



HAL
open science

Nonlinear time-domain wave-structure interaction: a parallel fast integral equation approach

Jeffrey C Harris, Emmanuel Dombre, Michel Benoit, Stephan T. Grilli,
Konstantin I Kuznetsov

► **To cite this version:**

Jeffrey C Harris, Emmanuel Dombre, Michel Benoit, Stephan T. Grilli, Konstantin I Kuznetsov. Nonlinear time-domain wave-structure interaction: a parallel fast integral equation approach. International Journal for Numerical Methods in Fluids, 2022, 94, pp.188-222. 10.1002/fld.5051 . hal-03385825

HAL Id: hal-03385825

<https://hal.science/hal-03385825>

Submitted on 19 Oct 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

RESEARCH ARTICLE

Nonlinear time-domain wave-structure interaction: a parallel fast integral equation approach

Jeffrey C. Harris*¹ | Emmanuel Dombre^{1,2} | Michel Benoit^{2,3} | Stephan T. Grilli⁴ | Konstantin I. Kuznetsov¹

¹LHSV, Ecole des Ponts, CEREMA, EDF R&D, Chatou, France

²EDF R&D, Laboratoire National d'Hydraulique et Environnement, Chatou, France

³Aix Marseille Univ., CNRS, Centrale Marseille, Institute de Recherche sur les Phénomènes Hors Equilibre (IRPHE) UMR 7342, Marseille, France

⁴Department of Ocean Engineering, University of Rhode Island, Narragansett, RI, USA

Correspondence

*Jeffrey C. Harris, LHSV, Ecole des Ponts, CEREMA, EDF R&D, Chatou, France.
Email: jeffrey.harris@enpc.fr

Abstract

We report on the development and validation of a new Numerical Wave Tank (NWT) solving fully nonlinear potential flow (FNPF) equations, as a more efficient variation of Grilli et al.'s NWT [Grilli et al., *A fully nonlinear model for three-dimensional overturning waves over arbitrary bottom*, *International Journal for Numerical Methods in Fluids* 35 (2001) 829-867], which was successful at modeling many wave phenomena, including landslide-generated tsunamis, rogue waves, and the initiation of wave breaking over slopes. This earlier NWT combined a three dimensional MII (mid-interval interpolation) boundary element method (BEM) to an explicit mixed Eulerian-Lagrangian time integration. The latter was based on second-order Taylor series expansions for the mesh geometry and Dirichlet free surface boundary condition for the potential, requiring high-order derivatives to be computed in space and time. Here, to be able to solve large scale wave-structure interaction problems for surface-piercing bodies of complex geometry, of interest for ocean engineering and naval hydrodynamics applications, the NWT is reformulated to use cubic B-spline meshes and the BEM solution is accelerated with a parallelized Fast Multipole Method (FMM) based on ExaFMM, one of the fastest open source FMM to date. The NWT accuracy, convergence, and scaling are first assessed for simple cases, by comparing results with those of the earlier MII-NWT as a function of mesh size and other model parameters. The relevance of the new NWT for solving the targeted applications is then demonstrated for surface piercing fixed cylinders, for which we show that results agree well with theoretical and experimental data for wave elevation and hydrodynamic forces.

KEYWORDS:

Boundary element; Free surface; Fluid-structure interaction; Hydrodynamics; Integral equations; Marine hydrodynamics; Multipole; Nonlinear dynamics; Parallelization; Potential flow

1 | INTRODUCTION

Numerical models based on potential flow theory, which assumes an irrotational (and thus kinematically inviscid) flow, usually solved in a boundary integral formulation by a Boundary Element Method (BEM), combined with a time integration of the (kinematic and dynamic) free surface boundary conditions, have been very successful in modeling non-breaking water waves in a variety of applications (e.g.,^{1,2,3,4,5,6,7}, and references herein), as well as their interactions with submerged or surface piercing structures. In the latter case, models are often implemented in the form of so-called Numerical Wave Tanks (NWTs), which have all the functionality of physical waves tanks (e.g., wave generation, propagation, and absorption)^{8,9}. As they do not suffer from the excessive numerical diffusion typical of Navier-Stokes models and thus yield faster and more accurate solutions at a given mesh resolution, such BEM models have become standard tools in ocean engineering and naval hydrodynamics; see, e.g., WAMIT¹⁰, AQUAPLUS¹¹, and AEGIR¹², which are industry standards. In BEM-NWTs that solve fully nonlinear potential flow (FNPF) equations in the time domain, an elliptic problem (Laplace's equation for mass conservation) must be solved at each time step^{1,4}, which is computationally intensive, particularly in three-dimensions (3D), considering the $O(N^3)$ numerical complexity of the BEM with earlier direct methods or $O(N^2)$ for the best iterative solvers (e.g., GMRES¹³), when discretizing the boundary with N nodes. Hence, for a time step Δt and a grid resolution Δx the computational time of a standard nonlinear 3D-BEM-NWT⁴ is at best $O[1/((\Delta x)^4 \Delta t)]$, which represents the main limitation of such methods, as the solution rapidly becomes prohibitive even for moderately large grids. This is in contrast with standard linear frequency domain solutions (e.g., WAMIT) where only one BEM solution is performed per frequency.

While it may be sufficient to only consider weakly nonlinear effects in many ocean wave propagation problems, for wave-structure interaction problems, whether designing offshore structures for extreme wave conditions or wave energy converters (WECs), which may be resonant systems, both large amplitude waves and body motions need to be simulated. Hence, while linear theory may be used for pre-design or parametric studies (e.g., Folley et al.¹⁴; Falnes¹⁵), nonlinear effects must be taken into account in order to perform accurate design. Navier-Stokes (NS) models can simulate large amplitude wave and body motions (e.g.,¹⁶), but they require large computational times to propagate incident waves or even perturbations over long durations^{17,18}. For extreme irregular sea states, as indicated before, the highly dissipative numerical schemes of NS models may cause excessive damping of incident waves before they interact with ocean structures; hence, for such problems, FNPF models are often preferred.

The challenge of obtaining both a fast and accurate BEM solution of FNPF problems has led to implementing fast $\sim O(N)$ methods for solving the discretized algebraic system (e.g.,¹⁹), or to new approaches (e.g., the Harmonic Polynomial Cell method²⁰), which for moderate-size problems may be promising even if they do not achieve an optimal asymptotic $O(N)$ complexity. Although some authors have suggested that optimized volume-type methods such as Finite Element Models (FEMs) may perform faster than non-accelerated BEM-type methods²¹, this was not based on comparing BEM and FEM solutions for

the same benchmark problem on the same computer, using optimized models in both cases; hence, it is difficult to properly assess this claim. It is noted, however, that Engsig-Karup²² has been able to achieve impressive speed with a finite difference approach with the FNPF code OceanWave3D²³, and in particular scaling well over many processors²⁴.

The implementation of FNPF-NWTs based on the BEM has undergone enormous progress in the past few decades, which paralleled the increasing power of computer systems. A few years after the initial computer applications of the BEM were published²⁵, simple FNPF models solved problems with as little as $N = 30$ collocation points in two dimensions (2D)²⁶. Even in 3D, problems at first were small enough that the linear system of equations could be solved directly, for example using a LU decomposition method or equivalent, with a numerical complexity of $O(N^3)$ at each time step^{27,28}. For some years, this was a limiting factor in NWT development, but soon iterative solvers were introduced for non-symmetric fully populated matrices, such as BiCGSTAB²⁹ or GMRES, which achieve a $O(N^2)$ numerical complexity at each time step, corresponding to the CPU time required to perform a matrix-vector multiplication. With this advance, the dominant numerical restriction in 3D-FNPF-NWT became the $O(N^2)$ solution time, combined with a $O(N^2)$ of the computer memory required to assemble the system matrix (e.g.,⁴). Consequently, later research focused on finding ways to obtain sparse matrices, whose solution time could asymptotically approach $O(N)$, by avoiding to compute some of the matrix terms resulting from the BEM, exactly or at all. One such approach proposed to do so is the *fast multipole method* (FMM), which achieves an $O(N)$ complexity for a single matrix-vector product. The FMM was first introduced in 2D by Greengard and Rokhlin³⁰ and later improved and widely implemented in 3D (e.g., Kormsmeier et al.³¹, Greengard and Rokhlin³²), with variants to run computations in parallel³³. Recent implementations of the FMM can be run on large parallel computer architectures with distributed memory; see Yokota³⁴ for a recent review.

With the Laplace equation being central to many fields of study, it is important to review what other methods have been developed, both inside and outside of hydrodynamics. While the FMM has been initially developed and applied outside of hydrodynamics, there are many alternate techniques, such as the pre-corrected FFT³⁵, adaptive cross approximation with H-matrices, or wavelet BEM. Although there exist many promising direct methods (e.g., recursive skeletization of Ho and Greengard³⁶, inverse FMM³⁷), their full benefit has only been achieved for problems where many solutions are required for the same geometry, as opposed to problems where solutions are only required once or a few times per time step. Algorithms of iterative methods are quite similar to those of N-body problems for particle interactions, of which FFT and FMM based methods are fastest³⁸ (and show similar performance).

Another approach to accelerate the BEM solution in NWTs has been to use higher-order elements, which allow achieving the same accuracy with a smaller number of nodes^{4,39}. To this effect Grilli et al.^{2,4}, for instance, introduced cubic mid-interval interpolation (MII) elements in their 2D and 3D NWTs. Using such higher-order elements, the NWT numerical solution complexity was reduced by describing the geometry with the smallest number of nodes possible. Grilli et al.'s⁴ accurate 3D-FNPF-MII-NWT, which followed earlier success in 2D^{1,40,2,41,8}, was used to model many wave phenomena, including landslide-generated

tsunamis⁴², the initiation of wave breaking over slopes⁵, and rogue waves⁶. As the BEM solution was based on GMRES, Grilli et al.'s 3D-NWT had a $O(N^2)$ complexity, making it computationally inefficient once computer memory size became large enough to consider large grids. To overcome this limitation, Fochesato and Dias¹⁹ implemented a FMM in the model (single CPU implementation), which theoretically provides a nearly $O(N)$ complexity, and Sung and Grilli^{43,44,45} verified this performance for grids with a few thousand nodes, for ship hydrodynamics problems. Using these advancements, a version of the model was recently used to study the onset of breaking in 3D focused waves⁴⁶. Alternatively, Nimmala et al.⁴⁷ extended Grilli et al.'s 3D-NWT with FMM to parallel computations, but due to complex details of the model algorithm, the method could only be implemented on a small shared memory cluster (8 CPUs), whereas it is necessary to use distributed memory to best utilize large modern computers; while a reasonable scalability with N was demonstrated, $O(N^{1.3})$ up to $N = 150,000$, the resulting calculation on 8 processors was only 2.5 times faster than on a single processor for this small cluster.

In terms of physical modeling, Guerber et al.⁴⁸ extended Grilli et al.'s 2D-FNPF-NWT^{1,2} to simulate wave interactions with 2D submerged bodies freely moving in waves. In this paper, with the eventual goal of efficiently simulating wave interactions with 3D submerged or surface piercing fixed/floating bodies, for large grid sizes, we first attempted to similarly extend Grilli et al.'s⁴ 3D-FNPF-NWT with FMM acceleration¹⁹ to surface-piercing bodies. However, the structure of the MII algorithm, while quite accurate and efficient for pure wave problems, was difficult to extend to 3D surface-piercing bodies of complex geometry, such as ships, offshore structures, and WECs. Hence, as a variation of Grilli et al.'s model, we developed a new NWT in which boundaries are discretized by arbitrary polynomial meshes, and then applied this using cubic B-spline elements, which have proven to be accurate for simulating linear ship hydrodynamics (e.g.,¹²) (higher-order elements with unstructured grids will be implemented in future work). Additionally, we implemented a more efficient FMM solution of BEM equations than in earlier work^{19,7,47}, based on the parallelized ExaFMM library⁴⁹, which is one of the fastest open source FMM codes to date. The application of B-splines as boundary elements, as well as connecting the BEM solver to the ExaFMM library, in parallel, are new developments here. Initial applications of this NWT have demonstrated its ability to predict nonlinear wave-induced forces on submerged bodies⁵⁰, and the expected favorable scaling of the FMM for large grids⁵¹.

Details of this new NWT approach, numerical implementation, and basic validation are given in Section 2, while validation of the Laplace solver and tangential derivatives, and applications are presented in Section 3. To assess the model accuracy, convergence, and numerical complexity, similar to Grilli et al.'s⁴, we first simulate a solitary wave propagating over constant depth, and evaluate numerical errors on energy and volume conservation, as compared to the very accurate steady state solution of Tanaka⁵²; we also compare results with those of Grilli et al.'s 3D-MII NWT in order to have a baseline of performance. As this is the wave forcing used in all the later applications, we then demonstrate the generation and absorption of fully nonlinear periodic waves in the NWT, similar to the method detailed in Grilli and Horrillo⁸ for their 2D-NWT. Finally, we simulate

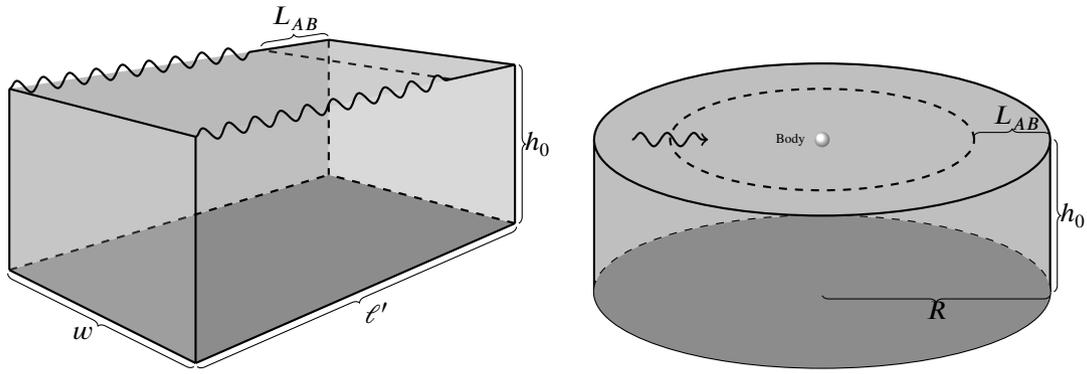


FIGURE 1 Definition sketch of NWT computational domain for wave interaction in either a rectangular domain (left; length l' by width w by depth h_0), or a circular domain (right; radius R with depth h_0). No-flow conditions are specified on lateral and bottom boundaries. Waves are imposed with Neumann boundary conditions for known velocity and acceleration either on the closest side (front left) for a rectangular domain, and damped to zero (for a rectangular domain), or imposed as a far-field boundary condition and damping the radiated waves (for a circular domain), using an absorbing beach (AB) of length L_{AB} .

wave interactions with both truncated and bottom-mounted surface-piercing cylinders, and compare results for the horizontal hydrodynamic force against experiments, theory, and other published numerical model results.

2 | METHODS

2.1 | Governing equations and boundary conditions

For an incompressible inviscid fluid with irrotational motion, mass conservation is equivalent to a Laplace's equation for the velocity potential, ϕ ,

$$\nabla^2 \phi = 0 \quad \text{in D} \quad (1)$$

$$\mathbf{u} = \nabla \phi \quad \text{in D} \quad (2)$$

with \mathbf{u} the flow velocity in domain D (e.g., Fig. 1). Based on Green's second identity, Eq. 1 is transformed into a Boundary Integral Equation (BIE) expressed over the domain boundary Γ , at a set of collocation points \mathbf{x}_i ($i = 1, \dots, N_\Gamma$),

$$\alpha(\mathbf{x}_i)\phi(\mathbf{x}_i) = \int_{\Gamma} \left[\frac{\partial \phi}{\partial n}(\mathbf{x})G(\mathbf{x} - \mathbf{x}_i) - \phi(\mathbf{x})\frac{\partial G}{\partial n}(\mathbf{x} - \mathbf{x}_i) \right] d\Gamma, \quad (3)$$

with α the interior solid angle made by the boundary at \mathbf{x}_i (e.g., for a smooth surface this would be 2π), \mathbf{n} the outwards normal vector to the boundary at point \mathbf{x} and G the 3D free space Green's function of Laplace's equation, based on the distance $r_i =$

$\|\mathbf{x} - \mathbf{x}_i\|$ from point \mathbf{x}_i to point \mathbf{x} on the boundary, i.e.,

$$G(\mathbf{x} - \mathbf{x}_i) = \frac{1}{4\pi r_i} \quad (4)$$

$$\frac{\partial G}{\partial n}(\mathbf{x} - \mathbf{x}_i) = -\frac{(\mathbf{x} - \mathbf{x}_i) \cdot \mathbf{n}}{4\pi r_i^3} \quad (5)$$

Momentum conservation can be integrated at all times into the generalized unsteady Bernoulli equation, which reads,

$$\frac{\partial \phi}{\partial t} = -gz - \frac{1}{2} \nabla \phi \cdot \nabla \phi - \frac{p}{\rho} \quad \text{in } D \quad (6)$$

with g the gravitational acceleration, z the vertical coordinate, p the fluid pressure (assumed to be zero on the free surface), and ρ the fluid density. Eq. 6 is the basis for the dynamic free surface boundary condition used in wave theories and models, in combination with a kinematic free surface boundary condition expressing that water particles on the free surface move with the free surface Γ_f ,

$$\frac{D\mathbf{r}}{Dt} = \frac{\partial \mathbf{r}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{r} = \mathbf{u} \quad \text{on } \Gamma_f \quad (7)$$

where the left-hand-side is the material derivative of a point \mathbf{r} located on the free surface.

On fixed submerged or surface-piercing bodies, the boundary condition is simply a no-flow condition on the body boundary (or hull) Γ_h , i.e., $\partial \phi / \partial n = 0$ (which also implies $\partial^2 \phi / \partial t \partial n = 0$). For moving bodies, with specified or free motion, which are not considered in this paper, the boundary condition expresses that the normal flow velocity matches that of the rigid body projected on the local normal direction. For freely moving bodies, such conditions would result from the balance of forces and moments acting on the body, in which the hydrodynamic components are obtained by integrating the hydrodynamic pressure along each body submerged boundary; see, e.g., Guerber et al.'s⁴⁸ boundary conditions for a similar 2D-NWT.

In the NWT, the BIE Eq. 3 is discretized by a BEM in an Eulerian coordinate system (x, y, z) , and the free surface boundary condition and geometry are time-integrated with a high-order time stepping scheme, following the mixed Eulerian-Lagrangian (MEL) methodology first proposed by Longuet-Higgins and Cokelet²⁶ and reviewed in section 2.2. Hydrodynamic forces and moments acting on the rigid body are computed by integrating the hydrodynamic pressure, which is obtained from Eq. 6 (and its moment) on the body surface. This requires calculating the time derivative of the potential at each time step, which also satisfies Laplace's equation. Here, as in Grilli et al.'s NWT^{1,4}, $\partial \phi / \partial t$ is also computed with a BIE, discretized by the BEM, for which boundary conditions are found based on the first BIE solution for the potential. For freely moving bodies, however, both BIEs for the potential and its time derivative are coupled through the unknown body motion, which requires implementing special procedures (see, e.g., Guerber et al.⁴⁸ for a review and details).

2.2 | Time integration in the NWT

At any time t , given well-posed boundary conditions, the BEM solution provides both the velocity potential and its normal derivative on the computational domain boundary, as discrete values at the N_Γ (collocation) points of the grid; see section 2.3.

This solution (both geometry and boundary conditions) is advanced in time on the free surface by integrating the free surface boundary conditions, Eqs. 6 and 7. Unlike in our earlier work⁴ which focused on overturning waves and thus required a Lagrangian updating of the free surface, here a semi-Lagrangian approach is used, for which free surface points are fixed in the horizontal direction. This leads to redefining the material derivative as,

$$\frac{\delta}{\delta t} = \delta_t = \frac{\partial}{\partial t} + \frac{\partial \zeta}{\partial t} \frac{\partial}{\partial z} \quad (8)$$

where $\zeta = \mathbf{r} \cdot \mathbf{k}$ (with \mathbf{k} the vertical unit vector) denotes the vertical position of the free surface.

Applying this new definition of the material derivative to the kinematic and dynamic free surface boundary conditions Eqs. 6 and 7 yields,

$$\frac{\delta \zeta}{\delta t} = \frac{\partial \phi}{\partial z} - \frac{\partial \phi}{\partial x} \frac{\partial \zeta}{\partial x} - \frac{\partial \phi}{\partial y} \frac{\partial \zeta}{\partial y} \quad (9)$$

$$\frac{\delta \phi}{\delta t} = -g\zeta - \frac{1}{2} \nabla \phi \cdot \nabla \phi + \frac{\partial \zeta}{\partial t} \frac{\partial \phi}{\partial z} \quad (10)$$

in which $(\partial \zeta / \partial x, \partial \zeta / \partial y)$ can be expressed as a function of the outward normal vector on the boundary, $\mathbf{n} = (n_x, n_y, n_z)$, as $(-n_x/n_z, -n_y/n_z)$.

With this semi-Lagrangian approach, if only surface piercing structures with vertical sidewalls are considered, there is no need for remeshing. For structures with curved walls, however, which are not considered here, both a different material derivative and remeshing should be used; see Sung and Grilli⁴⁵ or Zhang and Kashiwagi⁵³.

A third-order Runge-Kutta explicit scheme, similar to Gottlieb⁵⁴, is used to integrate Eqs. 9 and 10. With f denoting either ζ or ϕ , we have,

$$\begin{aligned} f^{(1)} &= f^{(n)} + \Delta t (\delta_t f^{(n)}) \\ f^{(2)} &= \frac{3}{4} f^{(n)} + \frac{1}{4} (f^{(1)} + \Delta t (\delta_t f^{(1)})) \\ f^{(n+1)} &= \frac{1}{3} f^{(n)} + \frac{2}{3} (f^{(2)} + \Delta t (\delta_t f^{(2)})) \end{aligned} \quad (11)$$

with Δt the time step.

In addition to the potential and its normal derivative, values of their time derivatives are required either for time-stepping or for computing forces on structures. As indicated before, in the NWT, the latter are obtained by solving a second Laplace equation for the time derivative of the potential $\partial \phi / \partial t$, as a BIE similar to Eq. 3. Additionally, values of the first- and second-order tangential derivatives of most of these fields must be computed on some parts of the boundary (i.e., the free surface and

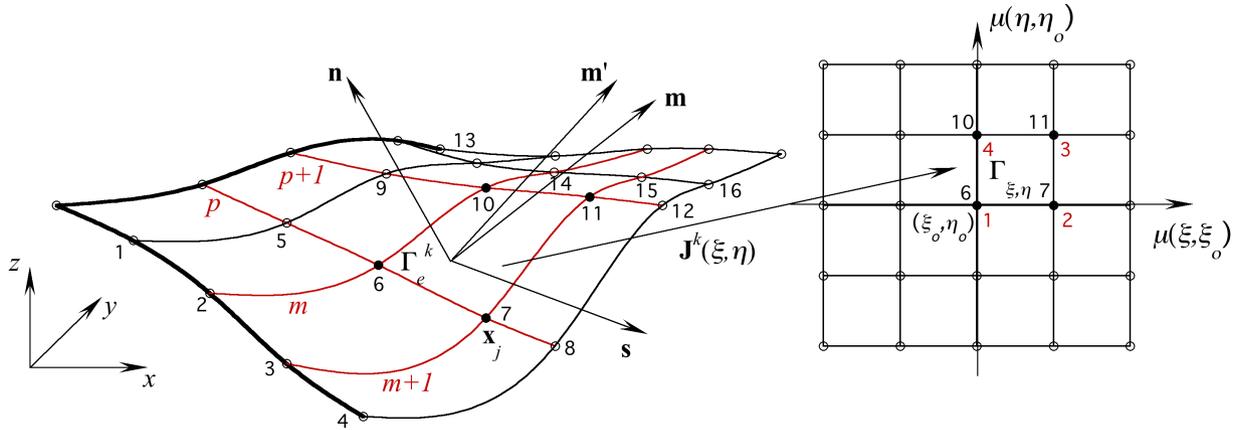


FIGURE 2 Sketch of a domain boundary section close to two edges (bold lines), that shows an example of a 16-node patch (1-16) used to define cubic MII or B-spline interpolations over quadrilateral Cartesian boundary element Γ_e^k (marked by intersecting red lines) and corresponding curvilinear reference element $\Gamma_{\xi,\eta}$ (nodes 6 (1), 7 (2), 11 (3), 11 (4)); the former is transformed into the latter with Jacobian $\mathbf{J}^k(\xi, \eta)$. The orthogonal curvilinear coordinate system ($\mathbf{s}, \mathbf{m}', \mathbf{n}$) is defined over the boundary, with the outward normal vector $\mathbf{n} = \mathbf{s} \times \mathbf{m}'$ being orientated based on nodes 1-4 numbering direction (e.g., anti-clockwise here). The lower left corner of the reference element is defined at (ξ_0, η_0) , as one in 9 possible positions within the 16-node patch. Red lines identify the one-dimensional cubic-isoparametric/spline interpolations that define the 2D MII/B-spline interpolations for element k .

moving or rigid bodies if any). As pointed out in earlier work⁴, the accurate computation of these derivatives is as important as that of the BEM solution, to reduce overall numerical errors during the time-stepping (e.g., on mass and energy conservation). Details of the higher-order free surface representation used in the NWT are given in the next section, as well as expressions of tangential derivatives.

Although we do not consider freely moving rigid bodies in this paper, it is worth noting that similar to the work of Guerber et al.⁴⁸ for submerged bodies, Dombre et al.⁵⁵ made use of the most recent 2D-NWT of Grilli et al.^{1,2} to simulate wave interactions with freely moving floating bodies. In their model, they used a symplectic-like time integration scheme to update both the free surface and body position, which was shown to improve long-term energy conservation. Although Dombre et al. only considered a first-order time-stepping, higher-order symplectic integrators have been proposed by others (e.g., Dias and Bridges⁵⁶). As for many applications it is not necessary to consider very long simulation times, the advantages of using a symplectic integrator may not be immediately apparent. In work in progress, the 3D-NWT was extended to more complex surface-piercing support structures used in offshore engineering⁵⁷.

2.3 | Representation of the boundary geometry in the NWT

For problems formulated as BIEs, the geometry is uniquely defined by that of the domain boundary, and the solution to certain field variables is found on this boundary. In BEM models, the boundary geometry is specified at N_Γ collocation points \mathbf{x}_j , and

M_Γ boundary elements are used to interpolate in between these points (Fig. 2). Within the domain, the solution can then be explicitly computed based on the boundary solution without solving additional equations.

Hence, in the NWT, the accurate representation of the boundary geometry and interpolation of field variables on the boundary are both key to the accurate solution of Laplace's equation. In the BEM, the piecewise representation of the boundary geometry can be based on elements of various order and number of nodes (e.g., linear, higher-order), with higher-order elements providing a more accurate and faster converging solution, but at a higher computational cost (Figs. 2 and 3). High-order interpolations are simpler to implement on structured grids, as one-dimensional interpolation methods can directly be extended and applied in two directions. In earlier work, Grilli and Subramanya² (in 2D, i.e., with a one-dimensional (1D) boundary), and Grilli et al.⁴ (in 3D, i.e., with a 2D boundary) introduced a cubic interpolation approach of the geometry and field variables on a structured grid in their BEM-NWT, referred to as mid-interval interpolation (MII). In 3D, MII elements are defined based on two one-dimensional cubic polynomial interpolations in each direction (ξ, η) (Figs. 2 and 3b), as the middle quadrilateral of a 16-node patch; for elements defined along boundary intersections, the MII uses the quadrilateral located off-center, which includes part of the boundary edge or intersection with surface piercing bodies. While this approach provides only C^0 continuity along the perimeter of each 16 node patch, it locally offers a higher-order solution while avoiding errors due to Runge's phenomenon.

In their 2D applications, Grilli and Subramanya² had also developed Mixed-Cubic-Interpolation (MCI) 1D elements, in which the geometry was interpolated by natural cubic splines and the field variables by a cubic MII. They found that both MII and MCI elements provided a similar accuracy, but as the MCI elements required computing inter-element slopes by solving an additional system of equations, they mostly used the MII elements in applications. When they extended simulations to 3D, Grilli et al.⁴ also used the MII approach. In many ocean and naval engineering applications of the BEM, however, a standard approach has been to use B-spline quadrilateral elements to interpolate the geometry and field variables on the boundary, which can be made arbitrarily smooth^{12,58}. Cubic B-splines, which have been widely used in computer aided design (CAD) to accurately describe complex surfaces, are similar to two-dimensional natural splines, but are more efficiently defined as a function of control points,

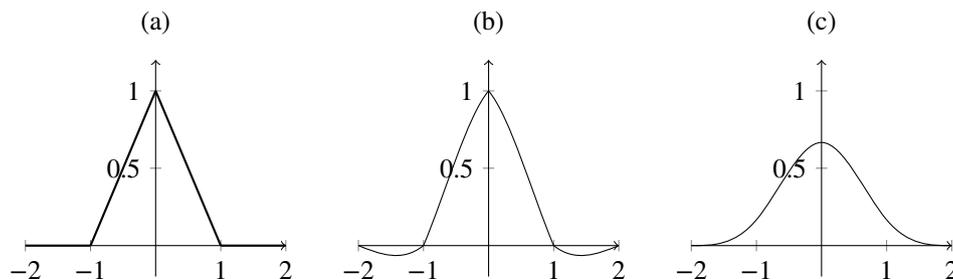


FIGURE 3 Equivalent one-dimensional (1D) basis function for the BEM elements considered in this paper: (a) linear isoparametric; (b) cubic MII⁴; (c) cubic B-spline. Note that linear elements can be applied to arbitrary, unstructured connectivities, while the MII elements cannot.

rather than inter-element slopes. Control points, in turn can be expressed as a function of nodal values of the geometry or field variables. Cubic B-spline elements will be used to discretize the boundary of the proposed NWT; details are provided below in section 2.3.3.

Additionally, the higher-order explicit time updating algorithm used in the NWT requires that tangential derivatives of the geometry and field variables be accurately computed along the free surface and body boundaries. While tangential derivatives on the boundary could also be directly computed with a BIE, this would require solving hypersingular equations; thus, for simplicity and higher efficiency of the model, similar to Grilli et al.'s NWT^{1,4}, tangential derivatives are computed analytically within a local approximation of the boundary geometry, which can be independent of that introduced in the BEM. Thus, in earlier work, Grilli et al.⁴, who used cubic MII elements in the BEM, locally computed first- and second-order tangential derivatives within 4th-order isoparametric sliding element patches (with 25 nodes). By contrast, in the proposed NWT, tangential derivatives will be computed using the same B-spline approximation as in the BEM.

With a piecewise high-order representation of the boundary geometry, tangential derivatives can be computed by defining a local coordinate system, with unit vectors $(\mathbf{s}, \mathbf{m}, \mathbf{n})$ pointing along 2 tangential directions (s, m) and the corresponding outward normal direction n (Fig. 2) in which, defining the curvilinear coordinates (ξ, η) ,

$$\mathbf{s} = \frac{1}{h_1} \left(\frac{\partial \mathbf{x}}{\partial \xi} \right) \quad \text{with} \quad h_1 = \left| \frac{\partial \mathbf{x}}{\partial \xi} \right| \quad (12)$$

$$\mathbf{m} = \frac{1}{h_2} \left(\frac{\partial \mathbf{x}}{\partial \eta} \right) \quad \text{with} \quad h_2 = \left| \frac{\partial \mathbf{x}}{\partial \eta} \right| \quad (13)$$

and thus, $\partial_s = \partial_\xi / h_1$, and $\partial_m = \partial_\eta / h_2$. The coordinate system $(\mathbf{s}, \mathbf{m}, \mathbf{n})$ is not necessarily orthogonal⁵⁹, thus defining $\kappa = \mathbf{s} \cdot \mathbf{m}$ and unit vector \mathbf{m}' such that $(\mathbf{s}, \mathbf{m}', \mathbf{n})$ is orthogonal (Fig. 2), we have,

$$\mathbf{n} = \mathbf{s} \times \mathbf{m}' = \frac{\mathbf{s} \times \mathbf{m}}{\sqrt{1 - \kappa^2}} \quad (14)$$

Using this coordinate system, we can compute $\mathbf{u} = \nabla \phi$ as⁵⁹,

$$\nabla \phi = \frac{\phi_s - \kappa \phi_m}{1 - \kappa^2} \mathbf{s} + \frac{\phi_m - \kappa \phi_s}{1 - \kappa^2} \mathbf{m} + \phi_n \mathbf{n} \quad (15)$$

The coordinate transformation of the boundary geometry defined in Cartesian coordinates to the curvilinear coordinates, $(x, y, z) \rightarrow (\xi, \eta)$ (with the third coordinate in the normal direction implied, but not used here as only the boundary is discretized) is defined by the Jacobian matrix (Fig. 2),

$$\mathbf{J} = \left(\frac{\partial \mathbf{x}}{\partial \xi}, \frac{\partial \mathbf{x}}{\partial \eta}, \mathbf{n} \right) \quad \text{with} \quad |\mathbf{J}| = \kappa h_1 h_2, \quad (16)$$

which can be used to transform integrals in the BIE, Eq. 3. Considering boundary element k , the piecewise interpolation of a scalar function f^k (e.g., $x, y, z, \phi, \phi_n, \dots$) over the Cartesian element boundary Γ_e^k is expressed by way of a set of N_e shape

functions, which are typically defined over the reference element $\Gamma_{\xi,\eta}$, $S_j(\xi, \eta)$ ($j = 1, \dots, N_e$), as,

$$f^k(\xi, \eta) = \sum_{j=1}^{N_e} S_j(\xi, \eta) \hat{f}_j^k \quad (17)$$

where \hat{f}_j^k denotes values related to the function f at different points, as described in the next section. Based on this representation, the tangential s - and m -derivatives of the function f can be computed on element k as,

$$\frac{\partial f^k(\xi, \eta)}{\partial(s, m)} = \frac{1}{(h_1, h_2)} \sum_{j=1}^{N_e} \frac{\partial S_j(\xi, \eta)}{\partial(\xi, \eta)} f_j^k \quad (18)$$

in which the ξ - and η -derivatives of the shape functions are typically analytically known, based on the considered element model.

Consistent with results presented later, where in some applications we compare the efficiency and accuracy of the newly proposed 3D-FNPF-NWT with FMM, based on cubic B-splines, to that of Grilli et al.'s⁴ based on MII elements, we detail below these two types of element models in sections 2.3.2 and 2.3.3, using a consistent set of notations. For completeness, we first discuss a linear piecewise interpolation based on an unstructured triangular grid, as this simpler approach was used in the initial version of the NWT^{51,50}.

2.3.1 | Linear interpolation on unstructured BEM grids

The linear interpolation of an arbitrary function $f(\mathbf{x}(\xi, \eta))$ over a triangular element k is defined with Eq. 17, for $N_e = 3$, using the linear shape functions (Fig. 3a), with $0 \leq \xi \leq 1$, and $0 \leq \eta \leq 1 - \xi$,

$$S_1(\xi, \eta) = 1 - \xi - \eta \quad ; \quad S_2(\xi, \eta) = \xi \quad ; \quad S_3(\xi, \eta) = \eta \quad (19)$$

and three nodal values of the function, \hat{f}_j^k ($j = 1, \dots, 3$) defined at the element vertices, located at $(0, 0)$, $(1, 0)$, and $(0, 1)$. This representation as well as other typical *isoparametric elements* have the advantage of only requiring that the mesh be a connected set of triangles, without any additional requirements on mesh connectivity (Fig. 3a). While quadratic triangular elements have been used in NWTs (e.g.,⁶⁰), higher-order Lagrange (C^0 continuous) elements are rare, because while high-order methods converge faster, high-order C^0 elements can have poor features at the edges between elements, which can negatively affect the solution, particularly when being part of a time-updating scheme.

An alternative for high-order unstructured meshes is to use subdivision surfaces, whereby the interpolation (or approximation) are determined in a recursive fashion. The limit surface, to which the method converges, in fact, is a smooth spline patch, except near extraordinary vertices (i.e., for quadrangular meshes, where the number of elements which meet at one point is not 4; for triangular meshes, where this number is not 6). Such subdivision elements have been used in finite element analyses⁶¹, and in ship design (e.g., DELFTship⁶²). Due to the additional complexity required at intersections, we do not consider this work here,

and focus instead on other issues (such as improving computational speed and scalability on computer clusters), leaving the use of higher-order unstructured meshes, although desirable, for future work.

2.3.2 | Mid-interval interpolation on structured BEM grids

As discussed above, the 3D-MII bi-cubic elements are defined as one of nine quadrilateral elements within a (4 x 4) 16-node sliding element patch, depending on their location on the boundary based on the quadrilateral's lower left node coordinates (ξ_0, η_0) (Fig. 2). The bi-cubic shape functions in Eq. 17, are constructed, for $N_e = 16$, as the product of two one-dimensional 4-node cubic shape functions $N'_c(\mu)$ ($c = 1, \dots, 4$) (Fig. 3b) where, in order for (ξ, η) to vary within $[-1, +1]$ over the element $\Gamma_{\xi, \eta}$ (i.e., within the middle interval of the 1D shape functions), the additional transformation from μ to the reference element intrinsic coordinates (ξ, η) is defined as,

$$\mu(\chi, \chi_o) = \chi_o + \frac{1}{3}(1 + \chi) \quad (20)$$

with $\chi = \xi$ or η , for each of the curvilinear direction in the reference element $\Gamma_{\xi, \eta}$, and $\chi_o = \xi_o$ or $\eta_o = -1, -1/3$ or $1/3$, depending on which of the 9 quadrilaterals defined is selected. Thus, for an MII element,

$$S_j(\xi, \eta) = N'_{b(j)}(\mu(\xi, \xi_o)) N'_{d(j)}(\mu(\eta, \eta_o)) \quad (21)$$

with $b, d = 1, \dots, 4$; $j = 4(d - 1) + b$, and the standard property of shape functions implying, for $i = 1, \dots, 4$,

$$N'_c(\mu_i) = \delta_{ic} \quad \text{with} \quad \mu_i = (2i - 5)/3 \quad (22)$$

Hence,

$$\begin{aligned} N'_1(\mu) &= \frac{1}{16}(1 - \mu)(9\mu^2 - 1) & ; & \quad N'_2(\mu) = \frac{9}{16}(1 - \mu^2)(1 - 3\mu) \\ N'_3(\mu) &= \frac{9}{16}(1 - \mu^2)(1 + 3\mu) & ; & \quad N'_4(\mu) = \frac{1}{16}(1 + \mu)(9\mu^2 - 1) \end{aligned} \quad (23)$$

where the range of integration for elements away from an edge are $-1/3 \leq \mu \leq 1/3$. Although this scheme is third order, we see that the slope of two adjacent elements are not necessarily the same, as $\partial_\mu N'_1(-1/3) \neq \partial_\mu N'_2(1/3)$. To create a smoother surface would require either using more nodes per element (i.e., a larger N_e), which incurs a significant computational time in the BEM approach, or choosing different basis functions for the interpolation (i.e., using B-splines, detailed next).

In some of the applications presented below, we will compare results of Grilli et al.'s 3D-NWT based on MII elements to those of the new proposed NWT based on cubic B-splines. The latter are detailed next.

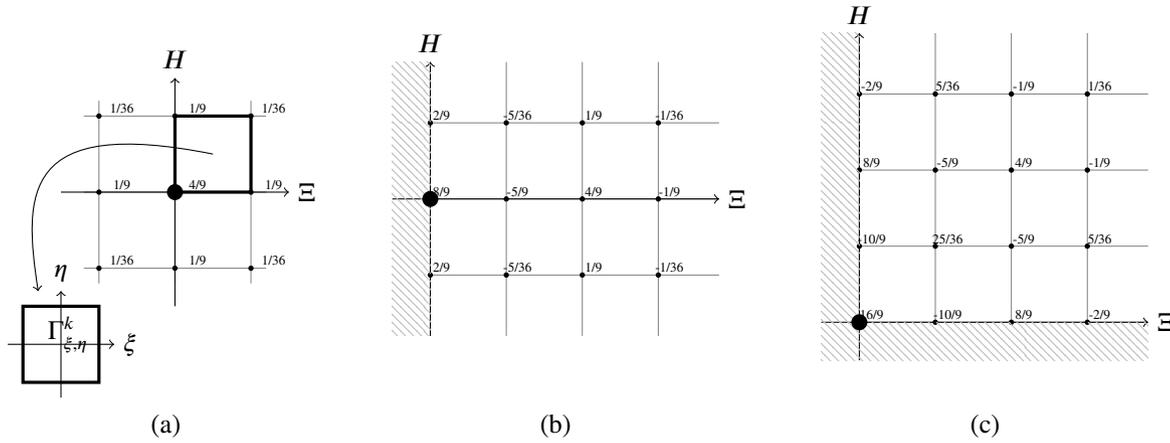


FIGURE 4 Coefficients relating 2D B-spline control points, \hat{f} , and the interpolated property $f(\xi(\mathbf{x}_i^k), \eta(\mathbf{x}_i^k)) = \sum_{b=1}^4 \sum_{d=1}^4 B'_b(\mu(\xi_i, \xi_0)) B'_d(\mu(\nu_i, \nu_0)) \hat{f}_j^k$ with $j = 4(d-1) + b$: (a) in the center of the 2D grid; (b) along an edge; (c) using not-a-knot conditions at a corner. Grayed out areas are beyond the edge of the 2D grid.

2.3.3 | Cubic B-splines on structured BEM grids

In the proposed NWT, bi-cubic B-splines are used to interpolate all field variables on the boundary (Figs. 2 and 3c), including coordinates, velocity potential, and its normal derivative (i.e., $f = x, y, z, \phi, \phi_n$ in Eq. 17). The goal of this interpolation is to obtain a smooth representation with continuous first- and second-derivatives, both on the surface of each element Γ_c^k and at each collocation node \mathbf{x}_i . This approach was first used in a 2D-BEM-NWT by Liu et al.⁶³. The application of more general polynomial expressions such as B-splines also makes it simpler to later consider using non-uniform rational B-splines (NURBS)⁶⁴ to model the NWT boundaries, and particularly the geometry of surface piercing structures such as ships. Abbasnia and Guedes Soares⁶⁵ have also done similar wave-body interaction simulations, purportedly with NURBS elements, though generally for cases which only require regular B-splines (and further their results are only parallelized on a small number of processors, and without FMM, as is done here).

For structured curvilinear grids made of quadrilateral elements, a bi-cubic B-spline interpolation can be defined over each element based on the product of two 1D cubic B-splines specified in directions ξ and η , respectively (Fig. 2). This can be written in the same notation as the MII elements previously:

$$\begin{aligned} B'_1(\mu) &= \frac{-1}{48} (3\mu - 1)^3 & B'_2(\mu) &= \frac{1}{48} (81\mu^3 - 27\mu^2 - 45\mu + 23) \\ B'_3(\mu) &= \frac{1}{48} (-81\mu^3 - 27\mu^2 + 45\mu + 23) & B'_4(\mu) &= \frac{1}{48} (3\mu + 1)^3 \end{aligned} \quad (24)$$

One can easily verify that this enforces, by construction, the continuity of the first- and second-derivatives, due to the C^2 continuity of the basis function.

At each edge and corner (i.e., boundary intersections in the NWT), high-order elements require additional information for interpolation, and one does not in general know both the value and the tangential derivative of the geometry and field variables,

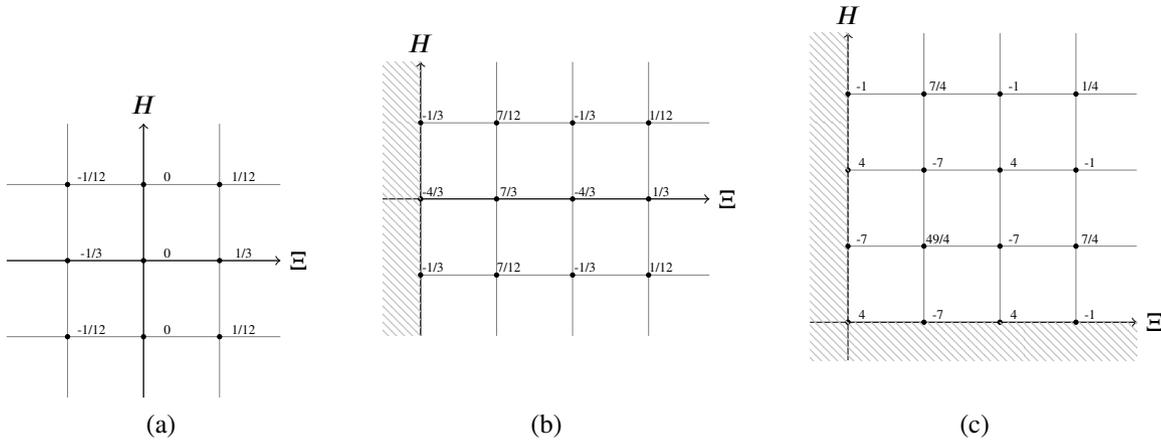


FIGURE 5 Several examples of coefficients relating 2D B-spline control points, \hat{f} , and the interpolated property derivative $(1/2)\partial_{\xi} f(\xi(\mathbf{x}_i^k), \eta(\mathbf{x}_i^k)) = (1/2) \sum_{b=1}^4 \sum_{d=1}^4 (\partial_{\mu} B'_b(\mu(\xi_i, \xi_0))) B'_d(\mu(v_i, v_0)) \hat{f}_j^k$ with $j = 4(d - 1) + b$ for, e.g.: (a) derivatives in the center of the 2D grid; (b) along an edge; (c) at a corner. Grayed out areas represent the area beyond the edge of the 2D grid.

so additional conditions must be specified. One standard method to specify such conditions, when no other information is known from the problem physics, is the so-called *not-a-knot* condition, forcing the third-derivatives in the two elements adjacent to a boundary to be the same; this is automatically satisfied with the μ -coordinate transformation above as used for the MII element.

Using these shape functions, however, we no longer have the property given in Eq. 22.

$$f[i] = \sum_{j=1}^N M_{ij}^{\mu} \hat{f}[j] \quad (25)$$

where each row of M_{ij}^{μ} is a representation of Eq. 17. More specifically, if a node i is part of an element k with local coordinates (ξ_0, η_0) , we can write the sum:

$$f[i] = f^k(\xi_0, \eta_0) = \sum_{j=1}^{N_e} S_j(\xi_0, \eta_0) \hat{f}_j^k \quad (26)$$

Given the properties of B-splines, this will be identical for any element chosen; the specific values can be seen in Fig. 4. Eq. 26 also shows that, while interpolation of a property on any boundary element will depend on all the nodal values, the matrix to solve for each grid line is very sparse and can be easily inverted to $(M_{ij}^{\mu})^{-1}$ (e.g., for an interior point, each row will contain 9 non-zero values, as seen in Fig. 4a).

In the NWT, since the grid connectivity does not change with time, this linear system is only constructed once at the beginning of each simulation, and then inverted at each time step as needed to interpolate different physical parameters over the mesh. The solution time for this sparse system of equations is very small compared to the total solution time (e.g., around 1 ms for the typical grids considered in this paper). In addition, this is only required twice per problem to solve, and not for every iteration, so the overall solution time is almost unaffected as compared to an MII element.

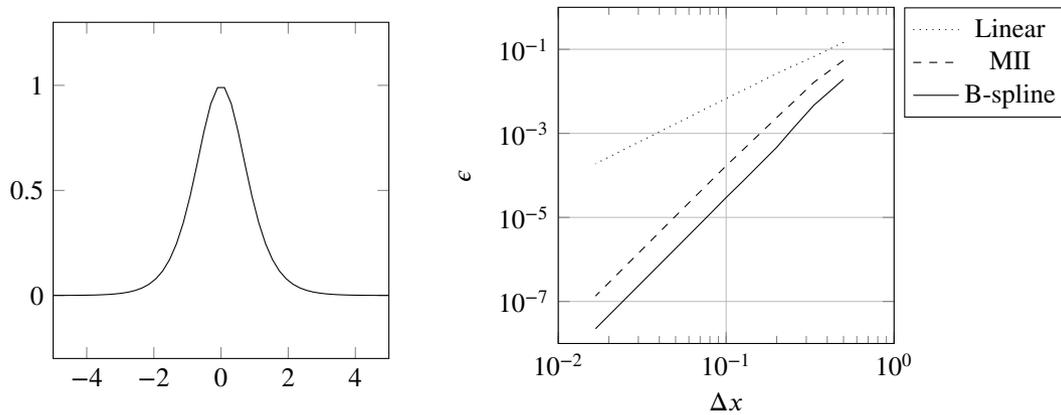


FIGURE 6 Convergence rate of tangential derivative for the function $\text{sech}^2(x)$ (left), showing errors (right) using linear elements, MII elements, and B-splines.

Tangential derivatives with B-splines

The derivatives at each node can be computed in the same manner, as a result of a direct application of Eq. 18. Example coefficients which correspond to these derivatives are shown in Fig. 5. As a simple example, we can simply take a surface (or equivalently, a line), and compute equivalent tangential derivatives using a linear, MII, and B-spline approach, and compare them. To avoid end-effects (discussed next), we can consider a smooth function, such as $\text{sech}^2(x)$, and look at the derivatives over an interval $[-5, 5]$ for different grid spacings, Δx . We find that for such functions, we obtain around 6 times more accurate tangential derivatives with B-spline than with MII elements, with a 4th-order convergence (Fig. 6).

Corner conditions in NWT geometric representation

A more complete study of B-splines versus MII elements can find cases when errors at corners are smaller for MII elements, and this is associated to the not-a-knot conditions. In any case, while not-a-knot conditions yield a well-posed linear system for computing B-spline control point values \hat{f}_j 's, more relevant boundary conditions can be expressed along edges based on physical constraints expressed for the velocity potential ϕ and the flow velocity \mathbf{u} (i.e., the normal and tangential derivatives of the potential) along the considered edge. For instance, in their 2D-NWT, Grilli and Svendsen⁶⁶ and Grilli and Subramanya² expressed such *extended compatibility conditions* for 1D quasi-spline or MII elements. Defining double nodes at corners, they expressed both continuity of the velocity potential and flow velocity on each side of the boundary, for various cases of Dirichlet/Neumann conditions; they then replaced one of the two double-node equations in the BIE algebraic system by this condition. Similar conditions (as first discussed for cubic B-splines in 2D by Sen⁶⁷) have been developed for the present NWT by Mivehchi et al.⁶⁸, which have been shown to improve the numerical accuracy and stability of the solution during time stepping.

2.4 | BEM solution of Laplace's equation with a FMM

In order to simulate problems with large number of gridpoints in a reasonable amount of time, some method of acceleration is typically required, such as the FMM. Here, we make use of the ExaFMM library developed by Yokota and Barba⁶⁹, which uses Cartesian multipole expansions of the Green's function (which is less common in FMM codes, but shown by Yokota³⁴ to be similar in speed to spherical harmonics, if not substantially faster, depending on the order of the expansion used), and the dual-tree traversal approach of Dehnen⁷⁰.

In terms of developments of the FMM-BEM, the ExaFMM library was used for computing all far-field interactions (M2M, M2L, L2L, L2P, as described in section 2.4.3), as well as inter-processor communication for far-field interactions, after the B-spline BEM code, developed here, computes local interactions and the multipole expansion for a given element. It should be noted that the ExaFMM library does not (at this point) have any mechanism for its application to boundary elements. This further limits the dependence of the BEM code on the specific FMM library in case other approaches become more efficient in the future. For completeness, the full algorithm is shown in section 2.4.3.

2.4.1 | Classic BEM solution

At each time step of the NWT solution, Laplace's equation is solved in domain D for the velocity potential and its time derivative (Eq. 1), based on the BIE Eqs. 3 to 5. To do so, these equations are discretized with a BEM, where the boundary geometry and field variables are piecewise-represented within M_Γ higher-order boundary elements, as detailed in the previous section. The BEM solution requires: (i) integrating complex integral kernels over each individual boundary elements Γ_e^k ($k = 1, \dots, M_\Gamma$), which become singular when $r_i \rightarrow 0$; and (ii) solving the resulting (typically N_Γ by N_Γ) linear system of algebraic equations.

Here, all regular integrals are computed with a 6-node (i.e., 36 nodes per element) Gauss integration and singular integrals (i.e., with a $1/r_i$ type weak singularity) are handled using a Duffy transformation⁷¹ for B-splines; singular integration methods for MII elements are detailed in⁴. Additionally, for boundary points close to a given element (but not on it), although not singular, the Green's function varies rapidly with r_i . This leads to so-called quasi-singular integrals, whose accuracy can be improved by adaptive element subdivision, as shown by Grilli and Subramanya^{72,2}. This two dimensional method was later extended by Grilli et al.⁴ to quadrilateral MII elements in their 3D-NWT. Here, a similar adaptive integration method is applied to B-spline elements when the distance from the center of an element being integrated and the collocation point being considered is within a factor of two of the element size (taken to be the largest distance between any two corners of the element). If the element is thus close to the collocation node, the boundary integral is split into four smaller Gauss integrals. This process is recursive and, for results presented here, it is applied up to four times (yielding a maximum of $4^4 = 256$ subdivisions).

Finally, coefficients α in the BIE Eqs. 3 are found by applying the rigid mode method (see, e.g., Brebbia⁷³ and^{1,4}), which expresses that, for a Dirichlet problem solved for a constant ϕ specified over the entire boundary Γ , the discretized BIE solution

must be $\partial\phi/\partial n = 0$ at all collocation points i ; based on this requirement, the α coefficients can be found as the residuals of this Dirichlet problem.

Discretizing Eq. 3 with M_Γ elements, representing the variation of each property (geometry and field variables) f^k over the elements Γ_e^k by Eq. 17, transforming the integral kernels to the reference element $\Gamma_{\xi,\eta}$, with $\xi \in [-1, 1]$ and $\eta \in [-1, 1]$ (Fig. 2)), and computing the influence coefficients by numerical integration, yields the linear algebraic system of equations (i and $j = 1, \dots, N_\Gamma$),

$$\alpha(\mathbf{x}_i) \phi(\mathbf{x}_i) = \sum_{k=1}^{M_\Gamma} \int_{\Gamma_e^k} \left[\frac{\partial\phi}{\partial n}(\mathbf{x}) - \phi(\mathbf{x}) \frac{\partial}{\partial n} \right] G(\mathbf{x} - \mathbf{x}_i) d\Gamma \quad (27)$$

$$= \sum_{k=1}^{M_\Gamma} \int_{\Gamma_{\xi,\eta}^k} \left[S_j(\xi, \eta) \frac{\partial\phi_j}{\partial n} - S_j(\xi, \eta) \phi_j \frac{\partial}{\partial n} \right] G(\mathbf{x}(\xi, \eta) - \mathbf{x}_i) |\mathbf{J}^k(\xi, \eta)| d\xi d\eta \quad (28)$$

$$= \sum_{j=1}^{N_\Gamma} \left(K_{ij}^d \frac{\partial\phi_j}{\partial n} - K_{ij}^n \phi_j \right) \quad (29)$$

and either the potential or its normal derivatives are specified as Dirichlet (d) or Neumann (n) boundary conditions, on complementary parts of the boundary.

The solution of the algebraic system of Eqs. 29 can efficiently be computed using GMRES¹³, a Krylov iterative solver with a $O(N_\Gamma^2)$ numerical complexity. Typically, however, the most computationally intensive part in such a standard BEM solution is the computation and assembling of the fully populated Neumann and Dirichlet system matrices, $[K_{ij}^n]$ and $[K_{ij}^d]$, respectively, which has a similar numerical complexity.

2.4.2 | Fundamental FMM assumptions

With the FMM, the BIE influence coefficients in matrices $[K_{ij}^n]$ and $[K_{ij}^d]$ are only integrated as stated for small r_i values (based on a FMM distance criterion), i.e., on the boundary part immediately surrounding point \mathbf{x}_i ; these are referred to as *local* or near-field FMM computations. For larger distances r_i , a multipole approximation of the Green's function is applied, which both simplifies and accelerates computations; these are referred to as *distant* or far-field FMM computations (Fig. 7).

Specifically, the FMM applies a divide-and-conquer strategy, in which far-field influence coefficients are approximated based on a truncated multipole (polynomial) expansion of the Green's function. If \mathbf{x}_j is a point belonging to a far-field boundary element with respect to collocation point \mathbf{x}_i , their distance is first decomposed into three parts (Fig. 8), by assuming that \mathbf{x}_j is close to the intermediate point \mathbf{x}_M , and \mathbf{x}_i to the intermediate point \mathbf{x}_Λ , as,

$$\mathbf{x}_i - \mathbf{x}_j = (\mathbf{x}_i - \mathbf{x}_\Lambda) + (\mathbf{x}_\Lambda - \mathbf{x}_M) + (\mathbf{x}_M - \mathbf{x}_j) \quad (30)$$

The ‘‘closeness’’ of all such pairs of boundary points is defined based on the multipole acceptance criterion (MAC) θ . Thus, if $(\|\mathbf{x}_i - \mathbf{x}_\Lambda\| + \|\mathbf{x}_M - \mathbf{x}_j\|)/\|\mathbf{x}_\Lambda - \mathbf{x}_M\| < \theta$ (Fig. 8), the Green's function for the distance vector $\mathbf{x}_i - \mathbf{x}_j$, i.e., between two points

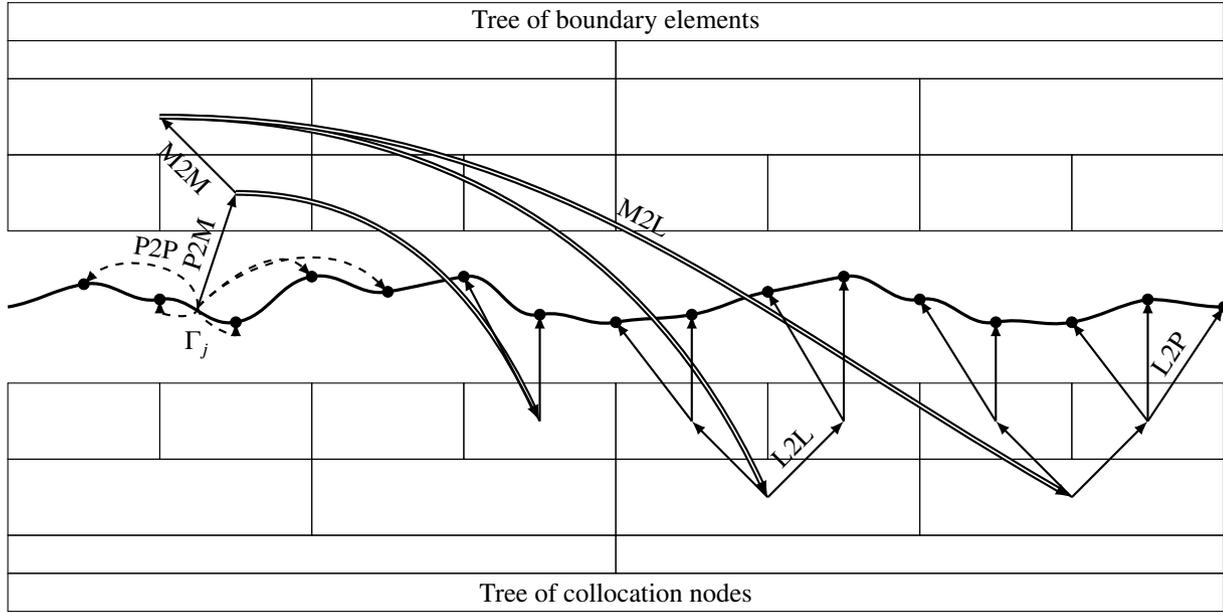


FIGURE 7 Schematic of 1D tree structure of how contributions of element Γ_j to the BIE, Eq. 3, are handled in the FMM for all boundary points. Local interactions (i.e., direct BIE computations) are computed as P2P terms (dashed) whereas distant interactions are computed with some combination of P2M, M2M, M2L, L2L, and L2P terms, while ensuring an $O(\theta^P)$ maximum error. Note that top layers are not used since they are always too close together according to the multipole acceptance criterion (MAC).

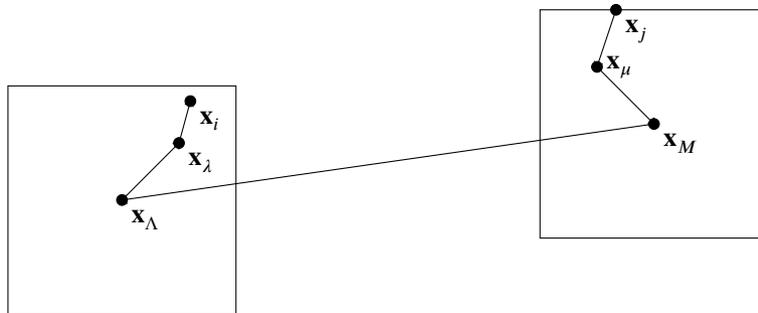


FIGURE 8 FMM structure of collocation points, showing how distance vectors are decomposed into five parts; the error on the Taylor series approximation of the Green's function is bounded by $O(\theta^P)$, with $\theta = (\|\mathbf{x}_i - \mathbf{x}_\Lambda\| + \|\mathbf{x}_M - \mathbf{x}_j\|) / \|\mathbf{x}_\Lambda - \mathbf{x}_M\|$.

in the neighborhood of intermediate points \mathbf{x}_Λ and \mathbf{x}_M , can be approximated by a polynomial expansion of order P ,

$$G(\mathbf{x}_i - \mathbf{x}_j) = \sum_{m_x=0}^{P-1} \sum_{m_y=0}^{P-m_x-1} \sum_{m_z=0}^{P-m_x-m_y-1} ([x_i - x_\Lambda] + [x_M - x_j])^{m_x} ([y_i - y_\Lambda] + [y_M - y_j])^{m_y} ([z_i - z_\Lambda] + [z_M - z_j])^{m_z} \frac{\partial_x^{(m_x)} \partial_y^{(m_y)} \partial_z^{(m_z)} G(\mathbf{x}_\Lambda - \mathbf{x}_M)}{m_x! m_y! m_z!} + O(\theta^P). \quad (31)$$

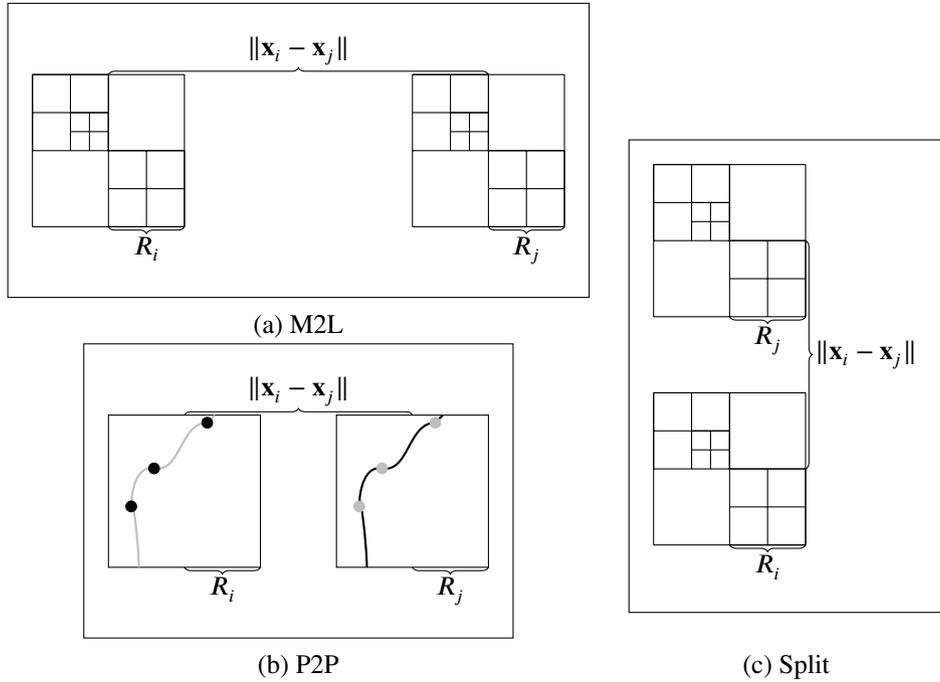


FIGURE 9 Dual-tree traversal: Given a pair of cells from the octree of elements and collocation nodes, with cell centers \mathbf{x}_i and \mathbf{x}_j , and radii R_i and R_j , then starting from the highest level: (a) if two cells are far apart ($\|\mathbf{x}_i - \mathbf{x}_j\|/(R_i + R_j) > \theta$), then the M2L kernel will be applied; (b) if two cells are close together ($\|\mathbf{x}_i - \mathbf{x}_j\|/(R_i + R_j) < \theta$) and do not consist of more sub-cells, then the direct BIE will be applied; and finally (c) if the two cells are close together but include additional levels to the octree, then the larger of the two cells will be split (if the two cells are of equal size, the target cell, C_i will be the one that is split).

Applying the binomial theorem and swapping loop orders, we get,

$$\begin{aligned}
 G(\mathbf{x}_i - \mathbf{x}_j) &= \sum_{k_x=0}^{P-1} \sum_{k_y=0}^{P-k_x-1} \sum_{k_z=0}^{P-k_x-k_y-1} \frac{(x_i - x_\Lambda)^{k_x} (y_i - y_\Lambda)^{k_y} (z_i - z_\Lambda)^{k_z}}{k_x! k_y! k_z!} \\
 &\quad \sum_{m_x=0}^{P-k_x-1} \sum_{m_y=0}^{P-k_y-m_x-1} \sum_{m_z=0}^{P-k_z-m_x-m_y-1} \left(\partial_x^{(m_x+k_x)} \partial_y^{(m_y+k_y)} \partial_z^{(m_z+k_z)} G(\mathbf{x}_\Lambda - \mathbf{x}_M) \right) \\
 &\quad \frac{(x_M - x_j)^{m_x} (y_M - y_j)^{m_y} (z_M - z_j)^{m_z}}{m_x! m_y! m_z!} + O(\theta^P), \tag{32}
 \end{aligned}$$

in which the first fraction is referred to as *local expansion* whereas the last two lines are referred to as *multipole expansion*. [See e.g., Dehnen⁷⁰ for a more compact form.] The truncation error over the $P(P+1)(P+2)/6$ terms of this expansion is function of the MAC, θ . Importantly, with this expansion, much of the computation of the Green's function no longer depends (and varies with) the original points \mathbf{x}_i and \mathbf{x}_j , but instead can be computed once for some intermediate points \mathbf{x}_Λ and \mathbf{x}_M .

2.4.3 | Application of FMM to BIE evaluation

In order to solve the BEM problem, we use an iterative approach (e.g., GMRES), and therefore ultimately the problem can be reduced to evaluating Eq. 29 for different guesses of ϕ and ϕ_n for a given geometry. We can split this equation into near- and

far-field contributions as (i and $j = 1, \dots, N$),

$$\alpha(\mathbf{x}_i)\phi(\mathbf{x}_i) = \left([K_{ij}^d]_{\text{near}} \frac{\partial \phi}{\partial n} \Big|_j - [K_{ij}^n]_{\text{near}} \phi_j \right) + \left([K_{ij}^d]_{\text{far}} \frac{\partial \phi}{\partial n} \Big|_j - [K_{ij}^n]_{\text{far}} \phi_j \right) \quad (33)$$

in which the near-field BEM influence coefficients are computed the classical way by direct integration of the Green's function and the far-field ones are approximated based on using Eq. 32. The complete solution will be discussed in section 2.5, but to apply the FMM to a single evaluation of the BIE, the work is divided into a number of stages:

Initialization of octree

As discussed above, the boundary Γ is discretized into a set of M_Γ elements, Γ_e^k , but is also made up of a set of N_Γ collocation nodes, \mathbf{x}_j . Each set is then subdivided into a number of subsets, equal to the number of computer processes available, by applying an orthogonal recursive bisection (ORB;⁷⁴; Fig. 10). [This is in contrast to hashed octrees⁷⁵, the other main approach for parallelization of tree-codes.] At each subdivision, an ORB determines which dimension (i.e., x , y , or z) is longest for the current subset, and further divides the subset in two pieces along this direction, to have an approximately equal number of nodes in each new subset. In our implementation, we assume that the number of processors available is a multiple of 2, but Yokota et al.⁷⁶ have shown how this can be extended to an arbitrary number of processors. For a given processor, we thus have two subsets, $\{\Gamma_e^k\}^{CPU=j}$ and $\{x_j\}^{CPU=j}$, and each processor will have similar amounts of computations.

Next, we construct two cubes, or bounding boxes, for each subset, which encompass all of the collocation nodes or all of the boundary elements attached to a processor. We then split each box into eight smaller boxes (i.e., an octree) and continue subdividing this structure until the number of nodes in a box is less than a pre-set critical value, N_{crit} . This parameter is a tunable parameter (which here we take to be 16). This parameter determines the number of levels to the octree created for the FMM; a small N_{crit} value will result in more levels, whereas a large value will result in fewer. One can see typical values for different distributions of points in Abdul-Jabbar et al.⁷⁷. For any given box, we store the center, size, and either the list of ‘‘child’’ boxes or collocation nodes or elements stored.

Noting that boundary elements are not point particles, after constructing the octree, we slightly increase the size of each cell to make sure that the MAC (the distance criterion) is correctly applied. At this stage, we do not yet have any information about which parts of the domain boundary are near or far, and we do not know anything about the multipole expansions; these are computed next.

Upward pass

Equations used to compute the far-field BEM influence coefficients in the FMM are derived from introducing Eq. 32 into Eq. 27.

Specifically, over element Γ_e^j belonging to the octree box of center \mathbf{x}_μ , the boundary integrals are computed based on the polynomial expansion in the last line of Eq. 32, as,

$$\mathbf{M}_j^{m_x, m_y, m_z}(\mathbf{x}_\mu) = \frac{1}{m_x! m_y! m_z!} \int_{\Gamma_e^j} \left(\frac{\partial \phi}{\partial n} - \phi n_x \frac{\partial}{\partial x} - \phi n_y \frac{\partial}{\partial y} - \phi n_z \frac{\partial}{\partial z} \right) \left((x_\mu - x)^{m_x} (y_\mu - y)^{m_y} (z_\mu - z)^{m_z} \right) d\Gamma, \quad (34)$$

which is referred to as a P2M (point-to-multipole) operation (Fig. 7). The first time this is done on a new geometry, the integrals are expressed in terms of their nodal values:

$$\mathbf{M}_j^{m_x, m_y, m_z}(\mathbf{x}_\mu) = \frac{1}{m_x! m_y! m_z!} \int_{\Gamma_{\xi, \eta}} \left(S_k(\xi, \eta) \frac{\partial \phi_k}{\partial n} - S_k(\xi, \eta) \phi_k \frac{\partial}{\partial n} \right) \left((x_\mu - x)^{m_x} (y_\mu - y)^{m_y} (z_\mu - z)^{m_z} \right) d\xi d\eta, \quad (35)$$

and therefore each P2M integral only has to be computed a single time for a single geometry.

The distance vector of Eq. 30 is then further decomposed as, $\mathbf{x}_M - \mathbf{x}_j = (\mathbf{x}_M - \mathbf{x}_\mu) + (\mathbf{x}_\mu - \mathbf{x}_j)$ (Fig. 8), in which point \mathbf{x}_μ is close to element Γ_e^k and thus also to point \mathbf{x}_M . Based on this, a formula is derived for combining multipole expansions from different boxes (referred to as M2M translation). Thus, applying the binomial theorem together with the additional distance decomposition yields,

$$\mathbf{M}_j^{m_x, m_y, m_z}(\mathbf{x}_M) = \sum_{k_x=0}^{m_x} \sum_{k_y=0}^{m_y} \sum_{k_z=0}^{m_z} \frac{1}{(m_x - k_x)! (m_y - k_y)! (m_z - k_z)!} \mathbf{M}_j^{k_x, k_y, k_z}(\mathbf{x}_\mu) \left((x_M - x_\mu)^{(m_x - k_x)} (y_M - y_\mu)^{(m_y - k_y)} (z_M - z_\mu)^{(m_z - k_z)} \right). \quad (36)$$

The P2M and M2M operations defined by Eqs. 34 and 36 are used to apply the respective multipole expansions to each boundary elements box of the octree, which is referred to as *upward pass*; starting with the largest box, C_j , the following algorithm is applied,

Algorithm 1 Upward pass (P2M, M2M)

- | | | |
|-----|--|---|
| 1: | procedure UPWARD PASS(C_j) | ▷ Get multipole expansion for all levels of C_j |
| 2: | if C_j has no child cells then | |
| 3: | Apply Eq. 34 | ▷ Compute multipole expansion with P2M |
| 4: | else | |
| 5: | for all child cells, C_{j2} do | |
| 6: | UPWARD PASS(C_{j2}) | |
| 7: | Compute Eq. 36 | ▷ Add to multipole expansion with M2M |
| 8: | end for | |
| 9: | end if | |
| 10: | return | |
| 11: | end procedure | |
-

Traversal

To convert the multipole expansions into local expansions, based on the second line of Eq. 32), we apply,

$$\mathbf{L}_j^{k_x, k_y, k_z}(\mathbf{x}_\Lambda) = \sum_{m_x=0}^{P-k_x-1} \sum_{m_y=0}^{P-k_y-m_x-1} \sum_{m_z=0}^{P-k_z-m_x-m_y-1} \mathbf{M}_j^{m_x, m_y, m_z}(\mathbf{x}_M) \left(\partial_x^{(m_x+k_x)} \partial_y^{(m_y+k_y)} \partial_z^{(m_z+k_z)} G(\mathbf{x}_\Lambda - \mathbf{x}_M) \right). \quad (37)$$

Note, Eq. 37 shows that arbitrary derivatives of the Green's function must be computed, which is performed here in a recursive fashion, following Visscher and Apalkov⁷⁸ or Zhang and Haas⁷⁹.

At this stage (Fig. 9), the distance criterion is applied to verify whether the approximation made in Eq. 37 is relevant. Considering two boxes in the octree, say C_i and C_j , of centers \mathbf{x}_i and \mathbf{x}_j , and radii R_i and R_j , the distance criterion is applied as detailed in Algorithm 2.

Algorithm 2 Dual Tree Traversal (M2L, P2P)

1:	procedure TRAVERSAL(C_i, C_j)	▷ Get local expansion for all levels of C_i
2:	if ($\ \mathbf{x}_i - \mathbf{x}_\Lambda\ + \ \mathbf{x}_M - \mathbf{x}_j\ / \ \mathbf{x}_\Lambda - \mathbf{x}_M\ < \theta$) then	
3:	Compute Eq. 37	▷ Compute local expansion with M2L
4:	else if both C_i and C_j have no child cells then	
5:	Store i and j indices in a list	▷ Remember which interactions are local
6:	else	
7:	Split C_i and/or C_j and traverse all child cells	▷ Fill lower levels of octree if needed
8:	end if	
9:	return	
10:	end procedure	

This step is referred to as *dual tree traversal*, at the end of which: (i) all the multipole expansions in the octree of boundary elements have been converted into local expansions in the octree of collocation nodes; (ii) the distance criterion has determined which parts of the domain are close to each other, for computing the near-field (local) sparse matrices $[K_{ij}^n]_{\text{near}}$ and $[K_{ij}^d]_{\text{near}}$.

Inter-processor communication

As indicated before, the parallel FMM solution over a series of sub-domains (or boxes) is divided into multiple processors (Fig. 10), which requires that information be efficiently exchanged between processors, in particular, the multipole expansions stored in each octree. In the previous phase, the dual-tree traversal, the far-field interaction between the elements and collocation nodes were stored on a given processor, $CPU = j$, that is, $\{\Gamma_e^k\}^{CPU=j}$ and $\{x_i\}^{CPU=j}$. Now the interaction between two different processors are considered, that is, $\{\Gamma_e^k\}^{CPU=j_1}$ and $\{x_i\}^{CPU=j_2}$, for all processors.

In order to prevent the amount of communication overhead from becoming prohibitive with an increasing number of processors, the amount of exchanged information can be *a priori* limited using a simplified distance criterion. Thus, for a remote

processor with a set of elements, $\{\Gamma_\epsilon^k\}^{CPU=j_1}$, a subset of the octree already produced is created, called the *local essential tree* (LET), which is transmitted between processors, instead of the entire octree, based on R , the distance to the edge of a domain considered by the remote processor.

For each part of the octree, the influence coefficients can be approximated by only one multipole expansion if it satisfies the criterion $2R_j/R < \theta$ (Fig. 10). In this case, the cell is distant enough that no additional detailed contribution is needed; i.e., finer details would not be used during the dual tree traversal, so the amount of information transmitted is reduced, but no approximation is applied. The dual tree traversal is then applied between all processors, but the amount of information transmitted is substantially reduced for more distant subdomains as compared to adjacent ones.

Downward pass

Similar to the upward pass, a method is developed to combine local expansions from different cells, by decomposing the distance vector as, $\mathbf{x}_i - \mathbf{x}_\Lambda = (\mathbf{x}_i - \mathbf{x}_\lambda) + (\mathbf{x}_\lambda - \mathbf{x}_\Lambda)$ (Fig. 8). Applying the binomial theorem, and swapping the loop order yields the L2L operation,

$$\mathbf{L}_j^{m_x, m_y, m_z}(\mathbf{x}_\lambda) = \sum_{k_x=m_x}^{P-1} \sum_{k_y=m_y}^{P-k_x-1} \sum_{k_z=m_z}^{P-k_x-k_y-1} \frac{1}{(k_x - m_x)! (k_y - m_y)! (k_z - m_z)!} ((x_\lambda - x_\Lambda)^{(k_x - m_x)} (y_\lambda - y_\Lambda)^{(k_y - m_y)} (z_\lambda - z_\Lambda)^{(k_z - m_z)}) \mathbf{L}_j^{k_x, k_y, k_z}(\mathbf{x}_\Lambda). \quad (38)$$

which expresses the combined contribution of local expansion terms. This contribution is converted into the far-field influence matrices, through the L2P operation,

$$[K_{ij}^d]_{\text{far}} \frac{\partial \phi}{\partial n} \Big|_j - [K_{ij}^n]_{\text{far}} \phi_j = \sum_{k_x=0}^{P-1} \sum_{k_y=0}^{P-k_x-1} \sum_{k_z=0}^{P-k_x-k_y-1} \frac{(x_i - x_\Lambda)^{k_x} (y_i - y_\Lambda)^{k_y} (z_i - z_\Lambda)^{k_z}}{k_x! k_y! k_z!} \mathbf{L}_j^{k_x, k_y, k_z}(\mathbf{x}_\lambda) \quad (39)$$

As a check of the various FMM contributions, we note that the combinations of Eqs. 34 to 39 is equivalent to substituting the original Taylor series expansion for the Green's function, Eq. 32, into Eq. 3. More specifically, the downward pass is computed as shown in Algorithm 3.

2.5 | Execution of the parallel FMM-BEM solution

At each time step, when a new domain geometry is considered, a far-field computation (Sec. 2.4.3) is first performed with $\phi = 1$ specified over the entire boundary, for which one has $\phi_n = 0$. In this process, as mentioned above, the dual-tree traversal will produce a list of near-field interactions, for which the standard BEM matrices are directly computed. Referring back to Eq. 33, we therefore compute α by the rigid mode method as,

$$\alpha(\mathbf{x}_i) = - \sum_j \left([K_{ij}^n]_{\text{near}} + [K_{ij}^n]_{\text{far}} \right) \quad (40)$$

Algorithm 3 Downward pass (L2L, L2P)

```

1: procedure DOWNWARD PASS( $C_i$ ) ▷ Get far-field influence
2:   if  $C_i$  has no child cells then
3:     Compute Eq. 39 ▷ Evaluate local expansion with L2P
4:   else
5:     for all child cells,  $C_{i2}$  do
6:       Compute Eq. 38 ▷ Add to local expansion with L2L
7:       DOWNWARD PASS( $C_{i2}$ )
8:     end for
9:   end if
10:  return
11: end procedure

```

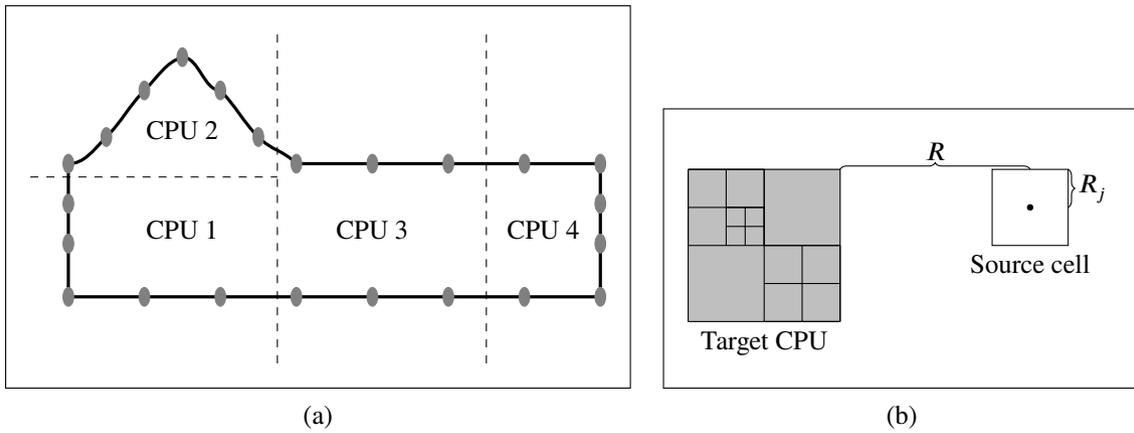


FIGURE 10 Parallelization, showing: (a) orthogonal recursive bisection, whereby each processor ends up with a similar number of collocation nodes or boundary elements; and (b) local essential tree criterion; for any given cell, if it is a leaf (containing boundary elements but no smaller cells), the element information is sent to the remote process (P2P interaction expected); if not, but $2R_j/R < \theta$, then the cell is far from the target partition, so the multipole expansion is sent; if the criterion is not satisfied, the source cell is split and the same procedure is applied to the eight smaller cells.

The computation of the near-field sparse matrices $[K_{ij}^d]_{\text{near}}$ and $[K_{ij}^n]_{\text{near}}$ is described next.

Assembly phase

After storing the near-field interaction list, $[K_{ij}^n]_{\text{near}}$ and $[K_{ij}^d]_{\text{near}}$ are computed using the standard BEM approach, which consists in directly integrating the boundary integral equation over each considered boundary element, based on the free space Green's function. However only $O(N_f)$ computations are required instead of $O(N_f^2)$, as these coefficients are stored as sparse matrices.

As in the classic BEM, in order to apply boundary conditions, Eq. 29 is further separated depending on the type of boundary condition into knowns (those with an overbar), and unknowns (those without):

$$\{C_{pl} + K_{pl}^n\}\phi_p - K_{gl}^d \frac{\partial \phi}{\partial n} \Big|_g = K_{pl}^d \overline{\frac{\partial \phi}{\partial n} \Big|_p} - \{C_{gl} + K_{gl}^n\} \overline{\phi_g} \quad (41)$$

where C is a diagonal matrix made of coefficients α_l , with $l = 1, \dots, N_\Gamma$; $g = 1, \dots, N_g$ refers to nodes with a Dirichlet condition on the free-surface boundary, Γ_f , and $p = 1, \dots, N_p$ refers to nodes on other boundaries with a Neumann condition. If we divide Eq. 41 into near- and far-field components, as in Eq. 33, we can also write it as:

$$\begin{aligned} & \{C_{pl} + [K_{pl}^n]_{near} + [K_{pl}^n]_{far}\} \phi_p \\ & - \{[K_{gl}^d]_{near} + [K_{gl}^d]_{far}\} \frac{\partial \phi}{\partial n} \Big|_g = \{[K_{pl}^d]_{near} + [K_{pl}^d]_{far}\} \overline{\frac{\partial \phi}{\partial n}} \Big|_p \\ & - \{C_{gl} + [K_{gl}^n]_{near} + [K_{gl}^n]_{far}\} \overline{\phi}_g \end{aligned} \quad (42)$$

After already determining the near-field coefficients, $[K_{ij}^n]_{near}$ and $[K_{ij}^d]_{near}$, and the coefficients α_l , the right-hand side in the above equation requires a second far-field FMM calculation, as part of the assembly phase.

Iterative solution with GMRES

After the assembly phase is completed, the BEM algebraic system (Eq. 33) is solved with the iterative solver GMRES. At each iteration, the error due to a solution “guess” is computed by applying the upward pass, traversal, and downward pass procedures detailed above, to compute the approximate matrix-vector products.

Convergence of the iterative solution is deemed attained when Eq. 3 is solved to within some prescribed error ϵ_{GMRES} , taken here to be 10^{-9} . When applying GMRES on a parallel CPU cluster, it is critical to limit the number of iterations and, as a result, the amount of inter-processor communication required, which penalizes the solution time. This is done here by applying a simple diagonal preconditioning to the system of equations. It is worth noting that, using all of the local (P2P) interactions that are already stored as a preconditioner is substantially more efficient on a single processor, but special attention is still required to minimize communication between processors. An assessment of the scalability of the current BEM-FMM solution is presented in the next section.

2.6 | Scalability of the BEM-FMM solution

The NWT BEM-FMM solver performance is first assessed for a steady state linear wave problem, for which an analytical solution is known (i.e., no time stepping is involved). Similar to Grilli and Svendsen⁶⁶, Grilli et al.⁴, or Shao and Faltinsen²⁰, a simple cuboid computational domain is used (Fig. 11a), for which mixed Dirichlet-Neumann boundary conditions are specified: (i) a Dirichlet condition for the potential: $\phi = \cos(kx)$, on the free surface; (ii) a Neumann boundary condition $\phi_n = 0$, on the lateral sides and on the bottom; and (iii) a Neumann boundary condition corresponding to $\mp \partial \phi / \partial x = \pm \sin(kx) \cosh(k(z + h)) / \cosh(kh)$, on the leftward and rightward boundaries. This problem represents an horizontally oscillatory potential, that is decaying with depth, for a linear wave of amplitude $a = \omega/g$, with angular frequency ω and wavenumber $k = 2\pi/\lambda$ satisfying the linear dispersion relationship, $\omega^2/g = k \tanh kh$ ⁸⁰.

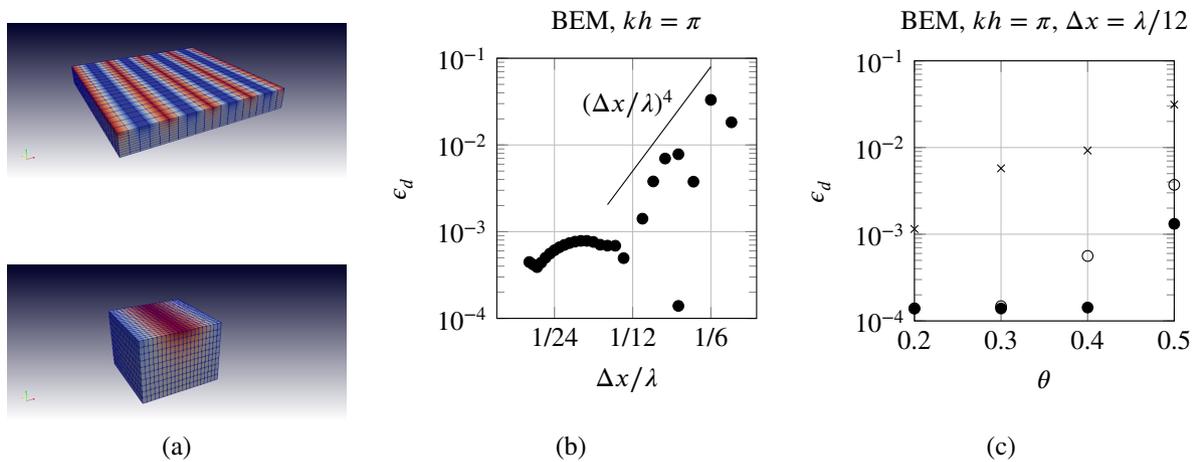


FIGURE 11 Convergence study of BEM numerical error for analytical wave benchmark: (a) two examples of computational domains, with 10 B-spline elements used vertically, and 20 B-spline elements used in each horizontal direction; (b) numerical error as a function of relative horizontal mesh size $\Delta x/\lambda$ without using the FMM; and (c) numerical error as a function of multipole acceptance criterion, θ , for different orders of FMM expansion (6 – \times , 10 – \circ , 14 – \bullet).

For this problem, it is easy to show that, on the free surface ($z = 0$), $\phi_z = \phi_n = \phi k \tanh kh$. Hence, the numerical error of the BEM-FMM solution is defined as the maximum of, $\epsilon_d = \phi_n - \phi k \tanh kh$, a free-surface error, which is assessed in results obtained both on a single processor as a function of the number of nodes N_Γ , with and without using the FMM, and then using the FMM on a cluster of M_C processors with distributed memory, for a given N_Γ value.

As the FMM is an approximate method, the error of the BEM solution is first assessed without using the FMM. Considering a domain with 20 B-spline elements horizontally in each direction, and 10 B-spline elements vertically, the geometry is scaled horizontally by maintaining a constant $kh = \pi$, which for linear waves corresponds to a deep water solution. Hence, by reducing the relative horizontal mesh size $\Delta x/\lambda$, one increases the number of nodes per wavelength used to solve the benchmark problem, and hence increases the numerical accuracy within the same total number of nodes N_Γ in the domain discretization. Fig. 11b shows that without the FMM the maximum error over the free-surface rapidly converges with a reduction in relative horizontal mesh size. As expected from the cubic elements that are used, this decrease in error is near order 4 with relative horizontal mesh size. However, there is a limit to the reduction in error (around 8×10^{-4}), due to other numerical parameters being kept constant, such as the vertical grid size $\Delta z/\lambda$.

The same problem is solved next using the FMM, and assuming different multipole acceptance criteria θ and order of expansion P . If θ is zero, then the result will be identical to the non-FMM model (though with additional computations to initiate the multipole expansions, which then would not be used), whereas if θ is large, most calculations will depend on the FMM approximation, with typically the ideal value of θ being between 0.1 and 1.0. As the FMM truncation error is $O(\theta^P)$, the same error can be obtained with different orders of expansion depending on the selected θ value. Note, the ideal choice of θ and P depends on the implementation and computer hardware. Fig. 11c shows that, for a low-order expansion, e.g., $P = 6$, significant

errors occur even for a multipole acceptance criterion as low as $\theta = 0.2$. Alternatively, for a high-order expansion, e.g., $P = 14$, the error in the free-surface solution is almost indistinguishable from that of the regular BEM solution. One could, however, obtain small (and similar) errors using $(\theta = 0.35, P = 10)$ or $(\theta = 0.45, P = 14)$; one would need a larger test case in order to see what effect these parameters have on CPU time.

Now, considering a similarly shaped domain which, instead of varying $\Delta x/\lambda$, this value is fixed at $\Delta x/\lambda = 1/10$, keeping $kh = \pi$, and the grid is discretized with a varying number of nodes N_Γ , Fig. 12a compares the CPU time for solving this problem on a single processor with the BEM model, with and without the FMM, as a function of N_Γ and for identical maximum errors in the solution of the algebraic system of equations (here a 10^{-12} relative error in the iterative solution with GMRES). Computations are performed on Porthos, an IBM NeXtScale nx360M5, using Xeon E5-2697v3 14C 2.6GHz processors, with an Infiniband FDR interconnect (ranked 407 on the TOP500 list as of June 2017). For small grids (10^3 nodes or less), as expected, the performance is not dramatically different, either with or without the FMM, but for 5×10^4 nodes or more, the classic BEM on a single processor becomes impractical, being at least one order of magnitude slower than when using the FMM. Compared to earlier works by¹⁹ or⁴³, the present method appears to be substantially faster, although this can be attributed mostly to improvements in processor performance and the differences in problem setup. Consistent with previously published results, we find a numerical complexity $O(N_\Gamma^2)$ without the FMM and $O(N_\Gamma^{1.2})$ with the FMM. The solution time with the FMM thus does not perfectly scale as $O(N_\Gamma)$, which may be because the number of GMRES iterations increases very slightly with problem size. A similar trend is observed for different order expansions, but lower-order expansions are slightly faster for the conditions tested here.

Fig. 12b shows the speed-up of the FMM solution as a function of the number of CPUs M_C , for a fixed number of nodes $N_\Gamma = 22,826$. For small numbers of processors, the problem scales very well, but the parallel efficiency drops off when using more than 16 processors. Although this performance is not ideal, it is not unexpected compared to recent works using this type of algorithm. The present speed-up is an order of magnitude better than that achieved in earlier work by Nimmala et al.⁴⁷ and consistent with Wang et al.⁸¹, who solved the Stokes equation using a similar computational approach and were able to achieve a 13 times speedup using 32 processors, for a problem with 36,504 nodes. A decrease in parallelization efficiency as the number of processors approaches the number of unknowns was shown in the first parallelization attempt of the FMM by Greengard and Gropp³³ as well.

Given that the first step of the parallel FMM solution is to assemble a local octree, it is logical that the scalability would be maintained when there are large numbers of collocation nodes per processor. Indeed results such as Yokota et al.'s⁴⁹ or Cao et al.'s⁸² show impressive scaling for thousands of processors, but using billions of particles, which only need to be solved a single time. With a fully nonlinear time-domain NWT, Laplace's equation must be solved multiple times per time step, and for thousands of time steps. Nevertheless, some modifications to our algorithm could be implemented to improve the scalability

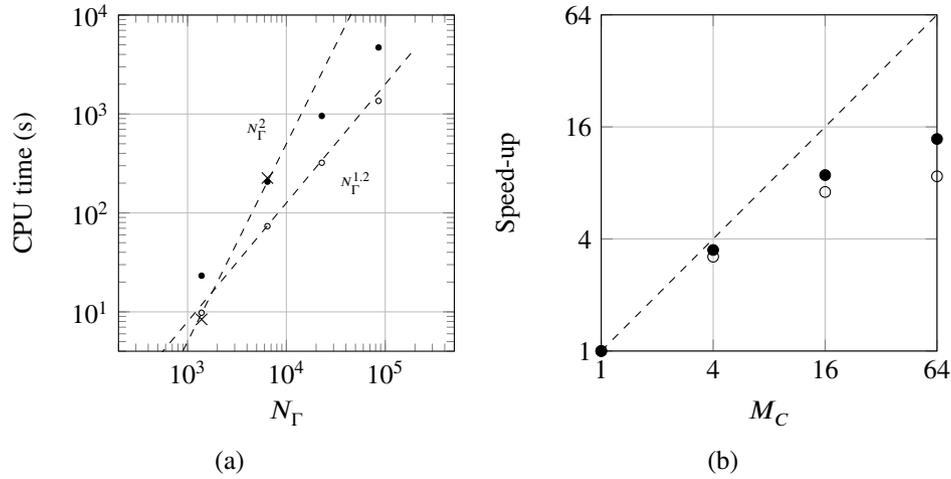


FIGURE 12 BEM solution of a mixed Dirichlet-Neumann problem in a box-like domain (Fig. 11a): (a) CPU time on a single CPU as a function of the number of nodes N_Γ , without the FMM (×), with the FMM, using $P = 10, \theta = 0.35$ (○), and with FMM, using $P = 14, \theta = 0.45$ (●); and (b) speed-up of the parallel solution, for $N_\Gamma = 22,826$, as a function of the number of CPUs M_C , using the FMM.

of the parallel FMM solution. For instance, our use of a diagonal preconditioner is overly simplistic; most production codes make use of a more sophisticated preconditioner such as ILU(0), or an algebraic multigrid approach, which converges faster. The NWT code is being refactored to consider more sophisticated approaches for future work.

2.7 | Filtering of the waterline

As will be seen in applications, the present FNPF-NWT yields a stable solution for the propagation of steep nonlinear waves. This is unlike some earlier nonlinear potential flow models (e.g., ^{83,20}), where sawtooth instabilities both occurred and required filtering over the entire surface. However, it is observed that, in some circumstances such as near and on the intersection of the free surface with surface piercing objects, instabilities are triggered for steep waves, that require filtering. This appears to be a result of nonlinear wave structure interactions that would normally result in local wave breaking or other physical effect.

To counteract this effect, a filter is applied at each time step to the three closest points (radially) near the waterline, based on the method introduced by Longuet-Higgins and Cokelet²⁶. In essence, this filter is designed to have a minimal effect on most waves, except those with a wavelength that is two elements long. Considering a free surface property f (i.e., ϕ, ζ), the filter applies the following 7-point formula, in the angular direction,

$$f_i = \frac{1}{32}(-f_{i-3} + 9f_{i-1} + 16f_i + 9f_{i+1} - f_{i+3}) \quad (43)$$

where i denotes the index of the grid points located around the perimeter of the surface-piercing body.

2.8 | Wave generation and absorption

Generation

Similar to the method introduced by Grilli and Horrillo's⁸ in their 2D-NWT, fully nonlinear periodic waves are generated here by specifying their kinematics along the NWT's leftward boundary (Fig. 1), based on the streamfunction wave theory⁸⁴. This theory was extended by Dalrymple⁸⁵ to include a depth-uniform current U , which Grilli et al.⁸ iteratively computed to balance the period-averaged mass transport velocity (i.e., Stokes drift), thus generating what they referred to as *zero-mass-flux* streamfunction waves, that do not in average modify the NWT volume (except for numerical errors). [Note, Fenton³ later showed that this current can be directly rather than iteratively computed.] More specifically, in this method, the streamfunction in depth h_0 is defined as a series expansion (with J terms) of trigonometric and hyperbolic functions, in a coordinate system moving at the wave celerity c ,

$$\psi(k(x - ct), z) = \sum_{j=1}^J \left\{ X(j) \sinh\{jk(h_0 + z)\} \cos\{jk(x - ct)\} \right\} - (U - c)z \quad (44)$$

based on which both velocity and acceleration can be easily computed over depth, and specified as Neumann boundary conditions for the ϕ and ϕ_t BIE problems. Note that, proceeding in a similar manner, fully nonlinear irregular incident waves could be specified on the NWT lateral boundary, provided these are described by a relevant nonlinear wave theory (e.g., high-order spectral method⁸⁶).

To prevent instabilities from being triggered at the wavemaker boundary at the beginning of the simulation, the velocity field is gradually ramped-up, say over one wave period T , by multiplying it by a tanh-like tapering function, $D(t)$; details can be found in Grilli and Horrillo⁸.

Absorption

On the far end of the NWT (e.g., rightward boundary in Fig. 1), also similar to Grilli and Horrillo's 2D work⁸, reflection is eliminated by damping waves in an absorbing beach of length L_{AB} (AB; Fig. 1). Although many methods have been proposed for doing so (see, e.g., the recent comparison of methods by Kim et al.⁸⁷), similar to Dombre et al.⁵⁵, dissipative terms are only added to the dynamic free surface boundary condition as,

$$\begin{aligned} \frac{\delta\zeta}{\delta t} &= \frac{\partial\phi}{\partial z} - \frac{\partial\phi}{\partial x} \frac{\partial\zeta}{\partial x} - \frac{\partial\phi}{\partial y} \frac{\partial\zeta}{\partial y} \\ \frac{\delta\phi}{\delta t} &= -g\zeta - \frac{1}{2} \nabla\phi \cdot \nabla\phi + \frac{\partial\zeta}{\partial t} \frac{\partial\phi}{\partial z} - \nu(x, y) \left(\frac{\partial\phi}{\partial z} - w_{ref} \right) \end{aligned} \quad (45)$$

with $\nu(x, y)$ a damping coefficient (Fig. 1) and w_{ref} a reference vertical velocity, either taken to be zero in rectangular domains (e.g., for wave propagation tests) or, for cylindrical domains, being the vertical velocity of the wave input (e.g., streamfunction wave theory); in this case, the AB is sometimes referred to as a relaxation zone.

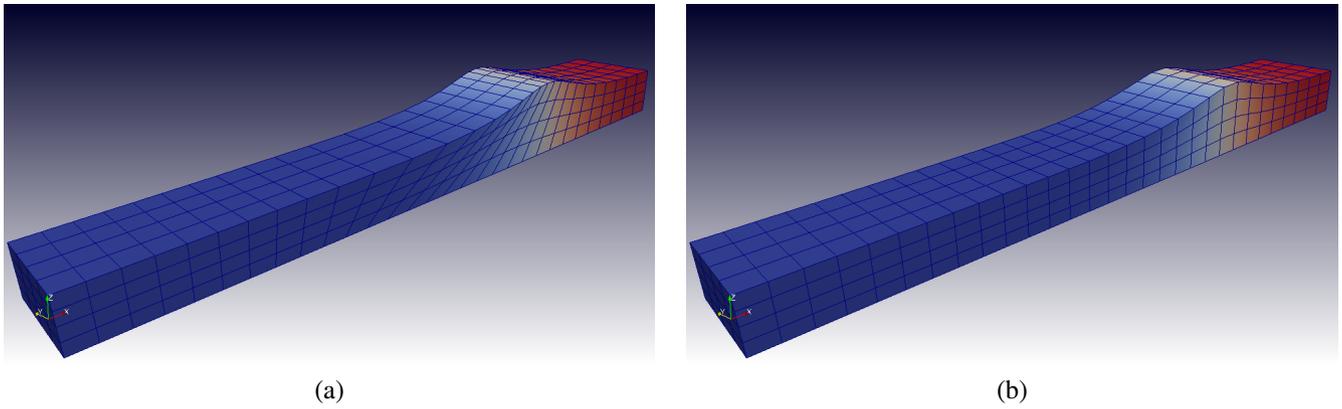


FIGURE 13 Propagation in the 3D-NWT of a large amplitude solitary wave with $H/h_0 = 0.6$, over constant depth h_0 , at $t' = t\sqrt{g/h_0} = 4.0$, with the boundary discretized with: (a) MII elements using fully Lagrangian free-surface updating; and (b) B-spline elements using semi-Lagrangian free-surface updating. Color scale is velocity potential, ϕ .

The damping coefficient, $v(x, y)$, smoothly increases from zero inside the domain, up to some maximum, v_0 :

$$v(l) = \begin{cases} v_0 \left(\frac{l-L_{AB}}{L_{AB}} \right)^2 & l > L_{AB} \\ 0 & l \leq L_{AB} \end{cases} \quad (46)$$

where for a rectangular domain, this just depends on the NWT longitudinal direction (i.e., $l = x$), or for a cylindrical domain, this just depends on the radius (i.e., $l = \sqrt{x^2 + y^2}$).

It should be noted that two more advanced aspects of the AB implemented by Grilli and Horrillo⁸ have not been used here. First, instead of using a constant coefficient v_0 , they optimized its value to best absorb the incident wave power at any given time. Second, on the rightward end wall of the NWT, they used an absorbing piston, as proposed by Clément⁸⁸, which can be more efficient than a dissipation term specified on the free surface (akin to a surface pressure) when absorbing long waves in shallow water (see details in⁸). For the applications presented here, however, both of these features are unlikely to have any significant effect, but in future studies (particularly for shallower water), it may be necessary to consider such methods.

3 | APPLICATIONS

3.1 | Solitary wave propagation

Similar to Grilli et al.⁴, the accuracy and convergence properties of the NWT are first assessed for the transient propagation of a nonlinear solitary wave over constant depth. The NWT has a length 15 times the water depth, a width twice the depth, and the wave height is such that $H/h_0 = 0.6$ (Fig. 13). The initial fully nonlinear solitary wave profile, free surface potential and normal velocity are computed using Tanaka's highly accurate method⁵². The wave crest is initially located at $x' = x/h_0 = 5.5$

and computations are performed until $t' = t\sqrt{g/h_0} = 4.0$ (prime variables are non-dimensional, with length being scaled by depth h_0 and time by $\sqrt{h_0/g}$). For such a permanent form wave, numerical errors are assessed in terms of conservation of wave mass and energy during the wave propagation, as compared to those computed with Tanaka's method (m_0, e_0) to within a 10^{-9} accuracy (note, here the reference level of the potential energy is defined at $z = 0$).

Thus, the numerical error on wave volume is defined as, $\epsilon_m = |(m(t) - m_0)/m_0|$, with the instantaneous solitary wave volume being computed in the NWT as,

$$m(t) = \int_{\Gamma_f} z n_z d\Gamma \quad (47)$$

with Γ_f denoting the free surface boundary and n_z the vertical component of the outward normal vector \mathbf{n} . Similarly, the numerical error on total wave energy is defined as, $\epsilon_e = |(e(t) - e_0)/e_0|$, with the instantaneous total energy being computed as the sum of kinetic and potential energy, as,

$$e(t) = \frac{1}{2}\rho \int_{\Gamma_f} \phi \frac{\partial \phi}{\partial n} d\Gamma + \frac{1}{2}\rho g \int_{\Gamma_f} z^2 n_z d\Gamma \quad (48)$$

In both Eqs. 47 and 48, computing the integrals over the free surface yields the volume and potential energy error corresponding to the wave only, which yields stricter relative errors than those using the entire NWT mass and energy.

Fig. 14 shows variations of these errors as a function of the grid size Δx and mesh Courant number, $C_0 = \Delta t\sqrt{gh_0}/\Delta x$. Overall, errors decrease with both Δx and C_0 , but for the latter, only up to a point, reaching a plateau or even slightly increasing, for small enough C_0 values in the finest grid sizes. This indicates that errors converge with grid size and, in the finer grids are

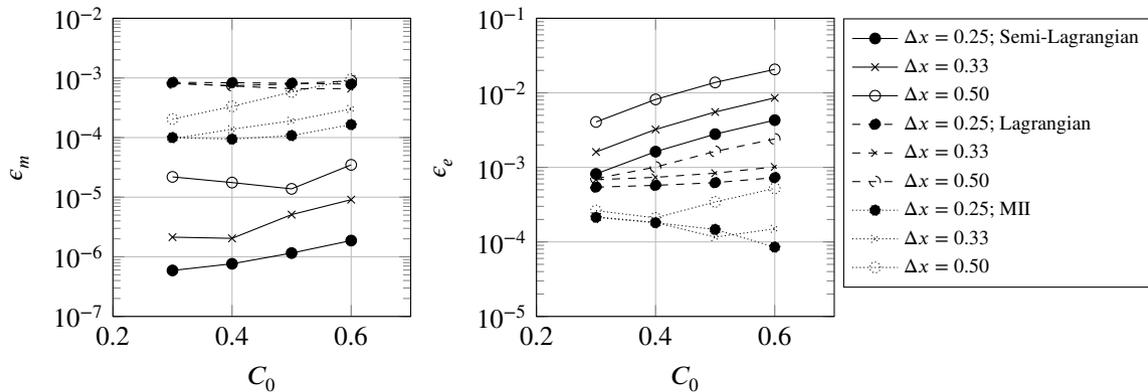


FIGURE 14 Maximum errors in wave volume (left) and energy (right) conservation for the propagation of a solitary wave with amplitude of $H/h = 0.6$ over constant depth, up to $t' = t\sqrt{g/h_0} = 4.0$. Results are provided as a function of mesh size Δx and Courant number $C_0 = \Delta t\sqrt{gh_0}/\Delta x$.

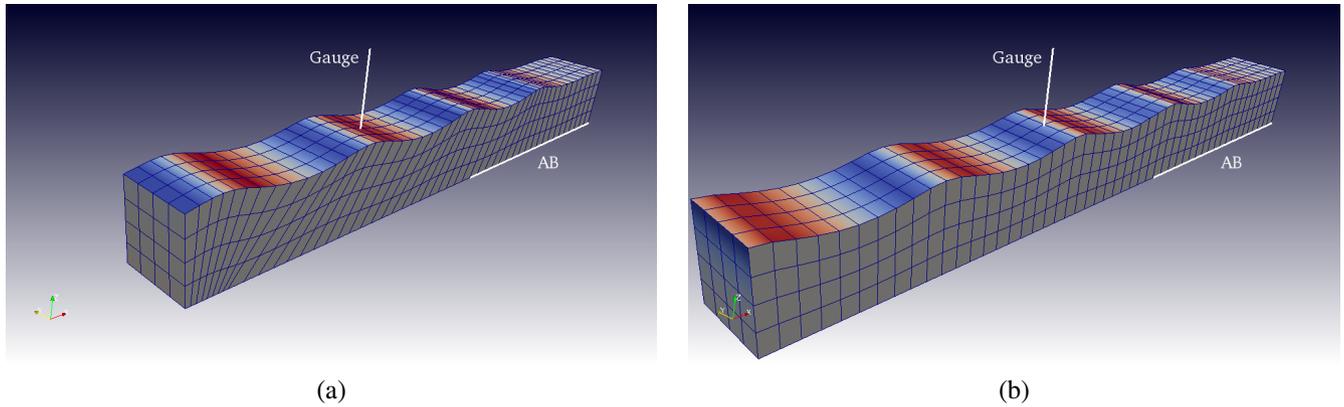


FIGURE 15 Simulations of the propagation of a streamfunction wave with height $H' = 0.15$ and period $T' = 3.5515$ ($L' = 2.07$, $H'/L' = 0.072$). Free surface elevation, shaded by ϕ_n , at $t' \approx 10.0T'$ (after achieving quasi-steady state) computed in: (a) the Lagrangian MII-NWT; and (b) the present semi-Lagrangian NWT with cubic B-splines.

minimum for an optimal mesh Courant number $C_0 \simeq 0.4 - 0.5$; this was also observed by Grilli et al.⁴ in their 3D-FNPF-NWT, using MII elements, whose results are reproduced here. Note that Büchmann⁵⁸ studied NWT stability and showed that the important nondimensional variable is $\beta = \sqrt{\Delta x/g}/\Delta t$, but as $C_0 = \beta^{-1} \sqrt{h_0/\Delta x}$, the two are equivalent.

In Fig. 14, the mass conservation error is substantially smaller when using the semi-Lagrangian time updating in this NWT, rather than the Lagrangian time-updating, but the trend is opposite for energy conservation. One likely reason for this reduction in energy conservation error is that a Lagrangian updating allows nodes to move closer together at the solitary wave crest, thus decreasing Δx as the simulation progresses. For the Lagrangian time-updating, we also see that results using B-splines are similar to those of Grilli et al.⁴ using MII elements, although errors level off near 10^{-4} . But that is expected due to no controls being applied to the velocity of the fluid at double nodes (i.e., intersections between the free-surface and the sidewalls); study of this effect will be subject to an upcoming work.

3.2 | Periodic wave propagation

The propagation of fully nonlinear periodic waves is simulated next, in a NWT similar to that of the previous case (Fig. 15). Similar to Grilli and Horrillo's⁸ 2D simulations, as detailed before, incident waves are generated on the leftward boundary of the tank as (zero-mass-flux) streamfunction waves of height $H' = H/h_0 = 0.15$ and period $T' = T\sqrt{g/h_0} = 3.5515$. With these values, the wavelength is $L' = L/h_0 = 2.07$ and steepness $H'/L' = 0.072$; hence, these are nearly deep water waves with significant nonlinearity.

As in the previous application, wave propagation is simulated in both the new 3D-FNPF-NWT and in Grilli et al.'s earlier NWT⁴, discretized here with 832 quadrilateral (either cubic MII or cubic B-spline) elements (1,070 nodes; grid spacing on the free surface $\Delta x' = 0.2$). Considering the small size of the discretization in this case, the FMM is not used as it would not

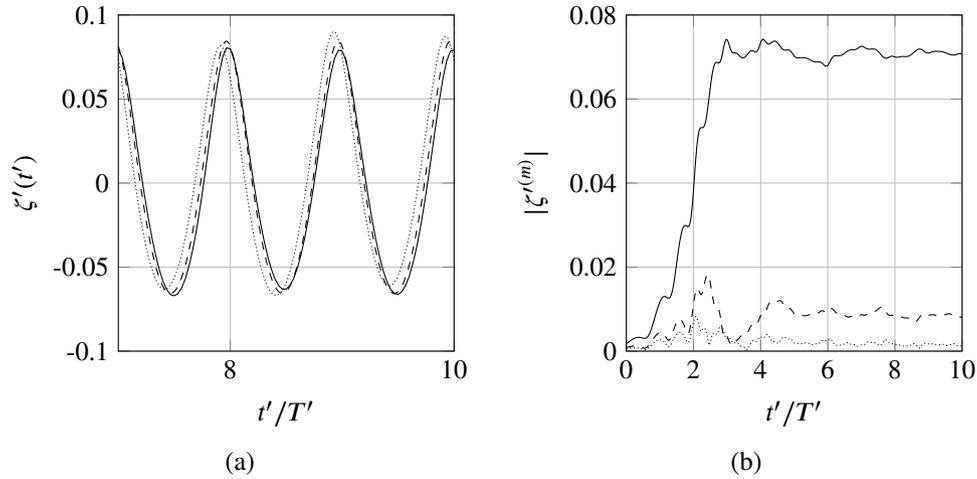


FIGURE 16 Results of streamfunction propagation simulations (case of Fig. 15): (a) time-series of streamfunction waves at $x' = 4.0$ (on the NWT axis), in B-spline NWT (—), theory (---), and MII-NWT (···); and (b) Short-term Fourier transform of B-spline results for 1st (—), 2nd (---), and 3rd (···) order harmonics ($m = 1, 2, 3$) in Eq. 49.

provide any advantage in speed. Larger applications using the parallel FMM are presented in the following sections. For the B-spline NWT, the time step is $\Delta t' = T'/30$, whereas for the MII NWT, the initial time step is $\Delta t' = T'/60$, which is adaptively adjusted to maintain a constant Courant number $C_0 = 0.45$; the finer time step is required to maintain accuracy for the duration of simulations, for the selected parameters. The domain length ℓ' is 10.0, or about five wavelengths long, with the absorbing beach (AB) starting at $x' = 6.0$, with a non-dimensional strength $\nu_o = 0.5$ (Eqs. 45 to 46). The startup duration is taken to be one wave period, $T'_{start} = T'$.

Fig. 15 shows free surface elevations computed in the MII and in the present NWTs, at $t' \approx 10.0T'$ after simulations have reached a quasi-steady state. Fig. 16a shows, for both NWTs, time series of incident waves elevation $\zeta'(t')$ computed at a numerical wave gauge located at $x' = 4.0$ on the tank axis, approximately two wavelengths away from the wavemaker, and one wavelength away from the start of the AB. Results of both numerical simulations are in good agreement with the theory, although slightly larger, with differences likely due to imperfect absorption of incident wave energy in the AB. Fig. 16b shows results of applying a short-time Fourier transform (STFT) to the time series computed in the B-spline NWT (for $m = 1, 2, 3$),

$$\zeta'^{(m)}(t) = \int_{t'}^{t'+T'} \zeta(t) e^{im\omega't} d\tau \quad (49)$$

where $\omega' = 2\pi/T'$ is the angular frequency of incident waves. Initially, due to the ramp-up of wave generation, the first three harmonics amplitudes both increase and oscillate, until simulations reach a quasi-steady state, for $t' > 7T'$ or so. Small residual amplitude oscillations remain beyond this time, due to imperfect absorption in the AB. Imperfect energy absorption is to be expected as, even the optimized AB of Grilli and Horrillo⁸ caused a few percent reflection. The STFT itself also appears slightly noisy, which could be alleviated, e.g., using a Hamming window to filter the input signal; or a wavelet transform could be used

instead of the STFT. Nevertheless, as seen in Table 1, for both NWTs, the value of the first three harmonics amplitudes in simulations is in good agreement with the theory, with a slight advantage for the present NWT.

In addition to the element type, as mentioned before, there is a slight difference in the wave generation methods used in the two NWTs. Indeed, the present NWT uses a semi-Lagrangian approach for the time integration of the free surface conditions, in which nodes are vertically updated; hence all free surface nodes keep a fixed horizontal location as a function of time. By contrast, in the MII-NWT, nodes are updated with a Lagrangian approach, following water particles on the free surface. Hence, as time increases, all the free surface nodes gradually move down the tank, towards $x > 0$, as a result of Stokes drift minus the opposite current that is specified in the wave generation; this is compensated by having the wavemaker boundary follow the first row of free surface nodes (see details in Grilli and Horrillo⁸).

In order to verify the effect of time-step or grid-size, we can adjust each and we will find a slight change to the time-series Fig. 17a, due to discretization effects. Further, one can verify the efficiency of the AB by taking a larger wave tank (of length 12.0), and looking at the spatial variation of the wave amplitude at different locations. For this comparison, we use the amplitude of the first harmonic, as computed using a least-squares fit, taking the free-surface elevation at each point to be a sum of sinusoids.

3.3 | Surface-piercing truncated cylinder

As a first case of wave-structure interaction, the diffraction of periodic deep water nonlinear waves is modeled around a truncated surface-piercing cylinder of radius R and draft D , with $D/R = 3$ (Fig. 18). Liu et al.³⁹ compared their second-order BEM results for this case, to the experiments of Krokstad and Stansberg⁸⁹. Kim and Yue⁹⁰ also studied this problem, up to second-order in the frequency domain, and Faltinsen et al.⁹¹ developed a small-body asymptotic theory for it.

Simulations are carried out in the present NWT with B-spline elements, with a domain geometry similar to that of Fig. 1 (rightward), for incident (zero-mass-flux) streamfunction waves of length L and amplitude A , with $kR = 0.22$ and $kA = 0.133$ ($k = 2\pi/L$). The outer radius of the NWT is $2L$, and its depth $h = L/2$ (i.e., these are deep-water waves). The absorbing beach (AB; here a relaxation zone) is one wavelength wide on all sides. The domain boundary was discretized with $N_{\Gamma} = 2,948$

TABLE 1 Case of Fig. 15. Non-dimensional amplitudes of first, second, and third harmonics computed on the time series of Fig. 16a, for incident (zero-mass-flux) streamfunction waves of height $H'_0 = 0.15$ and period $T' = 3.5515$, and results of NWT simulations at $x' = 4.0$; harmonics are computed by applying a short term Fourier transform (STFT) to time series between the 7th and 10th wave periods.

Model	$ \zeta^{(1)} /h_0$	$ \zeta^{(2)} /h_0$	$ \zeta^{(3)} /h_0$
Theory	7.32×10^{-2}	8.96×10^{-3}	1.66×10^{-3}
B-spline NWT	7.11×10^{-2}	8.52×10^{-3}	1.59×10^{-3}
MII-NWT	7.38×10^{-2}	9.89×10^{-3}	2.01×10^{-3}

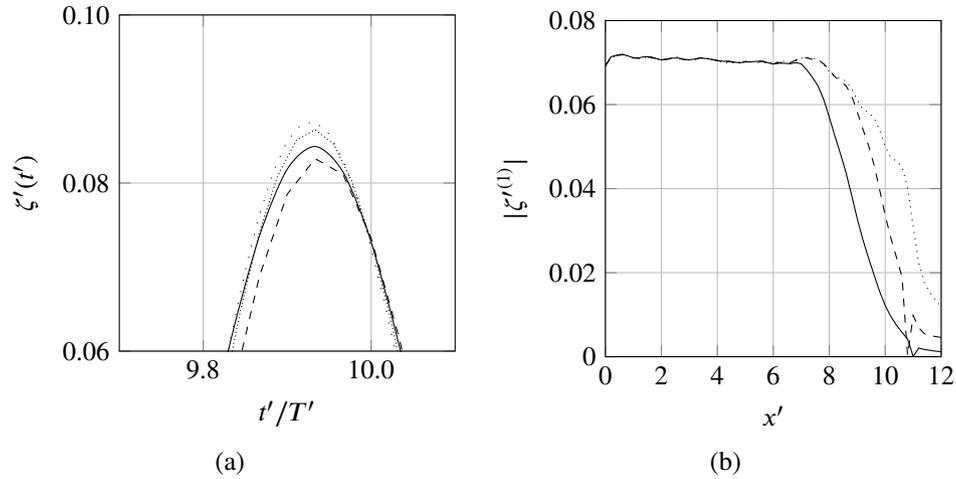


FIGURE 17 Results of streamfunction propagation simulations, showing: (a) time-series of streamfunction waves at $x' = 4.0$ (on the NWT axis) with a tank of length 10.0, in B-spline NWT for Δt of $T/30$ (---), $T/45$ (···), $T/60$ (· · ·), and reference solution (—); and (b) magnitude of the first harmonic in space for an AB that starts at $x' = 6$ (or $L_{AB} = 6$; solid), at $x' = 8$ (or $L_{AB} = 4$; dashed), and at $x' = 10$ (or $L_{AB} = 2$; dotted) for a tank of length 12.0.

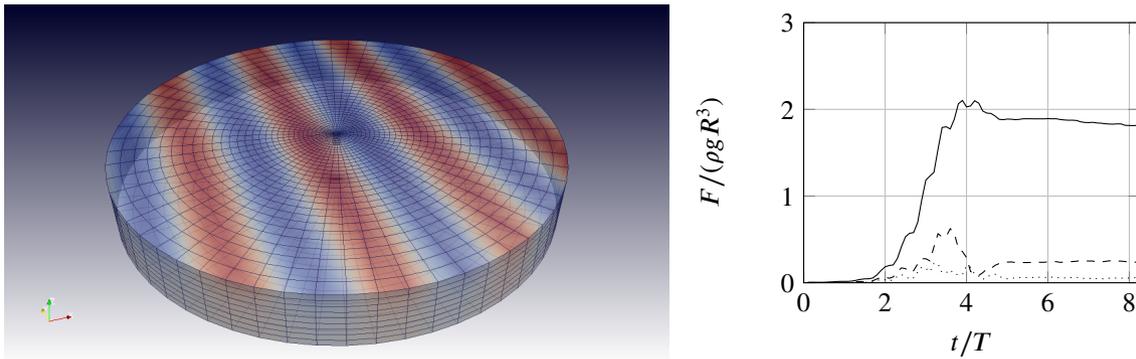


FIGURE 18 Diffraction around a surface piercing truncated cylinder. Short-time Fourier transform of longitudinal force measured on a truncated cylinder in waves, for the first (solid), second (dashed), and third (dotted) harmonics.

collocation nodes. To prevent instabilities, the filter detailed in Sec. 2.7 was applied to the free surface, near its intersection with the cylinder.

Wave diffraction was simulated for 11 wave periods T , during which the time series of the horizontal hydrodynamic force applied to the cylinder was computed as, $f(t) = \int_{\Gamma_h} p(t) n_x d\Gamma$ (where Γ_h denotes the cylinder's hull, and n_x the x -component of the normal vector, and the pressure is determined directly from Eq. 6); the non-dimensional force is further defined as, $f' = f/(\rho g R^3)$.

Although quasi-steady conditions are considered here, in offshore engineering, where truncated cylinders have often been used as part of support structures of oil rigs and are part of new designs of floating wind turbines, it is important to accurately estimate higher-order harmonics of the hydrodynamic force resulting from nonlinear wave diffraction. Indeed, these are associated with transient effects such as ringing (e.g., third-order forces; see Molin⁹²). Thus, Table 2 shows the first three harmonic amplitudes

of $f'(t)$ computed from $t = 6$ to $9T$, together with earlier experimental and numerical results. The agreement of most results with experiments is quite good for the 1st and 3rd harmonic and slightly less good for the second harmonic.

Given the small magnitude of the higher-order terms in the present case, however, it should also be pointed out that nonlinear viscous (drag) forces, neglected here, may significantly contribute to the higher-order harmonics of the total hydrodynamic force (Gentaz et al.⁹³). Adjustment of different grids show that the results for the first-order harmonic changes less than 3%, whereas higher-order harmonics vary more with changes in grid size, which is normal given the smaller wavelength of higher-order harmonics.

3.4 | Bottom-mounted vertical cylinder

Simulations were performed next for the interaction of deep water nonlinear periodic waves of various amplitudes A with a bottom-mounted vertical cylinder (i.e., monopile). Huseby and Grue⁹⁴ experimentally tested such a cylinder of radius $R = 3$ cm in depth $h = 0.6$ m. They generated incident waves with a piston wavemaker and placed the cylinder sufficiently far from it for the hydrodynamic force to be accurately measured, before spurious waves arrived (i.e., reflections, or parasitic, second-harmonic waves that are produced by a physical wavemaker).

3.4.1 | Results for $kR = 0.245$

Huseby and Grue's experimental conditions are modeled in the NWT using a computational domain similar to that in Fig. 1 (rightward), with an outer radius of 1.5 m and an AB (or relaxation zone) of 0.75 m on all sides. Similar to other authors^{83,20}, rather than modeling wave generation by a wavemaker, incident waves are simulated as in applications above, as zero-mass-flux streamfunction waves. A range of wave steepnesses kA are modeled, with $kR = 0.245$ (Fig. 19), and in each case simulations are performed for 10 wave periods T , with a quasi-steady state being reached for $t > 7T$.

TABLE 2 Case of Fig. 18. First three harmonic amplitudes of the non-dimensional horizontal hydrodynamic force $f'(t)$ applied to a truncated vertical cylinder, with $D/R = 3$, $kR = 0.22$, and $kA = 0.133$. Each amplitude was scaled by the corresponding power of wave steepness. NWT results are compared to earlier numerical and experimental results.

Result	Method	$\frac{ f'^{(1)} }{(kA)}$	$\frac{ f'^{(2)} }{(kA)^2}$	$\frac{ f'^{(3)} }{(kA)^3}$
Experiments	Krokstad and Stansberg (1995)	13.64	10.85	25.36
Second-order BEM	Liu et al. (2001)	13.41	12.97	30.99
Second-order freq.	Kim and Yue (1989)	13.28	14.86	26.30
Small-body asympt.	Faltinsen et al. (1995)	13.93	16.64	28.17
Present NWT	BEM (cubic B-spline)	13.92	13.79	23.11

The horizontal force $f(t)$ acting on the cylinder was computed as detailed in the previous application. Fig. 20 compares the computed force first three harmonics amplitudes to those of the experimentally measured force. For reference, the numerical results of Ferrant⁸³ are also shown. A good agreement of numerical results with experiments is observed, particularly for the first and third harmonics, with slightly larger errors for the second harmonics, although the expected trend with steepness kA is observed. The discrepancy in the second-order harmonics was also observed in Ferrant's⁸³ numerical results.

As with the previous application, filtering of the free surface was used (Sec. 2.7), near the intersection of the cylinder with the free surface. Additional simulations showed that instabilities only occurred for thin cylinders, i.e., when the surface Keulegan-Carpenter number is moderate. When running the model without filtering and at sufficiently high-resolution, wave patterns similar to those identified by Swan and Sheikh⁹⁵ as type-2 waves appeared near the cylinder, which are nonlinear waves that would eventually break, and discussion of the physics has been considered by Mockute et al.⁹⁶ for this type of problem.

3.4.2 | Results for $kR = 0.378$

In order to investigate the stability question further, tests for $kA = 0.05$ and $kR = 0.378$ were conducted, without using any filtering of the free-surface. In this case, for similar wave amplitudes, the test runs for $kR = 0.245$ would become unstable, but here (Fig. 21), we obtain stable results with the expected magnitude. To test the sensitivity of the FMM application, the model is run both with and without FMM, with a difference in the measured horizontal force less than 0.01%.

Others have also reported that filtering is required for modeling a vertical cylinder in regular waves has been required for moderate wave amplitudes and sufficiently thin cylinders; in particular, Ferrant⁸³ and Christou⁹⁷ both discussed this necessity for the same test case and parameters. Future work will look at physically accounting for the energy dissipation of such locally breaking waves.

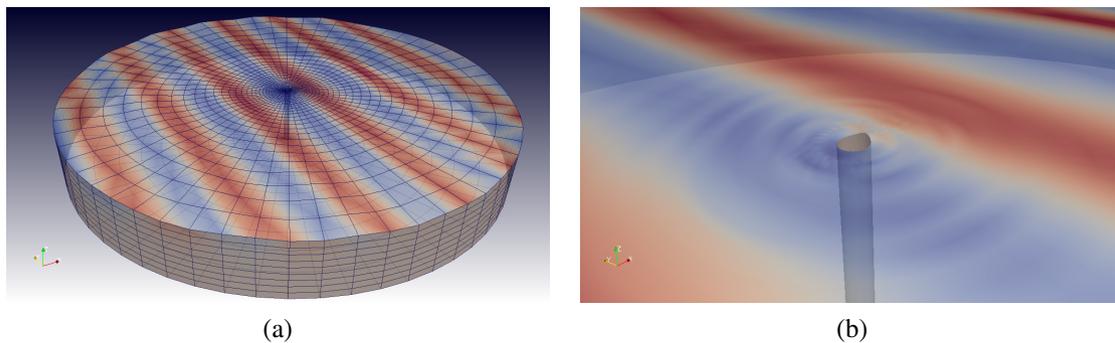


FIGURE 19 Snapshot of diffraction around a bottom-mounted cylinder, computed in the present NWT: (a) after achieving a quasi-steady state; and (b) close-up view showing diffracted waves. Incident waves are (zero-mass-flux) streamfunction waves in deep water, here with $kA = 0.10$ and $kR = 0.245$.

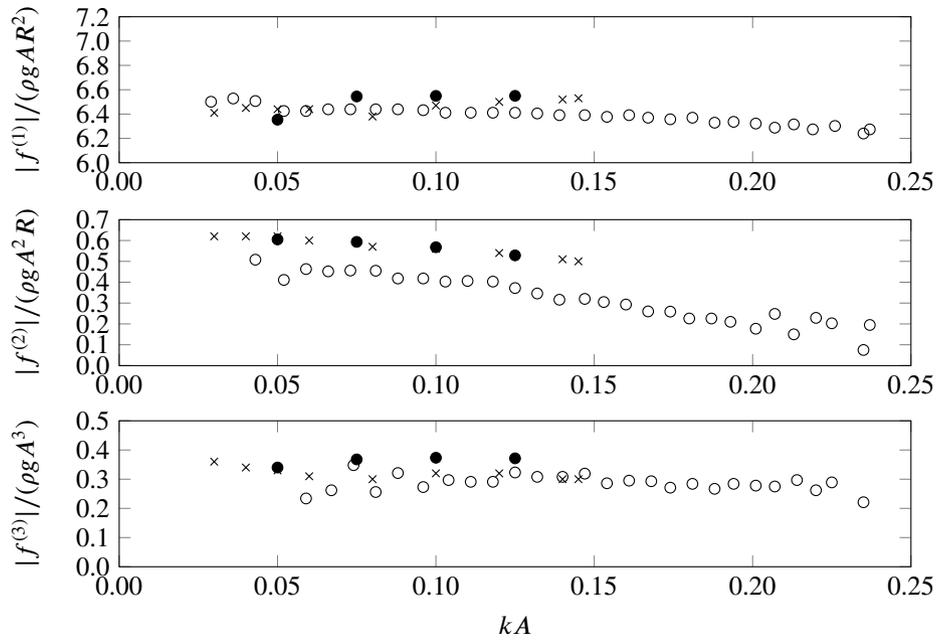


FIGURE 20 Non-dimensional amplitude of first- through third-order harmonic of the horizontal hydrodynamic force $f(t)$ applied to a bottom-mounted cylinder of radius R , for deep water periodic waves of wavenumber k , for different steepness kA , with $kR = 0.245$, in: (○) Huseby and Grue's⁹⁴ experiments; (×) Ferrant's⁸³ model results; and (●) computed in the present NWT.

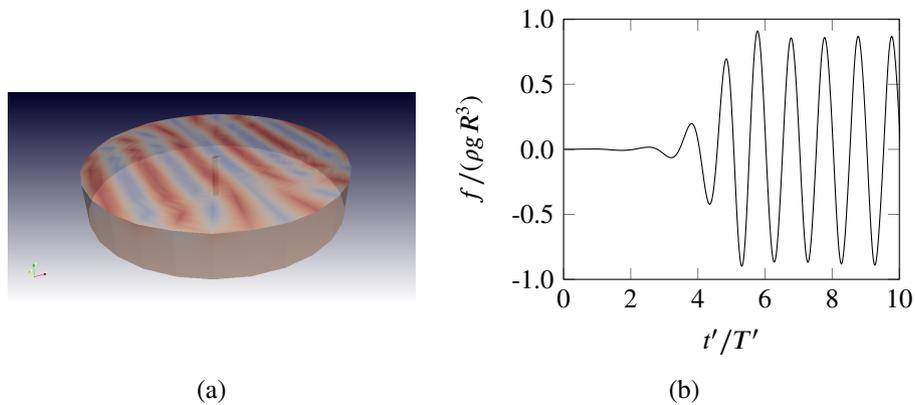


FIGURE 21 Snapshot of diffraction around a bottom-mounted cylinder, computed in the present NWT, without free-surface filtering: (a) after achieving a quasi-steady state; and (b) time-series of horizontal force. Incident waves are (zero-mass-flux) streamfunction waves in deep water, here with $kA = 0.05$ and $kR = 0.378$.

4 | CONCLUSIONS AND PERSPECTIVES

In this work, we reported on the development, validation, and application of a new parallelized, FMM-accelerated, three-dimensional (3D) Numerical Wave Tank (NWT) solving fully nonlinear potential flow equations (FNPF) for wave-structure interaction, based on a boundary element method (BEM). This NWT provides both a more efficient implementation (in terms

of numerical complexity) of Grilli et al.'s⁴ NWT and has the ability to use either unstructured grids, in the case of linear elements, or block structured grids, in the case of cubic B-spline elements. The latter is mathematically based on the same types of splines used in computer-aided designs for ships or other engineering structures.

While we focus on a semi-Lagrangian (i.e., vertical time-updating) treatment of the free surface, which is limited in some ways compared to cubic MII elements used in earlier work^{4,42,5,6} (e.g., no wave overturning can be modeled), applications presented here show that similarly acceptable numerical errors on conservation of mass and energy can be obtained, while achieving much faster computational times. This is due to the more advantageous numerical complexity of the present NWT. More specifically, in the applications presented here (e.g., solitary and periodic wave propagation), while providing the same or smaller numerical errors, the present B-spline NWT proved much more computationally efficient than that of Grilli et al.'s⁴ MII-NWT, which scales nearly as $O(N_r^2)$ in terms of CPU time and numerical complexity. Here, an $O(N_r^{1.2})$ complexity was achieved on a single processor, with an additional reasonable speed-up when using multiple processors (e.g. of nearly 4 for 4 processors and 16 for 64 processors). Beyond a few thousand collocation nodes, using the earlier NWT was deemed prohibitive. By contrast, with the present NWT, the combination of parallelization and FMM, makes it possible simulating transient wave structure interaction problems using large grids, of up to 100,000 nodes, while achieving a reasonable performance on a medium size cluster (desktop workstation). In particular, in the presented more advanced applications of wave interaction with a surface piercing cylinder, stable and accurate results were achieved for the nonlinear hydrodynamic force applied to the cylinder.

As described by Tanizawa⁹, there are common features to all time-domain potential flow nonlinear NWTs: the choice of a discretization; the development of a Laplace solver; the need to compute tangential derivatives (including at the edges and corners between multiple surfaces); the choice of a time-stepping method; and the application of methods for wave generation and absorption. Here, we show results using cubic B-spline elements, and we accelerate the Laplace solver by making use of the fast multipole method (FMM), in an efficient parallel implementation (ExaFMM), yielding a method which provides significantly better scaling than most traditional BEM solutions. Although, in some cases, we do not yet have some features of the NWT of Grilli et al.⁴, or the same speed as in the recent method of Engsig-Karup²², we expect that by improving the preconditioning of the algebraic system of equations, it will be possible to make significant improvements in all aspects of potential flow solution.

As a tool for offshore design, the present NWT still has a number of practical limitations, particularly when considering moving bodies. In this respect, recent improvements in the corresponding 2D-NWT of Grilli et al.^{1,2} by Guerber et al.⁴⁸ and Dombre et al.⁵⁵, concerning floating bodies, are being implemented in this 3D-NWT. Some results are reported in Dombre et al.⁵⁷. Also, given the physical limitations to what can be modeled with an inviscid NWT, it is logical to consider the possibility of coupling this NWT to a viscous model for flow around a body. We have shown in earlier work⁵¹ that obtaining the internal velocity quickly using the FMM is straightforward. Given the results shown here, as well as more recent results by Dombre⁹⁸

for forced and free motion of floating bodies, it is expected that the present formulation can be applied successfully both to cases of academic interest, as well as industrial designs, such as ship waves, wave energy converters, or floating offshore platforms.

An important limitation of this type of NWT design, even for flows which otherwise can be described almost entirely by potential flow, is the occurrence of local wave breaking, which in a perfectly inviscid description would result in overturning waves. In this case, the energy which would be lost from the waves due to this breaking can be absorbed by a pressure depending on local free-surface slope or curvature⁶⁸. Further, the interpolation on each boundary element is based here only on information on a given surface (e.g., the free-surface), but at intersections, one can obtain more accurate simulations by imposing additional constraints on the interpolation, as previously studied by Grilli and Svendsen⁶⁶ in 2D, and is now being considered in 3D⁶⁸. This work in progress will be incorporated in future versions of the model.

5 | ACKNOWLEDGEMENTS

The work of J. C. Harris and M. Benoit was partially funded as a part of the French ANR (Agence Nationale de la Recherche), project ANR11-MONU-018-01 MONACOREV, and the work of E. Dombre by the ANRT (Association Nationale de la Recherche et de la Technologie), CIFRE agreement #2011-1724. S.T. Grilli acknowledges support for his work from Grants N000-14-13-10687 and N000-14-16-12970 of the US Office of Naval Research (ONR) Code 333 (PM Kelly Cooper). We would like to thank Dr. Rio Yokota for useful discussions of the fast multipole method.

References

1. Grilli ST, Skourup J, Svendsen IA. An efficient boundary element method for nonlinear water waves. *Engineering Analysis with Boundary Elements* 1989; 6: 97–107. [https://doi.org/10.1016/0955-7997\(89\)90005-2](https://doi.org/10.1016/0955-7997(89)90005-2).
2. Grilli ST, Subramanya R. Numerical modeling of wave breaking induced by fixed or moving boundaries. *Computational Mechanics* 1996; 17: 374–391. <https://doi.org/10.1007/BF00363981>.
3. Fenton JD. Numerical methods for nonlinear waves. In: Liu PLF., ed. *Advances in Coastal and Ocean Engineering* World Scientific Pub. Co. Inc. 1999 (pp. 241–324). https://doi.org/10.1142/9789812797544_0005.
4. Grilli ST, Guyenne P, Dias F. A fully nonlinear model for three-dimensional overturning waves over arbitrary bottom. *International Journal for Numerical Methods in Fluids* 2001; 35: 829–867. [https://doi.org/10.1002/1097-0363\(20010415\)35:7<829::AID-FLD115>3.0.CO;2-2](https://doi.org/10.1002/1097-0363(20010415)35:7<829::AID-FLD115>3.0.CO;2-2).

5. Guyenne P, Grilli ST. Numerical study of three-dimensional overturning waves in shallow water. *Journal of Fluid Mechanics* 2006; 547: 361–388. <https://doi.org/10.1017/S0022112005007317>.
6. Fochesato C, Grilli ST, Dias F. Numerical modeling of extreme rogue waves generated by directional energy focusing. *Wave Motion* 2007; 44: 395–416. <https://doi.org/10.1016/j.wavemoti.2007.01.003>.
7. Grilli ST, Dias F, Guyenne P, Fochesato C, Enet F. *Series in Advances in Coastal and Ocean Engineering*. 11. ch. Progress in Fully Nonlinear Potential Flow Modeling of 3D Extreme Ocean Waves: 75–128; World Scientific Pub. Co. Inc. . 2010. https://doi.org/10.1142/9789812836502_0003.
8. Grilli ST, Horrillo J. Generation and absorption of fully nonlinear periodic waves. *Journal of Engineering Mechanics* 1997; 123: 1060–1069. [https://doi.org/10.1061/\(ASCE\)0733-9399\(1997\)123:10\(1060\)](https://doi.org/10.1061/(ASCE)0733-9399(1997)123:10(1060)).
9. Tanizawa K. The state of the art on numerical wave tank. In: *Proceeding of 4th Osaka Colloquium on Seakeeping Performance of Ships*. Osaka University. ; October 17–21, 2000; Osaka, Japan: 95–114.
10. Lee CH. WAMIT Theory Manual. tech. rep., MIT; Cambridge, MA, USA: 1995. Report 95-2, Department of Ocean Engineering.
11. Delhommeau G. Seakeeping codes AQUADYN and AQUAPLUS. In: *19th WEGEMT School on Numerical Simulation of Hydrodynamics Ships and Offshore Structures*. WEGEMT. ; September 20–24, 1993.
12. Kring DC, Korsmeyer FT, Singer J, Danmeier D, White J. Accelerated nonlinear wave simulations for large structures. In: *7th International Conference on Numerical Ship Hydrodynamics*. US Office of Naval Research. ; July 19–22, 1999; Nantes, France.
13. Saad Y, Schultz MH. GMRES: a generalized minimum residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing* 1986; 7: 856–869. <https://doi.org/10.1137/0907058>.
14. Folley M, Whittaker TWT, van't Hoff J. The design of small seabed-mounted bottom-hinged wave energy converters. In: *Proceedings of the 7th European Wave and Tidal Energy Conference*. EWTEC. ; September 11–13, 2007; Porto, Portugal. 10 pp.
15. Falnes J. *Ocean waves and oscillating systems: linear interactions including wave-energy extraction*. Cambridge University Press . 2002. ISBN 9781108481663.
16. Abadie S, Morichon D, Grilli ST, Glockner S. Numerical simulation of waves generated by landslides using a multiple-fluid Navier-Stokes model. *Coastal Engineering* 2010; 57: 779–794. <https://doi.org/10.1016/j.coastaleng.2010.03.003>.

17. Luquet R. *Simulation numérique de l'écoulement visqueux autour d'un navire soumis à une houle quelconque*. PhD thesis. Ecole Centrale de Nantes, Nantes, France; 2007.
18. Bhinder MA, Mingham CG, Causon DM, Rahmati MT, Aggidis GA, Chaplin RV. A joint numerical and experimental study of a surging point absorbing wave energy converter (WRASPA). In: *Proceedings of the 28th International Conference on Ocean, Offshore and Arctic Engineering*. American Society of Mechanical Engineers. ; May 31–June 5, 2009: 869–875. <https://doi.org/10.1115/OMAE2009-79392>.
19. Fochesato C, Dias F. A fast method for nonlinear three-dimensional free-surface waves. *Proceedings of the Royal Society A* 2006; 462: 2715–2735. <https://doi.org/10.1098/rspa.2006.1706>.
20. Shao YL, Faltinsen OM. A harmonic polynomial cell (HPC) method for 3D Laplace equation with application in marine hydrodynamics. *Journal of Computational Physics* 2014; 274: 312–332. <https://doi.org/10.1016/j.jcp.2014.06.021>.
21. Ma QW, Yan S. QALE-FEM method and its application to the simulation of free-responses of floating bodies and overturning waves. In: Ma Q., ed. *Advances in Numerical Simulation of Nonlinear Water Waves* World Scientific Pub. Co. Inc. 2009 (pp. 165–202). https://doi.org/10.1142/9789812836502_0005.
22. Engsig-Karup AP. Analysis of efficient preconditioned defect correction methods for nonlinear water waves. *International Journal for Numerical Methods in Fluids* 2014; 74: 749–773. <https://doi.org/10.1002/flid.3873>.
23. Engsig-Karup A, Bingham H, Lindberg O. An efficient flexible-order model for 3D nonlinear water waves. *Journal of Computational Physics* 2009; 228: 2100–2118. <https://doi.org/10.1016/j.jcp.2008.11.028>.
24. Engsig-Karup AP, Madsen MG, Glimberg SL. A massively parallel GPU-accelerated model for analysis of fully nonlinear free surface waves. *International Journal of Numerical Methods in Fluids* 2012; 70: 20–36. <https://doi.org/10.1002/flid.2675>.
25. Cheng AHD, Cheng DT. Heritage and early history of the boundary element method. *Engineering Analysis with Boundary Elements* 2005; 29: 268–302. <https://doi.org/10.1016/j.enganabound.2004.12.001>.
26. Longuet-Higgins MS, Cokelet E. The deformation of steep surface waves on water, I. A numerical method of computation. *Proceedings of the Royal Society A* 1976; 350: 1–26. <https://doi.org/10.1098/rspa.1976.0092>.
27. Isaacson M. Nonlinear effects on fixed and floating bodies. *Journal of Fluid Mechanics* 1982; 120: 267–281. <https://doi.org/10.1017/S0022112082002766>.
28. Romate J. The numerical simulation of nonlinear gravity waves. *Engineering Analysis with Boundary Elements* 1990; 4: 156–166. [https://doi.org/10.1016/0955-7997\(90\)90002-Q](https://doi.org/10.1016/0955-7997(90)90002-Q).

29. van der Vorst HA. Bi-CGSTAB: a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing* 1992; 13: 631–644. <https://doi.org/10.1137/0913035>.
30. Greengard L, Rokhlin V. A fast algorithm for particle simulations. *Journal of Computational Physics* 1987; 73: 325–348. [https://doi.org/10.1016/0021-9991\(87\)90140-9](https://doi.org/10.1016/0021-9991(87)90140-9).
31. Korsmeyer T, Yue DKP, Nabors K, White J. Multipole-accelerated preconditioned iterative methods for three-dimensional potential problems. In: *Boundary Element Methods XV Vol 1 Fluid Flow and Computational Aspects*. WIT Press. ; 1993: 517–527. <https://doi.org/10.2495/BE930351>.
32. Greengard L, Rokhlin V. A new version of the Fast Multipole Method for the Laplace equation in three dimensions. *Acta Numerica* 1997; 6: 229–269. <https://doi.org/10.1017/S0962492900002725>.
33. Greengard L, Groppe WD. A parallel version of the fast multipole method. *Computers and Mathematics with Applications* 1990; 20: 63–71. [https://doi.org/10.1016/0898-1221\(90\)90349-O](https://doi.org/10.1016/0898-1221(90)90349-O).
34. Yokota R. An FMM based on dual tree traversal for many-core architectures. *Journal of Algorithms and Computational Technology* 2013; 7: 301–324. <https://doi.org/10.1260/1748-3018.7.3.301>.
35. Yan H, Liu Y. An efficient high-order boundary element method for nonlinear wave–wave and wave-body interactions. *Journal of Computational Physics* 2011; 230: 402–424. <https://doi.org/10.1016/j.jcp.2010.09.029>.
36. Ho KL, Greengard L. A fast direct solver for structured linear systems by recursive skeletonization. *SIAM Journal on Scientific Computing* 2012; 34: A2507–A2532. <https://doi.org/10.1137/120866683>.
37. Ambikasaran S, Darve E. An $O(N \log N)$ fast direct solver for partial hierarchically semi-separable matrices. *Journal of Scientific Computing* 2013; 57: 477–501. <https://doi.org/10.1007/s10915-013-9714-z>.
38. Arnold A, Fahrenberger F, Holm C, et al. Comparison of scalable fast methods for long-range interactions. *Physical Review E* 2013; 88: 063308. <https://doi.org/10.1103/PhysRevE.88.063308>.
39. Liu Y, Xue M, Yue DKP. Computations of fully nonlinear three-dimensional wave-wave and wave-body interactions. Part 2. Nonlinear waves and forces on a body. *Journal of Fluid Mechanics* 2001; 438: 41–66. <https://doi.org/10.1017/S0022112001004384>.
40. Grilli ST, Subramanya R, Svendsen IA, Veeramony J. Shoaling of solitary waves on plane beaches. *Journal of Waterway, Port, Coastal and Ocean Engineering* 1994; 120(6): 609–628. [https://doi.org/10.1061/\(ASCE\)0733-950X\(1994\)120:6\(609\)](https://doi.org/10.1061/(ASCE)0733-950X(1994)120:6(609)).

41. Grilli ST, Subramanya R, Svendsen IA. Breaking criterion and characteristics for solitary waves on slopes. *Journal of Waterway, Port, Coastal and Ocean Engineering* 1997; 123(3): 102–112. [https://doi.org/10.1061/\(ASCE\)0733-950X\(1997\)123:3\(102\)](https://doi.org/10.1061/(ASCE)0733-950X(1997)123:3(102)).
42. Grilli ST, Vogelmann S, Watts P. Development of a 3D Numerical Wave Tank for modeling tsunami generation by underwater landslides. *Engineering Analysis with Boundary Elements* 2002; 26: 301–313. [https://doi.org/10.1016/S0955-7997\(01\)00113-8](https://doi.org/10.1016/S0955-7997(01)00113-8).
43. Sung HG, Grilli ST. A note on accuracy and convergence of a third-order boundary element method for three dimensional nonlinear free surface flows. *Journal of Ships and Ocean Engineering* 2005; 40: 31–41. (in Korean).
44. Sung HG, Grilli ST. Numerical modeling of nonlinear surface waves caused by surface effect ships dynamics and kinematics. In: Proceedings of the 15th International Offshore and Polar Engineering Conference. International Society of Offshore and Polar Engineers. ; June 19–24, 2005; Seoul, Korea.
45. Sung HG, Grilli ST. BEM computations of 3D fully nonlinear free surface flows caused by advancing surface disturbances. *International Journal for Offshore and Polar Engineering* 2008; 18: 292–301.
46. Barthelemy X, Banner ML, Peirson WL, Fedele F, Allis M, Dias F. On a unified breaking onset threshold for gravity waves in deep and intermediate depth water. *Journal of Fluid Mechanics* 2018; 841: 463–488. <https://doi.org/10.1017/jfm.2018.93>.
47. Nimmala SB, Yim SC, Grilli ST. An efficient parallelized 3-D FNPF numerical wave tank for large-scale wave basin experiment simulation. *Journal of Offshore Mechanics and Arctic Engineering* 2013; 135: 021104. <https://doi.org/10.1115/1.4007597>.
48. Guerber E, Benoit M, Grilli ST, Buvat C. A fully nonlinear implicit model for wave interactions with submerged structures in forced of free motion. *Engineering Analysis with Boundary Elements* 2012; 36: 1151–1163. <https://doi.org/10.1016/j.enganabound.2012.02.005>.
49. Yokota R, Bardhan JP, Knepley MG, Barba LA, Hamada T. Biomolecular electrostatics using a fast multipole BEM on up to 512 GPUs and a billion unknowns. *Computer Physics Communications* 2011; 182: 1272–1283. <https://doi.org/10.1016/j.cpc.2011.02.013>.
50. Letournel L, Harris JC, Ferrant P, et al. Comparison of fully nonlinear and weakly nonlinear potential flow solvers for the study of wave energy converters undergoing large amplitude motions. In: Proceedings of the 33rd International Conference on Ocean, Offshore and Arctic Engineering. American Society of Mechanical Engineers. ; June 8–13, 2014; San Francisco, CA, USA: 23912. <https://doi.org/10.1115/OMAE2014-23912>.

51. Harris JC, Dombre E, Benoit M, Grilli ST. Fast integral equation methods for fully nonlinear water wave modeling. In: Proceedings of the 24th International Offshore and Polar Engineering Conference. International Society of Offshore and Polar Engineers. ; June 15–20, 2014; Busan, Korea: 583–590.
52. Tanaka M. The stability of solitary waves. *Physics of Fluids* 1986; 29: 650–655. <https://doi.org/10.1063/1.865459>.
53. Zhang J, Kashiwagi M. Application of ALE to nonlinear wave diffraction by a non-wall-sided structure. In: Proceedings of the 27th International Ocean and Polar Engineering Conference. International Society of Offshore and Polar Engineers. ; June 25–30, 2017; San Francisco, CA, USA: 461–468.
54. Gottlieb S. On high order strong stability preserving Runge-Kutta and multi step time discretizations. *Journal of Scientific Computing* 2005; 25: 105–128. <https://doi.org/10.1007/BF02728985>.
55. Dombre E, Benoit M, Violeau D, Peyrard C, Grilli ST. Simulation of floating structure dynamics in waves by implicit coupling of a fully nonlinear potential flow model and a rigid body motion approach. *Journal of Ocean Engineering and Marine Energy* 2015; 1: 55–76. <https://doi.org/10.1007/s40722-014-0006-y>.
56. Dias F, Bridges T. The numerical computation of freely propagating time-dependent irrotational water waves. *Fluid Dynamics Research* 2006; 38: 803–830. <https://doi.org/10.1016/j.fluidyn.2005.08.007>.
57. Dombre E, Harris JC, Benoit M, Violeau D, Peyrard C. A 3D parallel boundary element method on unstructured triangular grids for fully nonlinear wave-body interactions. *Ocean Engineering* 2019; 171: 505–518. <https://doi.org/10.1016/j.oceaneng.2018.09.044>.
58. Büchmann B. Accuracy and stability of a set of free-surface time-domain boundary element models based on B-splines. *International Journal for Numerical Methods in Fluids* 2000; 33: 125–155. [https://doi.org/10.1002/\(SICI\)1097-0363\(20000515\)33:1<125::AID-FLD5>3.0.CO;2-Q](https://doi.org/10.1002/(SICI)1097-0363(20000515)33:1<125::AID-FLD5>3.0.CO;2-Q).
59. Fochesato C, Grilli ST, Guyenne P. Note on non-orthogonality of local curvilinear co-ordinates in a three-dimensional boundary element method. *International Journal for Numerical Methods in Fluids* 2005; 48: 305–324. <https://doi.org/10.1002/flid.838>.
60. Bai W, Eatock Taylor R. Higher-order boundary element simulation of fully nonlinear wave radiation by oscillating vertical cylinders. *Applied Ocean Research* 2006; 28: 247–265. <https://doi.org/10.1016/j.apor.2006.12.001>.
61. Cirak F, Ortiz M, Schröder P. Subdivision surfaces: a new paradigm for thin-shell finite-element analysis. *International Journal for Numerical Methods in Engineering* 2000; 47: 2039–2072. [https://doi.org/10.1002/\(SICI\)1097-0207\(20000430\)47:12<2039::AID-NME872>3.0.CO;2-1](https://doi.org/10.1002/(SICI)1097-0207(20000430)47:12<2039::AID-NME872>3.0.CO;2-1).

62. Delftship BV Hoofddorp, Netherlands *DELFTship user manual*. 2007. Version 3.1, 51 pp.
63. Liu PLF, Hsu HW, Lean MH. Applications of boundary integral equation methods for two-dimensional non-linear water wave problems. *International Journal for Numerical Methods in Fluids* 1992; 15: 1119–1141. <https://doi.org/10.1002/flid.1650150912>.
64. Hughes T, Cottrell J, Bazilevs Y. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computer Methods in Applied Mechanics and Engineering* 2005; 194(39–41): 4135–4195. <https://doi.org/10.1016/j.cma.2004.10.008>.
65. Abbasnia A, Soares CG. OpenMP parallelism in computations of three-dimensional potential numerical wave tank for fully nonlinear simulation of wave-body interaction using NURBS. *Engineering Analysis with Boundary Elements* 2020; 117: 321–331. <https://doi.org/10.1016/j.engabound.2020.05.008>.
66. Grilli ST, Svendsen IA. Corner problems and global accuracy in the boundary element solution of nonlinear wave flows. *Engineering Analysis with Boundary Elements* 1990; 7: 178–195. [https://doi.org/10.1016/0955-7997\(90\)90004-S](https://doi.org/10.1016/0955-7997(90)90004-S).
67. Sen D. A cubic-spline boundary integral method for two-dimensional free-surface flow problems. *International Journal for Numerical Methods in Engineering* 1995; 38: 1809–1830. <https://doi.org/10.1002/nme.1620381103>.
68. Mivehchi A, Harris J, Grilli S, et al. A hybrid solver based on efficient BEM-potential and LBM-NS models: recent BEM developments and applications to naval hydrodynamics. In: Proceedings of the 27th International Offshore and Polar Engineering Conference. International Society of Offshore and Polar Engineers. ; June 25–30, 2017; San Francisco, CA, USA: 721–728.
69. Yokota R, Barba L. A tuned and scalable fast multipole method as a preeminent algorithm for exascale systems. *International Journal of High-Performance Computing Applications* 2012; 26: 337–346. <https://doi.org/10.1177/1094342011429952>.
70. Dehnen W. A hierarchical $O(N)$ force calculation algorithm. *Journal of Computational Physics* 2002; 179: 27–42. <https://doi.org/10.1006/jcph.2002.7026>.
71. Duffy MG. Quadrature over a pyramid or cube of integrands with a singularity at a vertex. *SIAM Journal on Numerical Analysis* 1982; 19: 1260–1262. <https://doi.org/10.1137/0719090>.
72. Grilli ST, Subramanya R. Quasi-singular integrations in the modelling of nonlinear water waves. *Engineering Analysis with Boundary Elements* 1994; 13(2): 181–191. [https://doi.org/10.1016/0955-7997\(94\)90020-5](https://doi.org/10.1016/0955-7997(94)90020-5).
73. Brebbia CA. *The boundary element method for engineers*. John Wiley and Sons . 1978. ISBN 9780727302052.

74. Salmon JK. *Parallel hierarchical N-body methods*. PhD thesis. California Institute of Technology, Pasadena, CA, USA; 1991.
75. Warren MS, Salmon JK. A portable parallel particle program. *Computer Physics Communications* 1995; 87: 266–290. [https://doi.org/10.1016/0010-4655\(94\)00177-4](https://doi.org/10.1016/0010-4655(94)00177-4).
76. Yokota R, Barba LA, Narumi T, Tasuoka K. Petascale turbulence simulation using a highly parallel fast multipole method on GPUs. *Computer Physics Communications* 2013; 184: 445–455. <https://doi.org/10.1016/j.cpc.2012.09.011>.
77. Abduljabbar M, Markomanolis GS, Ibeid H, Yokota R, Keyes D. Communication reducing algorithms for distributed hierarchical N-body problems with boundary distributions. In: 32nd International Conference on High Performance Computing. ISC Group. ; June 18–22, 2017; Frankfurt, Germany: 79–96.
78. Visscher P, Apalkov D. Simple recursive implementation of fast multipole method. *Journal of Magnetism and Magnetic Materials* 2010; 322: 275–281. <https://doi.org/10.1016/j.jmmm.2009.09.033>.
79. Zhang W, Haas S. Adaptation and performance of the Cartesian coordinates fast multipole method for nanomagnetic simulations. *Journal of Magnetism and Magnetic Materials* 2009; 321: 3687–3692. <https://doi.org/10.1016/j.jmmm.2009.07.016>.
80. Dean RG, Dalrymple RA. *Water wave mechanics for engineers and scientists*. Advanced series on ocean engineering: Volume 2 World Scientific . 1991. <https://doi.org/10.1142/1232>.
81. Wang H, Lei T, Li J, Huang J, Yao Z. A parallel fast multipole accelerated integral equation scheme for 3D Stokes equations. *International Journal for Numerical Methods in Engineering* 2007; 70: 812–839. <https://doi.org/10.1002/nme.1910>.
82. Cao X, Mo Z, Liu X, Xu X, Zhang A. Parallel implementation of fast multipole method based on JASMIN. *Science China Information Sciences* 2011; 54: 757–766. <https://doi.org/10.1007/s11432-011-4181-3>.
83. Ferrant P. Fully nonlinear interactions of long-crested wave packets with a three-dimensional body. In: Proceedings of the 22nd Symposium on Naval Hydrodynamics. US Office of Naval Research. ; August 9–14, 1998; Washington, DC, USA: 403–416.
84. Dean RG. Stream function representation of nonlinear ocean waves. *Journal of Geophysical Research* 1965; 70: 4561–5472. <https://doi.org/10.1029/JZ070i018p04561>.
85. Dalrymple RA. Wave-induced mass transport in water waves. *Journal of the Waterways, Harbors and Coastal Engineering Division* 1976; 102: 255–264.

86. Dommermuth DG, Yue DKP. A higher-order spectral method for the study of nonlinear gravity waves. *Journal of Fluid Mechanics* 1987; 184: 267–288. <https://doi.org/10.1017/S002211208700288X>.
87. Kim MW, Koo W, Hong SY. Numerical analysis of various artificial damping schemes in a three-dimensional numerical wave tank. *Ocean Engineering* 2014; 75: 165–173. <https://doi.org/10.1016/j.oceaneng.2013.10.012>.
88. Clément A. Coupling of two absorbing boundary conditions for 2D time-domain simulations of free surface gravity waves. *Journal of Computational Physics* 1996; 126: 139–151. <https://doi.org/10.1006/jcph.1996.0126>.
89. Krokstad JR, Stansberg CT. Ringing load models verified against experiments. In: Proceedings of the International Conference on Offshore Mechanics and Arctic Engineering. American Society of Mechanical Engineers. ; June 18–22, 1995; Copenhagen, Denmark.
90. Kim HM, Yue DKP. The complete second-order diffraction solution for an axisymmetric body. Part I. Monochromatic incident waves. *Journal of Fluid Mechanics* 1989; 200: 235–264. <https://doi.org/10.1017/S0022112089000649>.
91. Faltinsen OM, Newman JN, Vinje T. Nonlinear wave loads on a slender vertical cylinder. *Journal of Fluid Mechanics* 1995; 289: 179–198. <https://doi.org/10.1017/S0022112095001297>.
92. Molin B. *Hydrodynamique des structures offshore*. Editions Technip . 2002. ISBN 9782710808152.
93. Gentaz L, Luquet R, Alessandrini B, Ferrant P. Numerical simulation of the 3D viscous flow around a vertical cylinder in non-linear waves using an explicit incident wave model. In: Proceedings of the 24th International Conference on Offshore Mechanics and Arctic Engineering. American Society of Mechanical Engineers. ; June 20–25, 2004; Vancouver, BC, Canada: 7 pp.
94. Huseby M, Grue J. An experimental investigation of higher-harmonic wave forces on a vertical cylinder. *Journal of Fluid Mechanics* 2000; 414: 75–103. <https://doi.org/10.1017/S0022112000008533>.
95. Swan C, Sheikh R. The interaction between steep waves and a surface-piercing column. *Philosophical Transactions of the Royal Society A* 2015; 373: 20140114. <https://doi.org/10.1098/rsta.2014.0114>.
96. Mockutè A, Marino E, Lugni C, Borri C. Comparison of nonlinear wave-loading models on rigid cylinders in regular waves. *Energies* 2019; 12: 4022. <https://doi.org/10.3390/en12214022>.
97. Christou M. *Fully nonlinear computations of waves and wave-structure interaction*. PhD thesis. Imperial College London, London, UK; 2009.

98. Dombre E. *Modélisation non-linéaire des interactions vague-structure appliquée à des flotteurs d'éoliennes off-shore*. PhD thesis. Université Paris-Est, Champs-sur-Marne, France; 2015.

