Earth-fixed reference frame, i.e. $U_g = g\eta$. This is allowable since U_g is a scalar quantity that does not depend on the adopted reference frame.

Appendix B – Temporal Convergence Study for Wave Tank with Periodic Boundary Conditions, IB Method

The cases in Table B. 1 are simulated for a time equal to two wave periods with the IB method in order to find a suitable time step Δt to use for the simulation cases in

Table 2.

Table B. 1 Simulation cases for time-step convergence study.

$kA\ (-)$	$n_{x,IB}\ (-)$	$n_{z,IB}\ (-)$	$dt\ (s)$	$T/dt\ \ (-)$
0.3	45	45	0.10	24
			0.05	48
			0.03	81
			0.02	121
			0.01	242

The convergence of the L_2 norms for the free-surface elevation $|L_{2\eta}|$, free-surface velocity potential $|L_{2\varphi,FS}|$ and velocity potential in the entire domain $|L_{2\varphi}|$ are considered to assess the results. These are plotted as a function of time in Figure B. 1, where the ordinate axis is in logarithmic scale. The results appear to be approximately converged for $\Delta t \leq 0.03s$. Since also cases with higher steepness and/or smaller grid spacing is considered in the main analysis, a time step $\Delta t = 0.01s$ is selected in order to be on the safe side. This gives between 235 and 252 time steps per oscillation period T (since T decreases slightly with increasing kA).

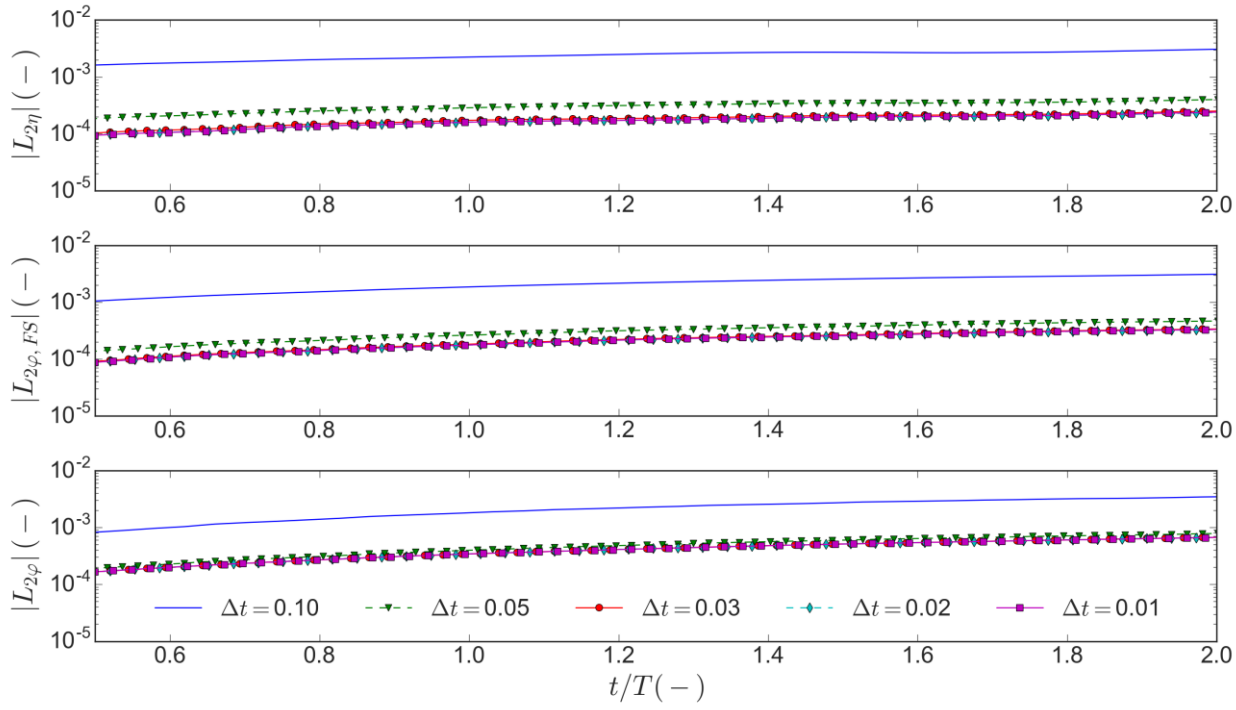


Figure B. 1 L_2 errors of free-surface elevation ($|L_{2\eta}|$), free-surface potential ($|L_{2\varphi,FS}|$) and potential in the entire fluid ($|L_{2\varphi}|$) for the time-step convergence cases in Table B. 1.