ENSTA Bretagne
Ifremer
15ÈMES JOURNÉES DE L'HYDRODYNAMIQUE
22 - 24 novembre 2016 - Brest

AVANCEES DE LA MODELISATION NON-LINEAIRE DES INTERACTIONS VAGUE-STRUCTURE

PROGRESS IN FULLY NONLINEAR WAVE MODELING FOR WAVE-STRUCTURE INTERACTION

J. C. HARRIS[(1)], E. DOMBRE[(1)], A. MIVEHCHI[(2)], M. BENOIT[(3)], S. T. GRILLI[(2)], C. PEYRARD[(1,4)]

(1) LHSV, Ecole des Ponts, CEREMA, EDF R&D, Université Paris-Est, Chatou, France
(2) Dept. of Ocean Engineering, University of Rhode Island, Narragansett, RI 02882, USA
(3) Institut de Recherche sur les Phénomènes Hors Equilibre (IRPHE), UMR 7342 (Aix-Marseille Univ., Ecole Centrale Marseille, CNRS), Marseille, France
(4) EDF R&D, Laboratoire National d'Hydraulique et Environnement, Chatou, France

Résumé

Ce papier présente les développements récents d'un canal à houle numérique. Dans ce modèle potentiel et complètement non-linéaire, utilisant un schéma d'avance en temps semi-lagrangien pour la surface libre, l'équation de Laplace sur le potentiel est résolue par la méthode des éléments de frontières à chaque pas de temps. L'utilisation pratique de ce type de modèle étant limitée par les temps de calcul, une méthode de multipôles rapide est utilisée et implémentée en mode parallèle. Elle montre une bonne scalabilité pour des nombres d'inconnues de 10^3 à 10^5 et sur plusieurs centaines de processeurs. L'intégration adaptative permet une résolution des vitesses dans le domaine fluide, y compris à proximité des parois. Le modèle est validé par une comparaison des efforts sur un cylindre vertical en interaction avec les vagues, et le sillage proche d'un profil NACA.

Summary

We present recent improvements in computational performance and capabilities of a numerical wave tank for fully nonlinear potential flows. This approach is based on the boundary element solution of the Laplace equation, and advanced in time with a mixed Eulerian-Lagrangian method. As computational time is a main limitation of this method, the fast multipole method is applied, in parallel, and is shown to scale well to 10^3–10^5 unknowns and hundreds of processors. In addition, internal velocities are computed efficiently in the same manner, which can be used for coupling to other models. In order to validate the approach for engineering applications, higher-harmonic wave forces on a vertical cylinder are considered as compared to experiments, as well as preliminary work with moving coordinate systems considering the wakes caused by a surface-piercing foil.

I – Introduction

Boundary integral methods have seen extensive use for wave-structure interaction problems since the work of Longuet-Higgins and Cokelet [16]. Although for small amplitude waves and linear motions there exist many models which produce practical engineering results, there are still many cases where nonlinear effects are important, and fully nonlinear potential flow models, when efficient, can be useful. The standard approach involves solving the Laplace equation for the velocity potential at each time step (and optionally for the time-derivative of the velocity potential), and then updating the mesh and free surface boundary conditions with a mixed Eulerian-Lagrangian approach.

We consider here a model based on the same approach as the wave model of Grilli et al. [7], which has been successful at modeling many wave phenomena, including landslide-generated tsunami, rogue waves, and the initiation of wave breaking caused by bathymetry. For the types of applications considered, Grilli et al. were able to consider structured grids, which enabled simpler approaches for setting up high-order (cubic) elements. In order to more easily tackle more complex grids and surface-piercing fixed or floating bodies, a variation has been developed in recent years [11] using unstructured meshes. Additionally, the model efficiency was improved by using the Fast Multipole Method (FMM [6]) working in parallel for larger grids [12]. The model was initially validated for wave propagation as well as radiation and diffraction from vertical cylinders and more recent improvements have been made to improve the accuracy of these results. For example, when computing the internal solution within the domain, which is required for coupling to other models, the numerical integration of the boundary integrals has high errors for points near the domain boundary, such as the free-surface or rigid bodies. Adaptively subdividing the elements in this case, we are able to maintain the accuracy of the solution throughout the domain. These advances also allow for higher-accuracy near corners and for considering elements with higher aspect ratios.

For other applications such as the wake of a ship, it is necessary to consider a moving coordinate system, as in Beck and Scorpio [1]. To validate such simulations, we compare results with experimental data, such as that of Metcalf et al. [18] for flow past a vertical surface-piercing NACA 0024 foil at moderate Froude numbers. Application of these improvements to study wave impact on offshore structures (e.g., gravity-based foundations, floating platforms for wind-turbines), and more advanced demonstrations of the model will be presented at the conference.

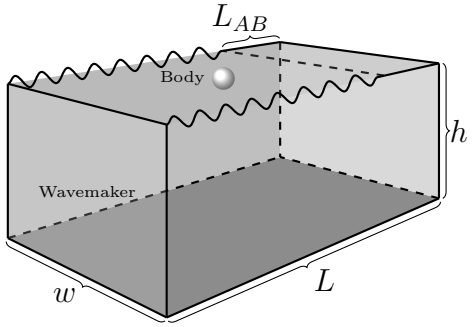


FIGURE 1 – Definition sketch of NWT computational domain for wave interaction with a rigid body (length L by width w by depth h). No-flow conditions are specified on lateral and bottom boundaries ; waves are generated on the leftward boundary (Neumann boundary condition for known velocity and acceleration) and are damped on the far end of the NWT over an absorbing beach (AB) of length L_{AB}.

II – Methods

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

$$\nabla^2\phi = 0 \qquad \text{in D} \tag{1}$$
$$\mathbf{u} = \nabla\phi \qquad \text{in D} \tag{2}$$

with $\mathbf{u}$ the flow velocity in domain D (e.g., Fig. 1). If we use a semi-Lagrangian approach on the free-surface where points are fixed in the horizontal direction, the material derivative for a point following the free-surface would be :

$$\frac{\delta}{\delta t} = \frac{\partial}{\partial t} + \frac{\partial \eta}{\partial t}\frac{\partial}{\partial z} \tag{3}$$

where η denotes the vertical position of the free surface. From this, we can fully express the kinematic and dynamic free surface boundary conditions in Cartesian coordinates, in a reference frame that is potentially moving in the x-direction at a speed, $U(t)$, as :

$$\frac{\delta \eta}{\delta t} = \frac{\partial \phi}{\partial z} - \frac{\partial \phi}{\partial x}\frac{\partial \eta}{\partial x} - \frac{\partial \phi}{\partial y}\frac{\partial \eta}{\partial y} - U(t)\frac{\partial \eta}{\partial x} \tag{4}$$
$$\frac{\delta \phi}{\delta t} = -g\eta - \frac{1}{2}\nabla\phi\cdot\nabla\phi + \frac{\partial \eta}{\partial t}\frac{\partial \phi}{\partial z} - U(t)\frac{\partial \phi}{\partial z} \tag{5}$$

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

On fixed, submerged, or surface-piercing bodies, the boundary condition is simply a no-flow condition on the body boundary (or hull) Γ_h, whereas for bodies moving relative to the coordinate system, with fixed 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. Numerically damping waves exiting at the edge of the domain is handled by adding a term $-\nu(x)\eta$ and $-\nu(x)\phi$ to the right side of Eq. 4 and 5, respectively, where $\nu = 0$ for all of the domain except for points $x \geq x_{AB}$, where $\nu(x) = \nu_0 \left(\frac{x-x_{AB}}{L_{AB}}\right)^2$.

Hydrodynamic forces and moments acting on the rigid body are computed by integrating the hydrodynamic pressure. 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 [7], $\partial\phi/\partial t$ is also computed with a BEM. For freely moving bodies, however, both BEMs 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. [10] for a review and details).

II – 1 BEM solution of the Laplace equation

As indicated above, Laplace's Eq. 1 is solved as a BIE expressed at each collocation point $\mathbf{x}_i$ (or $[x_i, y_i, z_i]$ for $i = 1, \ldots, N$),

$$\alpha(\mathbf{x}_i)\phi(\mathbf{x}_i) = \int \left[\frac{\partial \phi}{\partial n}(\mathbf{x})G(\mathbf{x}, \mathbf{x}_i) - \phi(\mathbf{x})\frac{\partial G}{\partial n}(\mathbf{x}, \mathbf{x}_i)\right] d\Gamma \tag{6}$$

where G is the free space Green's function based on the distance to the target point i, $\mathbf{r}_i = \|\mathbf{x}_i - \mathbf{x}\|$, α is the exterior solid angle made by the boundary at a collocation point i (e.g., for a smooth boundary this would be 2π), and $\mathbf{n}$ points in the direction of the local outwards normal vector to the boundary. In 3D, the free space Green's function of Laplace's equation and its normal derivative read :

$$G(\mathbf{x}, \mathbf{x}_i) = \frac{1}{4\pi\|\mathbf{r}_i\|} \tag{7}$$

$$\frac{\partial G}{\partial n}(\mathbf{x}, \mathbf{x}_i) = -\frac{1}{4\pi}\frac{\mathbf{r}_i \cdot \mathbf{n}}{\|\mathbf{r}_i\|^3} \tag{8}$$

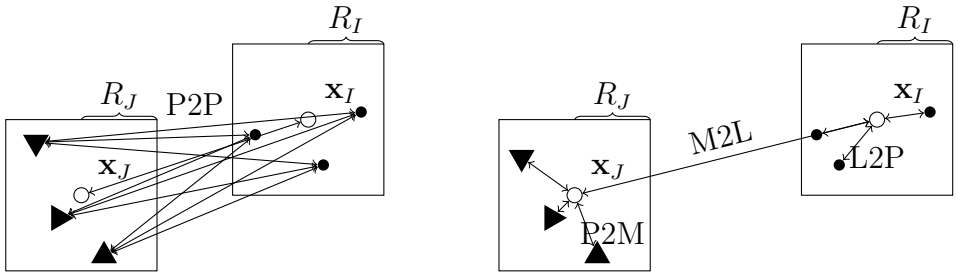
Solving Eq. 6 with a BEM discretization requires : (i) integrating complicated integral kernels over individual boundary elements, which become singular when $\mathbf{r}_i \to 0$; and (ii) solving the resulting (typically N by N) linear system of algebraic equations. The BEM integrals are performed over each triangular element using Dunavant's [4] rules, and quadrangular elements with a tensor product of Gauss integration. For linear triangular elements, singular integrals can then be analytically integrated. Although analytical solutions of non-singular integrals are known for linear triangular elements, to allow for a more straightforward extension of the existing formulation of the NWT to higher-order elements in future work (e.g., [7]), we compute these integrals numerically. The coefficients α in the BIE are found by applying the rigid mode method (e.g., [9]), which expresses that for a Dirichlet problem with $\phi = 1$ specified over the entire boundary of domain D, the discretized BIE must yield $\partial\phi/\partial n = 0$; the α coefficients are then found as the residuals of this Dirichlet problem. The solution of the BEM discretized algebraic system is then solved with BiCGSTAB, a Krylov iterative solver.

In the nonsingular integrals, as the free space Green's function Eq. 7 varies rapidly when a point is close to the element being considered (i.e., $\mathbf{r}_i \to 0$), an adaptive integration approach is used, both for collocation points belonging to the boundary discretization or for internal points. The method used is quite similar to that described by Grilli et al. [7] for the same purpose, but here we consider a simpler distance criterion : when the point under consideration is closer to the center of the element than twice the maximum element edge length, the element is recursively divided into four smaller elements, and this recursive process is done up to 16 times. For instance, if we consider the solution of the flow past a cylinder with a known theoretical boundary solution and compute the velocity potential for points very near the body boundary, the error is found to be quite significant for those, but becomes small when an adaptive subdivision is used.

In the FMM, the free space Green's function is approximated for "distant" points by a truncated (order P) multipole expansion.

$$G(\mathbf{x}, \mathbf{x}_i) \approx \sum_{m_x=m_y=m_z=0}^{m_x+m_y+m_z=P} \frac{(x-x_i)^{m_x}(y-y_i)^{m_y}(z-z_i)^{m_z}}{m_x!m_y!m_z!}\left[\left(\frac{\partial}{\partial x}\right)^{m_x}\left(\frac{\partial}{\partial y}\right)^{m_y}\left(\frac{\partial}{\partial z}\right)^{m_z} G(\mathbf{x}, \mathbf{x}_i)\right] \tag{9}$$

In this approach, both interactions (e.g., Fig. 2) that are "distant enough" are neglected (yielding a sparse matrix ; e.g. Fig. 3) and the full system matrix of the BIE does not have to be assembled, which is typically the most time consuming part of the NWT, as it has a $O(N^2)$ complexity. More specifically, to decide how to approximate (or even neglect) interactions, the FMM uses a divide-and-conquer strategy based on the distance between two points which is detailed in Fig. 2. Importantly, by assigning intermediate points



(a) $\|\mathbf{x}_I - \mathbf{x}_J\|/(R_I + R_J) < \theta$ (a) $\|\mathbf{x}_I - \mathbf{x}_J\|/(R_I + R_J) \geq \theta$

FIGURE 2 – The BEM problem can be considered as the interaction of many elements (triangles, above) and nodes (circles), independent of the mesh information. Given a partitioning of the grid, the BEM coefficients between a group of elements and nodes are computed with the FMM approximation when the center $\mathbf{x}_J$ of the group of elements and the center $\mathbf{x}_I$ of a group of nodes are far from each other, relative to the size of each group, R_I and R_J, comparing this relative distance, $\|\mathbf{x}_I - \mathbf{x}_J\|/(R_I + R_J)$, against a multipole acceptance criterion, θ, resulting in : (a) *local* computations where all interactions are computed by P2P (traditional BEM) ; (b) *distant* computations where all interactions are computed by multipole approximation, through P2M, M2L, and L2P operations.

(e.g., the centers of groups of nodes or elements of the boundary mesh) and applying the binomial theorem, we are able to manipulate multipole coefficients that only need to be computed once, instead of directly evaluating the BIE between each element and node.

Thus, in the FMM algorithm, for a cell (i.e. group) C_J containing a list of elements j, with a cell center x_J, we first construct the coefficients of the multipole series $\mathbf{M}^{m_x,m_y,m_z}$ ($0 \leq m_x + m_y + m_z \leq P$) of order P for all elements j (P2M) in the cell, based on introducing Eq. 9 into the BIE (Eq. 6), as,

$$\begin{aligned}\mathbf{M}^{m_x,m_y,m_z}(\mathbf{x}_J) = \sum_{j:\mathbf{x}_j \in C_J} \frac{1}{m_x! m_y! m_z!} \int \Big(\Big[(x_J - x)^{m_x} (y_J - y)^{m_y} (z_J - z)^{m_z} \Big] \frac{\partial \phi}{\partial n} \\ - \frac{\partial}{\partial n_x} \Big[(x_J - x)^{m_x} (y_J - y)^{m_y} (z_J - z)^{m_z} \Big] \phi \\ - \frac{\partial}{\partial n_y} \Big[(x_J - x)^{m_x} (y_J - y)^{m_y} (z_J - z)^{m_z} \Big] \phi \\ - \frac{\partial}{\partial n_z} \Big[(x_J - x)^{m_x} (y_J - y)^{m_y} (z_J - z)^{m_z} \Big] \phi \Big) d\Gamma_j \end{aligned} \quad (10)$$

Next, these multipole expansions are transformed into local expansions which correspond to the coefficients of a polynomial which approximates the BIE of a distant region. This is called the multipole to local (M2L) operator :

$$\mathbf{L}^{k_x,k_y,k_z}(\mathbf{x}_I) = \sum_{\substack{m_x=P-k_x \\ m_y=P-k_y \\ m_z=P-k_z}}^{m_x=m_y=m_z=0} \left[\left(\frac{\partial}{\partial x} \right)^{m_x+k_x} \left(\frac{\partial}{\partial y} \right)^{m_y+k_y} \left(\frac{\partial}{\partial z} \right)^{m_z+k_z} G(\mathbf{x}_I, \mathbf{x}_J) \right] \mathbf{M}^{m_x,m_y,m_z}(\mathbf{x}_J) \quad (11)$$

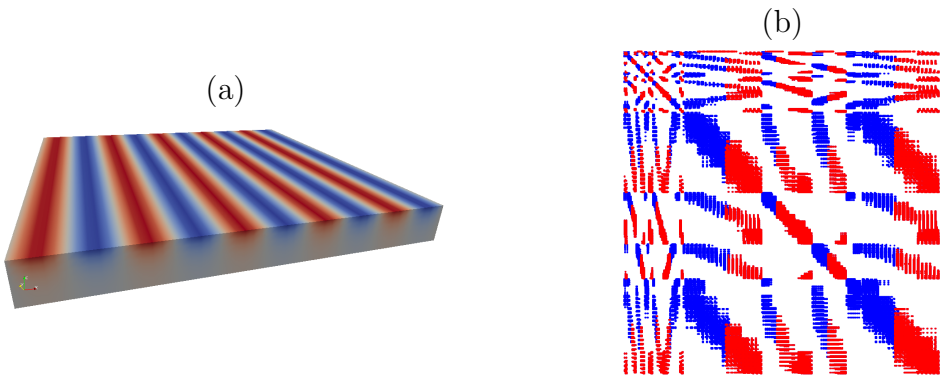


FIGURE 3 – (a) Typical domain for BEM calculations ; and (b) local system matrix structure, showing the non-zero values computed in the assembly phase of the Laplace solver, with values in red and blue corresponding to that computed on the first and second processors, for a problem split into two.

Finally, this local expansion (L2P) is evaluated as :

$$\Phi(\mathbf{x}_i) = \sum_{k_x=k_y=k_z=0}^{k_x+k_y+k_z=P} \frac{(x_i - x_I)^{k_x}(y_i - y_I)^{k_y}(z_i - z_I)^{k_z}}{k_x!k_y!k_z!}\mathbf{L}^{k_x,k_y,k_z}(\mathbf{x}_I) \tag{12}$$

BEM coefficients are computed with the FMM approximation when cells of boundary elements, with center $\mathbf{x}_J$ and cell radius R_J, and cells of observation points, with center $\mathbf{x}_I$ and cell radius R_I, are far from each other, based on a multipole acceptance criterion, $\|\mathbf{x}_I - \mathbf{x}_J\|/(R_I + R_J) > \theta$ (Fig. 2). The resulting error is thus $O(\theta^P)$.

In our model, we use the ExaFMM library [22], which computes all FMM interactions with a nearly order N scaling and has an efficient parallel implementation. Calculating interactions with Eq. 10 is in fact the only modification we make to the existing FMM algorithm, and this only requires integrating a polynomial expression on a BEM element.

III – Applications

III – 1 Scaling of Laplace solver

Typically, the computational time of the FMM scales as $O(N)$, and is faster than a direct calculations (which require $O(N^2)$ time with the best iterative solver) for problem sizes more than a few thousand unknowns. Several variations of the parallel FMM have been developed [22], but they generally rely on domain decomposition, where each processor first applies the FMM on some region of space, and then the results being combined. While this permits good scaling to hundreds of processors for a billion unknowns [23], for boundary element problems that require high accuracy, where one might have 10^5 unknowns and 10^2 processors, the approach described above, referred to as single-level FMM, can scale much more efficiently. This was shown by Waltz et al. [21] who compared it with many other parallelization attempts of the FMM-BEM and is due to the fact that while the number of unknowns is large enough for the FMM to be efficient, the number of unknowns per processor is low.

As the NWT domain moves at each time step, and the performance of the FMM depends on the geometry of the problem, to study more precisely the scaling of its numerical solution, we consider the two main phases of an iterative solver : (sparse) matrix assembly of the local system (e.g., Fig. 3), and a matrix-vector product that involves the FMM. We

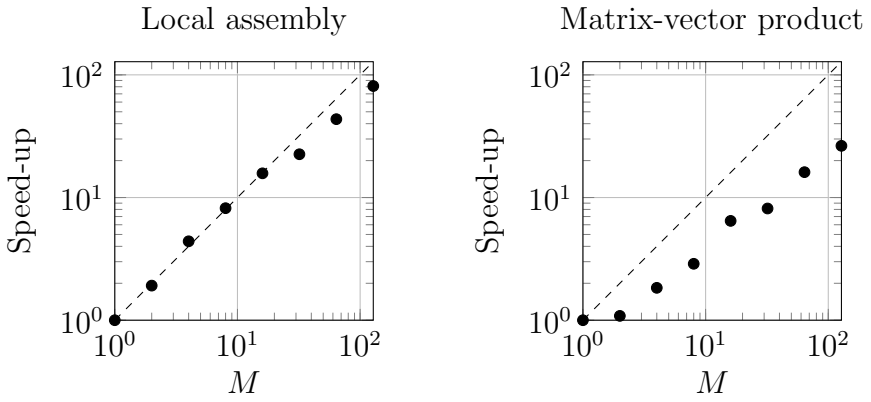


FIGURE 4 – FMM-BEM scaling of the assembly and matrix-vector products used in the iterative solution of the Laplace equation for a mesh with quadrangular elements and 79,202 nodes, with varying number of CPUs, M. The grid is partitioned into 1,024 subdomains, with $\theta = 0.4$ and a 15th order FMM expansion ; 100 integration points were used on each element.

do not consider the entire solver at once, as the number of iterations changes slightly with problem geometry and size, but is not affected by the parallelization ; typically around 100 iterations are required to converge to a solution, however.

If we take a quadrangular grid of 79,202 nodes, with dimensions $4.0 \times 2.0 \times 0.5$ (corresponding to a box of dimensions similar to that found in a wave propagation problem ; e.g., Fig. 3a), partitioned into 1,024 subdomains, we are able to show good scaling up to $M = 128$ processors (Fig. 4). Further study of the individual routines shows that each routine scales well with the division of computations between different processes, but eventually the amount of communication between processors penalizes the computational time for the matrix-vector product computations and the scaling with M deteriorates.

III – 2 Internal velocities

Based on the boundary integral equation (Eq. 6), which directly computes the velocity potential, we can write a similar BIE for the internal velocity :

$$\mathbf{u} = \nabla\phi(\mathbf{x}_i) = \int \left[\frac{\partial \phi}{\partial n}(\mathbf{x}) Q(\mathbf{x}, \mathbf{x}_i) - \phi(\mathbf{x}) \frac{\partial Q}{\partial n}(\mathbf{x}, \mathbf{x}_i) \right] d\Gamma \tag{13}$$

where we have :

$$Q(\mathbf{x}, \mathbf{x}_i) = \frac{\mathbf{r}}{4\pi \|\mathbf{r}_i\|^3} \tag{14}$$

$$\frac{\partial Q}{\partial n}(\mathbf{x}, \mathbf{x}_i) = -\frac{1}{4\pi} \frac{\mathbf{n} - 3(\mathbf{r}_i \cdot \mathbf{n})(\mathbf{r}/\|\mathbf{r}\|^2)}{\|\mathbf{r}_i\|^3} \tag{15}$$

which can be solved with the same method as used for the boundary solution.

One application of the computation of internal velocities from this NWT can be seen in certain types of model coupling (e.g., Harris and Grilli [13]). In order to verify that the solution produced is accurate, let us consider the flow around a Karman-Trefftz foil. In two dimensions (2D) the solution can be analytically expressed based on a conformal mapping :

$$Z = n\lambda \frac{(1 + \frac{\lambda}{z})^n + (1 - \frac{\lambda}{z})^n}{(1 + \frac{\lambda}{z})^n - (1 - \frac{\lambda}{z})^n} \tag{16}$$

which maps a cylinder of radius r centered at s on the horizontal axis x to a symmetric foil, where $n = 2-\beta/\pi$, with β the angle at the trailing edge, and $\lambda = r+s$; selecting $r = 1$ and the foil center at $s = -0.045$ we have $\beta = 8^o$. We consider a rectangular domain, with extension $-100 < x < 100$, $-100 < y < 100$, and solves this 2D flow in three dimensions in our model, using a transverse direction extending from, $-15 < z < 0$. Discretizing this domain with 15,488 quadrangular elements, and using 15th-order multipole expansions, we computed the internal velocity with Eq. 13 at 10,000 points located near the foil surface. Comparing the numerical results to the analytical solution we found maximum errors in velocity of 0.45%, occurring at the very tip of the foil on the boundary, and a L^2-error of 10^{-6} over the entire internal points (Fig. 5).

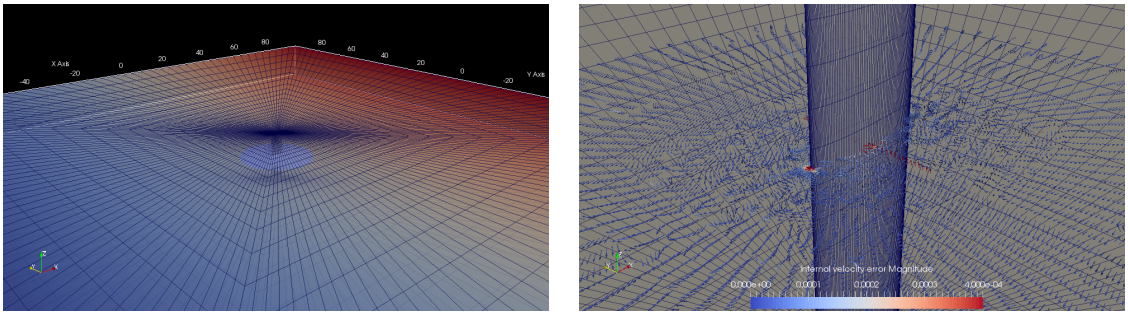


FIGURE 5 – Errors of velocities computed for flow around a Karman-Trefftz foil, using a grid of 15488 quadrangular elements.

III – 3 Wave-structure interaction

For time dependent problems such as wave-structure interaction, at any given time, the BEM solution provides both the velocity potential and its normal derivative on the computational domain boundary, as discrete values at the N (collocation) points of the grid. This solution (both geometry and boundary condition) is then advanced in time on the free surface by integrating the free surface boundary conditions, Eq. 4 and 5. Similar to Grilli et al. [9, 7], this in done using two explicit second-order Taylor series expansions for the boundary geometry and potential. Some quantities in these expansions (such as the velocity) require computing derivatives of the solution along the free surface mesh. For a given boundary point, this is done by averaging the values of the derivatives computed in all intersecting neighboring elements.

As an example, we consider the interaction of deep water nonlinear periodic waves with a bottom-mounted vertical cylinder, as tested by Huseby and Grue [14] in their laboratory experiments. They generated incident waves by 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., re-reflected by the wavemaker). In the NWT, similar to other authors [5, 19], we specify incident nonlinear periodic waves (as in other applications above), rather than model wave generation by the wavemaker itself. We simulate a range of wave steepness kA, for a cylinder of radius R, such that $kR = 0.245$ (Fig. 7), where waves have an amplitude A, a wavenumber $k = 2\pi/\lambda$, with λ is the wavelength. Similar to Grilli and Horrillo's [8] 2D-NWT, fully nonlinear periodic waves are generated in the 3D-FNPF-NWT by specifying their kinematics on the NWT's leftward boundary (Fig. 1), based on the streamfunction wave theory [3]. As in [8], these are generated over an opposite current with a velocity that is iteratively computed to create waves with a period-

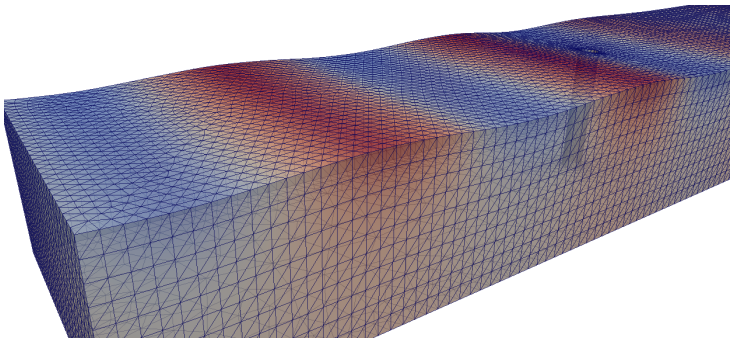


FIGURE 6 – Snapshot of wave diffraction around bottom-mounted cylinder, computed in the present NWT, after achieving a quasi-steady state ($t > 6T$). Incident waves are (zero-mass-flux) streamfunction waves in deep water, with $kA = 0.15$ and $kR = 0.245$.

averaged zero-mass-flux ; this allows creating the equivalent of an impermeable wavemaker boundary and keeping the mean NWT volume constant (except for numerical errors).

In the NWT, the computational domain is similar to that shown in Fig. 1, with dimensions $4.04\lambda \times 0.93\lambda \times 0.5\lambda$, a one-wavelength long AB, and a discretization with $N = 23,827$ collocation nodes and $M = 47,686$ triangular elements (Fig. 6) ; 8 wave periods are simulated. Note, the NWT depth does not exactly correspond to that in experiments, but since deep water incident waves are modeled, this should not affect the force computation. When simulations reach a quasi-steady state, for $t > 6T$, we compute the horizontal force $f(t)$ acting on the cylinder, and compare the amplitudes of its first three harmonics, for $t/T = 6$ to 8, to those of the experimentally measured force.

Fig. 7 shows a good agreement of numerical results with experiments, particularly for the first and third harmonics, with larger errors for the second harmonics, although the expected trend with kA is observed. The discrepancy in the second-order harmonics was also found in other numerical results (e.g., [5]).

III – 4 Flow past a surface-piercing foil

Next we consider the flow past a vertical surface-piercing NACA 0024 foil advancing with velocity U, at moderate Froude numbers Fr = 0.19 and 0.37, and compare numerical results with the experimental data of Metcalf et al. [18]. In the NWT we computed the flow past moving bodies, such as ships, in a relative axes, i.e., as a current moving past the bodies. This requires adjusting the free-surface boundary conditions Eqs. 4 and 5 to include the change of reference frame ; here we assumed a velocity ramp-up, $U(t) = U_0 \tanh t$, which gradually reaches a steady-state flow with uniform velocity U_0 in the x direction. Due to instabilities found when applying the second-order Taylor series integration for time-stepping, we follow the work of Buchmann [2] and consider an explicit-implicit Euler time-stepping scheme, similar to that used in the industrial code Aegir [15], where the kinematic free surface condition is satisfied through an explicit Euler time stepping, and the dynamic free surface condition is satisfied through an implicit Euler time stepping.

Further, as described in other NWTs (e.g. [20]), it was found that small instabilities occurred at the waterline, so we applied a simple smoothing algorithm to the free-surface at each time step. Considering the primitive filtering applied, we results shown in Fig. 8 for the free surface elevation appear reasonable as compared to experiments. The largest differences are seen in the area of the wake observed in experiments, which result from turbulent effects not represented in the potential flow model. Notably, small breaking waves appear for the largest Froude number (Fr = 0.37) that need to be damped ; the

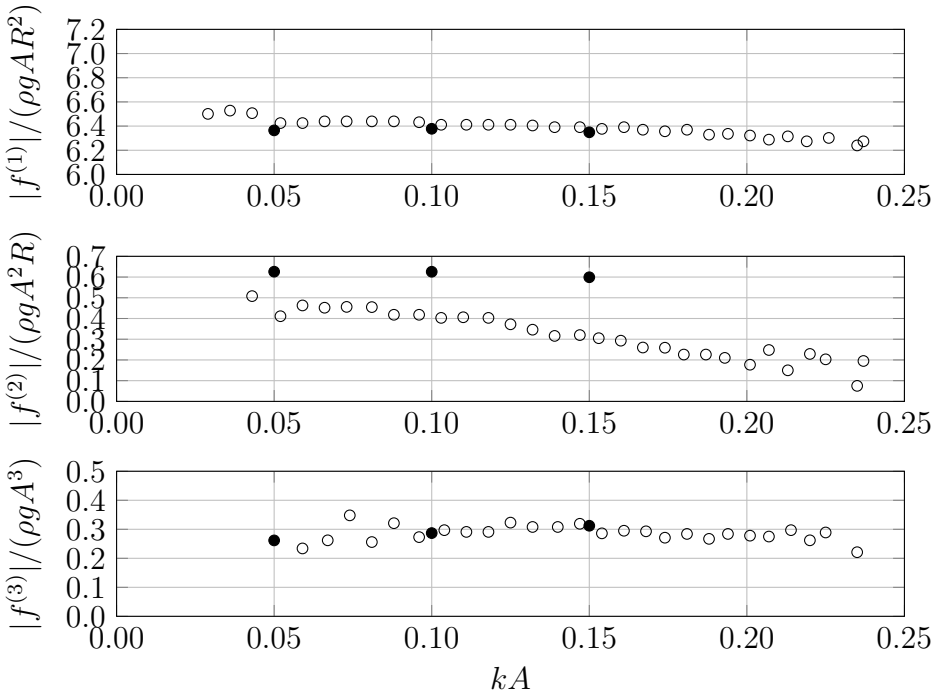


FIGURE 7 – Nondimensional amplitude of first- through third-order harmonic of the horizontal hydrodynamic force $f(t)$ applied to a bottom-mounted cylinder of radius R, computed for deep water periodic waves of wavenumber k, for different wave steepnesses kA, for $kR = 0.245$, in : ($\circ$) Huseby and Grue's [14] experiments ; and ($\bullet$) the present NWT.

inclusion of a smoothing of the velocity potential and free-surface elevation enables the model to produce reasonable wave profiles as well as pressures on the foil, but the strength of the damping needs to be calibrated using some physical criterion. Presently however ad hoc values are applied, resulting in too much damping of short waves and the wake far from the body, but too little damping at larger Froude numbers.

IV – Summary

In this paper, we detailed and illustrated recent advances in the implementation of a numerical wave tank (NWT), which efficiently solves fully nonlinear potential flows (FNPFs) using a FMM-BEM ; in particular regarding scaling of the numerical solution with problem size N and number of processors M in a parallel implementation. We presented results for simple flows past a submerged foil and wave-structure interaction (for fixed and moving surface piercing bodies). Internal velocities are efficiently computed by applying the same FMM-BEM approach, which could be used as part of the coupling of the NWT to other (e.g., Navier-Stokes) models in a perturbation approach solving the complete problem ; this is the object of a companion paper at the conference (O'Reilly et al.). By extending the NWT formulation to a moving coordinate systems and (later) to arbitrary geometries, we will be able to handle a more complex variety of applications of particular interest to Naval Hydrodynamics and ocean engineering. Specifically, although present results were obtained only for linear elements, the overall method is in the process of being extended to higher-order elements (e.g. [17]), which can be achieved without fundamental changes in the NWT formulation. Further examples and more recent results will be shown at the conference.

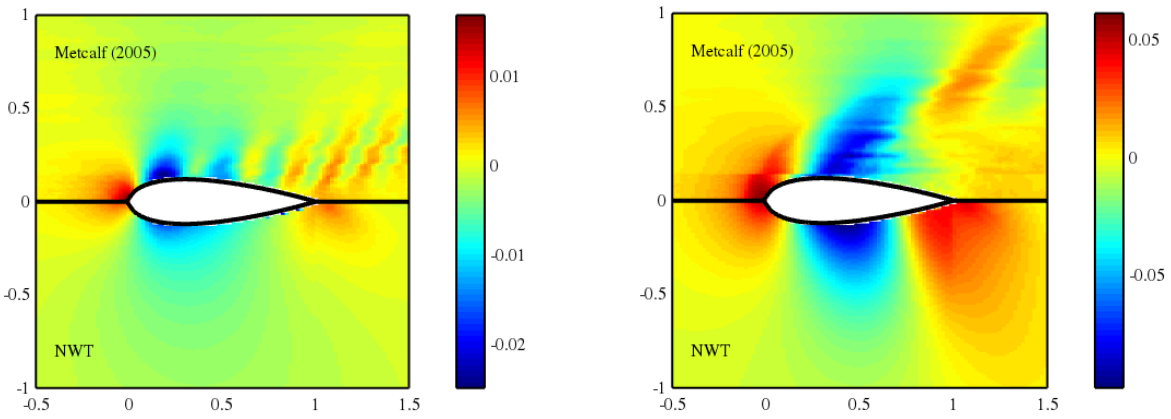


FIGURE 8 – Comparison of experimental data from Metcalf et al. and NWT results for free-surface elevation produced by a surface-piercing vertical NACA 0024 foil moving at different Froude numbers (Fr = 0.19 (left) ; 0.37 (right)).

V – Acknowledgements

Initial work on the project of J. Harris and M. Benoit was 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 and A. Mivehchi acknowledge support from grants N000141310687 and N000141612970 of the US Office of Naval Research (ONR). We would like to thank Dr. Rio Yokota for useful discussions of the fast multipole method.

References

[1] R. F. Beck and S. M. Scorpio. A desingularized boundary integral method for fully nonlinear water wave problems. In *Proceedings of the Twelfth Australasian Fluid Mechanics Conference*, pages 255–258, 1995.

[2] B. Büchmann. 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*, 33 :125–155, 2000.

[3] R. G. Dean. Stream function representation of nonlinear ocean waves. *Journal of Geophysical Research*, 70 :4561–5472, 1965.

[4] D. A. Dunavant. High degree efficient symmetrical Gaussian quadrature rules for the triangle. *International Journal for Numerical Methods in Engineering*, 21 :1129–1148, 1984.

[5] P. Ferrant. Fully nonlinear interactions of long-crested wave packets with a three-dimensional body. In *22nd Symposium on Naval Hydrodynamics*, pages 403–416, 2000.

[6] L. Greengard and V. Rokhlin. A fast algorithm for particle simulations. *Journal of Computational Physics*, 73 :325–348, 1987.

[7] S. T. Grilli, P. Guyenne, and F. Dias. A fully nonlinear model for three-dimensional overturning waves over arbitrary bottom. *International Journal for Numerical Methods in Fluids*, 35 :829–867, 2001.

[8] S. T. Grilli and J. Horrillo. Generation and absoption of fully nonlinear periodic waves. *Journal of Engineering Mechanics*, 123 :1060–1069, 1997.

[9] S. T. Grilli, J. Skourup, and I. A. Svendsen. An efficient boundary element method for nonlinear water waves. *Engineering Analysis with Boundary Elements*, 6 :97–107, 1989.

[10] E. Guerber, M. Benoit, S. T. Grilli, and C. Buvat. A fully nonlinear implicit model for wave interactions with submerged structures in forced of free motion. *Engineering Analysis with Boundary Elements*, 36 :1151–1163, 2012.

[11] J. C. Harris, E. Dombre, M. Benoit, and S. T. Grilli. A comparison of methods in fully nonlinear boundary element numerical wave tank development. In *Proceedings of the 13th Journees de l'Hydrodynamique*, 2014.

[12] J. C. Harris, E. Dombre, M. Benoit, and S. T. Grilli. Fast integral equation methods for fully nonlinear water wave modeling. In *Proceedings of the 24th International Offshore and Polar Engineering Conference*, pages 583–590, Busan, Korea, 2014.

[13] J. C. Harris and S. T. Grilli. A perturbation approach to large-eddy simulation of wave-induced bottom boundary layer flows. *International Journal for Numerical Methods in Fluids*, 68 :1574–1604, 2012.

[14] M. Huseby and J. Grue. An experimental investigation of higher-harmonic wave forces on a vertical cylinder. *Journal of Fluid Mechanics*, 414 :75–103, 2000.

[15] D. C. Kring, F. T. Korsmeyer, J. Singer, D. Danmeier, and J. White. Accelerated nonlinear wave simulations for large structures. In *7th International Conference on Numerical Ship Hydrodynamics*, Nantes, France, 1999.

[16] M. S. Longuet-Higgins and E. Cokelet. The deformation of steep surface waves on water, I. A numerical method of computation. *Proceedings of the Royal Society A*, 350 :1–26, 1976.

[17] J. Maestre, I. Cuesta, and J. Pallares. An unsteady 3D Isogeometrical Boundary Element Analysis applied to nonlinear gravity waves. *Computer Methods in Applied Mechanics and Engineering*, 310 :112–133, 2016.

[18] B. Metcalf, J. Longo, S. Ghosh, and F. Stern. Unsteady free-surface wave-induced boundary-layer separation for a surface-piercing NACA 0024 foil : Towing tank experiments. *Journal of Fluids and Structures*, 22 :77–98, 2006.

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

[20] K. Tanizawa. The state of the art on numerical wave tank. In *Proceeding of 4th Osaka Colloquium on Seakeeping Performance of Ships*, pages 95–114, 2000.

[21] C. Waltz, K. Sertel, M. A. Carr, B. C. Usner, and J. L. Volakis. Massively parallel fast multipole method solutions of large electromagnetic scattering problems. *IEEE Transactions on Antennas and Propagation*, 55(6) :1810–1816, 2007.

[22] R. Yokota. An FMM based on dual tree traversal for many-core architectures. *Journal of Algorithms and Computational Technology*, 7 :301–324, 2013.

[23] R. Yokota, J. P. Bardhan, M. G. Knepley, L. A. Barba, and T. Hamada. Biomolecular electrostatics using a fast multipole BEM on up to 512 GPUs and a billion unknowns. *Computer Physics Communications*, 182 :1272–1283, 2011.