Available online at www.sciencedirect.com
SCIENCE DIRECT®
Journal of Computational Physics 210 (2005) 368–385
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

Unstructured MEL modelling of nonlinear unsteady ship waves

Q.X. Wang *

Nanyang Technological University, Maritime Research Centre, School Civil and Environmental Engineering, 50 Nanyang Avenue, Singapore 630798, Singapore

Received 21 January 2004; received in revised form 14 April 2005; accepted 21 April 2005
Available online 17 June 2005

Abstract

An *unstructured* mixed-Eulerian–Lagrangian model (MEL) is described for the simulation of the nonlinear wave–body interaction, as opposed to the structured model used widely in this field. In present model, the free surface is interpolated accurately using a polynomial scheme coupled with the moving least-squares method, and is triangulated with a local interrogation method coupled with the advancing front method to provide a high quality mesh. A new method is devised to update the free surface intersection with the body, based on the desingularized boundary-integral method. *Double desingularized point sources* are used for a control point at the intersection, with one inside the body and the other above the free surface, to satisfy both the body and free surface boundary conditions over there. The desingularized point sources are positioned in terms of the element sizes as well as solid angles of the boundary surface, so as to avoid possible overcrowding of the sources at the corners of the surface. Solid angles on the surface are computed analytically using the spherical triangle theory. As an illustration, the algorithm is used to simulate the wave generation for a Wigley hull, which impulsively starts to move at a constant speed on a calm water surface. The anticipated Kelvin ship-wave pattern is well simulated. The wave profiles along the hull agree well with the measured results of Nobeless and McCarthy [Proceedings of the 2nd DTNSRDC Workshop, Maryland, 1983, pp. 5–35]. As compared to a structured approach, the unstructured model reduces the CPU time and memory requirements and, being robust in handling complex geometries, is more versatile in practical applications.

Keywords: Wave–body interaction; Kelvin ship-wave; Potential flow theory; Mixed-Eulerian–Lagrangian method; Desingularized boundary-integral method; Unstructured mesh of curved surfaces; Advancing front method; Moving least-squares method

* Tel.: +65 65630309.
E-mail address: cqxwang@ntu.edu.sg.

0021-9991/$ - see front matter © 2005 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2005.04.012

1. Introduction

The interaction between the free surface wave and a floating body poses an interesting theoretical problem of practical importance in naval architecture and ocean engineering. The linearized water wave theory is often sufficient for routine design. But the nonlinear effects must be considered in the case of the extreme loads and motions, which are essential to the critical performance, safety, and even survival of marine vessels and offshore structures (cf. [2–4]).

For a body moving steadily in a calm water surface, nonlinear computations can be performed using an iterative technique to solve a series of linearized boundary value problems, in which each iterative solution is linearized to the solution of the previous iteration [5–8]. It has the advantage of converging to the fully nonlinear solution in lesser CPU time than a time-domain method, and is one of the major tools used for predicting the wave resistance for ships. However, this modelling approach is only suitable for simulating the steady problems.

Longuet-Higgins and Cokelet [9] invented a mixed-Eulerian–Lagrangian method (MEL) to model the transient steep water waves. This method has been applied to a wide variety of water wave problems, both with and without a floating body present. MEL was implemented for the computations of two-dimensional water waves [10–12]. It was developed for the simulations of three-dimensional water waves by Dommermuth et al. [13], Cao et al. [14], Maskew [15–17], Beck et al. [18–21], Liu et al. [22], and Xue et al. [23], among the others. The time-domain scheme is more powerful, since it can be used to simulate the nonlinear transient wave–body interactions. Compared to the three-dimensional domain approaches [24–26], MEL reduces the dimensions of the problem by one and avoids the tedious volumetric mesh. To date, the nonlinear transient three-dimensional results are still limited by theoretical complexity and computational burden. Reviews on this topic may be found in [3,4].

The above MEL studies for both the steady and unsteady water wave problems were based on a structured mesh. Although a structured mesh is relatively easy be generated, it is difficult to obtain a good quality mesh and to generate it automatically for a complex surface. An unstructured approach is presented in this work alternatively based on the following considerations. Firstly, the simulation of ship waves is associated with a *large computational domain of the free surface*, around 5×3 hull lengths in the stream-wise and span-wise directions, respectively. In addition, it is a *multi-scaled problem*, because the length of a ship hull is usually one order larger than its draft. The physical scale of the free surface wave near the hull is of the ship's draft, where the free surface wave is steep, and is of the wavelength of the ship Kelvin wave far away, which is at the order of the ship's length. The free surface can be triangulated with a smooth and variable mesh density distribution. The element size of the free surface near a ship hull can be chosen being small compared to the draft, gradually becoming coarser away from the hull, and being small compared with the wavelength far away. Secondly, an unstructured mesh is generally more robust than a structured one; a complex surface can be triangulated automatically [27]. An unstructured approach is therefore more versatile in practical applications.

In the MEL model, the free surface has to be interpolated and meshed at every time step. As the process needs to be repeated hundreds of times, a high quality mesh and an accurate interpolation for the free surface are necessary to keep the accuracy. In the present work, a polynomial scheme coupled with the moving least-squares method is implemented to interpolate accurately the free surface. A local interrogation method coupled with the advancing front method is devised to generate the high quality unstructured mesh of the free surface.

Proper updating of the intersection between the body and free surface is critical to the prediction of the wave run-up along and wave loads on the body. Both the body and free surface boundary conditions need to be satisfied at the intersection and, consequently, the problem becomes overdetermined over there. Maskew [15–17] avoided the problem by using boundary elements with constant distributions of sources and doublets and choosing the control points at the centres of the elements. The accuracy of this approach

is however compromised at the intersection. Cao et al. [14], Beck [21], and Xue et al. [23], among others used double control points at a grid at the intersection, satisfying the body and free surface boundary conditions, respectively. Their approach, in principle, results in the discontinuity of the velocity potential across the intersection.

An alternative method is devised here to update the intersection based on the desingularized boundary-integral method for ship waves [14,21]. *Double desingularized point sources* are chosen for a control point at the intersection, one inside the body and the other above the free surface. Their strengths are determined by enforcing both the body and free surface boundary conditions at the control point. The resulting solution of the velocity potential is smooth across the intersection, and the overdetermined problem is resolved.

The remainder of the paper is organized as follows. The mathematical formulation of the water wave problem is briefed in Section 2. The unstructured MEL model is described in Section 3, in which particular attentions are paid to the new numerical techniques introduced in this work. The interpolation and triangulation of the free surface are described in Sections 4 and 5, respectively. In Section 6, the algorithm is used to simulate the wave generated by a Wigley hull piercing the water surface. The wave profiles along the hull are compared with the testing result. Section 7 contains the summary and conclusions.

2. Mathematical formulation

Consider a ship moving on a free surface, as shown in Fig. 1. A Cartesian coordinate system o-xyz fixed to the body is built, with its z-axis in the opposite direction of gravity, plane $z = 0$ at the undisturbed free surface, and x-axis in the longitudinal direction towards the stern of the ship. We assume that the fluid is inviscid and incompressible, and the flow induced is irrotational. A velocity potential exists in the fluid domain, bounded by the wetted body surface and free surface. The potential can be decomposed as, $\mathbf{V}_b \cdot \boldsymbol{r} + \varphi(\boldsymbol{r}, t)$, where $\boldsymbol{r} = (x, y, z)$, and $\boldsymbol{V}_\mathrm{b} = (U_\mathrm{b}, V_\mathrm{b}, W_\mathrm{b})$ is the body velocity. The disturbance velocity potential φ satisfies the Laplace equation in the fluid domain

$$\Delta\varphi = 0, \tag{1}$$

subjected to the following boundary conditions. φ vanishes at infinity. It satisfies the impermeable boundary condition on the wetted surface of the body S_B

$$\frac{\partial\varphi}{\partial n} = \boldsymbol{V}_\mathrm{b} \cdot \boldsymbol{n}, \quad \text{as } \boldsymbol{r} \text{ on } S_\mathrm{B}, \tag{2}$$

where $\boldsymbol{n}$ is the unit outward normal vector of the fluid domain on the body surface.

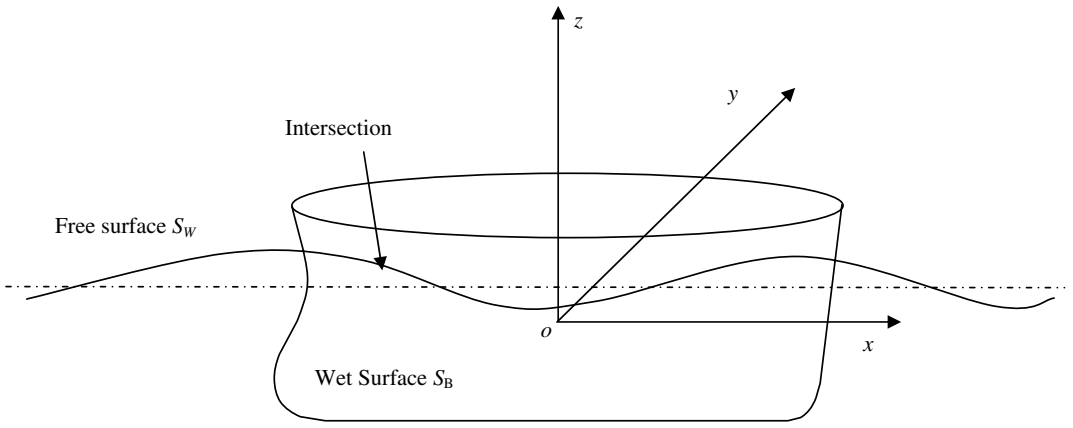

Fig. 1. The Cartesian coordinate system o-xyz and notations for a ship hull moving at a free surface.

The evolution of the free surface S_F is governed by the kinematic and dynamic boundary conditions, requiring a point on the surface to remain on the surface and the pressure to be continuous across the surface, respectively,

$$\frac{d\boldsymbol{r}}{dt} = \boldsymbol{V}_b + \nabla\varphi, \quad \text{as } \boldsymbol{r} \text{ on } S_F, \tag{3}$$

$$\frac{d\varphi}{dt} = \frac{1}{2}|\nabla\varphi|^2 - gz \quad \text{as } \boldsymbol{r} \text{ on } S_F, \tag{4}$$

where g is the gravitational acceleration.

An oval shape computational domain of the free surface is chosen in the unstructured approach, having only upstream and downstream edges, as shown in Fig. 2. Compared to a rectangle truncated free surface used in a structured approach, as shown by dotted lines in the figure, the relatively farer portions are further truncated, leaving a smaller computation domain. Following Cao et al. [14], zero free surface elevation and zero disturbance velocity potential are imposed on the upstream edge of the computational domain. The grids are dropped, when they move out of the computational domain in the downstream.

Suitable initial conditions are required in the MEL modelling. The quiescent initial condition is usually set for a body piercing on a calm water surface

$$\varphi(\boldsymbol{r}, 0) = 0. \tag{5}$$

Apparently, the initial condition (5) is not compatible with the impermeable body boundary condition (2). The incompatible problem exists too for a wave diffraction problem, where the initial condition is usually set at the prescribed incident wave field [23]. To resolve this difficulty, two treatments were implemented and evaluated: (i) allowing the body speed to accelerate from rest to the actual value [14,21]; and (ii) allowing the body boundary to possess an initial permeability which gradually vanishes [17,23].

Following Maskew [17], we implemented a sieve technique to allow the body boundary to be fully permeable initially at $t = 0$, and to become impermeable gradually over a short period. The boundary condition (2) on the wetted surface of the body S_B is adjusted as

$$\frac{\partial\varphi(\boldsymbol{r}, t)}{\partial n} = S_v(t)\boldsymbol{V}_b \cdot \boldsymbol{n}, \quad \text{as } \boldsymbol{r} \text{ on } S_B, \tag{6}$$

where the sieve function $S_v(t)$ is chosen as a smooth function

$$S_v(t) = \begin{cases} 3\left(\frac{t}{t_c}\right)^2 - 2\left(\frac{t}{t_c}\right)^3, & \text{as } 0 \leqslant t \leqslant t_c, \\ 1, & \text{as } t > t_c, \end{cases} \tag{7}$$

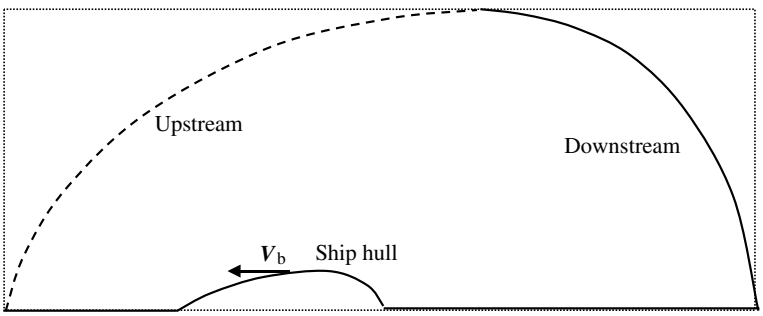

Fig. 2. Sketch of an oval-type computational domain of the free surface, with only upstream and downstream edges, for a ship moving on a water surface.

and t_c is the end time of the sieve treatment, which, in this work, is set at $t_c = 0.3L/U_b$. Under this setting, the ship hull gradually becomes impermeable after it moves $0.3L$.

3. Unstructured MEL method

The MEL model consists of two sub-steps at every time step. In sub-step (i), the wetted body surface S_B and free surface S_F are known with the solutions of the previous time step. The potential φ on the free surface S_F equals to the solution of the previous time step $\varphi_F^n(\boldsymbol{r}, t)$

$$\varphi(\boldsymbol{r}, t) = \varphi_F^n(\boldsymbol{r}, t), \quad \text{as } \boldsymbol{r} \text{ on } S_F. \tag{8}$$

φ is then obtained by solving the Cauchy problem (1), (6), and (8) using the boundary-integral method (BIM). In sub-step (ii), the free surface S_F and potential φ on it are updated by integrating the kinematic and dynamic boundary conditions (3) and (4) on the free surface in time.

3.1. Double-source technique for modelling intersections

The updating of the intersection between the body and free surface is a challenging problem in the MEL model. Not only the body boundary condition (6) but also the free surface boundary condition (8) has to be satisfied at the intersection. As the result, the problem is overdetermined over there. A choice has to be made between satisfying the body boundary condition alone at the intersection and satisfying the free surface condition alone. Maskew [15–17] avoided this problem using the boundary elements with uniform distributions of sources and doublets, and choosing the control points at the element centres. Having avoided the direct modelling of the intersection, his algorithm is quite stable, but the accuracy is compromised at the intersection, where the physical quantities have to be obtained by extrapolation.

Cao et al. [14], Beck [21], and Xue et al. [23], among others used double control points at a grid at the intersection, a body control point and a free surface control point. The potential at the body control point, satisfying the body boundary condition, is obtained by solving the Cauchy problem of (1), (6), and (8). The potential at the free surface control point, satisfying the free surface boundary condition, is obtained by integrating the dynamic boundary condition of the free surface (4) in time. As the result, the potential is discontinuous across the intersection. In principle, this is incompatible with the basic continuity assumption of the velocity potential.

An alternative method is presented here to update the intersection, based on the desingularized BIM. In line with the conventional BIM, the single-layer potential is introduced over the boundary surface of the fluid domain S

$$\varphi(\boldsymbol{r}, t) = \frac{1}{4\pi} \int_S \frac{\sigma(\boldsymbol{r}_s, t)}{|\boldsymbol{r} - \boldsymbol{r}_s|} \, \mathrm{d}S, \tag{9}$$

where σ is the unknown source density, $\boldsymbol{r}$ is the field point, and $\boldsymbol{r}_s$ is the source point.

In a desingularized BIM, the integral surface S in (9) is positioned slightly off the boundary surface outside the fluid domain. After the boundary surface is meshed, all the grids on it are chosen as the control point $\boldsymbol{r}_i^c$. For each control point $\boldsymbol{r}_i^c$, a desingularized point source $\boldsymbol{r}_i^s$ is placed slightly off the fluid domain, as shown in Fig. 3. Double desingularized point sources are chosen for a control point $\boldsymbol{r}_c$ at the intersection, one at $\boldsymbol{r}_{sb}$ inside the body and the other at $\boldsymbol{r}_{sf}$ above the free surface. Their strengths are determined by enforcing both the body and free surface boundary conditions at the control point $\boldsymbol{r}_c$. Consequently, the overdetermined problem is resolved, the potential solution is smooth across the intersection, and both of the two boundary conditions are satisfied over there simultaneously.

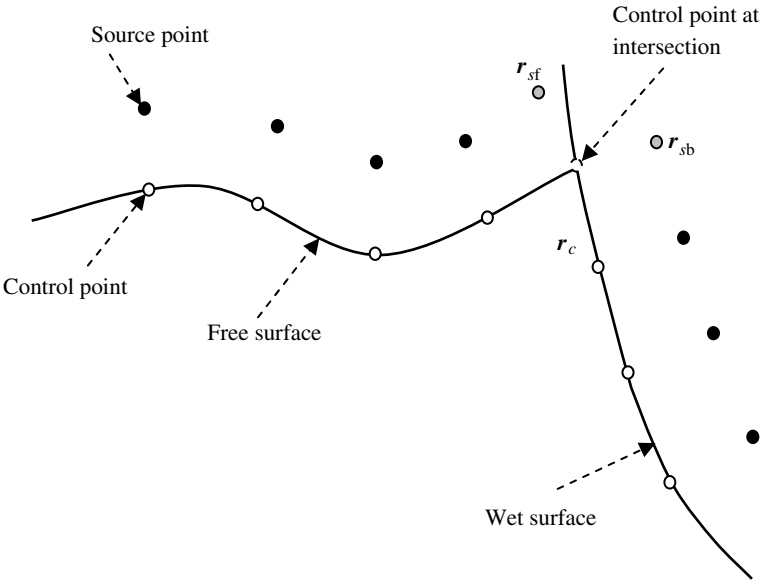

Fig. 3. Sketch of the desingularized point sources, with double point sources (marked in gray circles) used for a control point at the intersection between the body and free surface, one at r_{sb} inside the body, and the other at r_{sf} above the free surface.

The desingularization replaces the integral (9) by a summation of point sources

$$\varphi(\boldsymbol{r},t)=\frac{1}{4\pi}\sum_{j=1}^{N+N_{\mathrm{I}}}\frac{\sigma_j}{|\boldsymbol{r}-\boldsymbol{r}_j^{\mathrm{s}}|}, \tag{10}$$

where N is the number of the grids on the boundary surface, N_{I} is the number of the grids at the intersection. The strengths σ_j are determined by the boundary conditions on the wetted body surface (6) and free surface (8)

$$\frac{1}{4\pi}\sum_{j=1}^{N+N_{\mathrm{I}}}\frac{\sigma_j}{|\boldsymbol{r}_i^{\mathrm{c}}-\boldsymbol{r}_j^{\mathrm{s}}|}=\varphi_i,\quad \text{as } \boldsymbol{r}_i^{\mathrm{c}} \text{ on } S_{\mathrm{F}}, \tag{11a}$$

$$\frac{1}{4\pi}\sum_{j=1}^{N+N_{\mathrm{I}}}\frac{\boldsymbol{n}_i\cdot\left(\boldsymbol{r}_i^{\mathrm{c}}-\boldsymbol{r}_j^{\mathrm{s}}\right)}{\left|\boldsymbol{r}_i^{\mathrm{c}}-\boldsymbol{r}_j^{\mathrm{s}}\right|^3}\sigma_j=-S_{\mathrm{v}}(t)\boldsymbol{V}_{\mathrm{b}}\cdot\boldsymbol{n}_i,\quad \text{as } \boldsymbol{r}_i^{\mathrm{c}} \text{ on } S_{\mathrm{B}}. \tag{11b}$$

When the control point $\boldsymbol{r}_i^{\mathrm{c}}$ is at the intersection, both (11a) and (11b) are satisfied, hence (11) has $N+N_{\mathrm{I}}$ equations with $N+N_{\mathrm{I}}$ unknowns, and are solvable. ScaLAPACK was used in the algorithm for solving the linear algebra equations (11) for a high accuracy. Its CPU time is at the order of $\mathrm{O}(N^3)$.

No numerical integration is required in the desingularized BIM. After the source strengths σ_j are determined from (11), the material velocity on the free surface can be obtained by direct differentiation of (10). In contrast, the calculation of the material velocity on the free surface is a tedious task in the conventional BIM [28].

3.2. Positioning desingularized point sources

The desingularized point source $\boldsymbol{r}_j^{\mathrm{s}}$, corresponding to a control point $\boldsymbol{r}_j^{\mathrm{c}}$, is placed at

$$\boldsymbol{r}_j^{\mathrm{s}}=\boldsymbol{r}_j^{\mathrm{c}}+d_{\mathrm{s}}\boldsymbol{n}_j, \tag{12}$$

where $\boldsymbol{n}_j$ is the unit outward normal vector of the boundary surface at $\boldsymbol{r}_j^{\mathrm{c}}$. The shift distance d_{s} is chosen at the order of the local element size

$$d_{\mathrm{s}} = \beta d_{\mathrm{m}}, \tag{13}$$

where d_{m} is the mean distance of the surrounding grids to $\boldsymbol{r}_j^{\mathrm{c}}$. Parameter β was chosen at 0.85, since the accuracy and convergence of the solution is insensitive to β in the range of 0.8–0.9.

The normal vectors $\boldsymbol{n}_j$ of the surface at control point $\boldsymbol{r}_j^{\mathrm{c}}$ need in (12) to position the desingularized sources, but are undefined, after the surface has been meshed. To resolve this problem, $\boldsymbol{n}_j$ is calculated by an weighted average of the normal vectors of its surrounding triangles $\boldsymbol{n}_k^{\mathrm{e}}$

$$\boldsymbol{n}_j = \frac{\sum_{k=1}^{M_{\mathrm{s}}} w_k \boldsymbol{n}_k^{\mathrm{e}}}{\sum_{k=1}^{M_{\mathrm{s}}} w_k}, \tag{14}$$

where M_{s} is the number of the surrounding triangles, and w_k is the weighted function for each of the triangles. The element normal $\boldsymbol{n}_k^{\mathrm{e}}$ can be regarded as the approximate normal vector of the surface at the triangle centre $\boldsymbol{r}_k^{\mathrm{e}}$. w_k thus should decrease with the distance between $\boldsymbol{r}_j^{\mathrm{c}}$ and $\mathbf{r}_k^{\mathrm{e}}$, and is chosen to decease exponentially with the distance

$$w_k = \exp\left(-\frac{|\boldsymbol{r}_k^{\mathrm{e}} - \boldsymbol{r}_j^{\mathrm{c}}|}{d_{\mathrm{M}}}\right), \tag{15}$$

where $d_{\mathrm{M}} = \max_k |\boldsymbol{r}_k^{\mathrm{e}} - \boldsymbol{r}_j^{\mathrm{c}}|$, for $k = 1, 2, \ldots, M_{\mathrm{s}}$.

We then concern about positioning the desingularized sources at corners of the body surface. A ship hull may have corners at its bow, stern, and bottom, etc. The isolated sources may be overcrowding at the corners. Fig. 4(a) illustrates the isolated sources near a corner, where C_j and S_j, $j = 1, 2, 3, \ldots$, denote the control points and corresponding sources in sequence starting from the corner. One can see that the source S_1 for the control point C_1 at the corner is very close to its neighbouring source S_2. Consequently, the coefficient matrix formed for the linear algebra equations of (11) is poorly conditioned, and the solution becomes inaccurate.

Cao et al. [14] remedied the problem by reducing the disingularized distances to $0.1d_{\mathrm{s}}$ for the control points at the sharp bow and stern, and improved their simulations significantly. Their technique is suitable for two-dimensional sharp corners. As a generalization of this technique for three-dimensional corners and/or non-sharp corners, the shift distance $\bar{d}_{\mathrm{s}}$ is chosen as follows

$$\bar{d}_{\mathrm{s}} = 0.1d_{\mathrm{s}} + 0.9d_{\mathrm{s}}\frac{\alpha}{2\pi}, \tag{16}$$

where α is the solid angle of the meshed surface at the control point considered. The solid angle α at a point on a surface represents the openness of the surface over there, and it is of 2π as the surface is smooth at the point. The crowding of the sources near corners is lessened with the revised desingularized distance (16), as shown in Fig. 4(b).

3.3. Calculation of solid angles

We further consider the calculation of solid angles needed in (16). For a closed surface, solid angles are available as a subset of the influence coefficients in the BIM modelling. The calculation of solid angles on an open surface is a tedious task. One can calculate solid angles by adding some artificial surfaces to make it a closed surface, but considerable CPU time is required. This is because the mesh of the artificial surfaces must be fine enough to provide a reasonable result. Zhang et al. [29] calculated a solid angle involving elliptic functions, and subsequently they calculated a solid angle at a grid by summing the solid angles subtended by the triangles to the grid [30]. Actually, a solid angle can be

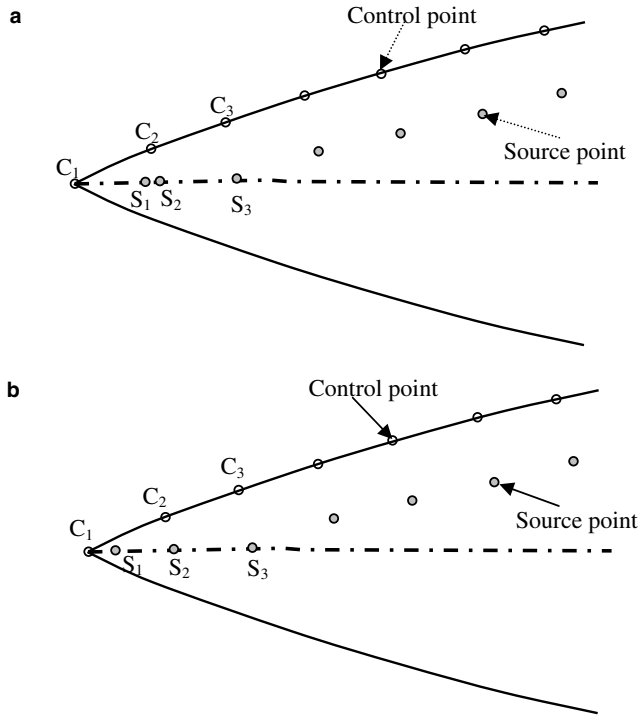

Fig. 4. Sketch of desingularized point sources near a corner when (a) the conventional shift distance is defined by (13), and (b) the revised shift distance is defined by (16).

calculated straightforwardly using the following formula, which was derived from the spherical triangle theory

$$\alpha = \sum_{k=1}^{M_s} \delta_k + (2 - N_b) \cdot \pi, \tag{17}$$

where M_s is the number of the surrounding elements to a grid considered, and δ_k is the angle between every two connected surrounding elements, as shown in Fig. 5. The angle between the positive sides of the two elements, in terms of their outer normal directions, is chosen as δ_k. By the way, Eq. (17) for the solid angle is also applicable for a structured mesh.

3.4. Lagrangian time integration

With the material velocity on the free surface calculated, the free surface and velocity potential on the surface can be updated by integrating the kinematic and dynamic boundary conditions, (3) and (4), on the free surface in time. As the remeshing is performed at every time step, the first order Euler–Cauchy method is used for the time integration.

To update the intersection between the ship hull and free surface, the hull surface is represented by a set of curves distributed longitudinally. The position and velocity potential at a grid at the intersection are first updated by integrating (3) and (4) in time. The cross points between the hull curves and updated free surface are then solved as the intersection, using the free surface interpolation to be described in Section 4. The

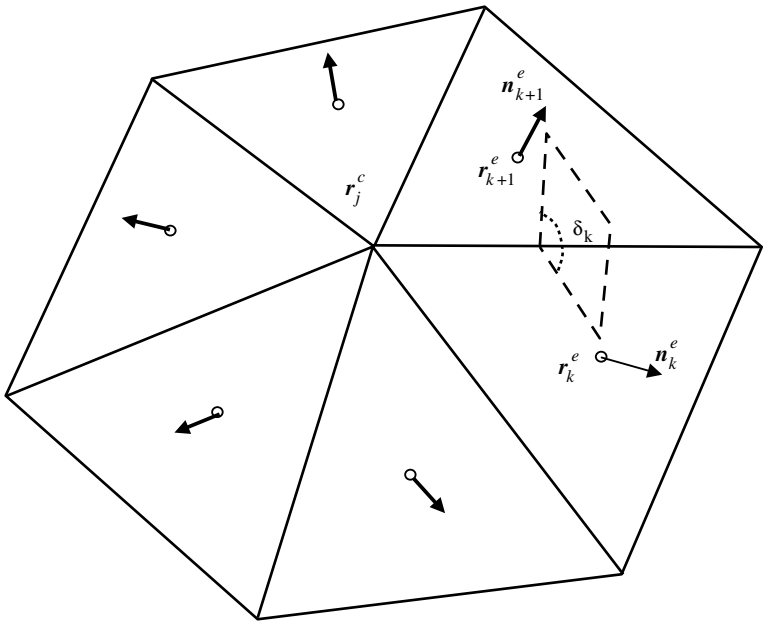

Fig. 5. Sketch of the surrounding elements to a control point r^c_j, the outward normal n^e_k and geometry centres r^e_k of the elements, and the angles δ_k between the positive sides of every two connected triangles.

velocity potentials at the intersection are obtained using the free surface interpolation too. Each hull curve is re-meshed to fit to the new intersection, and the wetted surface of the body S_B is thus updated.

4. Free surface interpolation

At every time step, the free surface needs to be interpolated before meshing. As the process is repeated hundreds of times, the interpolation must be accurate to keep the accuracy. A linear interpolation was used initially, and the free surface waves were damped out by the numerical damping involved as the waves developed. A higher order interpolation scheme is required. We implemented and compared three high order interpolation approaches, i.e., a conventional polynomial scheme, a polynomial scheme coupled with the moving least-squares method, and a second order shape function of nine grids [30]. The polynomial scheme coupled with the moving least-squares method proved to be most accurate and robust, which is described as follows.

4.1. Polynomial coupled with weighted moving least-squares method

The free surface may be steep or even parallel to the z-axis locally, which causes a polynomial interpolation to be poor or even singular. A local Cartesian co-ordinate system O-XYZ is thus introduced, with its origin O at the point considered and its Z-axis in the normal direction of the element to which the point belongs. The second order polynomial is implemented for both the free surface shape and velocity potential distribution on the surface

$$Z = F(X, Y) = a_1 + a_2X + a_3Y + a_4X^2 + a_5XY + a_6Y^2. \tag{18}$$

By the way, Zinchenko et al. [31] and Zhang et al. [30] made the Z-axis exactly at the normal direction of the surface using an iteration method. However, the Z-axis is not necessary to be at the exact normal direction for the interpolation, and neither the iteration method improves the accuracy of the interpolation.

The six coefficients a_i, $i = 1, 2, \ldots, 6$, for the polynomial of (18) are determined from the nearest neighbouring grids within $2d$ from the point considered, where d is the local mesh size. Denote the nearest neighbouring grids set as (X_k, Y_k, Z_k), $k = 1, 2, \ldots, N_b$, where N_b is the number of the grids in the set. The coefficients are determined by the weighted moving least-squares method with the error function given as

$$\sigma(a_1, a_2, a_3, a_4, a_5, a_6) = \sum_{k=1}^{N_B} W_k [F(X_k, Y_k) - Z_k]^2, \quad N \geqslant 6, \tag{19}$$

where W_k is the weight function for the neighbouring grid $\boldsymbol{r}_k$, which should decrease with the distance between $\boldsymbol{r}_k$ and the point considered $\boldsymbol{r}_0$. W_k is chosen as

$$W_k = \exp\left(-\frac{|\boldsymbol{r}_k - \boldsymbol{r}_0|}{2d}\right). \tag{20}$$

Let $\frac{\partial \sigma}{\partial a_j} = 0$ using (21), one can obtain the linear algebra equations for determining a_j

$$\sum_{j=1}^{6} A_{ij} a_j = B_i, \quad \text{for } i = 1, 2, \ldots, 6, \tag{21}$$

where A_{ij} and B_i are given in terms of W_k and (X_k, Y_k, Z_k), for $k = 1, 2, \ldots, N_b$

$$A_{ij} = \sum_{k=1}^{N_b} W_k \beta_{kj} \beta_{ki}, \quad B_i = \sum_{k=1}^{N_b} W_k Z_k \beta_{ki}, \tag{22a}$$

$$\left.\begin{array}{lll} \beta_{k1} = 1, & \beta_{k2} = X_k, & \beta_{k3} = Y_k \\ \beta_{k4} = X_k^2, & \beta_{k5} = X_k Y_k, & \beta_{k6} = Y_k^2 \end{array}\right\} \quad \text{for } k = 1, 2, \ldots, N_b. \tag{22b}$$

The above interpolation scheme holds for the velocity potential distribution on the free surface too, except that Z_k in (19) and (22a) should be replaced by the velocity potential φ_k.

4.2. Treatments of special situations

To make the interpolation robust, attentions must be paid for a few special situations, where the above polynomial interpolation scheme is inaccurate or does not work. Firstly, when a point near or at the boundaries of the truncated free surface, including the intersection between the body and free surface, its neighbouring grids are not evenly distributed. The following degenerate polynomial scheme is used instead

$$Z = F(X, Y) = a_1 + a_2 X + a_3 Y + a_4 X^2, \tag{23}$$

where the local X-axis is chosen along the tangential direction of the boundary of the truncated free surface. The polynomial is therefore the second order along the boundary, and linear along the direction perpendicular to the boundary.

Secondly, the neighbouring grids defined above are occasionally less than six. Thirdly, the matrix A_{ij} in (21) is occasionally poorly conditioned or even singular. Under the two situations, the above polynomial interpolation scheme is inaccurate or does not work; hence a linear interpolation is used instead to avoid the problems. As these two situations seldom happen, the linearization does not cause significant loss of accuracy.

Note here that the polynomial interpolation scheme described in this section for an unstructured mesh is a local interpolation. Its interpolation results are poorer for a coarse mesh, compared with a global cubic interpolation for a structured mesh.

5. Free surface triangulation

At every time step of the MEL simulation, the free surface needs to be remeshed to conform to the updated body/free surface intersection. One of the main efforts of this work is to develop an triangulation program for curved surfaces. A curved surface can be triangulated using the advancing front method [27,32,33]. To provide a high quality mesh, Anastasiou and Chan [34] triangulated a curved surface in two separated steps, i.e., planting all grids by an interrogation method and then connecting the grids by the advancing front method. In the first step, potential locations of grids are interrogated at a small interrogation step, which is one order smaller than the mesh size. As a result, this step needs the CPU time at the order of $O(10^2N)$.

5.1. Grid generation

A smoothed curved surface is considered, which may be multi-connected. A piecewise smooth surface can be handled by dividing it into several smooth patches. Following Anastasiou and Chan [34], a curved surface is triangulated here in the two separated steps, i.e., planting all grids and then connecting the grids. But we have devised a local interrogation method coupled with the advancing front method to generate grids to reduce the CPU time.

We first discuss the generation of grids on the curved surface. As in the conventional advancing front method, the initial front consists of all the grids at the boundaries of the surface, including both its external and inner boundaries. Consider a segment F_iF_{i+1} on the front. We draw two circles on the local tangential plane of the surface, with their centres at F_i and F_{i+1} and their radii at the local mesh size d, as shown in Fig. 6. Denote the cross point of the two circles on the side to be meshed as C_i. We further draw two groups of circles on the local tangential plane, with their centres at F_i and F_{i+1} too and radii from $d_{\min}$ to $d_{\max}$. Here $d_{\min}$ and $d_{\max}$ are the minimum and maximum acceptable local mesh sizes. The cross points of the two groups of the circles on the side to be meshed provide the local interrogation points for the front segment F_iF_{i+1}. $d_{\min}$ and $d_{\max}$, in this work, are chosen at $0.9d$ and $1.1d$ in the inner part of the surface, and at $0.85d$ and $1.15d$ on its boundary.

The interrogation points are then examined in the order of their distances to the point C_i, starting with the point C_i. A point among them is chosen, when no existing grid is within the sphere centred at the point with radius $d_{\min}$. This point is then projected to the curved surface to generate a new grid N_i. The segment F_iF_{i+1} is removed from the front segment set after it has been interrogated, and the segments F_iN_i and N_iF_{i+1} are added to the set, if the grid N_i is generated. This grid generation process terminates, when the front segment set becomes empty.

5.2. Surface triangulation

After all the grids have been generated, they are connected to form an unstructured mesh using the advancing front method. To save the CPU time in connecting grids, the neighbouring grid set is sorted for every grid. The initial front again consists of all the grids on the boundaries of the surface. Consider a segment F_iF_{i+1} on the front. A grid M_i, among the neighbouring grids of F_i and F_{i+1}, is chosen to form a triangle with the segment F_iF_{i+1} based on the following two criteria. Firstly, the newly formed triangle

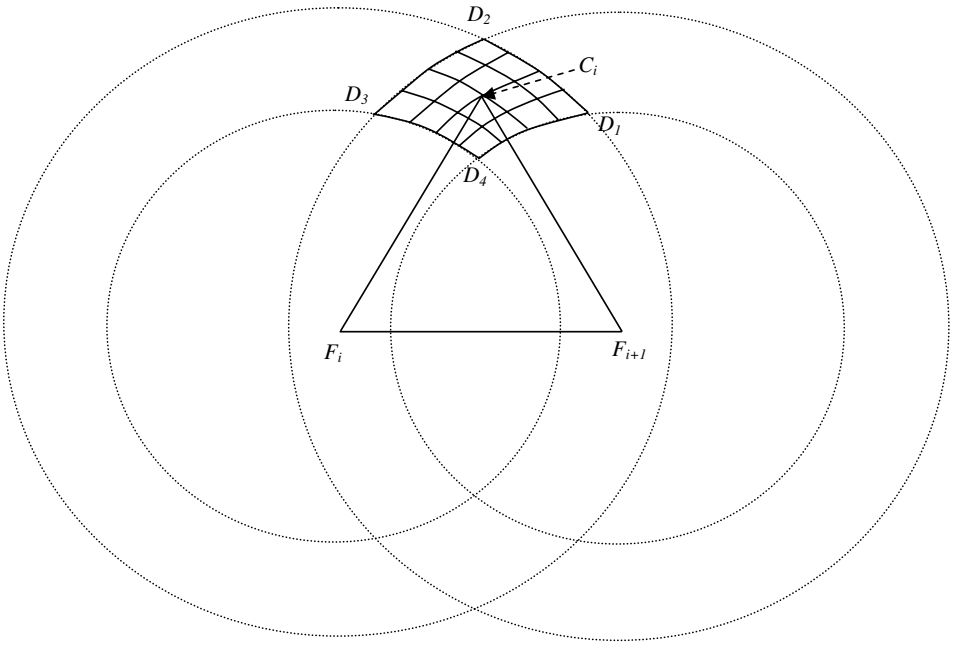

Fig. 6. The generation of the local interrogation points in the zone $D_1D_2D_3D_4$ corresponding to a segment F_iF_{i+1} on the advancing front.

does not overlap with any existing triangles. Secondly, the triangle possesses the best obtainable shape factor. The shaper factor of a triangle is defined as

$$\gamma = 2R_{\text{in}}/R_{\text{out}}, \tag{24}$$

where R_{in} and R_{out} are the radii of the incircle and circumcircle of a triangle. The shape factor of an equilateral triangle achieves the maximum value of 1.

While selecting M_i for the front segment F_iF_{i+1} to form the triangle $F_iF_{i+1}M_i$, we consider not only the shape factor of the current triangle, but also the shape factors of the two triangles to be possibly formed from F_iM_i and M_iF_{i+1}. As the result, the determined shape factor of the triangle to be connected to the front F_iF_{i+1} is defined as

$$\gamma = \gamma_0\sqrt{\gamma_1\gamma_2}, \tag{25}$$

where γ_0 is the shape factor of the triangle $F_iF_{i+1}M_i$, γ_1 and γ_2 are the shape factors of the two triangles to be possibly formed from F_iM_i and M_iF_{i+1}, respectively.

After the triangulation, the Laplace smoothing technique is used to further improve the mesh quality, whereby each of the interior grids is shifted to the centre of its surrounding polygon [27]. The iteration for smoothing is terminated when no significant improvement is achieved for the overall average shape factor measure $\prod_{i=1}^{N_e}(\gamma_i)^{1/N_e}$, where γ_i is the shape factor of triangle i, and N_e is the number of the elements generated.

Fig. 7 shows the unstructured mesh of the truncated free surface pierced by a Wigley hull. The mesh is extended 1.5 hull lengths behind the stern, a half hull lengths upstream before the bow, and 1.2 hull lengths in the transverse direction. The mesh density of the free surface is fine near the hull, and becomes coarse gradually far away. The mesh density has also been set finer near the edges of the truncated free surface, to prepare enough grids for improving the local interpolation over there.

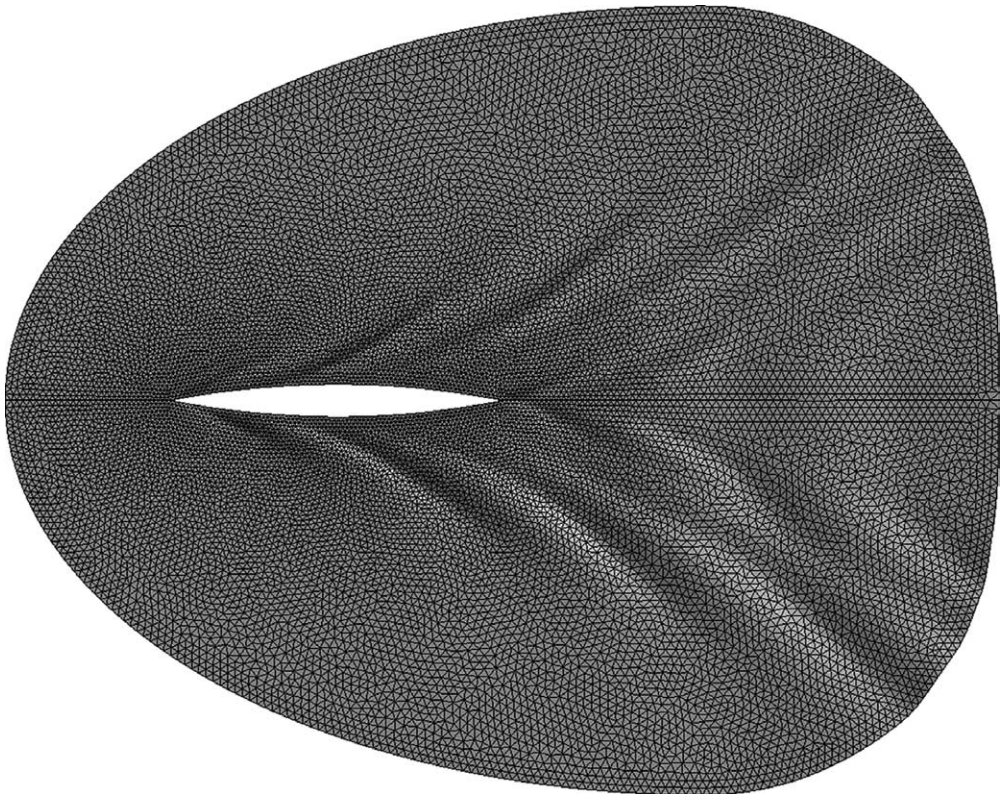

Fig. 7. The triangulation of the truncated free surface for a Wigley hull piercing on a water surface.

By the way, a half of the wetted surface of a ship hull is a rectangle type surface, hence it is first meshed into rectangles, and each of the rectangles is then divided into two triangles.

6. Results and discussions

A FORTRAN code, named UFS (unstructured free surface), was developed based on the techniques described in Sections 2–5. As an illustration, we simulate the Wigley hull, which impulsively starts to move at a constant speed on a calm water surface. The Wigley hull is a mathematical hull from, whose geometry is defined as

$$y(x,z)=\begin{cases}\frac{B}{2}\left(1-\frac{4x^2}{L^2}\right)\left(1-\frac{z^2}{T^2}\right), & \text{as } -D\leqslant z\leqslant 0,\\ \frac{B}{2}\left(1-\frac{4x^2}{L^2}\right), & \text{as } z>0,\end{cases}\qquad \text{for } -L/2\leqslant x\leqslant L/2, \tag{26}$$

where L, B and D are the length, beam and draft of the hull, respectively. To compare with the experiment of Noblesse and McCarthy [1], we choose $L/B = 10$ and $B/D = 1.6$. The hull is set to move at the Froude number $F_r = U_b/\sqrt{gL} = 0.316$.

The problem is symmetric to the centerline of the hull, so that only a half of the geometry needs to be modelled. We use about 500 grids on half of the hull and 4500 grids on half of the truncated free surface, as

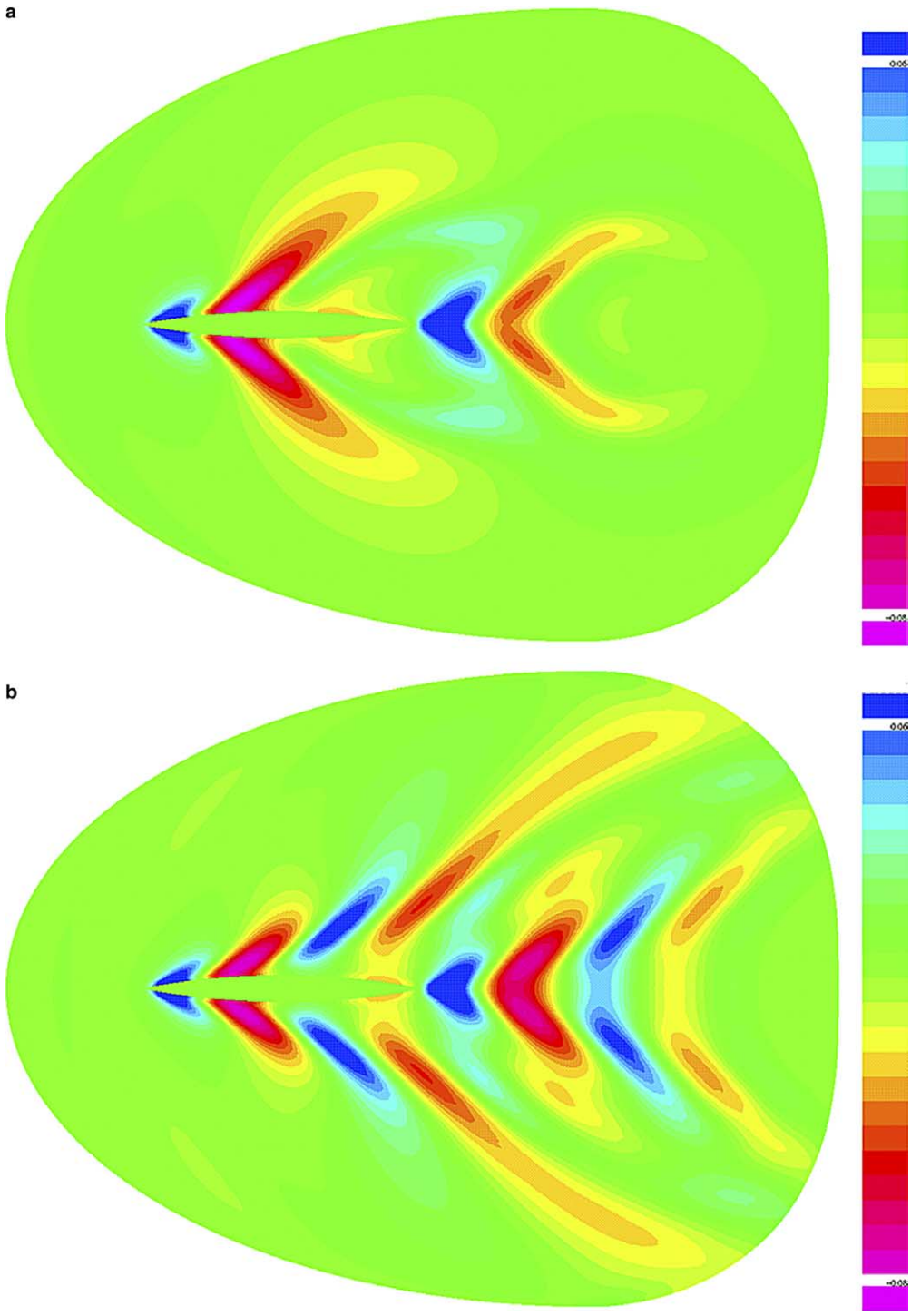

Fig. 8. The generation of the Kelvin ship-wave for a Wigley hull, impulsively started to move at $F_r = 0.316$ through a calm water, at the normalized time of $U_b\ t/L$ = (a) 1.0, (b) 2.0, and (c) 3.0, respectively.

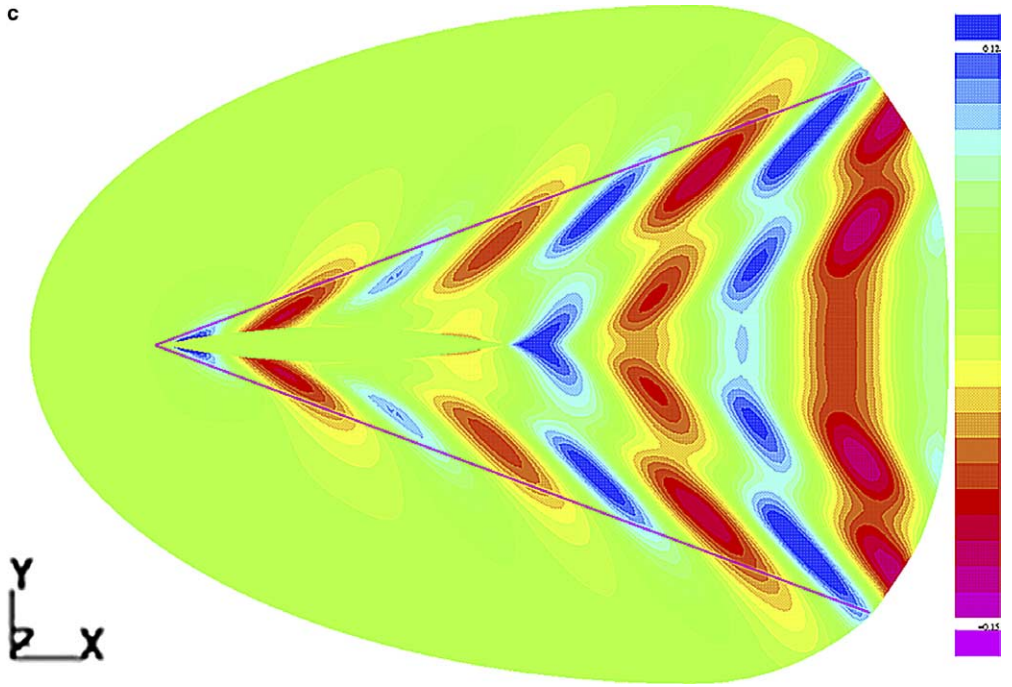

Fig. 8 (*continued*)

shown in Fig. 7. The grid number changes with time, since the mesh is performed every time step. The minimum mesh size of the free surface is chosen as $0.02L$, which is attained at the intersection between the body and free surface. The time step is set to $0.02L/U_b$, i.e., the hull moves a minimum mesh size of the free surface every time step.

We first consider the wave generation process. The free surface elevation contours simulated are shown in Figs. 8(a)–(c) at the time steps when the hull has moved for 1.0, 2.0, and 3.0 hull lengths, respectively. The free surface wave is smooth and physically reasonable at all time steps. When the hull has moved for one hull length (Fig. 8(a)), the free surface rises up near the bow and falls down before the amidships, and it first falls down and then rises up behind the stern. When the hull has moved for two and three hull lengths, the wave propagates downstream (Figs. 8(b) and (c)). The free surface wave is well developed, after the hull has moved for three hull lengths (Fig. 8(c)), and does not change significantly after that. The anticipated Kelvin ship-wave pattern is well simulated. The half angle of the Kelvin wave wedge is close to the analytical solution of 19.5° (cf. [35, Chapter 3]), denoted by in the pink lines in Fig. 8(c).

We then consider the wave elevation along the hull. Fig. 9 shows the comparison of the wave elevations along the hull simulated in the present work against the testing result of Nobelesse and McCarthy [1]. The computational results are chosen after the hull has moved for three hull lengths, when the free surface wave is stabilized. To illustrate the convergence of the algorithm, the computational results are provided with about 5000 grids (solid line) and 3500 grids (dash line). One can see that the numerical results are convergent well with the mesh size, and in good agreement with the testing result. The bow wave height computed agrees with that of the measurement too, which is typically underpredicted by the linear theories [21]. However, there are discrepancies between the computed and measured wave height in the vicinities of the bow and stern, which may be due to viscous effects neglected in the present modelling, and/or the presence of the spray in the physical experiment. It may be also due to the insufficient resolution of the free surface mesh

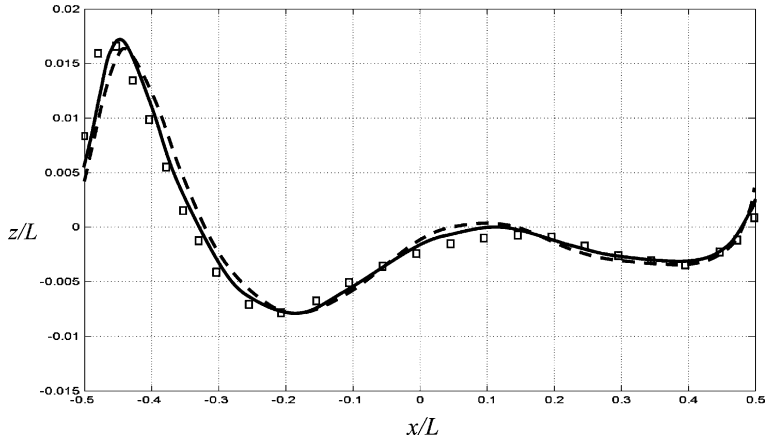

Fig. 9. The comparison of the computed and measured wave elevations along the Wigley hull at $F_r = 0.316$. The results shown are: the unstructured MEL profile with about 5000 grids (solid line) and 3500 grids (dash line), and the measurement result of Nobelesse and McCarthy [1] (□).

near the bow and stern. In principle, the physical size of the free surface wave near the bow (stern) should be the curvature radius of the bow (stern), which is very small.

The author performed the simulation using USAERO/FSP, a commercial software for ship waves based on the structured MEL method [15–17]. At the same minimum mesh size, about 8000 grids were used in USAERO/FSP. Simulations of the two algorithms, on an IBM POWER3 RS/6000 workstation with a CPU clock rate of 375 MHz, showed that about two thirds of the CPU time were saved by the unstructured approach.

By the way, the unstructured MEL model described here had been earlier on used to simulate the evolution of bubbles near a wall, and their interaction with a free surface. More validations on the convergence and accuracy of the algorithm are referred to [28,36].

7. Summary and conclusions

The simulation of ship waves is associated with *a large computational domain of the free surface*, and it is a *multi-scaled* problem. An unstructured approach appears ideal for simulating this problem. The mesh size of the free surface near a ship hull can be chosen as being small compared with the draft, where the free surface wave is steep, becoming gradually coarser away, and being small compared with the wavelength of the Kelvin ship-wave far away. This paper describes an unstructured MEL modelling for the simulation of the nonlinear wave–body interaction, which may be summarized as follows:

(i) An *unstructured* MEL model is described for the simulation of the wave–body interaction.
(ii) *Double desingularized point sources* are used for a control point at the body/free surface intersection to satisfy both the body and free surface boundary conditions over there. As a consequence, the solution of the velocity potential is smooth across the intersection.
(iii) The desingularized point sources are positioned in terms of the element sizes and solid angles of the boundary surface, which can effectively avoid possible overcrowding of the sources at the corners of the surface. Solid angles on the surface are computed analytically.

(iv) The free surface is interpolated accurately using a polynomial scheme coupled with the moving least-squares method, and is triangulated with a local interrogation method coupled with the advancing front method to provide a high quality mesh.
(v) Based on the above techniques, a Fortran code, named UFS, is developed to simulate the nonlinear unsteady ship waves.

The algorithm is used to simulate a Wigley hull piercing on a clam water surface. It is shown to be convergent with the mesh size. The wave profiles along the hull calculated agree well with the testing results of Noblesse and McCarthy [1]. The unstructured MEL model is found costing less CPU time and memory requirements compared to a structured one.

Being robust in handing complex and/or time varying geometries, the unstructured approach is versatile in practical applications. In particular, it has a potential to be developed to simulate and study the following problems, which have not been well handled by the structured one.

(i) The interaction of two ships moving in close proximity.
(ii) Ship motions in confined water, such as: a ship moving near a bank, berthing to a quay wall, or moving in shallow water or in a harbor (marine traffic simulation).
(iii) The wave run-up along multi-columns of offshore platforms.

In those problems, the geometries of the free surface and the wetted surface of the body are complex, multi-connected, and varying with time at large amplitudes. The free surface and wetted surfaces of the bodies need to be meshed at every time step, to conform to the new body/free surface intersections.

Acknowledgements

This work was partially carried out under the support of the DSTA funding, while the author was at the DSO National Laboratories. He wants to express his thanks to Dr. J.K. Tan, Dr. Y.W. Chan, Mr E.K. Png, Dr. H.M. Tsai, and Mr N.H. Teng at DSO, and Dr. B. Maskew, Dr. Y. Cao, Dr. M. Hughes, and Mr. B. Murray at Analytical Methods, Inc., Prof. S. K. Tan, Prof. C.Y. M. Lo, Dr. X.L. Wang, and Ms. S.C.C. Lee at NTU, for stimulating discussions. He thanks the two referees of this paper for their valuable comments.

References

[1] F. Noblesse, J.H. McCarthy, Ship wave-resistance computations, in: Proceedings of the 2nd DTNSRDC Workshop, Maryland, 1983, pp. 5–35.
[2] C.C. Mei, Applied Dynamics of Ocean Surface Waves, World Scientific, Singapore, 1992.
[3] W.T. Tsai, D.K.P. Yue, Computation of nonlinear free-surface flows, Annu. Rev. Fluid Mech. 28 (1996) 249–278.
[4] R.F. Beck, A.M. Reed, Modern seakeeping computations for ships, in: Proc. 23rd Symp. on Naval Hydro., 2001, pp. 1–45.
[5] H.C. Raven, A practical nonlinear method for calculating ship wavemaking and wave resistance, in: Proc. 19th Symp. on Naval Hydro., 1992, pp. 349–370.
[6] D.C. Scullen, E.O. Tuck, Nonlinear free surface flow computations for submerged cylinders, J. Ship Res. 39 (3) (1995) 185–193.
[7] M.J. Hughes, Application of CFD to the prediction of wave height and energy from high speed ferries, in: Proc. CFD'97 – Intl. CFD Conf., Ulsteinvik, Norway, 1997.
[8] D.C. Wyatt, Development and assessment of a nonlinear wave prediction methodology for surface vessels, J. Ship Res. 44 (2) (2000) 96–107.
[9] M.S. Longuet-Higgins, E.D. Cokelet, The deformation of steep surface waves on water. I. A numerical method of computation, Proc. R. Soc. Lond. A 350 (1976) 1–26.

[10] G.R. Baker, D.I. Meiron, S.A. Orszag, Generalized vortex methods for free-surface problems, J. Fluid Mech. 123 (1982) 477–501.
[11] R. Zhao, O.M. Faltinsen, Water entry of two-dimensional bodies, J. Fluid Mech. 246 (1993) 593–612.
[12] A. Clement, Coupling of two absorbing boundary conditions for 2D time-domain simulations of free surface gravity waves, J. Comput. Phys. 126 (1) (1996) 139–151.
[13] D.G. Dommermuth, D.K.P. Yue, W.M. Lin, R.J. Rapp, E.S. Chan, W.K. Melville, Deep-water plunging breakers: a comparison between potential theory and experiments, J. Fluid Mech. 189 (1988) 423–442.
[14] Y. Cao, W.W. Schultz, R.F. Beck, A three-dimensional desingularized boundary integral method for potential problems, Intl. J. Numer. Meth. Fluids 11 (1991) 785–803.
[15] B. Maskew, USAERO/FSP: a time-domain approach to compute free-surface problems, in: Proc. Symp. of High-Speed Marine Vehicles, Naples, Italy, 1991.
[16] B. Maskew, Prediction of nonlinear wave/hull interaction on complex vessels, in: Proc. 19th Symp. on Naval Hydro., 1992.
[17] B. Maskew, Prediction of non-linear hydrodynamic characteristics of complex vessels using a numerical time-domain approach, in: Proc. 6th Intl. Conf. on Numl. Ship Hydro., Iowa City, 1993.
[18] R.F. Beck, Time-domain computation for floating bodies, Appl. Ocean Res. 16 (5) (1994) 267–282.
[19] R.F. Beck, Y. Cao, S.M. Scopio, W.W. Schultz, Nonlinear ship motion computations using the desingularized method, in: Proc. 20th Symp. on Naval Hydro. Santa Barbara, CA, 1994, p. 227.
[20] R.F. Beck, A.R. Magee, Time-domain analysis for predicting ship motions, in: W.G. Price, P. Temarel, A.J. Keane (Eds.), Dynamics of Marine Vehicles and Structures in Waves, Elsevier, Amsterdam, 1994, pp. 49–64.
[21] R.F. Beck, Fully nonlinear water wave computations using a desingularized Euler–Lagrange time-domain approach, in: O. Mahrenholtz, M. Markiewicz (Eds.), Nonlinear Water Wave Interaction, Advances in Fluid Mechanics Series, WIT Press, 1999, pp. 1–58.
[22] Y. Liu, M. Xue, D.K.P. Yue, Computations of fully nonlinear three-dimensional wave–wave and wave–body interactions. Part 2. Nonlinear waves and forces on a body, J. Fluid Mech. 438 (2001) 41–66.
[23] M. Xue, H. Xü, Y. Liu, D.K.P. Yue, Computations of fully nonlinear three-dimensional wave–wave and wave–body interactions. Part 1. Dynamics of steep three-dimensional waves, J. Fluid Mech. 438 (2001) 11–39.
[24] T.Q. Li, P. Troch, J. De Rouck, Wave overtopping over a sea dike, J. Comput. Phys. 198 (2) (2004) 686–726.
[25] F.S. deSousa, N. Mangiavacchi, L.G. Nonato, A. Castelo, A front-tracking/front-capturing method for the simulation of 3D multi-fluid flows with free surfaces, J. Comput. Phys. 198 (2) (2004) 469–499.
[26] D.C. Lo, D.L. Young, Arbitrary Lagrangian–Eulerian finite element analysis of free surface flow using a velocity-vorticity formulation, J. Comput. Phys. 195 (1) (2004) 175–201.
[27] Joe F. Thompson, Bharat Soni, Nigel P. Weatherrill, Handbook of Grid Generation, CRC Press, Boca Raton, FL, 1998.
[28] Q.X. Wang, Numerical simulation of violent bubble motion, Phys. Fluids 16 (5) (2004) 1610–1619.
[29] Y.L. Zhang, K.S. Yeo, B.C. Khoo, W.K. Chong, Simulation of three-dimensional bubbles using desingularized boundary integral method, Int. J. Numer. Meth. Fluids 31 (1999) 1311–1320.
[30] Y.L. Zhang, K.S. Yeo, B.C. Khoo, C. Wang, 3D jet impact and toroidal bubbles, J. Comput. Phys. 166 (2001) 336.
[31] A.Z. Zinchenko, M.A. Rother, R.H. Davis, A novel boundary-integral algorithm for viscous interaction of deformable drops, Phys. Fluids 9 (1997) 1493.
[32] D.J. Mavriplis, An advancing front Delaunay triangulation algorithm designed for robustness, J. Comput. Phys. 117 (1) (1995) 90–101.
[33] D. Rypl, P. Krysl, Triangulation of 3D surfaces, Eng. Comput. 13 (1997) 87–98.
[34] I. Anastasiou, C.T. Chan, Automatic triangular mesh generation scheme for curved surfaces, Commun. Numer. Meth. Eng. 12 (1996) 197–208.
[35] M.J. Lighthill, Waves in Fluids, Cambridge Press, Cambridge, 1978.
[36] Q.X. Wang, The evolution of a gas bubble near an inclined wall, Theor. Comput. Fluid Dyn. 12 (1998) 29–51.