Free-Surface Viscous Flow Solution Methods for Ship Hydrodynamics

**J. Wackers · B. Koren · H.C. Raven · A. van der Ploeg ·
A.R. Starke · G.B. Deng · P. Queutey · M. Visonneau ·
T. Hino · K. Ohashi**

Abstract The simulation of viscous free-surface water flow is a subject that has reached a certain maturity and is nowadays used in industrial applications, like the simulation of the flow around ships. While almost all methods used are based on the Navier-Stokes equations, the discretisation methods for the water surface differ widely. Many of these highly different methods are being used with success.

We review three of these methods, by describing in detail their implementation in one particular code that is being used in industrial practice. The descriptions concern the principle of the method, numerical details, and the method's strengths and limitations. For each code, examples are given of its use. Finally, the methods are compared to determine the best field of application for each.

The following surface descretisation methods are reviewed. First, surface fitting/mesh deformation in PARNAS-SOS, developed by MARIN; the description focuses on the efficient steady-state solution method of this code. Then surface capturing with Volume-of-Fluid in ISIS-CFD, developed by CNRS/Ecole Centrale de Nantes; the main topic of this review are the compressive flux discretisation schemes for the volume fraction that are used in this code. And finally, the Level Set method in SURF, developed by NMRI;
this description contains a modified formulation of the Level Set method that is optimised for ship flow computation.

1 Introduction

The use of Reynolds-Averaged Navier Stokes (RANS) methods for the simulation of the water flow around ships has reached a first level of maturity. During the last fifteen years, much progress has been made in the development of robust and accurate computational strategies able to predict flows that contain both viscous and turbulent effects and a free water surface. While this development continues unabated, the application of these methods to full-complexity real-life problems is entering industrial practice. The current methods can provide a good evaluation of resistance and wave forces on ships and marine structures, accurate predictions of the flow field useful for improving the ship's hull form design, and local information on the flow enabling the analysis and improvement of appendices and propulsive systems.

The main particularity of flow solvers for marine applications is the need to consider the water surface. If the water and the air above it are considered inmiscible, the water surface appears as a clearly defined interface between them. Besides a flow discretization, a numerical method must contain a model for this surface.

It is remarkable that several different discretization methods for the free water surface, of a highly different nature, coexist. While most practically used methods are based on similar principles for the RANS flow discretization (usually the finite-volume method is used, combined with a one- or two-equation turbulence model), fundamentally different principles are used to model the water surface. Moreover,

J. Wackers (✉) · G.B. Deng · P. Queutey · M. Visonneau
CNRS/Ecole Centrale de Nantes, Nantes, France
e-mail: jeroen.wackers@ec-nantes.fr

B. Koren
CWI/Leiden University, Amsterdam/Leiden, The Netherlands

H.C. Raven · A. van der Ploeg · A.R. Starke
MARIN, Wageningen, The Netherlands

T. Hino · K. Ohashi
National Maritime Research Institute, Tokyo, Japan

these different methods all work well in practice, each having its specific advantages.

Three such approaches, the surface fitting, volume-of-fluid/monofluid, and level set technique, are reviewed in this paper. For each method, we first provide a short overview of the existing works. Then, instead of a general discussion on all these implementations, both established and experimental, we have chosen to explain in detail the particular features of a single marine hydrodynamics code that has proven its value in industrial practice. The presentation for each code is done by the codes' developers. Sections 2, 3 and 4 are devoted to each of the codes, the paper ends with a short conclusion in Sect. 5 where the suitability of each method for specific problems is discussed.

1.1 Ship Flow Problems

Before continuing the numerical description of the water surface models, we will give an overview of the physical problem that is to be solved with these models—the computation of ship flow—and of the specific numerical aspects associated with this problem. The performance of a ship in operation can be decomposed into different aspects. The most important are:

- ship resistance: the drag force on a ship moving at constant speed through still water;
- ship propulsion: the behavior of the propeller behind the ship, including cavitation; the engine power to be delivered to the propeller to produce the required thrust;
- seakeeping: the movement of a ship in incoming waves ('seaway');
- manoeuvring: the ability of the ship to keep its course and to perform turns at different speeds.

These aspects will be illustrated in some of the examples later on. Each has its own importance in the process of ship design.

A ship travelling in a seaway is subject to a hydrodynamic resistance, and thus requires engine power and consumes fuel. These quantities depend quite strongly on the ship hull form and its speed. It is, therefore, essential to be able to predict the resistance of the ship at the required speed in an early stage of the design; and to minimize it by a proper hull form design. While the seaway has an effect, the resistance and power in practice are largely determined by the resistance of the ship moving through still water. In addition, propeller performance is affected by hydrodynamic interaction of hull, propeller and rudder. The inflow to the propeller which operates in the viscous wake field of the ship is an important aspect determining the propulsive efficiency and cavitation. This inflow again depends on details of the hull form.

The hull form also plays a role in the ship's motion in response to incoming waves. Large movements must be prevented by proper hull design, to ensure ship stability, safety, and the comfort of crew and passengers. For manoeuvring performance, there is a pronounced effect of the hull form and a wish to predict the manoeuvring properties in an early stage to ensure compliance with the rules.

The main ship performance aspects mentioned are typically predicted on the basis of model tests. Large models of ships, varying between 3 and 12 m length, are tested in model basins ('towing tanks') and the flow and behavior are determined. However, due to the reduced scale there is a significant difference in Reynolds number between model and ship, and differences in all flow properties affected by viscosity. Semi-empirical corrections for these so-called 'scale effects' are being made, necessarily simplified. Instead, detailed computational methods can predict the flow at the full scale of the ship, and provide much more detail than model experiments. Therefore, for the principal aspects of ship performance there is a clear role for CFD tools in ship design, in combination with model testing.

In still water, the forces exerted by the flow on the ship's hull and appendages consist of a wave component coming from the pressure variations associated with the wave pattern that the ship generates, as well as a viscous component related with the boundary layer and wake. These two components are primarily governed by different parameters, and by different length scales in the flow. The conventional approximation has been to consider viscous flow and wave making as separate phenomena, and to disregard their interactions. This decomposition underlies the experimental prediction of ship resistance from a model test, but has also long been used in computational predictions: wave pattern computations in practical ship design are routinely made by free-surface potential flow methods, e.g. [75, 76], while the viscous flow around the hull is computed using RANS methods, either with an undisturbed, flat, water surface, or under a wave surface computed by a potential-flow code [77]. Seakeeping and manoeuvring simulations are usually performed by potential codes only.

Still, interactions exist between both physical aspects; viscous effects on the wave making and wave effects on the viscous flow do occur. Our objective to predict free-surface flow by solving the RANS equations means that the interactions of the viscous and turbulent flow with wave making can be fully taken into account.

For the solution, accuracy is of prime importance. In most ship flow cases, the viscous and wave making effects are weakly coupled, so their coupled solution is only of interest:

- If it is accurate enough to really represent the limited interaction effects, thus improving upon the separate predictions;

- If it works for ships at full scale, not just for model scale, as there is no known way to relate the interaction effects at model scale with those at full scale;
- If it is accurate enough to predict scale effects on the wave system, i.e. the small differences in wave pattern between model test and full scale.

These are significant requirements. Some of the earlier results contained numerical errors in the predicted wave pattern that exceeded the viscous effects on that pattern, or dominated the computed scale effects. Even now, several methods exist that are not able to predict these aspects accurately. In particular the demand to work well at full scale, for Reynolds numbers (based on ship length and speed) up to 5×10^9, is not easy to meet.

Also, the solution must be robust with respect to breaking waves. For ship flows in general, breaking waves frequently occur, either as plunging breakers appearing at sharp bows or as spilling breakers for stern and blunt bow flow. For most practical applications it is not necessary to model this wave breaking exactly but one would aim at predicting correctly the global effect of the breaking waves on the wave field and forces. Also, the method should not break down when breaking occurs. The importance of these aspects increases for high-speed ships for which stronger breaking occurs.

And finally, a general method needs to account for the motion of the ship. In seakeeping and manoeuvring simulations, ultimately the full unsteady motion of the ship is to be resolved together with the flow field. However, several useful models are known that can be based on simpler simulations.

Later on, when discussing applications of the numerical models discussed, we will come back to these requirements.

1.2 A Classification of Water Surface Models

References to computational methods aiming at computing interfacial flows can be found first in the early sixties. Since then, many different varieties have been proposed. Classifying these varieties is not straightforward; the literature is even ambiguous, as some category names have different meanings in different papers. A classification that is often used, is the division into two categories:

- Fitting methods, where the computational mesh is deformed to make a boundary coincide with the water surface.
- Capturing methods, where the water surface is located in the interior of the mesh.

Also, following the classification first introduced in [28] and reused in [93], methods can be classified as:

- Surface methods, for which the interface is explicitly represented and boundary conditions are applied on the surface.

- Volume methods, for which the two fluids are distinguished by particles or a specific indicator function associated with each fluid; no explicit interface model is used.

Ambiguity appears in the use of the term 'tracking methods'. Some authors (e.g. [1]) use this term to denote methods that are capturing and surface methods according to the classifications above, others have used it to denote fitting methods. Accordingly, the term 'capturing methods' has different meanings for different authors. Readers of the literature should beware of this.

We consider that the most logical classification of water surface models is to combine the two classifications given above into a division in three categories, similar to the one in [103]:

- Fitting methods, where the grid is deformed and free-surface boundary conditions are applied to a boundary of the grid. These methods are usually solved in an iterative process, where alternately the flow field is computed and the grid is deformed to match the current shape of the wave surface. An example of a fitting method is given in Sect. 2.
- Capturing methods with reconstruction. For these methods the grid is not necessarily deformed; the interface is defined as a surface that cuts through the grid. Initially, this surface was defined by convecting marker particles on the surface with the flow field [15, 65]. Later, variants of the volume-of-fluid (VoF) method used the convected value of the water volume fraction in cells to determine the surface location [66, 106]. The latest addition to these methods is the level set method, where the plane is defined by a convected continuous function [68, 84]. Section 4 is devoted to the level set technique.
- Capturing methods without reconstruction. For these methods, like the original VoF method [40], a volume fraction equation determines the amount of each fluid in the cells and local fluid properties are set as a mixture of the two pure-fluid properties according to this volume fraction. No attempt is made to reconstruct the interface, instead it appears as a numerical discontinuity in the volume fraction. Such capturing methods are detailed in Sect. 3.

This division in three categories corresponds to the three codes that will be presented in the following sections.

1.3 The Three Codes

PARNASSOS is a structured multiblock RANS solver with a surface fitting algorithm. The free-surface method has been developed at MARIN since 1998, its unique surface fitting technique was created in cooperation with CWI. The main development of the code having been completed, it is

currently the principal method of ship viscous flow calculation at MARIN and has been licenced to other institutes. The code PARNASSOS is described in Sect. 2.

ISIS-CFD is an unstructured face-based finite-volume solver that computes free-surface flow with a surface capturing approach without reconstruction, using compressive discretization schemes. Started in 1999, the code is developed at Ecole Centrale de Nantes (ECN) by researchers from ECN and CNRS. Since 2006, the code is distributed commercially by NUMECA International as a part of the FINE/Marine computing suite. ISIS-CFD is described in Sect. 3.

SURF is an unstructured finite-volume solver with a level-set discretization of the free surface. The code has been developed since 1994 at Center for CFD Research, National Maritime Research Institute, Japan. This code is commercialized by NMRI themselves and distributed to many shipyards in Japan. SURF is described in Sect. 4.

2 A Free-Surface Fitting Technique in Steady Form

2.1 Introduction

The first category of free-surface viscous flow calculation methods that we address is that of 'free-surface fitting' formulations. The example we consider is the code PARNASSOS, which is used extensively in practical ship design at the Maritime Research Institute Netherlands since many years. Its free-surface option is more recent and is being used in practical applications since a few years.

The focus of the present section is on the free-surface fitting property of the method, but also on its particular solution strategy which is enabled by the free-surface fitting formulation. On this point it differs from the two other methods described in this paper (and from almost all other free-surface viscous flow methods). This solution method makes it accurate and very efficient for steady-flow problems; but it is more specialized in its applicability than the others.

The free-surface code builds on the original PARNASSOS code for computing viscous flow around ship hulls. The foundation for this method was laid in the 1980's and it has been further developed by MARIN and IST in Lisbon, Portugal [42, 43, 98]. It has been used in practical ship design since about 1990.

In 1998 the development of a free-surface viscous flow code was started on this basis. At that time some methods to solve viscous free-surface ship flows had already been proposed by others. All of these used a time-dependent process for computing the steady state, and typically that steady state was approached quite slowly, leading to less accurate final solutions and large computation times. Because of this it was decided to aim at an alternative formulation that omits all time-dependence and solves the steady problem by iteration.

A particular form of the free-surface boundary conditions is then required for a successful algorithm. First numerical experiments with this form were made in 1998 in joint research of CWI and MARIN, by Van Brummelen. In some simple 2D test cases the method quickly fulfilled its promise by showing a very rapid convergence of the free surface shape [78, 96]. Much analysis work on the 2D method has been done by Van Brummelen [95, 97], while Lewis, in a subsequent research project, made other improvements and extensions to 3D cases [54]. The first applications to actual ship forms were done in 2002 [77]. The method has been refined and extended since, and more detailed applications and validations have been done. Today it is used in practical applications at MARIN, and already its results have contributed substantially to the insight in the physics of viscous effects on ship wave patterns [80].

The next subsection discusses free-surface fitting techniques for water wave problems more generally, indicating global advantages and limitations. Then, based on a consideration of the context and objective of the development, choices are made on the formulation, which is then described. Some of the analyses on the fundamentals of the method are briefly described in Sect. 2.3. Applications and validations are discussed in Sect. 2.4. Section 2.5 summarizes the main points.

2.2 Description of the Method

2.2.1 Free-Surface Fitting Methods

In free-surface fitting methods the free surface is considered as a boundary of the flow domain that moves under the influence of the flow. On that boundary, free-surface boundary conditions (FSBC's) are imposed. The dynamics of the air flow above the water surface is not computed and its influence on the water flow is disregarded: the pressure at the free surface is assumed to be constant (atmospheric) and no shear stress acts on the surface. In view of the limited flow speeds in the air and the large difference in density this is a justified approximation for almost all practical purposes envisaged.

The free surface shape is to be determined in the solution process. Typically a stepwise (time-dependent or iterative) procedure is used, in which alternatively the flow field and the free surface are updated. Normally, at each step the grid is adjusted such that it matches the current free-surface estimate, although there are exceptions [55]. Thus, boundary conditions are imposed on a previous estimate of the wave surface.

A very early example of a free-surface fitting method is that of Coleman and Haussling in 1981 [11, 12], for

a free-surface potential-flow problem solved with a finite-difference method. When, around 1995, the development of solution methods for free-surface viscous ship flow problems was taken up by more groups, free-surface fitting techniques were the preferred choice. Several methods were already in use for computing viscous flow around ship hulls without free surface, i.e. in which the deformation of the water surface was disregarded or computed separately using potential-flow methods. The free-surface fitting techniques added the need to impose FSBC's, and to update the free surface and the grid; but otherwise they formed a natural extension of existing tools. Examples of such methods can be found in [21, 27, 31, 69, 89, 92]. Today some of the better-known ship flow solvers are free-surface fitting methods, e.g. ICARE [2, 3], NEPTUNE [38], UNCLE [7], and the PARNASSOS method described here; but most recent new developments are for methods in the free-surface capturing category.

Advantages of surface fitting methods are that the form and location of the free surface is precisely known; there is no 'smearing' of the interface and it is a sharp and well-defined boundary. This increases the accuracy of force computations and flow details near the wave surface. With a limited amount of grid refinement, detailed phenomena such as free-surface boundary layers or surface tension effects could be modelled. Moreover, as it appears the inherent numerical damping of surface waves is less in free-surface fitting methods than in capturing methods, unless in the latter special techniques are used to minimize the effect, as e.g. described in Sect. 3. Numerically it can also be an advantage that there is no need to discretize across a density jump of a factor of 800.

However, free-surface fitting clearly has limitations as well. A rather fundamental problem in a surface fitting method is to deal with topology changes. In general free-surface flows, these can occur for so-called 'plunging breakers', i.e. overtopping waves that fall back onto the water surface and enclose an air region that breaks up in smaller bubbles; or by formation of spray, or by breaking up of cavitation bubbles. Imposing FSBC's at many rapidly changing free surfaces in such flows would be impractical or impossible. If one wants to deal with general free-surface ship flows, including ships in a seaway with breaking waves, free-surface fitting may not be a practical choice.

A more practical question is whether a good grid can be made matching the wave surface. While free-surface gravity waves away from boundaries typically have relatively mild wave slopes, more extreme wave shapes can occur directly adjacent to the hull. In certain cases locally large slopes or thin sheets of water can occur, e.g. in slamming problems, that would make gridding difficult. One should keep in mind however that surface fitting is not confined to structured grids.

In most algorithms it is also required that during the computation the grid follows the evolution of the wave surface while keeping its topology. Grid deformation then is a preferred technique, for which a variety of methods has been proposed, based on transfinite interpolation of boundary movements, or linear and torsional spring analogies in 2D or 3D, or analogies with an elastic solid. Some of these are quite powerful and applicable to large deformations but, if applied at each time step, might add substantially to computing cost.

In summary, while free-surface fitting methods have advantages for accuracy, the occurrence of topology changes and the need to generate free-surface conforming grids may be serious limitations for quickly evolving free-surface flows with large free-surface deformations.

2.2.2 Context and objective

Let us now consider to what extent these advantages and limitations apply to the class of problems we aim at with the method described in the present section. It is primarily aimed at solving the ship resistance problem; the prediction of the resistance of the ship moving at constant speed through still water. As mentioned before, the resistance and power in practice are to a large extent determined by the still-water resistance, which often forms the main hydrodynamic aspect determining the hull form design and the principal demand for CFD capabilities. The existing PARNASSOS code already was mainly dedicated to this problem.

As pointed out before, to correctly predict the weak interaction between viscous effects and wave making and to precisely account for scale effects, the accuracy of the numerical method is of the highest importance. Also from a practical point of view a very high accuracy of resistance predictions is often requested (differences as small as 1% need to be indicated). Moreover, computational results for several cases usually need to be obtained in a rather limited time (a few weeks for the entire hull form design process) to fit in a typical merchant ship design project.

On the other hand, some simplifications can be made to the problem. The cases we want to deal with are usually rather streamlined ship hulls, causing smooth flow fields. Massive flow separation is normally avoided. Flow fields are generally steady (apart from turbulent fluctuations). Steady ship wave patterns have fairly mild wave slopes away from the hull, presenting no difficulty for grid generation. Again locally at the intersection of the wave surface with the hull situations can occur that complicate gridding; but the fact that in steady flow the wave elevation is bounded by the stagnation height $\zeta_{max} = \frac{1}{2}V^2/g$ limits the occurrence of extreme wave shapes.

An often-quoted limitation of free-surface fitting methods is that they are not able to model wave breaking. As

mentioned this is actually the case for plunging breakers as these involve topology changes. In steady ship waves these usually occur only locally at sharp ship bows at higher speed; Fig. 28 shows an example. But most wave breaking in steady ship wave patterns is of the 'spilling-breaker' type, characterized as a volume of (aerated) water riding on top of the front face of a wave of the underlying flow. Such a breaker has a rather simple shape with limited wave slope and can in principle be dealt with by a fitting method. Still there are modelling and resolution issues that would need to be solved. However, a precise representation of wave breaking is in most cases not considered to be essential for ship resistance and flow predictions. We hope to accommodate a simplified modelling of wave breaking effects later.

Summarizing, the degree of generality we aim for is limited, and the accuracy and efficiency are of paramount importance. Therefore, free-surface fitting methods are a category to be considered.

As a further simplification, the free-surface shapes we need to consider are normally single-valued functions of the horizontal coordinates, or if they are not that is a very local effect that is not necessarily predicted precisely.

2.2.3 Solution Methods for Free-Surface Viscous Flow Problems

In free-surface fitting methods, at the upper boundary of the computational domain, which is (an approximation of) the wave surface, free-surface boundary conditions (FSBC's) are imposed. If we denote the velocity components (in an (x, y, z)-coordinate system fixed to the ship, with x positive aft and z upward) by u, v, w, the wave height by $\zeta(x, y)$, and non-dimensionalise all quantities using ship speed U, a reference length L (normally, ship length), and gravity acceleration g, the free-surface boundary conditions are:

- a kinematic condition that the free surface moves with the flow:

$$\zeta_t + u\zeta_x + v\zeta_y - w = 0 \quad \text{at } z = \zeta; \tag{1}$$

- a normal component of the dynamic condition, requiring that at the surface the pressure p^* is atmospheric ($p^* = 0$); in which surface tension effects have been disregarded, being insignificant for ships;
- two tangential components of the dynamic condition, requiring that no shear stress is exerted on the water surface.

As ζ is unknown, the free-surface boundary conditions are non-linear and must be imposed on a surface not known beforehand. A solution algorithm presents itself from the form of the kinematic FSBC (1), which can well be integrated in time to find a new wave elevation. A time-dependent solution procedure, for free-surface fitting methods but also for free-surface capturing methods with reconstruction, can thus be followed, in which in each time step:

1. The RANS equations are solved subject to the dynamic conditions imposed at the current wave surface;
2. The wave surface is updated by integrating in time the kinematic free-surface boundary condition; and if applicable, the grid is updated to match the wave surface.

Capturing methods without reconstruction actually follow a closely comparable evolution algorithm. Instead of step 1, the RANS equations are solved in the entire domain (water and air), for a given distribution of density and viscosity reflecting the current estimate of the wave surface location. In view of the small density of the air this corresponds closely with imposing dynamic boundary conditions at the free surface. Next, the volume fraction function, level set function or whatever indicator of the free surface location is updated by solving a convection equation, corresponding with the kinematic FSBC.

This time-dependent formulation involves the physics of transient ship waves. If the objective is to compute the steady flow and wave pattern, the transients are of no interest but can cause a substantial delay of the process:

- A variety of transient waves is generated by the initial acceleration of the ship, or whatever other start of the computation. Before a steady result can be obtained, these have to leave the domain.
- As is analyzed in [97], in an unbounded domain the asymptotic decay of the transient waves is determined by waves that have group velocity equal to the ship speed. Their effect is frequently observed as persistent, slowly decaying oscillations of the wave resistance, with nondimensional period $\Delta T.V/L = 8\pi V^2/(gL)$.
- Physically it takes substantial time before a steady wave system has established in an area around the ship, as a result of the fact that in deep water, wave energy travels with half the speed of the wave itself.

Therefore, the physical simulation time needed before a steady result is obtained, is substantial; in practical applications often around 10 to $15L/V$.

One might hope to make large time steps in the initial stages to limit the drawback. However, algorithms that essentially uncouple the dynamic conditions (imposed to the RANS solution) and the kinematic free-surface conditions (used to update the wave surface) in each time step, often impose time step restrictions.

Of course there are ways to alleviate some of the difficulties mentioned, and not all methods suffer equally. Nevertheless, tens of thousands of required time steps are frequently reported for 3D ship cases.

2.2.4 Formulation of the Steady Iterative Approach

To avoid that transient waves lead to slow convergence to steady state, it is desirable to omit all time-dependent terms

in the method and to solve a strictly steady form of the problem directly by iteration. The problem then excludes any transients at startup, only admits wave solutions that satisfy the steady dispersion relation, guarantees that waves are confined to the 'Kelvin wedge' thus reducing reflection problems at artificial boundaries, and eliminates contact line problems.

The free-surface boundary conditions to be satisfied, in steady form, are:

- a kinematic condition

$$u\zeta_x + v\zeta_y - w = 0 \quad \text{at } z = \zeta, \tag{2}$$

- a normal dynamic condition, $p = 0$ at $z = \zeta$. In terms of the non-dimensional hydrodynamic pressure

$$p = (p^* + \rho gz)/(\rho U^2),$$

this boundary condition, neglecting surface tension and viscous contributions, becomes:

$$Fn^2 p - \zeta = 0 \quad \text{at } z = \zeta, \tag{3}$$

- a tangential dynamic condition

$$\vec{t}.\tau.\vec{n} = 0. \tag{4}$$

Having dropped ζ_t from the kinematic FSBC we cannot use it any more to advance ζ in time, so a substantially different iteration process is needed.

The dynamics of ship waves are essentially governed by the normal component of the dynamic condition, (3), and the kinematic condition (2). None of these two conditions introduces any wave character by itself: wave solutions, a dispersion relation, group velocity and all other properties of free-surface waves only arise from the combination of both conditions. Therefore it is important not to impose these two conditions alternatingly in a two-step iterative process. The key to a successful iterative solution of this steady RANS/FS problem is to impose a *combined* form of the kinematic and normal dynamic FSBC. This is obtained by substituting the wave elevation from the dynamic condition into the kinematic condition, which yields:

$$Fn^2(up_x + vp_y + wp_z) - w = 0 \quad \text{at } z = \zeta. \tag{5}$$

This combined condition needs to be supplemented by the three dynamic conditions (3) and (4), to give a set of conditions that corresponds exactly with the original set.

An iterative solution procedure is then defined as, per iteration step:

- I. The RANS equations are solved subject to the combined condition (5) and the tangential dynamic conditions (4), imposed at the current wave surface;
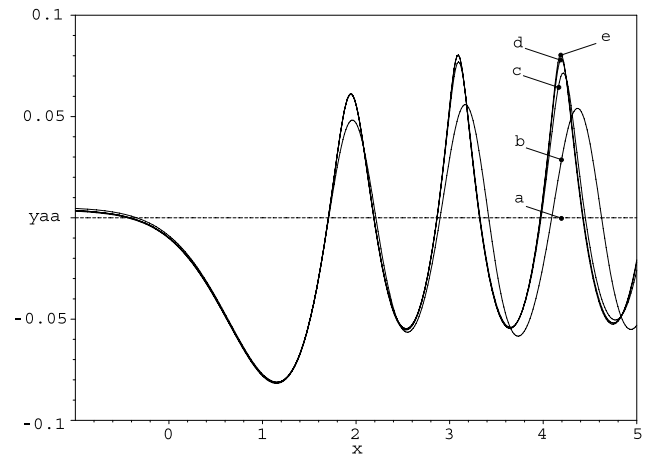
Fig. 1 Wave profile caused by a bottom bump, evolution in the course of the iteration process

- II. Next, the wave surface and grid are updated using the normal dynamic condition (3), which for given p is an explicit expression for ζ.

Upon convergence the pressure deviation, normal velocity and shear stress vanish at the actual wave surface and the solution of the steady RANS/FS problem has been obtained.

As shown in [77] the combined FSBC has a form comparable with usual FSBC's for steady free-surface potential flow methods. In itself it already embodies the proper wave-like behavior: even without any free-surface update we already find physically plausible solutions for the wave pattern in a single step. The free-surface updates are needed for some of the non-linear terms and add accuracy.

As a first illustration of how this unfamiliar formulation works, we show here some results obtained by Van Brummelen in an early stage of his research [78]. It is for 2D flow of a river over a bottom bump that disturbs the water surface and causes a trailing wave pattern. The evolution of the wave surface in the course of the iterative solution is shown in Fig. 1. The computation is started with a flat water surface (a) and a grid conforming to it, and the combined FSBC is imposed. The first RANS solution immediately produces a trailing wave system (b) with nearly the correct length and amplitude. In the next iteration, the free surface is adjusted, the grid deformed, and again the combined FSBC imposed. As the figure shows, the second estimate (c) is very close to the final result and full convergence is obtained in 3 to 9 iterations, depending on the height of the bottom bump and resulting wave steepness. This very fast convergence and favorable properties made us pursue the steady iterative approach and extend it to more general, 3D applications.

2.2.5 Implementation—The RANS Solver

The code used is PARNASSOS, which solves the RANS equations for steady, 3D incompressible flow around a ship
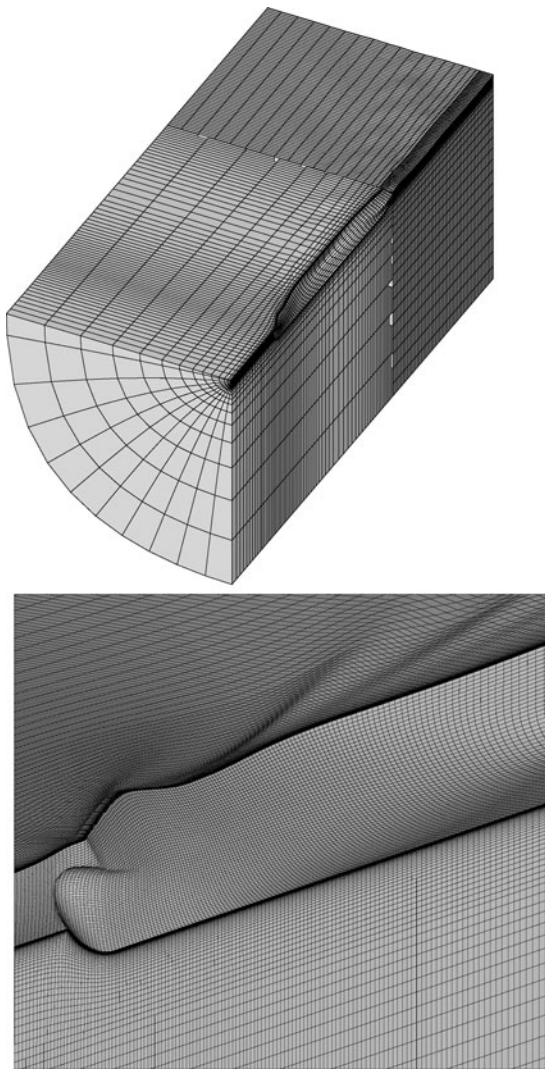
Fig. 2 Example of grid topology (only every 5th grid line shown); and close-up of grid around bow

hull. Various 1- and 2-equation turbulence models are available. Body-fitted HO-type grids are used, which are strongly contracted towards the hull; Fig. 2 shows an example of the grid layout.

No-slip conditions are imposed at solid walls, symmetry conditions at the ship's centerplane; given velocity components at the inlet. At the external boundary we impose Dirichlet conditions for the pressure and for the velocity components tangential to the boundary, derived from a free-surface potential flow calculation. At the outlet, Neumann conditions for pressure and velocities are imposed, the former again derived from the potential-flow computation.

The discretization is of finite-difference type. For the convective terms and in the continuity equation we use second or third-order upwind schemes in streamwise direction, and third-order schemes for the normal and girthwise direction. For the pressure derivatives in the momentum equations we use third-order schemes, which for stability and consistency must have a bias opposite to that of the velocity derivatives in the continuity equation; therefore a downstream-biased pressure derivative scheme is applied in mainstream direction.

A particular feature that plays a role in the free-surface approach is the treatment of the pressure. In incompressible flows, the pressure does not occur in the continuity equation, and most RANS methods therefore reformulate that equation, e.g. as a pressure-correction equation or a time-dependent artificial-compressibility formulation. Instead, in PARNASSOS the full coupling of the momentum and continuity equations is maintained, and after discretization this gives rise to a matrix equation containing 4×4 blocks. This matrix equation is solved using GMRES with ILU-preconditioning. This fully coupled solution has been found to be robust and quite insensitive to the mesh aspect ratio, and most suitable for external flows at high Rn.

The entire system of equations would still be very large. Its size is reduced by subdividing the domain into subdomains, each consisting of a group of (typically 4 to 8) transverse grid planes. In a 'global iteration' process, a block Gauss-Seidel type of solution procedure is used in which the subdomains are addressed in a downstream sequence. The preconditioned GMRES solver acts on the equations for each subdomain. This global iteration exploits the character of the problem, which has a predominant flow direction, and still takes into account all influences in upstream direction. The convergence of the global iteration is enhanced by grid-sequencing, i.e. we start the computation on a mesh coarsened in the mainstream and girthwise direction and refine repeatedly by a factor of 2. In addition, between the downstream sweeps simple algebraic upwind sweeps are performed in which only the pressure is updated.

More about the solution algorithms can be found in [98].

2.2.6 Implementation—The Free-Surface Boundary Conditions

The combined FSBC is transformed to the curvilinear system which is conforming to the boundaries of the domain. In (5) the pressure gradient ∇p and vertical velocity component w are taken implicitly in the iteration process; while the other velocities are taken from the previous step of the global iteration. Unlike in the momentum equations, in this FSBC p_x must be modelled by an upstream-biased discretization. Normally a third-order 4-point scheme is used.

Since the combined FSBC is trivial at the wall and also an update of the free surface using the dynamic FSBC can be applied arbitrarily close to the ship hull, there is no 'contact-line problem' and there should be no need for special approximations along the waterline. However, in the vicinity of the hull the grid is strongly contracted towards the

hull surface, while the vertical grid spacing is several orders of magnitude larger. In this region, transverse oscillations of the pressure occurred at the free surface, which took the form of sharply diverging waves which could propagate away from the waterline and affect the wave pattern. Such short waves should have a very quick decay in vertical direction, but this decay cannot be resolved since the vertical mesh spacing is much larger. Therefore the short waves will be severely distorted, which may explain their occurrence in the computation. This problem could be largely resolved by a local adjustment of the discretization and a vertical contraction of the grid. In practice, a local approximation of the pressure variation normal to the hull surface is used in a narrow region along the waterline, to avoid the oscillations and resulting robustness problems.

2.2.7 Free-Surface Updates

In each free-surface iteration the local wave elevations are updated from the dynamic FSBC, and the grid is updated to the new free-surface shape. This is done by deforming the grid to match the new shape of its upper boundary. Some different grid deformation tools are available, based on Trans Finite Interpolation (e.g. [24]) of the boundary movements, or on a torsional-spring analogy [5, 19, 26]; either applied in 2D form to the grids in transverse planes, or in a 3D form. Alternatively, in case of larger deformations the grid can be regenerated using the same parameter settings as for the original grid. Next, the computation goes on simply using the previous solution, without any spatial interpolation towards the new locations.

2.2.8 Solution Strategy

Summarizing, we have four iteration processes:

- The grid-sequencing. In this process the grid spacing is halved several times in longitudinal and girthwise direction;
- The free-surface iteration in which the wave elevations and grid are updated;
- The global iteration, in which subdomains are addressed in a downstream order;
- The iterative solution of the systems of linear equations for the subdomains, using preconditioned GMRES.

It is to be noted that there is no need to combine the free-surface iteration and the global iteration. It is possible to continue the global iteration until convergence within each free-surface iteration, thus solving the RANS problem completely subject to the combined FSBC. After convergence one adjusts free surface and grid, and restarts. Full convergence of the wave pattern then is usually achieved in as little as 10 free-surface iterations. However, in practice it has been found to be more robust to make more frequent free-surface updates (e.g. every 20th global iteration), with some under-relaxation.

The grid sequencing as the outer loop provides an initial estimate of the wave pattern on very coarse grids in little computation time. This is a useful option in case one starts with an undisturbed free surface as initial estimate. However, since free-surface potential-flow computations provide good results for much of the wave pattern in just minutes of CPU time, we often use this wave pattern, generate a grid under it, and start the RANS computation with that. In that case no free-surface updates are made on the coarsened grids, only once the finest grid has been reached.

2.2.9 Dynamic Trim and Sinkage

The hydrodynamic pressure distribution around the ship hull at forward speed exerts forces and moments causing a change of attitude of the ship. Typically, larger ships at normal speeds move slightly downwards ('sinkage') and may get some forward or aft angle ('trim'). These in turn affect the flow and the resistance, so it is important to take them into account in the computation. For a steady flow condition we include this in the iteration process. Simultaneous with the free-surface and grid updates we update the attitude of the ship hull. The imbalance is determined between the vertical force and pitching moment exerted by the pressure distribution over the hull, and the weight of the ship and its distribution. Trim and sinkage changes are then determined such that the resulting change of the hydrostatic force and moment cancel that imbalance. The hydrodynamic forces will then change as well; this is taken care of in the iteration process. This procedure is straightforward, converges very well and causes no delay of the computation.

2.3 Theoretical Analyses

Important theoretical analysis work on the present formulation has been done primarily by Van Brummelen [95, 97]. Among other analyses, he has considered conditions for convergence of the iterative formulation; and has derived how the number of time steps in a time-dependent method, needed to obtain a steady result, compares with the number of iterations in the steady iterative formulation; concluding that the dependence on mesh density is much smaller in the latter. We limit ourselves to giving the reference here, and just consider a study on the numerical accuracy.

Numerical Dispersion and Damping RANS/FS computations for steady ship wave patterns are not easy to make grid-independent, and frequently display a too quick decay of wave elevations with distance to the ship, as a result of numerical damping. Therefore it is important to

study the wave propagation properties of the method analytically, and such an analysis seems easier to do for surface-fitting methods. A theoretical analysis of the numerical errors was proposed by Van Brummelen [94], who essentially derived the numerical dispersion and damping of a variant of the present method. The analysis was again applied to the present method by Lewis [54]. Subsequently, in [79] the analysis has been extended, the asymptotic behavior has been studied, and the role of various difference schemes has been demonstrated analytically and numerically.

In simple terms, the analysis considers small perturbations of a uniform flow in 2D, represented by Fourier components e^{ikx+sz}. This perturbation must satisfy linearized RANS equations. A set of eigensolutions of these equations is derived. Substitution into the free-surface boundary condition provides a single wave component. For the continuous problem, this represents the steady 2D (or transverse) wave, which has a nondimensional wave number $k = k_0 = 1/Fn^2$ and $s = k_0$. For the discretized RANS equations, the difference schemes acting on the Fourier components introduce Discrete Fourier Transforms, which are different for the various schemes used; e.g. $\hat{D}_x^-$ is the DFT of the backward 3-point scheme for the convective terms, $\hat{D}_{x,fs}$ the DFT of the backward scheme for p_x in the combined FSBC, etc. Derivation of the eigenmodes and substitution in the FSBC then leads to the expression:

$$-\hat{D}_{x,fs}.\hat{D}_x^- - k_0\hat{D}_z^+ = 0,$$

in which $\hat{D}_z^+$ is a function of s which in turn depends on k by a relation dependent on mesh density.

Corresponding to the steady transverse wave of the continuous problem, this expression determines the solution of a single wave component as found from the discrete formulation. From this expression we can evaluate the ratio k/k_0 for that wave, which depends on the relative mesh densities $\varepsilon \equiv k\Delta x$ and $\beta = s\Delta z$. The real part of $k/k_0 - 1$ represents a *numerical dispersion*, i.e. an error in the dispersion relation that links wave length and wave speed. The imaginary part is a *numerical damping*, visible as a too quick decay of waves on a coarser grid. The numerical dispersion and damping thus appear to depend on all difference schemes used, in momentum and continuity equations and in the FSBC. The expression obtained has been evaluated asymptotically for vanishing mesh spacings. For a 4-point upwind-biased third-order scheme for p_x in the FSBC this yields for instance:

$$k/k_0 = 1 - \frac{1}{6}\varepsilon^2 - \frac{1}{12}\beta^3 + \frac{1}{4}i\varepsilon^3 + \mathcal{O}(\varepsilon^4).$$

With the results of this analysis it has been possible to design a 'balanced' difference scheme for the FSBC such that, in
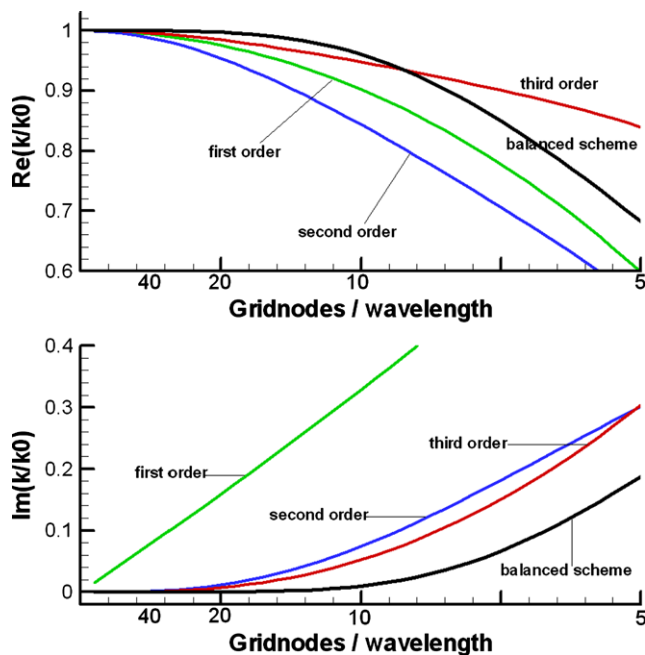
**Fig. 3** Wavenumber ratios for various difference schemes in the combined FSBC

the combination with all other discretizations used in PARNASSOS, the second-order numerical dispersion and third-order damping just cancel, yielding:

$$k/k_0 = 1 - \frac{1}{12}\beta^3 + \mathcal{O}(\varepsilon^4).$$

Dispersion is then reduced to a small third-order term in Δz, and numerical damping is reduced to fifth order.

With a simplification, the expressions for k/k_0 have also been evaluated numerically for finite mesh sizes. Figure 3 shows the result of the latter, indicating the numerical dispersion and damping for given grid densities. The different lines represent various difference schemes used in the combined FSBC, in combination with all other difference schemes used in the code. Clearly, the third-order scheme normally used already gives a good accuracy, but the balanced scheme designed with this analysis is still significantly more accurate for denser meshes.

All properties thus derived analytically have essentially been confirmed in numerical experiments. Figure 4 shows the computed wave trains generated by a travelling free-surface pressure distribution, for different grid densities. The two-point first-order scheme (top figure) is very poor, giving still a much too low wave amplitude for a grid density of 50 cells per wavelength—as a matter of fact, first-order errors in free-surface viscous flow codes often lead to unacceptable inaccuracies. The second-order three-point scheme is much better but has appreciable numerical dispersion. The third-order scheme gives an almost grid-independent result on the finest grids. However, the balanced scheme already
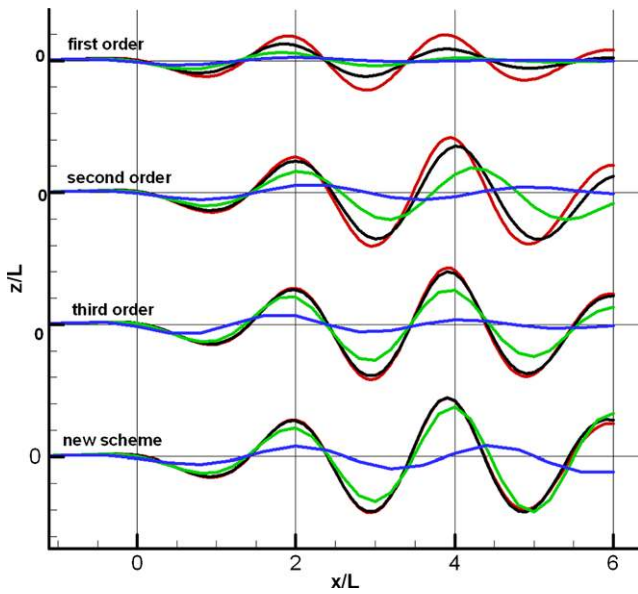
Fig. 4 Grid dependence study for various difference schemes in the combined FSBC. Pressure patch, longitudinal cut at $y/L = 1$, for grids with 6, 12, 25 and 50 cells per wavelength


Fig. 5 Convergence of the wave profile along the hull in successive free-surface updates
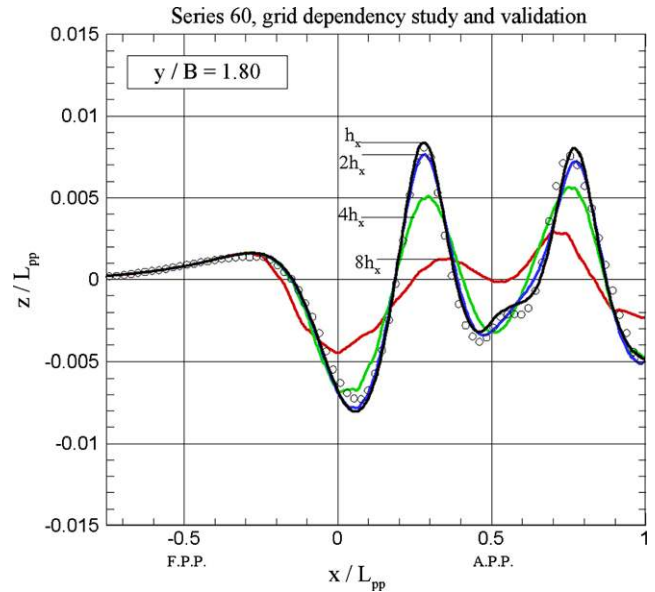
Fig. 6 Effect of longitudinal grid spacing on wave cut at $y/L = 0.2395$; and comparison with experimental data; for Series 60 model

produces a (visually) grid-independent solution with just 25 cells per wavelength.

This convincingly illustrates the importance of details of the discretization on the accuracy of computed wave patterns. It also shows the value of a theoretical accuracy analysis to improve RANS/FS methods without simply 'adding more cells'. If equal accuracy can be achieved on a coarser grid, this means a vast improvement in efficiency and applicability.

2.4 Applications

Various applications have been shown in [77, 79, 80, 88, 99]. We include some examples here to illustrate the capabilities and limitations of the method.

2.4.1 Series 60 Ship

A first ship case addressed was the so-called Series 60 $Cb = 0.6$, a standard hull form for which many validation data are available, for a model scale Reynolds number $Rn = 3.4 \times 10^6$ and Froude number $Fn = 0.316$. The computational mesh consisted of $321 \times 121 \times 45 = 1.8$ M grid nodes in the streamwise, wall-normal and girthwise direction, respectively.

Figure 5 shows the convergence of the hull wave profile, the intersection of the wave surface with the hull. In this computation the global iteration was continued until full convergence within each free-surface iteration. The computation was started with an undisturbed free surface, the first RANS solution immediately yields a most reasonable wave

profile, and after just a few free-surface iterations the wave pattern has converged.

Figure 6 compares the results for a longitudinal wave cut at $y/L_{pp} = 0.2395$, (where L_{pp} is the standard ship reference length) obtained on four meshes that were successively refined by a factor two, in streamwise direction only in this case. The number of grid nodes in streamwise direction is equal to about 13, 26, 52 and 103 per transverse wave length, respectively. On the coarsest grid the wave amplitude is strongly under-estimated, even though the wave length is
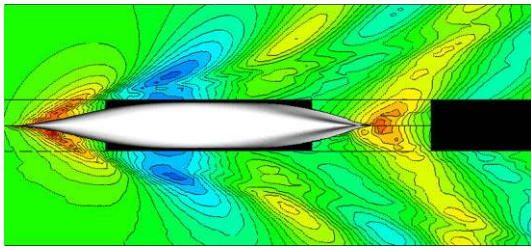
Fig. 7 Comparison of computed (*top*) and measured (*bottom*) wave pattern for Series 60 model

Fig. 8 Comparison of computed and measured longitudinal wave cut at $y/L = 0.184$, for Hamburg Test Case model

captured reasonably well. Upon grid refinement the solution rapidly improves and differences between the finest and the second finest mesh are small. The fine-grid results are in good agreement with the experimental data from [91]. Figure 7 compares the computed and measured wave pattern and shows overall good agreement, with only locally small differences in the shorter wave components.

2.4.2 Hamburg Test Case

A next case we show is the so-called Hamburg Test Case, a containership for which extensive measurement data are available. This ship was the subject of a computation workshop in the EU-funded project VIRTUE. Computations have been made for the model scale (1:24) and for the actual ship. Dynamic trim and sinkage were included in the computation. The challenge here was to have results accurate enough to distinguish the scale effects on the wave pattern and wave resistance coefficient; which, as mentioned, classically are supposed to be absent.

The Froude number was $Fn = 0.238$, the Reynolds number 11.77×10^6 for model scale, 1.2×10^9 for full scale. The grids used had 2.3 M cells for model scale ($440 \times 100 \times 52$), 3.2 M cells for full scale ($440 \times 140 \times 52$). No wall functions were used.

The workshop provided interesting comparisons between surface-fitting and surface-capturing approaches. Surface-fitting here was found to have advantages for the type of problem to be solved: a steady flow, and a high required accuracy. Free-surface capturing methods used by other workshop participants had a similar number of grid cells, but had to spend part of that for the air region, and for the dense grid needed in the free-surface region in order to limit the amount of interface smearing. Except for the resulting somewhat lower numerical accuracy the results of the method of Sect. 3 were actually very similar to those shown here [99].

The required CPU-times for model and full scale, respectively, were 48 and 72 hours on a one-processor PC for the present method. The time-dependent capturing methods applied to the same case needed up to 56 times more computation time.
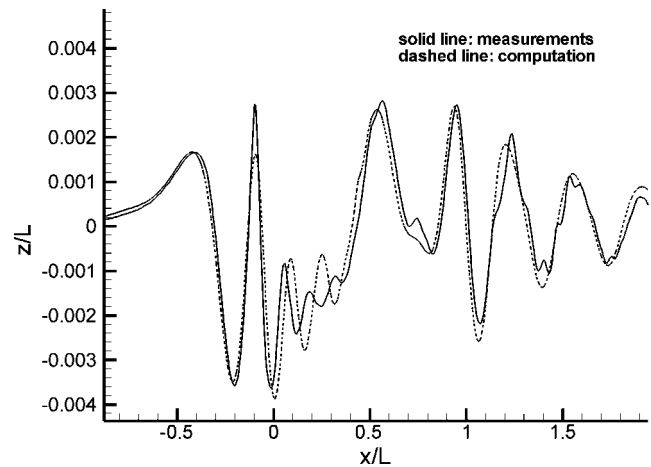
Figure 8 shows a comparison with experimental data for one wave cut, confirming that the model wave pattern computed is realistic. Figure 9 compares the wave pattern for the ship at full scale (left side of the figure) and at model scale (right side). The two are almost identical, with the exception of the stern wave system generated at the aft part of the ship, where viscous effects (e.g. boundary layer displacement) on the wave system are significant. Figure 10 shows these scale effects on the stern wave system by a longitudinal wave cut at the centerline, and compares with inviscid results. The magnitude of the viscous reduction of the stern wave is substantial locally.

Clearly, the possibility to solve free-surface viscous flow problems provides new, detailed information hard to obtain otherwise. In [80] these results have been further analyzed, and practical implications shown. Specifically, in the usual prediction of the resistance and engine power of a ship based on the measured resistance at model scale, the effect of the wave-viscous interaction on the resistance is disregarded, or rather, implicitly compensated by empirical corrections. However, from computations this interaction effect appears to differ significantly between ships, and cannot be well corrected empirically. Thus it should be possible to improve the accuracy of the prediction of ship resistance by a sensible use of CFD results in the process.

2.4.3 2D Transom Stern Flows

A detailed study with the present method has been done on transom stern flows [88]. A transom stern is the cut-off aft end of a ship, as is common for most large ships today. At the lower edge of the transom the free surface often detaches cleanly from the ship hull surface; this is the so-called dry-transom regime. However, if that lower edge is at a too low position, a flow separation with a recirculation area aft of
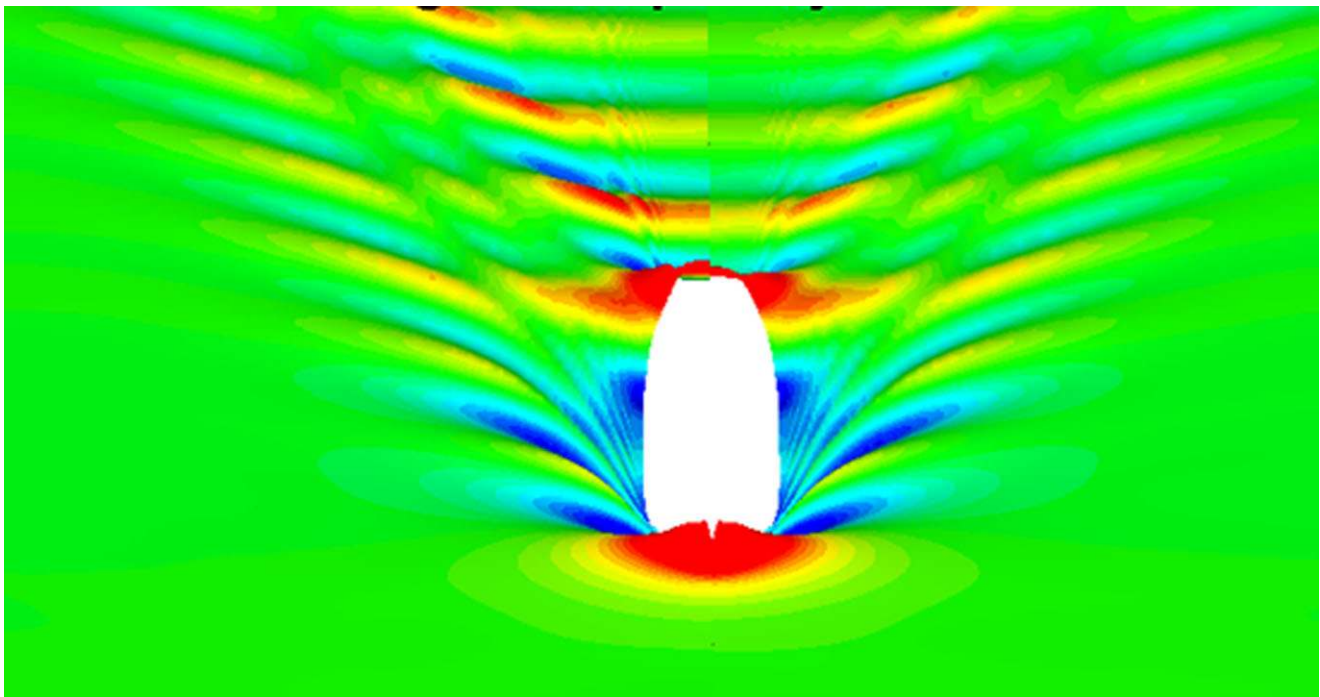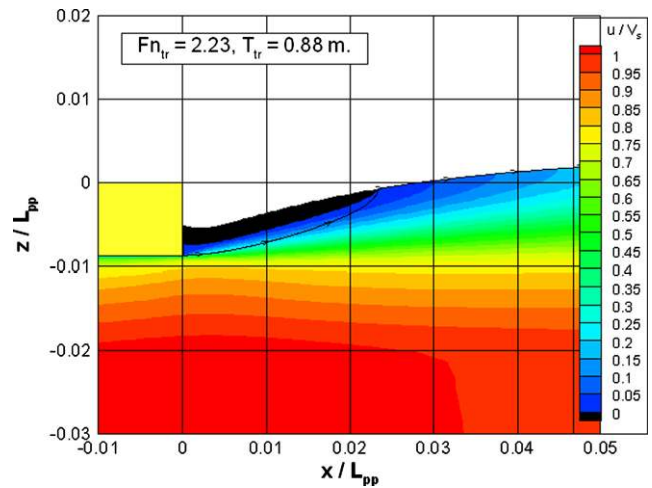
Fig. 9 Computed wave pattern for Hamburg Test Case at full scale (*left*) and model scale (*right*). Wave heights 5 times magnified
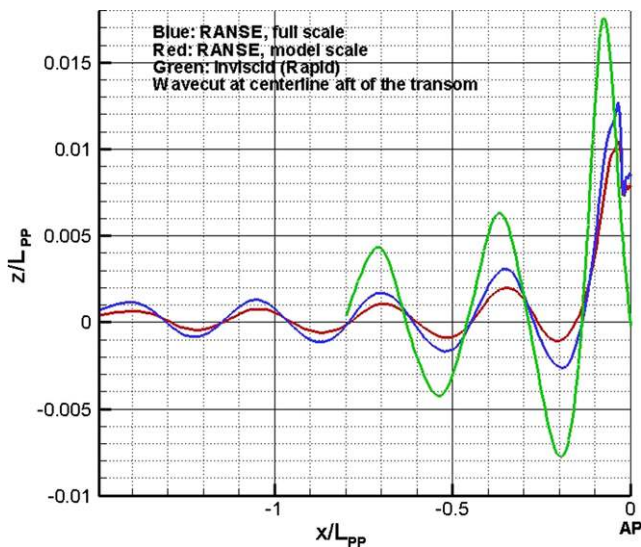
Fig. 10 Hamburg Test Case. Wave cut at centerline aft of the stern. Full scale, model scale and inviscid

Fig. 11 2D wetted-transom stern flow. Contour plot of longitudinal velocity component, showing recirculating flow aft of transom

the transom may occur; a 'wetted-transom flow'. In an intermediate regime, the free surface detaches cleanly from the transom edge, but aft of it a forward-directed spilling breaker occurs.

The transom flow regime is relevant for the resistance; but it is not easy to foresee which regime will be realized, as it depends strongly on viscous effects and the history of the boundary layer: low momentum in a thick boundary layer favors transition to a wetted transom flow. Thus the flow

regime depends on the interplay of viscous and wave effects and can only be predicted using free-surface viscous flow computations. However, it is not evident that the present method will be adequate in all cases: A single-valued description of the wave surface is used; the combined free-surface condition is rather weak in a recirculation area; and the method is in completely steady form and disregards the unsteadiness which is pronounced in the transition regimes.

In our 2D study we consider the aft end of an infinitely wide, flat-bottomed ship with a straight transom. From the inflow boundary the flow passes under the bottom of the
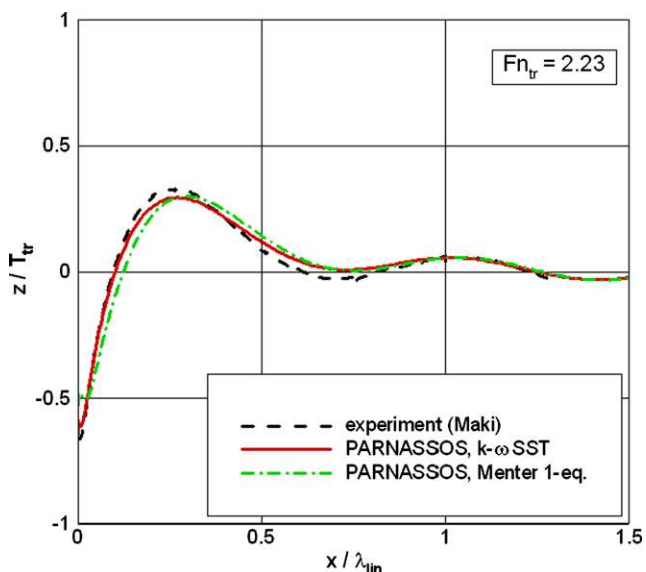
Fig. 12 2D wetted-transom stern flow. Comparison of computed and measured wave profile aft of the transom. Experimental data from Maki [59]

Fig. 13 2D dry-transom stern flow with spilling breaker on stern wave. Contour plot of longitudinal velocity component and streamlines, showing flow reversal in breaking area

Fig. 14 Containership. Wave pattern aft of transom stern, and illustration of the block structure used. (Hull surface only shown as surface grid on its wetted part)

ship, then detaches from the transom edge, either with a recirculation or as a dry-transom flow. For a given speed, the flow regime obtained depends on the depth of the ship.

For the dry-transom flow regime, accurate solutions were obtained for full scale, for model scale and for inviscid flow [88]. By systematically increasing the depth of the transom for a given speed it proved possible to determine the transom depth at which inception of wave breaking would take place, as indicated by the vanishing of the longitudinal velocity at the crest of the first wave aft of the transom. Scale effects on this breaking inception, which initiates the intermediate regime, could thus be established.

Figures 11 and 12 shows one of the results for a case with wetted transom, with a clear recirculation region aft of the transom face. The wave pattern is in reasonable agreement with the experimental data.
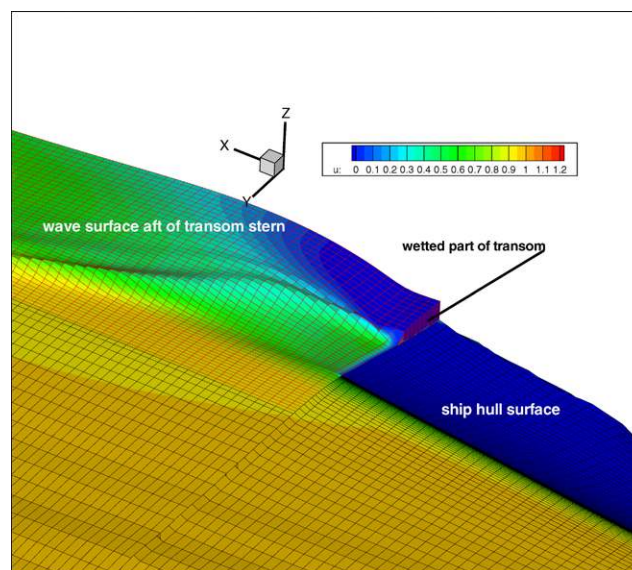
In the intermediate regime the free surface detaches from the transom edge but a spilling breaker occurs some distance aft of it. Figure 13 illustrates the computed flow for one case. While the transom is dry, just aft there is a sort of jump in the wave elevation, followed by an area with recirculation and a relatively flat wave surface aft of it. This is in good qualitative agreement with reality for a spilling breaker. Even so, resolution in these computations was still insufficient; but the results suggest that the occurrence of wave breaking of this type in itself need not be a fundamental problem for surface-fitting methods.

2.4.4 Container Ship

The final application we show is a container ship model with a transom stern. The transom flow regime being unknown,

for this computation we use a multiblock grid in which there is a transverse block junction at the transom. Aft of it is a block that covers the flow aft of the transom, with or without recirculation. The top boundary is along the wave surface, and dependent on the transom flow regime that boundary starts at the transom edge or the transom face. Thereby the transom flow regime is automatically found in the computations. In the course of the free-surface iteration process a flow is found with a narrow wetted transom part near the centerline, and a smooth dry-transom flow off the transom further aside; separated by a rather steep wave slope as we also see in experiments. Figure 14 illustrates the grid after the automatic deformation in the course of the iterative so-

lution, and the computed stern wave pattern. Clearly we find the main transom flow regimes on the same ship next to each other. A comparison with measured wave cuts aft of the transom confirmed the accuracy of the solution for this case.

2.5 Summary and Perspectives

The present section has discussed a free-surface viscous flow computation method based on a free-surface fitting technique. The method is dedicated to steady flow around ship hulls, and steady wave patterns. It uses multiblock structured grids that are repeatedly deformed in the course of the computation to match the free surface. Various grid deformation tools are available, based on Transfinite Interpolation of free-surface changes, or on torsional-spring analogies in 2D or 3D.

In the method discussed in this section, a particular iterative solution method for the steady problem is used, which distinguishes it from virtually all other methods. Owing to the special formulation of the boundary conditions a very fast convergence of the free-surface shape is obtained. In addition, the steady formulation eliminates various problems related to reflection of transient waves, persistent unsteadiness and contact line problems.

The present method is being applied in practical ship design projects and has already proven its value. Some applications have been discussed, indicating the possibilities, the numerical properties, the accuracy achieved, and the use of the results. The availability of highly-accurate viscous free-surface flow solutions for model and full scale has appeared to provide new insights that may have an impact on ship power prediction and hull form design in a near future.

3 Surface Capturing Based on Compressive Discretizations

3.1 Introduction

For volume methods (surface capturing), the fluids on either side of the interface are marked by particles or a specific indicator function; this means that the interface between fluids is never explicitly known throughout the computations. Therefore, special reconstruction techniques have to be developed to capture and locate accurately the interface, which is the main deficiency of this technique. In 1965, the Marker and Cell (MAC) method was introduced by Harlow and Welch [30]. In this first implementation, a cell without marker was considered as empty and the interface was defined as the set of cells containing a marker adjacent to empty cells. During each time step, the marker particles are moved according to the neighboring fluid velocities. Daly [14] first and then Hirt and Shannon [41], refined the

MAC method to take into account two-fluid computations by defining different markers for different fluids. In that formulation, a cell containing both marker particles was considered to be located across the interface. All these methods belong to the so-called particle-in-fluids category. A second category, called volume fraction methods, emerged naturally in the 80s. Instead of using markers, a scalar indicator function taking values between zero and one (the volume fraction) is used to distinguish between two different fluids, a value of zero indicating the first fluid and a value of one the occurrence of the second one, respectively. A volume fraction between these two limits points out the presence of the interface with a direct indication of the relative proportion of fluid occupying the computational cell. The great advantage brought by this formulation is that only one transport equation has to be solved to determine the proportion of fluid in each cell. However, the accuracy of this approach will depend on discretization schemes which should not be too diffusive in the vicinity of the interface. Therefore, it is clear that volume methods are more robust on one hand but there is still a need for specific techniques to improve the determination of the location of the interface. Noh and Woodward [66] proposed in 1976 the SLIC algorithm (Simple Line Interface Calculation) in which the interface in each computational cell was approximated as a line parallel to the Cartesian axis according to any particular kind of fluid distribution in the cell. This idea was generalized by Youngs [106] with PLIC (Piecewise Linear Interface Construction) and Lötstedt [58] who proposed to replace parallel lines by oblique segments. The SLIC method was furthermore improved by Ashgriz and Poo [4] who introduced in 1991 the Flux Line-segment model of Advection and Interface Reconstruction (FLAIR). More recently, Pillod and Tuckett [71] introduced a least-squares procedure while Lopez et al. [57] proposed reconstructions based on splines. The main drawback of all these methods is that they are clearly limited to structured grids composed of rectangular or parallelepipedic control volumes. Consequently, the generalization to control volumes bounded by an arbitrary number of faces appears problematic.

The donor-acceptor formulation was first introduced by Ramshaw and Trapp in 1976 [74]. In this formulation, the volume fraction value of the downwind cell (acceptor cell) of a cell face is used to evaluate the level of volume fraction value transported through it during a time step. Using downwind information to build a numerical scheme leads to instability problems associated in that case to unboundedness which occurs when the volume fraction values become larger than unity or lower than zero. Such situations may occur when a volume fraction of one fluid requested by the acceptor cell is larger than the one available in upwind (donor) cell. It is therefore mandatory to ensure boundedness. which is achieved by taking into account the availability of fluid in

the donor cell to modify the value predicted by the acceptor cell. This first clever formulation was later improved by Hirt and Nichols [40] with their famous 1981 Volume-of-Fluid (VOF) formulation by including a dependency with respect to the slope of the interface. Later, Lafaurie et al. [50] built the so-called SURFER scheme which was further refined to simulate merging and fragmenting of interfaces. The use of VOF for multidimensional problems was done with the operator splitting strategy, which intrinsically limited it to structured grid discretizations. Moreover, Ashgriz and Poo and Lafaurie et al. established that the shape of an interface was not kept unchanged when convected in a square domain with an oblique velocity field. There was therefore a need for higher-order discretization schemes which could be used on fully-unstructured grids.

A more convincing approach was initialized by Davis [18] in the early 90s. Instead of trying to represent geometrically the interface between fluids, this strategy aimed at discretizing more accurately a contact discontinuity convected by a transport equation with the help of higher order or blended discretization schemes. The difference between a less diffusive scheme (reducing the smearing of a profile) and a compressive scheme (removing any diffusion in the vicinity of the interface) was stated for the first time and opened the way to a variety of new compressive discretization schemes. Most of these schemes are based on the Normalized Variable Diagram (NVD) introduced by Leonard [52] in 1991. Ubbink [93] in 1997 proposed the CICSAM scheme (Compressive Interface Capturing Scheme on Arbitrary Meshes), a blending of Hyper-C and ULTIMATE-QUICKEST determined by the orientation of the interface and flow directions. Another interesting scheme was proposed by Jasak et al. [47] with IGDS (Interface Gamma Discretization Scheme), a compressive version of their GDS scheme [45]. Muzaferija and Peric [64] proposed another variation called HRIC (High Resolution Interface Capturing). More recently, Queutey and Visonneau [73] built a new blended scheme called BICS for Blended Interface Capturing Scheme, which is further refined in this section under the acronym BRICS which stands for Blended Reconstructed Interface Capturing Scheme. All these schemes try (i) to keep constant the width of the interface on the smallest number of control volumes by reducing the numerical diffusion and dispersion, (ii) to ensure a monotonic change of the volume fraction (boundedness criterion). Moreover, they never introduce a geometrical representation of the interface but try to satisfy the aforementioned criteria with the help of properly designed discretization schemes. All these recent discretization schemes share the following properties and differ in the details of their mechanisms:

- they are based on the NVD diagram and use blending of pure downwind and upwind second-order discretization,

- the blending is controlled by a local Courant number, which means that the reduction of smearing is paid through an increase of the computational effort, especially for steady free-surface flows,

- to be used in the fully unstructured framework, these schemes provide only a flux reconstruction based on three points, the two natural neighbors of the face and an additional one located in the upwind direction,

- the slope of the interface is taken into account to avoid spurious oscillations which appear when it is aligned with the flow direction.

This section describes the surface capturing technique, the way it has been implemented in the solver ISIS-CFD, developed at Ecole Centrale de Nantes (ECN) by researchers from ECN and CNRS. Sections 3.2 and 3.3 introduce the governing flow equations and the way they are discretized in this solver. Section 3.4 describes the cell face reconstruction of the pressure near the water surface. The key Sect. 3.5 introduces the numerical implementation of compressive interface capturing schemes and the details of the BRICS scheme used in ISIS-CFD. Section 3.6 shows, how the surface capturing approach fits in the overall algorithm of the flow solver. Finally, Sect. 3.7 gives results for test cases that cover a wide range of ship hydrodynamic flow applications, illustrating the generality and robustness that can be obtained with the surface capturing approach.

3.2 Governing Equations

3.2.1 Flow Equations

The ISIS-CFD flow solver resolves the incompressible Unsteady Reynolds-Averaged Navier Stokes equations. In the case of turbulent flows, additional transport equations for modelled variables are solved in a form similar to the momentum equations and they can be discretized and solved using the same principles.

In the multi-phase continuum, for an incompressible flow of viscous fluid under isothermal conditions, mass, momentum and volume fraction conservation equations can be written in Cartesian coordinates, on a possibly moving domain, as (using the generalized form of Gauss' theorem):

$$\frac{\partial}{\partial t} \int_V \rho \, dV + \int_S \rho (\overrightarrow{U} - \overrightarrow{U}_d) \cdot \overrightarrow{n} \, dS = 0, \tag{6}$$

$$\frac{\partial}{\partial t} \int_V \rho U_i \, dV + \int_S \rho U_i (\overrightarrow{U} - \overrightarrow{U}_d) \cdot \overrightarrow{n} \, dS$$
$$= \int_S (\tau_{ij} I_j - p I_i) \cdot \overrightarrow{n} \, dS + \int_V \rho g_i \, dV, \tag{7}$$

$$\frac{\partial}{\partial t} \int_V c_i \, dV + \int_S c_i (\overrightarrow{U} - \overrightarrow{U}_d) \cdot \overrightarrow{n} \, dS = 0, \tag{8}$$

where V is the domain of interest, or control volume, bounded by the closed surface S moving at the velocity $\overrightarrow{U}_d$ with a unit normal vector $\overrightarrow{n}$ directed outward. $\overrightarrow{U}$ and p represent, respectively, the velocity and pressure fields. τ_{ij} and g_i are the components of the viscous stress tensor and the gravity vector, whereas I_j is a vector whose components vanish, except for the component j which is equal to unity. c_i is the ith volume fraction for fluid i and is used to distinguish the presence ($c_i = 1$) or the absence ($c_i = 0$) of fluid i. Since volume fractions between 0 and 1 indicate the presence of a mixture, the value of $1/2$ is selected as a definition of the interface.[1]

The effective flow physical properties (viscosity μ and density ρ) are obtained from the physical properties of the constituent fluids (μ_i and ρ_i) with the following constitutive relations:

$$\rho = \sum_i c_i \rho_i, \qquad \mu = \sum_i c_i \mu_i, \qquad 1 = \sum_i c_i. \qquad (9)$$

When the grid is moving, the so-called *space conservation law* must also be satisfied:

$$\frac{\partial}{\partial t} \int_V dV - \int_S \overrightarrow{U}_d \cdot \overrightarrow{n} \, dS = 0. \qquad (10)$$

Then, a simplified form, more practical in the context of incompressible flows, of the general mass conservation equation (6) can be obtained when considering incompressible phases with constant densities ρ_i. With a few manipulations the mass conservation simplifies to

$$\int_S \overrightarrow{U} \cdot \overrightarrow{n} \, dS = 0, \qquad (11)$$

or, in a non-integral form using the divergence operator $\nabla \cdot$,

$$\nabla(\overrightarrow{U}) = 0. \qquad (12)$$

3.2.2 Turbulence Modelling

Several turbulence models ranging from one-equation models to Reynolds stress transport models are implemented in ISIS-CFD. Most of the classical linear eddy-viscosity based closures like the Spalart-Allmaras one-equation model [86] and the two-equation $k - \omega$ SST model by Menter [61], for instance are implemented. Two more sophisticated turbulence closures are also implemented in the ISIS-CFD solver, an explicit algebraic stress model (EASM) [20] and a Reynolds stress transport model [25].

3.3 Finite-Volume Discretization

The software is based on the finite volume method to build the spatial discretization of the transport equations. The discretization is face-based, which means that the fluxes are constructed face by face; therefore, cells with an arbitrary number of arbitrarily-shaped faces are accepted by the code. The discretization and the segregated solution method are based on the cell centered co-located Rhie and Chow SIMPLE [82] algorithm (i.e. Strongly Implicit Method for Pressure-Linked Equation): the velocity field is obtained from the momentum conservation equations (7) and the pressure field is extracted from (6), the mass conservation constraint or continuity equation, transformed into a pressure-equation.

Key features of the ISIS-CFD discretization, as compared to standard SIMPLE algorithms, are the reconstruction of the states on the cell faces in the neighborhood of the water surface, notably the computation of the pressure and pressure gradient, and the compressive discretization for the volume fraction. These procedures are detailed in the following two subsections. All numerical details of gradient approximation and reconstruction on faces can be found in [73].

3.4 Pressure Reconstruction

The major difficulty when solving both air and water in the same continuum is to obtain a perfect equilibrium between the pressure gradient and the gravity term to prevent the growth of parasitic currents due to gravity, even when neglecting surface tension and viscosity effects in jump conditions. This subsection explains the reconstruction of pressure and pressure gradients at the cell faces, in order to deal correctly with gravity and large density variations.

3.4.1 Pressure Equation

In the spirit of the Rhie and Chow method, the velocity fluxes $\mathscr{F}(\overrightarrow{U})$ on the faces, in the discretization of (6), are taken from the partial discretization of the momentum equations (7) in the face centers and can be symbolically written as:

$$\mathscr{F}(\overrightarrow{U}) = \mathscr{F}(\overrightarrow{\tilde{U}}) - C_p\left(\mathscr{F}\left(\frac{\overrightarrow{\nabla} p}{\rho}\right) - \mathscr{F}(\overrightarrow{g})\right), \qquad (13)$$

having defined the operator $\mathscr{F}(\cdot)$ that provides the flux of a vector through the face by

$$\mathscr{F}(\overrightarrow{Q}) = \overrightarrow{Q_f} . \overrightarrow{S_f} = S_f \overrightarrow{Q_f} . \overrightarrow{n_f}, \qquad (14)$$

where $\overrightarrow{S_f}$ is the oriented surface vector $\overrightarrow{S_f} = S_f \overrightarrow{n_f}$. The vector $\overrightarrow{\tilde{U}}$ and the coefficient C_p contain all contributions to
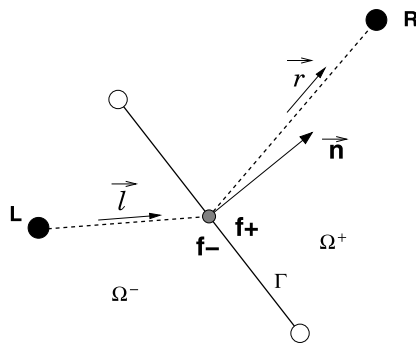
Fig. 15 Cell face notations

the momentum equations except the pressure gradient and the gravity contribution. All terms in $\vec{U}$ and C_p are rebuilt from their cell values. The pressure itself is then solved (see Sect. 3.6) from the matrix assembled from all control volumes using the specific discretization of the normalized pressure gradient through the face, as explained below. The gravity term must be kept along with the pressure gradient so that a pure hydrostatic equilibrium will be exactly satisfied if the continuous hydrostatic equilibrium $\vec{\nabla} p/\rho = \vec{g}$ is satisfied.

3.4.2 Basis of Pressure Reconstruction

In continuous flow, the water surface appears as a density discontinuity. Over such a discontinuity, the following jump conditions hold:

$$[p] = 0, \tag{15a}$$

$$\left[\frac{\vec{\nabla} p}{\rho}\right] = \vec{0}. \tag{15b}$$

$\vec{\nabla} p$ itself, however, is discontinuous. In the discretized situation, with large density jumps appearing across the cell faces near the water surface, it is therefore logical to base face reconstructions on the quantities in (15). Moreover, reconstructing $\frac{\vec{\nabla} p}{\rho}$, instead of $\vec{\nabla} p$, is the only sure way to exactly satisfy hydrostatic equilibrium, and thus to avoid the generation of parasitic currents in undisturbed flow.

3.4.3 Face Reconstruction of p

From Taylor series expansion on both sides of the face, combined with the jump conditions (15), a reconstruction of the pressure on the face can be established [73] in the following compact form involving left and right side cell-centered data only (see Fig. 15 for notations). This pressure reconstruction will be used in the momentum equations.

$$p_f = \frac{h^+\rho^+ p_L + h^-\rho^- p_R}{h^+\rho^+ + h^-\rho^-}$$

$$+ \boxed{\frac{\rho^+\rho^-}{\hat{\rho}}\left(\frac{h^-\vec{E}^+ - h^+\vec{E}^-}{h}\right)}$$

$$\times \boxed{\left\{\frac{h^+}{h}\left(\frac{\vec{\nabla} p}{\rho}\right)_L + \frac{h^-}{h}\left(\frac{\vec{\nabla} p}{\rho}\right)_R\right\}}. \tag{16a}$$

The framed term is kept explicit in each solution step, while the non-framed term is implicited in the solver. Geometrical vectors $\vec{E}^{\pm}$ are introduced so that the framed term contribution goes to zero when the grid becomes orthogonal ($\overrightarrow{Lf}.\vec{n} = \overrightarrow{fR}.\vec{n} = 0$):

$$\vec{E}^- \triangleq (\overrightarrow{Lf}.\vec{n})\vec{n} - \overrightarrow{Lf},$$
$$\vec{E}^+ \triangleq (\overrightarrow{fR}.\vec{n})\vec{n} - \overrightarrow{fR}. \tag{16b}$$

Distances used are the projected distances to the face $h^{\pm}$ and the projected distance h between the L and R cell centers:

$$h^- = \overrightarrow{Lf}.\vec{n}, \qquad h^+ = \overrightarrow{fR}.\vec{n},$$
$$h = h^- + h^+ = \overrightarrow{LR}.\vec{n}. \tag{16c}$$

The quantity $\hat{\rho}$ homogeneous with ρ is defined by:

$$\hat{\rho} = \frac{h^-\rho^- + h^+\rho^-}{h}. \tag{16d}$$

3.4.4 Normal Gradient to the Face

The discretization of the pressure gradient in (13) is obtained with a reconstruction following the same rules as for the quantity on the face. The continuous term through the face is the gradient normal to the face normalized by ρ: $(\vec{\nabla} p.\vec{n}/\rho)_f$:

$$\left(\frac{\vec{\nabla} p.\vec{n}}{\rho}\right)_f = \frac{1}{\hat{\rho}}\frac{p_R - p_L}{h}$$

$$+ \boxed{\left\{\frac{\vec{\nabla} p_L.\vec{E}^- + \vec{\nabla} p_R.\vec{E}^+}{\hat{\rho}h}\right\}}. \tag{17}$$

Here again, the framed (explicit) term contribution goes to zero when the grid becomes orthogonal. The non-framed term is the implicit part that goes into the matrix for the pressure equation.

3.5 Compressive Discretization

This section concerns the face reconstruction of the volume fraction c_i in the conservation equation (8), as this reconstruction is central to the water surface modelization in ISIS-CFD. We focus on an improved scheme, BICS, based on the robust Gamma Differencing Scheme (GDS) [45] as used

in ISIS-CFD for both momentum and turbulence equations, and the Inter-Gamma scheme [46] for its compressive characteristics properties. To guarantee accuracy, stability and boundedness of the solutions, the scheme is developed in the Normalized Variable Diagram (NVD), which is first introduced. The description of the recently developed scheme BRICS, which is based on BICS but has an improved treatment of upwind information, concludes the section.

3.5.1 NVD Diagram and Boundedness Considerations

When constructing a face reconstruction scheme, in order to avoid unrealistic oscillations, especially in the volume fraction c_i which is discontinuous in nature, the search for an acceptable compromise between accuracy and boundedness is a key point [17, 47, 72]. A practical way to introduce a new numerical implementation is to consider the Normalized Variable Diagram (NVD) analysis [16, 51] and to follow the rules that enforce local monotonicity and the Convection Boundedness Criterion (CBC) [29]. The NVD is introduced in general terms here. In the following, the generic quantity Q can be thought of as the volume fraction c_i.

The NVD diagram was originally introduced on structured, one-dimensional grids. On such a grid, in the neighborhood of a face f, a normalized variable $\tilde{Q}$ in the NVD diagram is defined as

$$\tilde{Q} = \frac{Q - Q_U}{Q_D - Q_U}, \tag{18}$$

where points U, C and D are selected according to the flow direction on the face and represent the downwind, central, and upwind cell centers, respectively (see Fig. 16). The idea of the NVD is to represent a reconstruction scheme for the face value by $\tilde{Q}_f$, the normalized value on the face, as a function of $\tilde{Q}_C$.

The CBC criterion [29] corresponds to an area in the NVD in which a scheme must lie in order to be stable and monotone; this area is shaded in Fig. 17a. The shape of this area implies that the first-order upwind differencing scheme (UDS) is the only scheme which unconditionally satisfies the boundedness criterion. The second-order centered differencing scheme (CDS) is only useful in the range $0 \leqslant \tilde{Q}_C \leqslant 1$, as is the first-order downwind differencing scheme (DDS). Practical schemes are often designed as blendings of these elementary schemes, to obtain certain desired properties while remaining inside the CBC area.

3.5.2 NVD on Unstructured Grids

On arbitrary unstructured grids, the far upstream node U is not known explicitly (C and D are still chosen as the centers of the two cells next to the face, see Fig. 17b). It is not even certain that a node exists in the position where U should
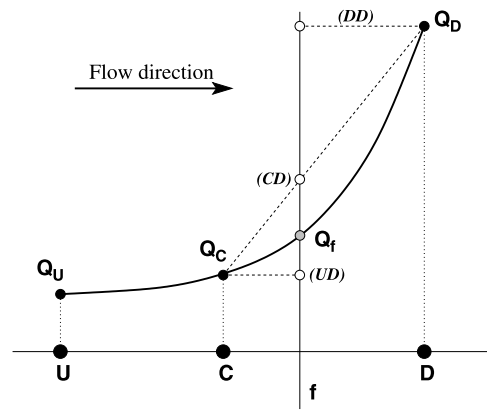

Fig. 16 1D variation of Q in physical space

lie. Therefore, an alternative is chosen: an imaginary nodal quantity Q_U is defined by the use of the gradient projection method [73] in such a way that

$$Q_U = Q_C - \overrightarrow{CU} \cdot \overrightarrow{\nabla} Q|_C, \quad \text{with } \overrightarrow{CU} \triangleq -\overrightarrow{CD}. \tag{19}$$

The location of the imaginary point U is found as the point that is the mirror image of D relative to point C.

For continuous quantities, this procedure is entirely satisfactory. However, since the volume fraction is discontinuous, the reconstructed Q_U may be far away from the actual solution. We will see later how to improve such a reconstruction of the upstream value without the need for the volume fraction gradient.

3.5.3 The BICS Scheme

In order to keep a sharp water surface, the scheme for the volume fraction should ideally be compressive: it should sharpen any gradient into a discontinuity. However, because the solutions are discontinuous, boundedness is of the highest importance as well. And, ideally, there should be no limitation on the cell Courant numbers Co, in order to permit the simulation with reasonable time steps. Our BICS scheme fulfils these requirements by blending together different schemes.

One such scheme, the GDS scheme illustrated in the NVD diagram by Fig. 18a provides a continuous switch between UDS and CDS; it establishes a smooth blending between UDS and CDS over the interval $0 < \tilde{Q}_C < \beta_m$. The GDS scheme is not appropriate for maintaining the sharpness of an interface convected by the transport equation (8), since the use of upwind-differencing produces numerical diffusion which can result in a very strong smearing of the interface. On the other hand, the GDS scheme does not suffer from any cell Courant number limitation and could be used without any time step limitation. The coefficient β_m is

Fig. 17 Basic schemes in NVD diagram and upstream node

(a) UDS, CDS, and DDS schemes

(b) Far imaginary upstream node U for NVD construction
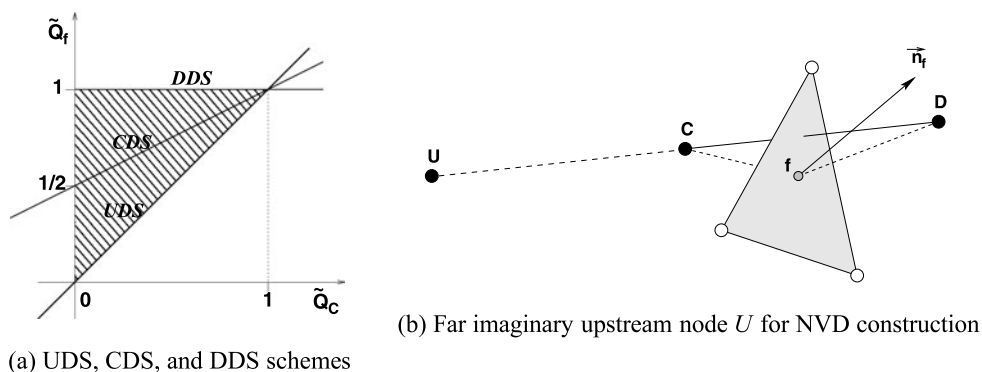
Fig. 18 Basic BICS scheme from GDS and IGDS schemes
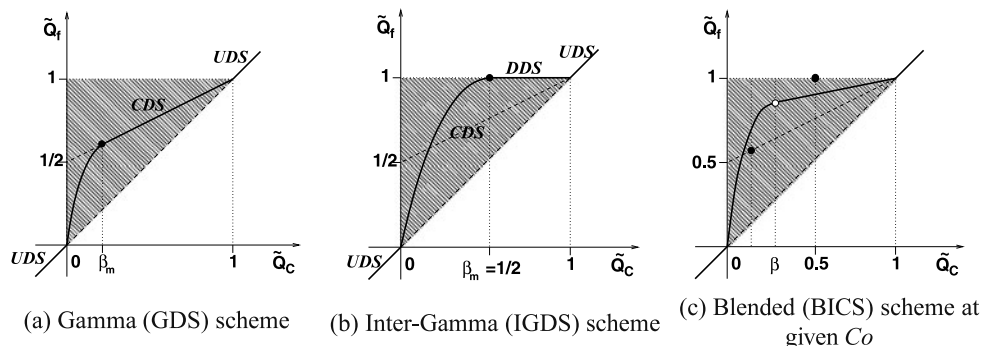
(a) Gamma (GDS) scheme

(b) Inter-Gamma (IGDS) scheme

(c) Blended (BICS) scheme at given Co

usually around 1/6 [29], and to achieve an accurate resolution of sharp profiles, the default value is set to 0.1 in the flow solver.

Since compressive characteristics are absolutely required to build an accurate sharp interface capturing scheme, we also retain the Inter-Gamma IGDS scheme [46], Fig. 18b, which is a straightforward extension of the GDS scheme to downwind differencing (DDS). The switch to DDS starts with $\beta_m = 0.5$. The main disadvantage of the IGDS scheme, with respect to the GDS scheme, is in a cell Courant number limitation, $Co < 0.3$ in multidimensional cases.

The construction of the new Blended Interface Capturing Scheme (BICS) starts with a blending of the GDS and IGDS scheme depending on the local Courant number Co as illustrated in the NVD diagram of Fig. 18c. In that figure, the white point at $\tilde{Q}_C = \beta$ defines the matching point between the two parts: for $\tilde{Q}_C \leqslant \beta$ a quadratic behavior is retained according to the GDS scheme, for $\tilde{Q}_C \geqslant \beta$ a linear variation is retained. The location of this matching point depends on the Courant number: for low Co, this point moves to the right black (IGDS) point, and for high Co it goes toward the left black (GDS) point.

We use the Courant number Co dependency to define an exponential variation for the slope p of the linear (right) part in the NVD diagram

$$p(Co) = \alpha_p(Co) \cdot p_{IGDS} + (1 - \alpha_p(Co)) \cdot p_{GDS}, \quad (20a)$$

Table 1 Basis of the BICS scheme implementation in NVD variables

$\tilde{Q}_C$	$\tilde{Q}_f$	Note
$]-\infty, 0]$	$\tilde{Q}_C$	UDS
$]0, \beta[$	$\frac{1-p}{\beta^2} \tilde{Q}_C^2 + (p + \frac{2(1-p)}{\beta}) \tilde{Q}_C$	Quadratic Blending (GDS-like)
$[\beta, 1[$	$p\tilde{Q}_C + (1 - p)$	Linear (toward DDS)
$[1, +\infty[$	$\tilde{Q}_C$	UDS

with $\alpha_p(Co) = 1$ for $Co \leqslant 0.3$,

$$\alpha_p(Co) = \frac{Co - 0.3}{e^{(Co-0.3)} - 1} \quad \text{for } Co > 0.3, \qquad (20b)$$

and $p_{GDS} = 1/2; \qquad p_{IGDS} = 0,$ (20c)

and for the β abscissa position of the matching point we use a linear dependency on the slope p

$$\beta(p) = a_0 + a_1 \cdot p(Co), \qquad (21a)$$

with $a_1 = (\beta_{GDS} - \beta_{IGDS})/(p_{GDS} - p_{IGDS});$

$$a_0 = \beta_{IGDS} - a_1 \cdot p_{IGDS}, \qquad (21b)$$

with $\beta_{GDS} = 0.1; \qquad \beta_{IGDS} = 0.5.$ (21c)

The basis of the BICS scheme is summarized in Table 1. An illustration of its behavior for different Co can be found in Fig. 19.
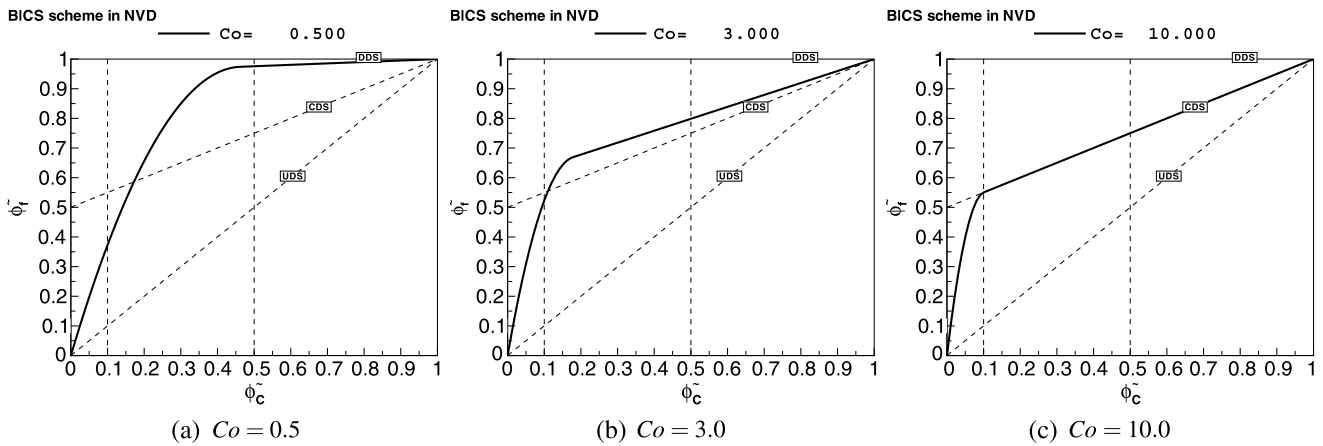
(a) $Co = 0.5$ (b) $Co = 3.0$ (c) $Co = 10.0$

Fig. 19 Basic BICS scheme behavior for various Co

A special case appears when the interface is aligned with the flow direction. Then, too much compression of the interface should be avoided since it causes oscillations in the interface shape. Therefore, a correction C_{θ_f} is added, (22), so that discretization depends on the interface velocity direction and the angle θ_f it makes with the cell face (see Fig. 20). We adopted the classical angle correction [64].

$$\tilde{Q}_f = C_{\theta_f} \tilde{Q}_f + (1 - C_{\theta_f}) \tilde{Q}_{f(GDS)}, \tag{22a}$$

with $C_{\theta_f} = \sqrt{|Cos(\theta_f)|}$, $\tag{22b}$

and $\quad |Cos(\theta_f)| = \dfrac{|\overrightarrow{\nabla} c_i . \overrightarrow{CD}|}{\|\overrightarrow{\nabla} c_i\| \|\overrightarrow{CD}\|}.$ $\tag{22c}$

The result is, that the BICS scheme reduces to the GDS scheme when the interface is parallel to the flow.

3.5.4 From BICS to BRICS

As noted before, for discontinuous solutions, the problem with standard NVD schemes on unstructured grids lies in the treatment of the upwind point U, where the solution is reconstructed using the gradient in the point C (19). As this gradient is by definition a poor approximation to a (nearly) discontinuous solution, the reconstructed value Q_U for BICS will frequently lie outside the physical interval [0, 1] for c_i. Even though we limit the output values Q_f to lie in this interval, the simulated water surface with BICS is often jagged and sometimes unstable.

Therefore, the Blended Reconstructed Interface Capturing Scheme (BRICS) was developed to be the same as BICS, but with a better reconstruction of Q_U. Since we prefer to avoid the use of the volume fraction gradient, the value Q_U is found based on the position of the point U in (19), by first searching the cell closest to this point, and then using a
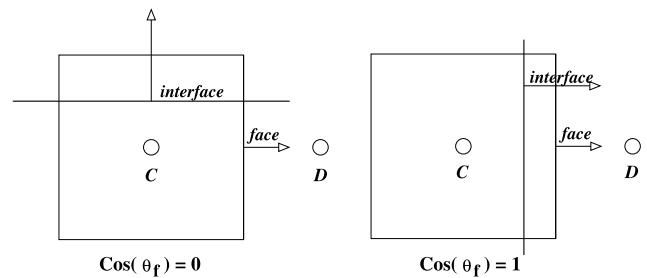

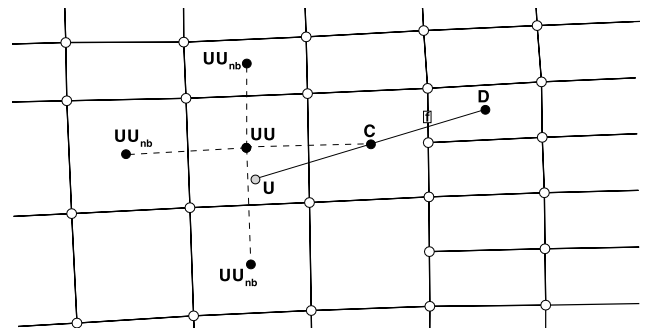**Fig. 20** Face-Interface correction angle

Fig. 21 Improved reconstruction of quantity at the far imaginary upstream point U

weighted interpolation of the state in this cell and its neighbors to define Q_U, see Fig. 21.

The first step in the BRICS scheme consists of running a fast search path algorithm to identify the cell UU containing the imaginary known point U. This search, starting from cell C, consists of a loop over the faces of a cell to identify the face center that lies closest to the point U. The search then goes to the neighbor cell adjacent to this face, where a new loop over the faces is started. The algorithm ends when all scalar products of the vectors center face—U with the outward normals of the faces are < 0, indicating that U lies

in the cell. The cells UU are pre-computed for all the cell faces.

Once the UU container cell is identified, the value at U is computed from a weighted interpolation based on UU and its neighboring cells UU_{nb}:

$$Q_U = \sum_{N=UU,UU_{nb}} \frac{Q_N}{\|\overrightarrow{NU}\|} \bigg/ \sum_{N=UU,UU_{nb}} \frac{1}{\|\overrightarrow{NU}\|}. \qquad (23)$$

Finally, Q_f is computed with the scheme introduced for BICS. The result is a much smoother interface description and more stable and robust computations.

3.6 Algorithm

The discretization of the volume fraction conservation equation, together with the mass and momentum equations, yields a set of algebraic equations: one for each control volume and for each transport/conservation equation. These non-linear and coupled equations are solved by a segregated algorithm. This subsection gives a short overview of the solution algorithm, to indicate how the resolution of the volume fraction equation fits in the overall algorithm.

1. Initialize the flow field quantities Q^0 at $t = t^0$,
2. New time step $t = t + \Delta t$,
3. Start the iterative procedure with $Q = Q^0$,
4. Compute the volume fraction for each fluid phase and update the fluid properties ρ and μ,
5. Compute the turbulent quantities from the field of step 3,
6. Solve the momentum equations to obtain a new prediction for the velocities,
7. Solve the pressure equation to obtain a new pressure field,
8. Update the velocity face fluxes and correct the velocity components with the new pressure field,
9. If the nonlinear residuals are not low enough, go to step 3 and update the iteration counter within the time step,
10. Go to step 2 and update the time, t.

Due to the complexity of the B(R)ICS discretization scheme involved in step 4, a classical defect correction method is used to solve the current volume fraction field, with the implicit part built from the (UDS) differencing scheme.

Concerning the linear solver used in steps 4, 5 and 6 for phase concentration, turbulent quantities, and velocity components, respectively, about 20 Gauss-Seidel iterations are enough to converge when diagonal dominance is increased by 50% with the help of the local time stepping artifact. While this point-wise solver works perfectly on linear systems arising from transport equation discretizations, its efficiency is dramatically reduced on the pressure operator. This operator is singular in nature, having an infinite number of solutions (to a constant pressure). Moreover,

the corresponding linear system is ill-conditioned for highly stretched grids. This is why the flow solver uses an efficient CGSTAB algorithm and an Incomplete LU(k) preconditioning with two levels of filling ($k = 2$). In all cases, the Compressed Sparse Row (CSR) format is the basic format of linear systems adopted in ISIS-CFD.

When running the code on parallel machines, the computational domain is split into multiply-connected domains having approximately the same number of unknowns. This is done with the help of the METIS [48, 62] partitioning algorithm. Communication of faces data between domains is performed according to the Message Passing Interface standard [63].

3.7 Test Cases

The main advantage of the surface capturing formulation for the water surface discretization is its absolute generality. Without modification, the method can handle such problems as ship resistance computation in calm water and diffraction problems for fixed and freely moving ships, in head waves or oblique waves. For all these problems, strong breaking waves and wave interaction with highly complex structures can be modelled. The following section presents three test cases that illustrate the different capabilities of the method.

3.7.1 DTMB 5512 in Head Waves

The first test case illustrates the capacity of the RANS surface capturing method to model the wave diffraction problem. The test case is the US Navy combatant DTMB 5512 at model scale in two conditions: moderate speed in long and smooth waves and high speed in short and steep waves, for which a complex non-linear breaking wave phenomenon is observed. The computations are conducted with the hull in fixed position. Experiments for these cases have been published by IIHR [8, 9].

Grid Generation   The computational domain extends from $-2.0L < x < 3.5L$, $0.0 < y < 2.0L$ and $-1.50L < z < 0.5L$, where the symmetry condition is taken into account at $y = 0$. The ship axis is located along the x-axis with the bow located at $x = 0$ and the stern at $x = L$. The free-surface at rest lies at $z = 0$. The incident wave is generated at the inlet according to the first order Stokes theory, wall function is used at the ship wall and various far-field and symmetry boundary conditions are used at the other boundaries of the computational domain. The unstructured hexahedral grid is generated with HEXPRESS. A local zone of refinement is created near the free surface in the entire domain, to ensure a small enough z-grid spacing. A second refinement zone in the vicinity of the body ensures small x- and y-spacing there to properly capture details of the free surface. Correct

Fig. 22 DTMB5512—$Fn = 0.28$—Local grid distribution in the vertical plane of symmetry

(a) Bow   (b) Stern

(a) Resistance   (b) Heave
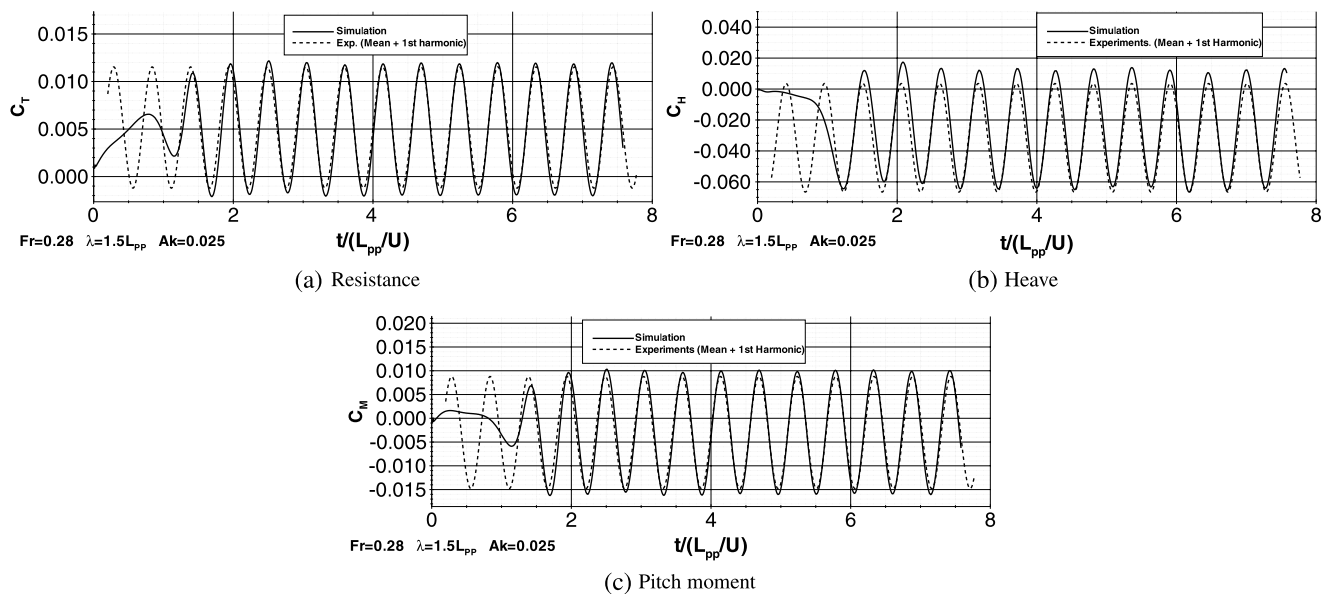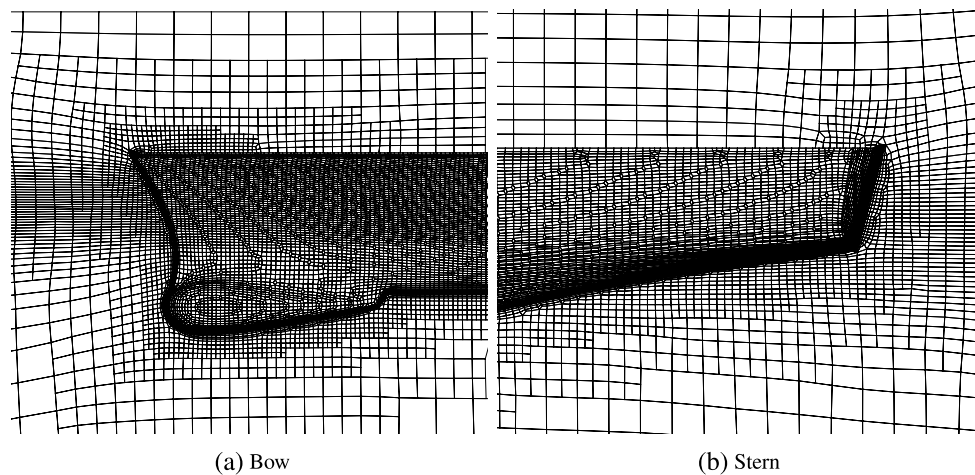
(c) Pitch moment

Fig. 23 DTMB5512—$Fn = 0.28$—Time evolution of the resistance, heave and pitch moment compared with experiments

propagation of the incoming waves is assured by the use of a refinement box starting from the inlet and ending after the ship.

Figures 22 illustrates the local grid distributions close to the bow and stern. This grid is composed of 2.0 million cells with about 40,000 cells located on the hull. The first point close to the hull is located at $y^+ = 30$, approximately. About 20 to 25 points are located vertically in the layer where the free-surface is supposed to move and a wave length is described with 40 to 50 cells.

For the high-speed case, three additional refinement boxes are included in order to improve the simulation of bow and stern breaking waves. Two boxes are placed close to bow to improve the simulation of the first and second bow breaking waves. Another box of refinement is added at the stern to improve the capturing of stern breaking waves. This

grid is composed of 4.0 million cells with about 75,000 cells located on the hull.

Long Waves at Moderate Speed The first test case has $Fn = 0.28$, $\lambda = 1.5L$ and $Ak = 2\pi \frac{A}{\lambda} = 0.025$.

Figure 23 provides a comparison between the temporal evolution of experimental and computed forces and moment coefficients. The experiments are rebuilt with their mean and first harmonic. Agreement with experimental signals is good even if the maximum of the heave force is slightly overestimated. For this case of moderate Froude number and wave amplitude but long wave length, the periodic behavior is rapidly captured after four wave interactions. No phase analysis has been performed and the rebuilt experimental signals are simply time adjusted to match the maximum of the computed force coefficients. It should be noticed that the ship starts at rest with an acceleration to reach its nominal
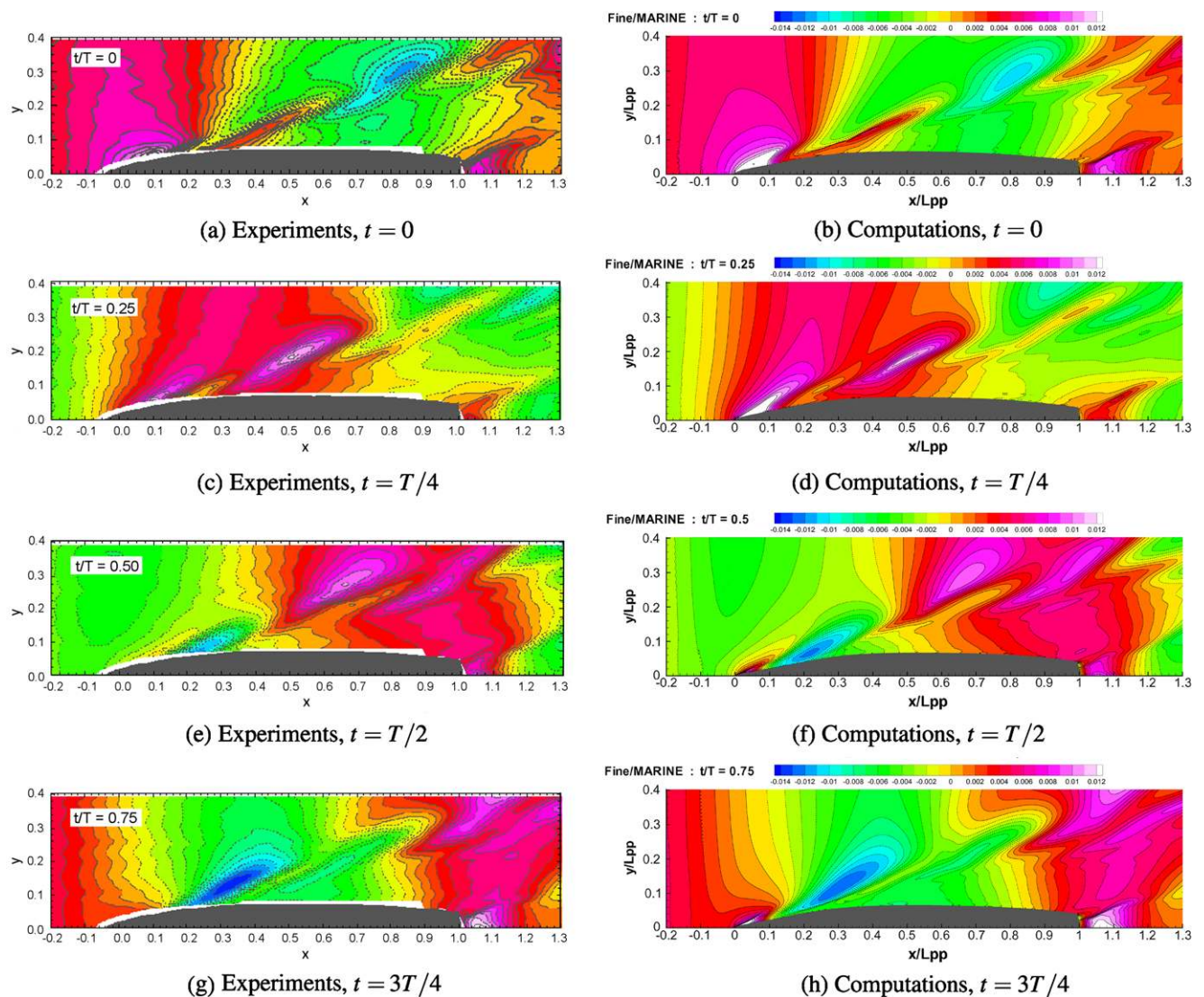
Fig. 24 DTMB5512—$Fn = 0.28$—Phase-averaged free-surface elevations

speed at time $t = 2s$ following a quarter-sine law to prevent a too high response of the free-surface to high acceleration.

Figure 24 compares the unsteady wave pattern obtained by numerical simulation with results from experiments. Four selected times are retained during one encounter period: time $t/T = 0$ corresponds to the time when the wave crest reaches $x/L = 0$. Quantitative agreement is achieved at any of the four specific times for both the locations and the amplitudes of the waves. The compressive property of the surface capturing scheme is evident when looking at the shape of the very crisp bow and stern waves.

Short Waves at High Speed The high-speed case has $Fn = 0.41$, $\lambda = 0.5L$ and $Ak = 0.075$. A first interesting result is the time needed to get a periodic signal for the resistance. It was found that 12 wave periods are needed

to get a signal of good periodicity. The FFT analysis of the resistance is shown in Fig. 25. We notice a very good agreement between the computed and experimental signals, and on the other hand, the very rich content of the computed signals in terms of frequency. The first harmonic is of course equal to the encounter frequency but the second harmonic, in excellent agreement with the measurements, is quite strong (about 23% in terms of amplitude). There is evidence that this harmonic is associated with the breaking wave phenomena [101]. The agreement on the amplitudes is very satisfactory on the heave and pitch and one notices a slight overestimation of the amplitude of the first harmonic of the resistance force.

Figures 26 and 27 provide perspective views of the computed wave profile at three specific times, T_b is the time at which breaking of the main bow wave occurs, T_{min} and T_{max} are the instants corresponding to the minimum and

(a) Resistance (b) Heave (c) Pitch moment
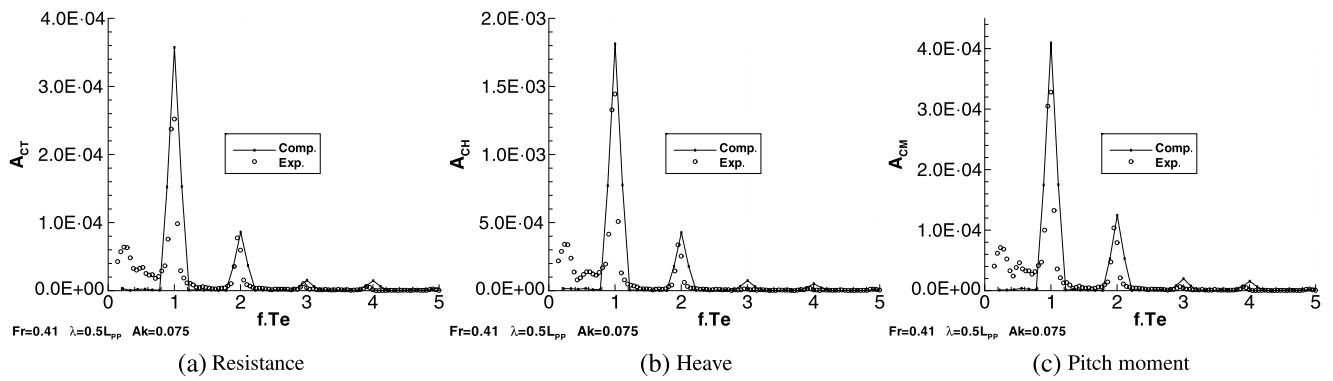
Fig. 25 FFT analysis of the resistance, heave and pitch moment for the fixed ship—$Fn = 0.41$, $Ak = 0.075$ case

Fig. 26 Phase-averaged free-surface elevations close to the bow at times T_{min}, T_{max} and T_b for the fixed ship—$Fn = 0.41$, $Ak = 0.075$ case


(a) Tmin (b) Tmax (c) Tb
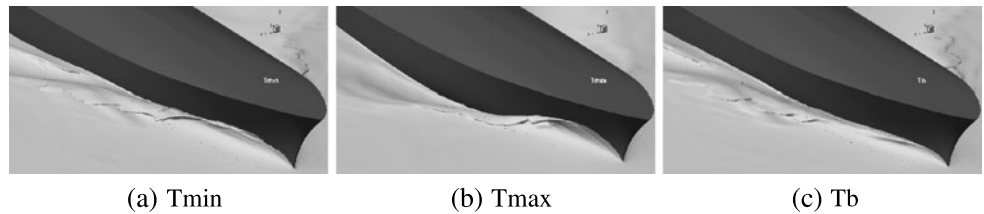
Fig. 27 Phase-averaged free-surface elevations close to the stern at times T_{min}, T_{max} and T_b for the fixed ship—$Fn = 0.41$, $Ak = 0.075$ case

(a) T_{min} (b) T_{max} (c) T_b
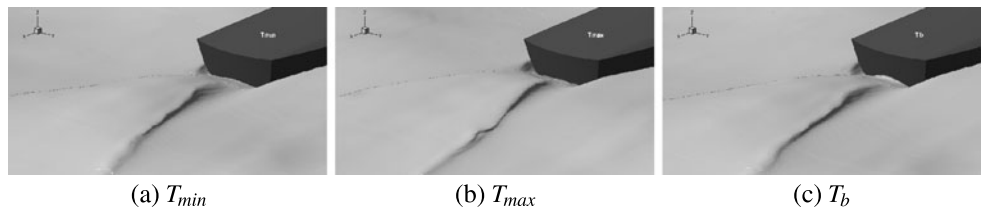
maximum resistance force, respectively. The main breaking phenomena is clearly illustrated and one can see the secondary shoulder breaking wave occurring at T_{min}. Near the stern, the rooster-tail breaking wave is also well defined behind an almost entirely dry transom since only a very small portion of the bottom of the transom appears to be periodically wetted.

3.7.2 Planing Hull at High Fn

Modern fast sailing and motor boats operate at high Froude numbers (>0.5) where massive wave breaking and foam creation form an essential part of the flow field. Surface capturing is at this moment the only RANS technique that can reasonably approximate these flows. Here, a study for French yacht designers Finot-Conq is presented, that concerns the drag and heave force prediction on an IMOCA Open 60 sailing yacht.

Computations are performed at different Fn for a hull at fixed attitude in calm water. A relatively coarse mesh is used with 800,000 cells for the half ship. As the computations are compared with experiments in a small towing tank, the bottom of the tank is taken into account for the computations.
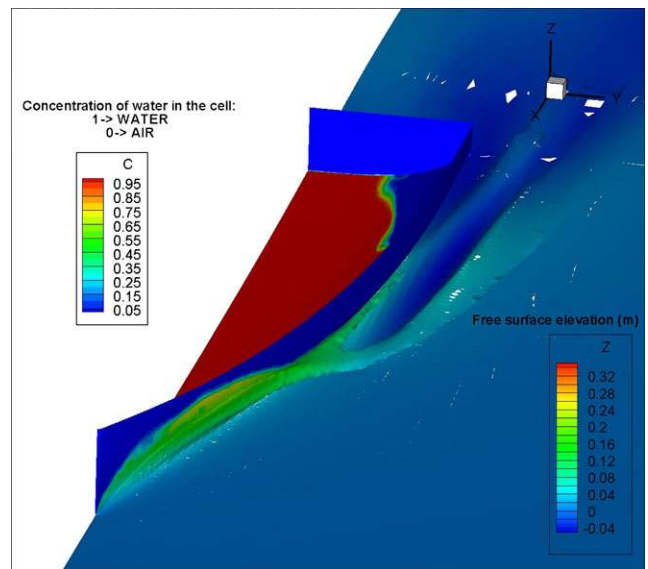
Fig. 28 Dynamic wetted surface and free-surface elevation for the Open 60 hull at $Fn = 0.96$ (25 kts)

Figure 28 shows the water surface at $Fn = 0.96$. At this speed, a strong, well-developed breaking bow wave is observed, that even produces a secondary breaking wave be-
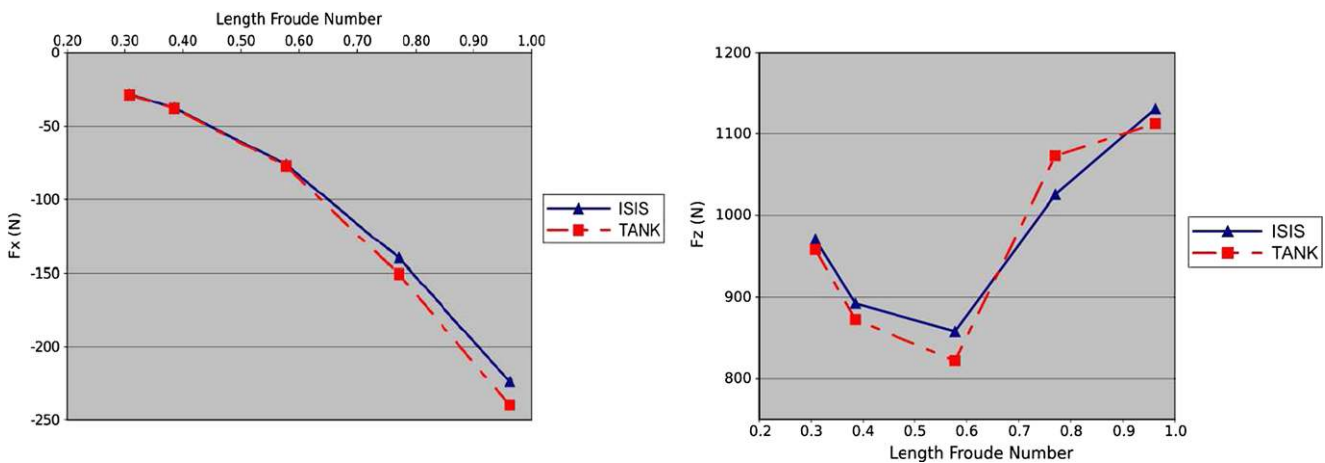
Fig. 29 Drag and heave force for the Open 60 hull at different Froude numbers

hind it. A part of the hull is sustained by foam and bubbly water, that is represented in the capturing model by water concentrations between 0 and 1.

Forces on the hull are given in Fig. 29. The agreement with measurements is excellent, especially when considering that the grid is not extremely fine. The only large difference that is observed, is in F_z at $Fn = 0.96$. As this corresponds to a depth Froude number in the tank of nearly 1, the difference is likely due to choking effects in the tank. In all the other cases, the errors are 1–2% in F_z and less than 4% in F_x.

3.7.3 Free Container Ship in Head and Oblique Waves

Seakeeping investigates wave load and wave induced ship motion in seaways. Under extreme wave conditions, strong nonlinear interaction between waves and the moving ship occurs. The surface capturing method is well able to simulate these cases. An example, taken from the CRS Forward Speed Benchmark workshop [60], is presented in this section.

Case Setup A model scale container ship is simulated in the present study. Measurement data were provided by MARIN during the CRS workshop. Ship length is $L_{pp} = 4.9$ m and the ship speed is $U = 1.7$ m/s (24.5 knots for full scale ship), leading to a Froude number $Fn = 0.245$ and a Reynolds number $Rn = 8.54e6$. One wave height $H = 0.08$ m and two wave directions are simulated, namely 180° (head waves), and 225° (oblique waves on port bow). Wave length ranges from $0.23L_{pp}$ to $2.5L_{pp}$, the highest wave steepness being $Ak = 0.224$.

To capture correctly the incoming wave in a RANS simulation, the free-surface needs to be refined sufficiently in the wave propagation direction. Previous experiences show that 60 grid nodes per wave length is an appropriate choice.
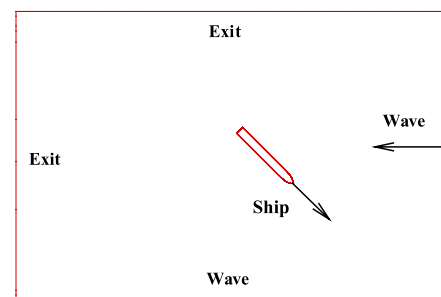
**Fig. 30** Boundary conditions for container ship in oblique waves

In the region covered by the wave, at least 10 nodes in the vertical direction are required. To satisfy this criterion, 4 different meshes containing 1.5 M, 2.0 M, 2.8 M and 3.8 M nodes, respectively, have been generated for different wave frequencies.

For the ship motion, the horizontal displacements are obtained by a block movement of the mesh, while vertical displacement and all rotations are treated with analytical weighted mesh deformation [53].

For the head wave case, pressure is prescribed at the upper boundary, while a slip boundary condition is applied to the bottom boundary. Mirror and exit boundary conditions are applied to the side and outflow boundary. No special treatment at the exit is applied for wave absorption. The setup for oblique waves computation requires special attention. In the present study, we use a computational domain that is parallel to the wave propagation direction, as shown in Fig. 30. To capture the incoming waves, these meshes need to be refined in the x-direction only. A wave boundary condition is applied at the two upstream boundaries, while exit conditions are applied at the other two boundaries.

Seakeeping Simulation with Head Waves Figure 31 compares the predicted heave and pitch motion amplitude in the

Fig. 31 Heave and pitch motion at the center of gravity for the container ship in head waves
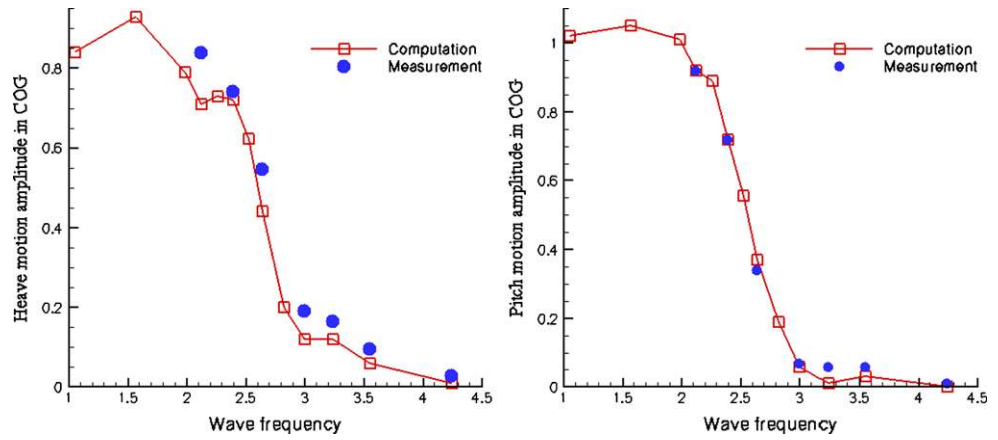
Fig. 32 Added resistance and resistance time evolution for the container ship in head waves
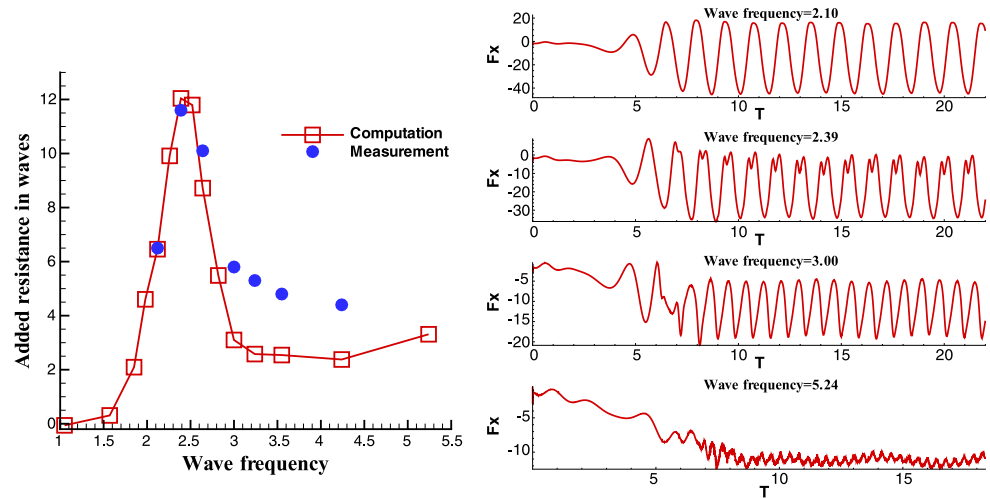
center of gravity with the measurement data obtained by MARIN. At low wave numbers, when the wave length is more than two times larger than the ship length, the motions are strong; the heave motion amplitude is very close to the wave amplitude. At high wave numbers, when the wave length is about one third of the ship length, the motions are nearly zero. Very good agreement is observed compared with measurement data except at low frequency.

Added resistance in waves is shown in Fig. 32. At low wave frequencies, the numerical prediction is in good agreement with the measurement. At high wave frequency, predicted added resistance in wave is about 50% smaller than the measurement data, which corresponds to about 14% of the total resistance.

Figure 32 shows the time evolution of ship resistance in waves for 4 selected wave frequencies. At low frequency, there is a nearly linear response to waves. As wave frequency increases, wave steepness increases. Interaction of steeper waves with ship motion results in bow wave breaking, which increases the added resistance in waves considerably. At even higher wave frequencies, the effect of incoming waves becomes smaller due to the shorter wave

length. However, this wave response is non-linear, multiple frequencies can be observed both in ship motion and in wave load. Higher added resistance in waves is observed for the case with highest frequency (5.24) simulated in the present study is due to the existence of a low frequency motion.

Seakeeping Simulation with Oblique Waves For the case with oblique waves, sway and yaw motion are frozen, while heave, roll and pitch motions are free. Beside the special boundary condition setup, the other aspects of the numerical simulation are similar to the case with head waves. Pitch and roll motion amplitudes are presented in Fig. 33. The pitch motion amplitude agrees reasonably well with measurement data, although there is a systematic over-estimation. Roll results show very good agreement. It was noted that bow wave breaking occurs at two of the simulated wave frequencies, 2.64 and 3.0.

3.8 Summary and Perspectives

This section has presented a surface capturing method, in which the volume fraction conservation law is discretized
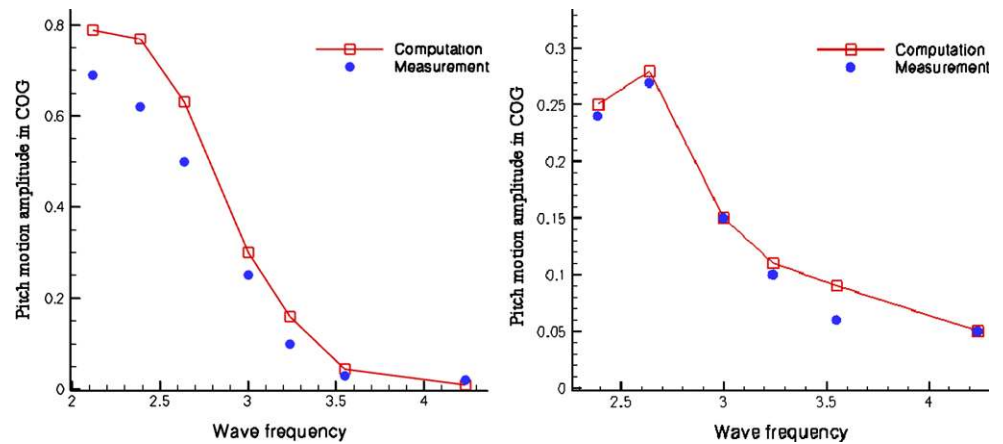
with a specific compressive scheme. No reconstruction of the water surface or application of free-surface boundary conditions are needed.

The main advantage of this method is its high generality: it gives satisfactory results for cases that cover most of the application domain for naval hydrodynamics, i.e. still-water resistance, wave diffraction and free-motion problems, displacement and planing hulls, as well as a large range of Froude numbers. The method is robust, as it continues to work well even in the presence of very strong wave breaking. The test cases presented confirm this robustness and generality.

A point of attention for the method is the simulation of wave propagation. Preserving a wave that travels over a long distance requires fine grids at the water surface, as became clear in the seakeeping test presented. Techniques under consideration for improving the wave propagation are higher-order accurate discretizations and the use of automatic grid refinement.

4 The Level Set Method

4.1 Introduction

One of the interface capturing methods for free surface flow simulations, a level set method [67, 68, 84, 85] is described here. In this method, the level set function which is the signed distance from the interface is used as an indicator of the interface location.

As indicated in the Introduction, the main characteristic of interface capturing methods is that a computational grid does not fit to the interface shape and that the interface is "captured" somewhere between the grid points. An advantage of the interface capturing approaches in general is that they can be used when interface deformation is large or topology of interface changes, such as overturning or breaking of waves occurs, while the interface fitting methods

would encounter difficulties of gridding in these situations. Another advantage of capturing is that a grid needs not be regenerated in order to follow the interface deformation. This would be a big benefit in case that grid generation requires large efforts such as grids around a complex geometries. On the other hand, numerical accuracy of the capturing method degrades to some extent, because there are no grid points at the exact location of the interface where the boundary conditions should be satisfied.

There are many methods which can be categorized as interface capturing methods. Among them, the volume-of-fluid (VOF) method [40] which has been introduced in the previous section and the level set method which is described here are two major implementations. The VOF method employs a volume fraction of each fluid in a cell as the indicator of the interface, while the level set method utilizes the level set function, the signed distance from the interface, to locate the interface. In the VOF method, it is essential to use a specially designed convection scheme which has good conservation property and prevents smearing of the interface due to numerical dissipation as described in the previous section, since a volume fraction is a discontinuous function. On the other hand, the level set function is a linearly varying function and its convection can be handled in a simpler way, although the conservation property of the level set method is not as good as the VOF method since the level set function does not guarantee that fluid volumes are preserved in the interface.

Nowadays, computations by use of a level set method are found in many publications. The level set approaches can be categorized into two groups. One is a two-phase fluid approach in which both water and air regions together with an interface movement between them are solved by the multi phase Navier-Stokes equations coupled with a level set equation. This is the same way as the original level set approach [90].

Cura Hochbaum et al. [13] used a two-phase fluid level set method for the simulations of free surface flows. This

code *NEPTUN* is a multi block solver based on the SIMPLE algorithm. Its applications range from resistance to maneuvering or seakeeping performance of ships.

Vogt et al. [102] presented the results for two-dimensional free surface flows around a submerged hydrofoil using a two-phase level set method.

Yang et al. [105] applied a Cartesian grid method coupled with two-phase level set to Large Eddy Simulation (LES) for ship flows by their code *CFDSHIP-IOWA Ver.6*.

Another method which adopts the Cartesian grid with level set method is developed by Dommermuth et al. [23]. They used the unique interface capturing method (CLSVOF) which couples both the level set method and VOF method and simulated free surface flows around a ship and ship-like bodies.

The other approach is a single-phase approach in which a flow field is analyzed only in a water region and flow variables in an air region are extrapolated using free surface boundary condition. This approach seems to become a majority in level set methods for marine hydrodynamics.

The National Maritime Research Institute (NMRI) group [34, 35] adopts a single-phase method in their unstructured grid based Navier-Stokes solver, *SURF*. The algorithm of this particular method is shown in the following section.

The research group at the University of Iowa also uses a single-phase level set method in the recent versions of their code *CFDSHIP-IOWA* [10, 104]. The flow solver is based on PISO for velocity-pressure coupling and a block structured grid system including overset grids is adopted. Many applications in ship hydrodynamics are reported using this method.

The group at INSEAN (Italian Ship Model Basin) also develops a program based on a single-phase level set approach. Their flow solver *χship* [22] adopts an artificial compressibility approach for pressure velocity coupling. Overset multiblock structured grids are used for complex geometry cases.

Another unstructured grid based solver with single-phase level set method is developed by Burg [6]. It is an extended version of U^2NCLE code developed at Mississippi State University [44].

4.2 Numerical Procedure of *SURF*

In this section, the Navier-Stokes solver *SURF* ("<u>S</u>olution algorithm for <u>U</u>nstructured <u>R</u>aNS with <u>F</u>VM") [32–34], which is under development toward a practical ship design tool at National Maritime Research Institute, is introduced.

4.2.1 Flow Solver

The governing equations are the three-dimensional Reynolds-Averaged Navier-Stokes equations for incompressible flows. In order to couple pressure with a velocity field, artificial compressibility is introduced into the continuity equation at the expense of time accuracy. (Unsteady flow computation is also possible by use of dual time stepping [81].) The final form using non-dimensional variables can be written as follows:

$$\frac{\partial \overrightarrow{q}}{\partial t} + \frac{\partial(\overrightarrow{e} - \overrightarrow{e}^v)}{\partial x} + \frac{\partial(\overrightarrow{f} - \overrightarrow{f}^v)}{\partial y} + \frac{\partial(\overrightarrow{g} - \overrightarrow{g}^v)}{\partial z} = 0,$$

(24)

and

$$\overrightarrow{q} = \begin{bmatrix} p \\ u \\ v \\ w \end{bmatrix},$$

where p is the modified pressure from which hydrostatic component is extracted, i.e.,

$$p = p^* + \frac{z}{Fn^2},$$

where p^* is the original pressure and Fn is the Froude number with z being the vertical coordinate. By this modification of pressure, the gravitational acceleration term can be omitted from the z-momentum equations. The velocity components in the (x, y, z) direction are expressed as (u, v, w).

The inviscid fluxes $\overrightarrow{e}$, $\overrightarrow{f}$ and $\overrightarrow{g}$ are defined as

$$\overrightarrow{e} = \begin{bmatrix} \beta u \\ u^2 + p \\ uv \\ uw \end{bmatrix}, \qquad \overrightarrow{f} = \begin{bmatrix} \beta v \\ vu \\ v^2 + p \\ vw \end{bmatrix},$$

$$\overrightarrow{g} = \begin{bmatrix} \beta w \\ wu \\ wv \\ w^2 + p \end{bmatrix},$$

where β is a parameter for artificial compressibility. $\overrightarrow{e}^v$, $\overrightarrow{f}^v$ and $\overrightarrow{g}^v$ are the viscous fluxes defined as follows:

$$\overrightarrow{e}^v = \begin{bmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ \tau_{zx} \end{bmatrix}, \qquad \overrightarrow{f}^v = \begin{bmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ \tau_{yz} \end{bmatrix}, \qquad \overrightarrow{g}^v = \begin{bmatrix} 0 \\ \tau_{zx} \\ \tau_{yz} \\ \tau_{zz} \end{bmatrix},$$

where

$$\tau_{ij} = \left(\frac{1}{Rn} + v_t \right) \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right),$$

and $(x_x, x_y, x_z) = (x, y, z)$ and $(u_x, u_y, u_z) = (u, v, w)$. In the above expressions all the variables are made dimensionless using the reference density ρ, velocity U and length L.
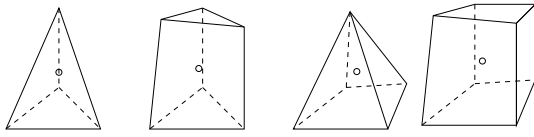
Fig. 34 Cell shapes

Rn is the Reynolds number defined as UL/ν where ν is the kinematic viscosity. ν_t is the non-dimensional kinematic eddy viscosity determined by a turbulence model.

Since the numerical procedure for the Navier-Stokes equations is described in the references [32, 33], only the brief outline is given here. A finite-volume method is adopted for spatial discretization. First, a computational domain is divided into unstructured polyhedral cells. Cell shapes which can be used in the present solver are tetrahedron, prism, pyramid or hexahedron and face shapes of these cells are either triangular or quadrilateral as shown in Fig. 34. Flow variables (pressure, velocity and eddy viscosity) are stored in the center of each cell.

For the inviscid fluxes (convection terms and pressure gradient terms), the second order upwind scheme based on the flux-differencing splitting of Roe [83] with the MUSCL method are employed. The viscous fluxes are evaluated by the second order central scheme. Thus, the overall accuracy in space is second order.

The backward Euler scheme is used for the time integration. The linear equations derived from the time linearization of the fluxes are solved by the Symmetric Gauss-Seidel (SGS) iteration.

The turbulence models implemented are the one-equation model by Spalart and Allmaras [87] and its modified version [39].

4.2.2 Free Surface Treatment

The free surface is an interface between air and water in the present context. Free surface conditions consist of dynamic and kinematic conditions and they are implemented in the interface capturing framework.

The kinematic condition is the condition that fluid particles on a free surface remain on an interface. This is written in a mathematical form as follows:

$$\frac{\mathrm{D}H}{\mathrm{D}t} \equiv \frac{\partial H}{\partial t} + u\frac{\partial H}{\partial x} + v\frac{\partial H}{\partial y} + w\frac{\partial H}{\partial z} = 0, \quad (25)$$

where a free surface shape is defined as

$$H(x, y, z; t) = 0.$$

In the present scheme, this kinematic condition is formulated based on the localized level set method [70] which improves the efficiency of the original level set approach [90].

The level set function ϕ is defined as the signed distance from the interface, i.e.,

$$\phi \begin{cases} > 0 & \text{in water,} \\ = 0 & \text{on the interface,} \\ < 0 & \text{in air.} \end{cases} \quad (26)$$

Since $\phi(x, y, z; t) = 0$ defines the free surface shape, the kinematic condition can be satisfied if the following equation is used to update ϕ:

$$\frac{\mathrm{D}\phi}{\mathrm{D}t} \equiv \frac{\partial \phi}{\partial t} + u\frac{\partial \phi}{\partial x} + v\frac{\partial \phi}{\partial y} + w\frac{\partial \phi}{\partial z} = 0. \quad (27)$$

In the localized version of the level set method, the two parameters γ_1 and γ_2 where $0 < \gamma_1 < \gamma_2$ are introduced. The signed distance function is rewritten as $d(x, y, z; t)$ and the definition of the level set function is modified as

$$\phi = \begin{cases} \gamma_2 & \text{if } d > \gamma_2, \\ d & \text{if } |d| \leq \gamma_2, \\ -\gamma_2 & \text{if } d < -\gamma_2. \end{cases} \quad (28)$$

Thus, the level set function is localized within the bandwidth $2\gamma_1$ from the interface. The transport equation (27) is also modified as

$$\frac{\partial \phi}{\partial t} + C(\phi)\left(u\frac{\partial \phi}{\partial x} + v\frac{\partial \phi}{\partial y} + w\frac{\partial \phi}{\partial z}\right) = 0, \quad (29)$$

where $C(\phi)$ is the cut-off function defined as

$$C(\phi) = \begin{cases} 1 & \text{if } |\phi| \leq \gamma_1, \\ \frac{(|\phi|-\gamma_2)^2(2|\phi|+\gamma_2-3\gamma_1)}{(\gamma_2-\gamma_1)^3} & \text{if } \gamma_1 < |\phi| \leq \gamma_2, \\ 0 & \text{if } \gamma_2 < |\phi|, \end{cases} \quad (30)$$

in such a way that the update of ϕ is performed only in the region where $|\phi| \leq \gamma_2$.

The numerical solution method for (29) is identical to the one for the flow equations (24). The cell centered finite-volume discretization applied for the cell i yields

$$V_i \frac{\partial \phi_i}{\partial t} + \sum_j C(\phi_i)(\phi_{(i+j)/2}U_{(i+j)/2}) = 0, \quad (31)$$

where

$$U_{(i+j)/2} \equiv u_i S_{x,(i+j)/2} + v_i S_{y,(i+j)/2} + w_i S_{z,(i+j)/2},$$

V_i is the cell volume and j is the neighbor cells of the cell i. The subscript $(i + j)/2$ means the cell face between the cells i and j and (S_x, S_y, S_z) are the area vectors of cell faces. $\phi_{(i+j)/2}$, the value of ϕ on the cell face, is extrapolated from the cell centered values in the second order upwind manner.

The gradient of ϕ at the cell center used in the extrapolation above is obtained by the least squares method. The time integration is carried out by the Euler backward scheme.

Special care should be taken in the construction of the numerical flux $\phi_{(i+j)/2}U_{(i+j)/2}$. Suppose that the interface is undisturbed on $z = 0$ and the flow field is uniform and $(u, v, w) = (1, 0, 0)$. The initial value of ϕ is $\phi = -z$. This is apparently a steady state solution of the analytical equation (27). The discrete equation (31), however, does not necessarily gives $\partial\phi/\partial t = 0$, unless

$$\sum_j (\phi_{(i+j)/2}S_{x,(i+j)/2}) = 0, \tag{32}$$

is satisfied. Since the current function $\phi = -z$ is a linearly varying function, the extrapolation of the face value $\phi_{(i+j)/2}$ from the cell centered values can be performed to give the exact value. However, the one point quadrature $\phi_{(i+j)/2}S_{x,(i+j)/2}$ has a certain error when applied to quadrilateral faces. The higher order quadrature which gives exact results will increase the computational cost considerably. The approach taken here to remedy this problem is to introduce a new variable $\tilde{\phi}$ which is defined as

$$\tilde{\phi} = \phi + z, \tag{33}$$

and to solve $\tilde{\phi}$ instead of ϕ. Thus, (27) is modified as

$$\frac{\partial\tilde{\phi}}{\partial t} + C(\phi)\left(u\frac{\partial\tilde{\phi}}{\partial x} + v\frac{\partial\tilde{\phi}}{\partial y} + w\frac{\partial\tilde{\phi}}{\partial z} - w\right) = 0, \tag{34}$$

and in the discrete form

$$V_i\frac{\partial\tilde{\phi}_i}{\partial t} + \sum_j C(\phi_i)(\tilde{\phi}_{(i+j)/2}U_{(i+j)/2})$$
$$- C(\phi_i)V_i w_i = 0. \tag{35}$$

For the initial field of $\phi = -z$ and $(u, v, w) = (1, 0, 0)$, $\partial\tilde{\phi}/\partial t = 0$ because $\tilde{\phi} = 0$ everywhere and the fluxes are evaluated as zero, regardless of the quadrature.

In order to avoid reflection of free surface waves in the outer boundaries of a computational domain, the wave damping method which has been proved to be effective in the interface fitting approach [37] is also applied to the level set method. The damping term which makes $\tilde{\phi}$ approach to zero is added to the level set equation (35) as follows:

$$V_i\frac{\partial\tilde{\phi}_i}{\partial t} + \sum_j C(\phi_i)(\tilde{\phi}_{(i+j)/2}U_{(i+j)/2})$$
$$- C(\phi_i)V_i w_i + V_i W\tilde{\phi}_i = 0. \tag{36}$$

$W(x, y, z)$ is weight function defined as

$$W(x, y, z) = A \times \max(W_x(x), W_y(y)),$$

$$W_x(x) = \begin{cases} (\frac{x-x_d}{x_o-x_d})^2 & \text{if } x_d \leq x \leq x_o, \\ 0 & \text{otherwise,} \end{cases}$$

$$W_y(y) = \begin{cases} (\frac{y-y_d}{y_o-y_d})^2 & \text{if } y_d \leq y \leq y_o, \\ 0 & \text{otherwise,} \end{cases}$$

where x_d, y_d are the coordinates from which the damping region starts and x_o, y_o are the location of outflow and side boundaries. The parameter A controls the amount of damping. The added term $V_i W\tilde{\phi}_i$ is proved to effectively make $\tilde{\phi}$ approach to zero in the damping region and thus dissipate outgoing waves.

There is a singular behavior of the interface in the region close to a solid wall. The no-slip condition imposed on a wall prevents the interface movement there, while the interface in the outer region moves following the fluid motion. It causes the large deformation of ϕ near a solid wall. The extrapolation approach is employed here to remove this singularity, in which the value of ϕ for the cells close to the wall is extrapolated from the outer cell.

It is easy to find the outer cell for a particular cell in the structured grid case due to the alignment of cells along a grid line. In the unstructured grid case, however, it is not a trivial task. In order to carry out the extrapolation, the distance to the closest wall d_i (where i is the cell index) is computed and stored for each cell first. In the region where extrapolation is needed, ϕ is extrapolated from outside in such a way that it is constant in the direction normal to the wall. Thus, the pseudo convection equation

$$\frac{\partial\phi}{\partial\tau} - \frac{\nabla d}{|\nabla d|} \cdot \nabla\phi = 0, \tag{37}$$

is solved in the pseudo time τ in the extrapolation region. Note that the convection velocity $-\nabla d/|\nabla d|$ is the unit vector normal to the wall from outside to a wall. It ensures the extrapolation from outside when an upwind scheme is used for the discretization. Upon convergence, ϕ becomes constant in the direction normal to the wall as desired.

4.2.3 Re-initialization

The re-initialization of the level set function is an important step in the level set method, since the level set function is no longer a distance function after the convection. The re-initialization process can be done using the partial differential equation as in [90] or [70].

The equation to be solved is

$$\frac{\partial\phi}{\partial\tau} + S(\phi_0)(|\nabla\phi| - 1) = 0. \tag{38}$$

Where ϕ_0 is the initial value and S is the sign function

$$S(\phi) = \begin{cases} -1 & \text{if } \phi < 0, \\ 0 & \text{if } \phi = 0, \\ 1 & \text{if } \phi > 0. \end{cases}$$

Upon convergence, ϕ becomes the distance function again, since $|\nabla\phi| = 1$.

In a numerical process, $S(\phi)$ is approximated as

$$S_\varepsilon(\phi) = \frac{\phi}{\sqrt{\phi^2 + \varepsilon^2}}, \tag{39}$$

where ε is a typical grid spacing.

Equation (38) is rewritten as

$$\frac{\partial \phi}{\partial \tau} + S_\varepsilon(\phi_0)\frac{\nabla\phi}{|\nabla\phi|} \cdot \nabla\phi = S_\varepsilon(\phi_0), \tag{40}$$

which can be viewed as the convection equation for ϕ with the convection velocity being $S_\varepsilon(\phi_0)\nabla\phi/|\nabla\phi|$.

The numerical procedure for (40) is similar to the convection equation (36). First, $\nabla\phi$ is computed using the least squares method for each cell center. The convection velocity $S_\varepsilon(\phi_0)\nabla\phi/|\nabla\phi|$ is then evaluated. The second $\nabla\phi$ in (40) is discretized by the first order upwind scheme in a finite volume framework. The pseudo time integration is made by the Euler implicit scheme.

4.2.4 Flow Variable Extrapolation

Since most ship hydrodynamics applications require a flow field of water region only, a one-phase flow approach is used, i.e., the flow equations are solved only in a water region. Flow variables in an air region are extrapolated from a water region in such a way that the dynamic condition on the free surface boundary is satisfied. This method also has an advantage that it is not necessary to cope with the large density difference between air and water. At this point, the present method differs from the original level set method [90] where a two-phase flow approach is employed.

The dynamic free surface condition can be approximated by the following two conditions. First, the velocity gradients normal to the free surface are zero. Second, the pressure on the free surface is equal to atmospheric pressure. In order to satisfy the first condition, the velocity components are extrapolated in the direction normal to the interface. Following the localized level set method [70], this is achieved by solving the following equation in an air region where $\phi < 0$ for the pseudo time τ.

$$\frac{\partial \overrightarrow{q}}{\partial \tau} - \frac{\nabla\phi}{|\nabla\phi|} \cdot \nabla \overrightarrow{q} = 0. \tag{41}$$

Note that the quantity $-\nabla\phi/|\nabla\phi|$ is the unit vector normal to the interface whose direction is from water to air.
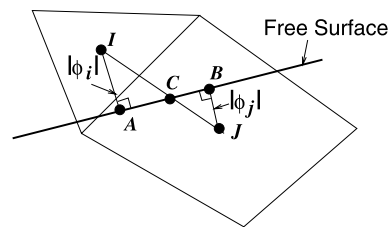
Fig. 35 Pressure condition on free surface

In the region away from the interface where ϕ is constant, $-\nabla\phi/|\nabla\phi|$ is replaced by the vector $(0, 0, 1)^T$.

The pressure boundary condition is written as

$$p = \frac{\zeta}{Fn^2} \quad \text{on the free surface,} \tag{42}$$

where atmospheric pressure is assumed to be zero and ζ is the z-coordinate of the interface. For an air cell which is next to a water cell, pressure is extrapolated in the following way. Suppose that the cell i is the air cell for which the pressure must be extrapolated and the cell j is the neighboring water cell as shown in Fig. 35. From the definition, $|\phi|$ is the distance to the interface with $\phi_i < 0$ and $\phi_j > 0$. The interface is supposed to be located between the cell centers i and j.

As shown in Fig. 35, the interface is locally approximated by a flat surface (a straight line in 2D) and the closest point on the interface from the cell center i (point I in Fig. 35) is denoted as A and the closest point from the cell center j (J in Fig. 35) is denoted as B. Also the intersection of the interface and the line connecting the cell centers i and j is denoted as C. Since the triangle IAC is similar to the triangle JBC and the length $\overline{IA} = |\phi_i|$ and $\overline{JB} = |\phi_j|$. The Z_C, z-coordinate of the point C, is given by

$$Z_C = \frac{|\phi_i|Z_J + |\phi_j|Z_I}{|\phi_i| + |\phi_j|}.$$

Thus pressure is extrapolated as

$$p_i = \frac{(Z_C/Fn^2)(|\phi_i| + |\phi_j|) - p_j|\phi_i|}{|\phi_j|},$$

in such a way that

$$p_C = \frac{Z_C}{Fn^2}.$$

Note that this procedure uses only the value of ϕ and the actual free surface shape does not need to be constructed.

In case that an air cell has several adjacent water cells, the pressure value is obtained by taking the average of the extrapolated values from each water cell. In the remaining air region, pressure is extrapolated using (41).

Fig. 36 Computational grid around KRISO Container Ship

Grid near bow   Grid near stern
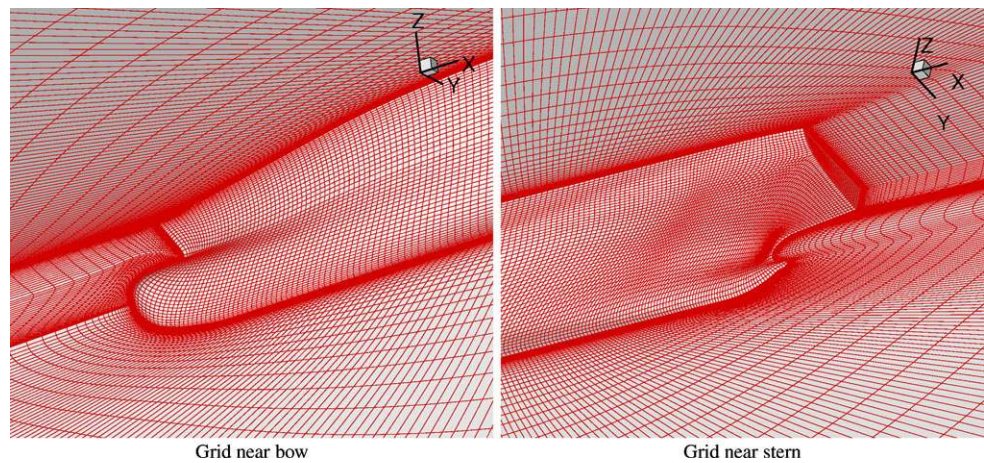
4.3 Test Cases

4.3.1 Flows Around a Ship in Straight Ahead Condition

Ship Model and Computational Grid The approach described above is applied to the free surface flow simulations of a practical ship hull form [35]. A ship model used in this exercise is KRISO Container Ship (KCS) which was conceived at then KRISO (Korea Research Institute for Ships and Ocean Engineering) to provide data for both explication of flow physics and CFD validation for a modern container ship with bulbous bow and transom stern [100].

The computational grid generated is based on the structured grid of an O–O type. Number of grid points are $257 \times 49 \times 81$ covering the half domain in the port side. The solution domain is set

$$-1.5 \le x \le 3.5, \qquad -2 \le y \le 0, \qquad -1.98 \le z \le 0.018,$$

and the averaged minimum spacing near the solid wall is 1.2×10^{-6}. The partial views of the grid are shown in Fig. 36.

Free Surface Flow Computations A free surface flow around a ship hull is computed. The Froude number is set $Fn = 0.26$ and the Reynolds number is 1.4×10^7. The turbulence model used is the modified Spalart-Allmaras model.

The computation in the even-keel condition is carried out first. Figure 37 shows the comparison of the computed wave contour with the data measured at KRISO [100]. Although the computed waves away from the ship hull are dissipated where the grid resolution is not fine enough, the wave systems near the hull are reasonably well captured. Figure 38 is the comparison of wave profiles between measurement at SRI (then Ship Research Institute, Japan) [49] and the present computation. Except for the slight discrepancy in the bow shoulder wave, agreement is fairly well. Figure 39
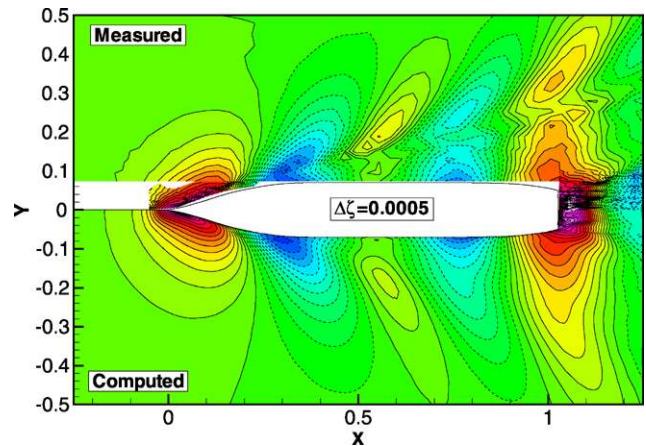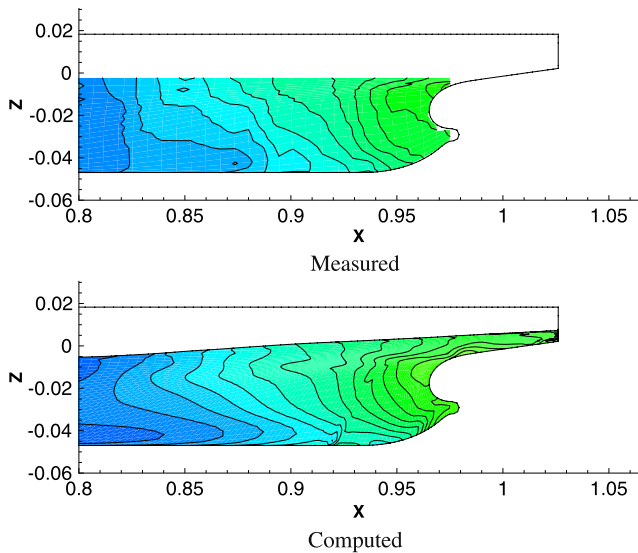

Fig. 37 Comparison of measured (at KRISO, *top*) and computed (*bottom*) wave contours in even-keel condition. Contour interval $\Delta\zeta = 0.0005$
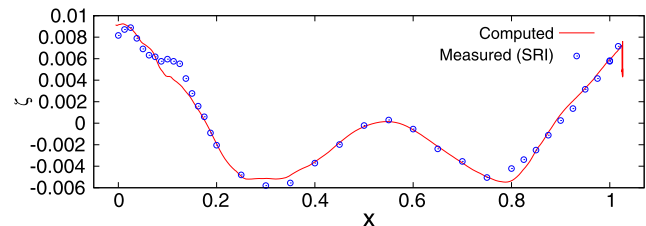
Fig. 38 Comparison of measured (at SRI, *marks*) and computed (*line*) wave profiles on a ship hull surface

shows the comparison of measured (also at SRI) and computed hull surface pressure distributions. Pressure patterns in general are similar between two results.

Next, trim and sinkage of a ship are estimated from the computed flow field of even-keel condition. The dipping at FP and AP are compared with measured (at KRISO) values in Table 2. They are well predicted although the dipping at AP is slightly underestimated. Note that the amount of dipping at AP is quite small (3.2 mm for a model of 7.2786 m).

Fig. 39 Comparison of measured (at SRI, *top*) and computed (*bottom*) surface pressure distributions. Contour interval $\Delta C_P = 0.025$

Table 2 Comparison of measured (at KRISO) and computed dipping at FP and AP for KCS at $Fn = 0.26$

	Dipping at FP	Dipping at AP
Measured	0.339×10^{-2}	0.0443×10^{-2}
CFD	0.337×10^{-2}	0.0262×10^{-2}

Table 3 Measured (at KRISO) and computed resistance coefficients of KCS at $Fn = 0.26$ (coefficients are $\times 10^3$ and normalized by $0.5\rho U^2 S$)

	Cd	$1 + K$	Cw
Measured (Fixed)	3.56	1.10	0.447
Measured (Free)	3.66	1.10	0.547
Computed (Fixed)	3.49	1.09	0.405
Computed (Free)	3.62	1.09	0.535

Further iterations between the attitude estimation and flow field computation will improve the accuracy.

With the estimated trim and sinkage, the computational grid is modified and the flow field is computed using a new grid with trim and sinkage.

The computed and measured [100] resistance coefficients are listed in Table 3. Also listed are the wave resistance coefficient Cw estimated using the corresponding form factors, the computed form factors are estimated by the additional double model flow computation. The trend of the increased resistance in the trim-free condition is captured by the numerical computation.

The wave contours with the even-keel and free conditions are compared in Fig. 40. Although the trim and sinkage are varied, the wave fields appear almost the same. Similarly
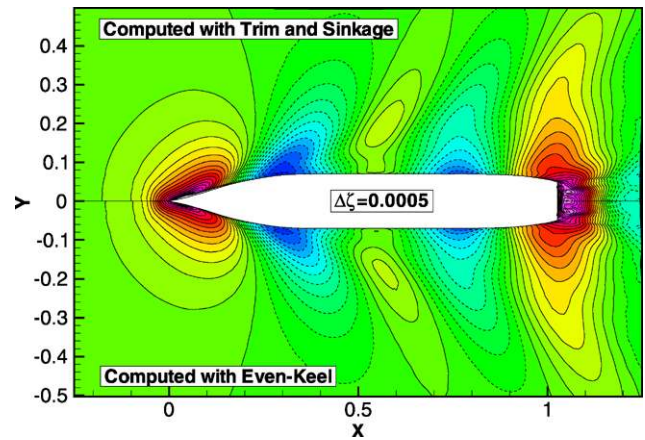
Fig. 40 Comparison of computed wave contours of free condition (*top*) and even-keel condition (*bottom*). Contour interval $\Delta \zeta = 0.0005$
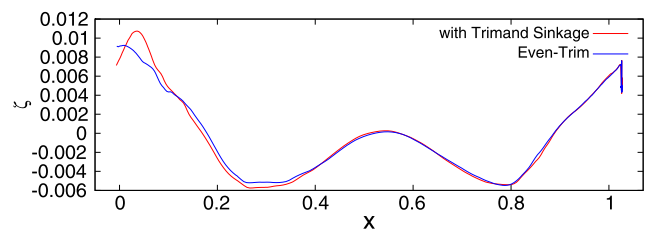
Fig. 41 Comparison of computed wave profiles on a ship hull surface. *Red line*: with trim and sinkage, *blue line*: even-keel

the comparison of the wave profiles in Fig. 41 shows only small difference in the bow. However, this slightly higher bow waves of the trim-free case together with the change of attitude yield the larger wave resistance (see Table 3).

4.3.2 Flows Around a Ship in Steady Drift and Turning Conditions

Ship Model and Computational Grid Free surface flow simulations for steady drift and steady turning conditions are carried out in order to estimate maneuvering properties of a ship.

A ship model is Series 60 ($Cb = 0.6$) which has been used for various CFD validations. Flow conditions are as follows: the Froude number is $Fn = 0.316$ and the Reynolds number is 4.0×10^6. In case of steady drift, the drift angle β is 10 degrees, while in steady turning, non-dimensional angular velocity $r'(= rL/U)$ is 0.4. Note that the validation data is available for the steady drift case [56]. The Spalart-Allmaras model is used as a turbulence model.

A computational grid of O–O type with $129 \times 129 \times 81$ points is generated for both sides of a ship hull. The solution domain is set

$$-1.5 \le x \le 3, \qquad -2.5 \le y \le 2.5, \qquad -2.45 \le z \le 0.05,$$

Fig. 42 Computational grid around Series 60 ($Cb = 0.6$)

Total view

Magnified view
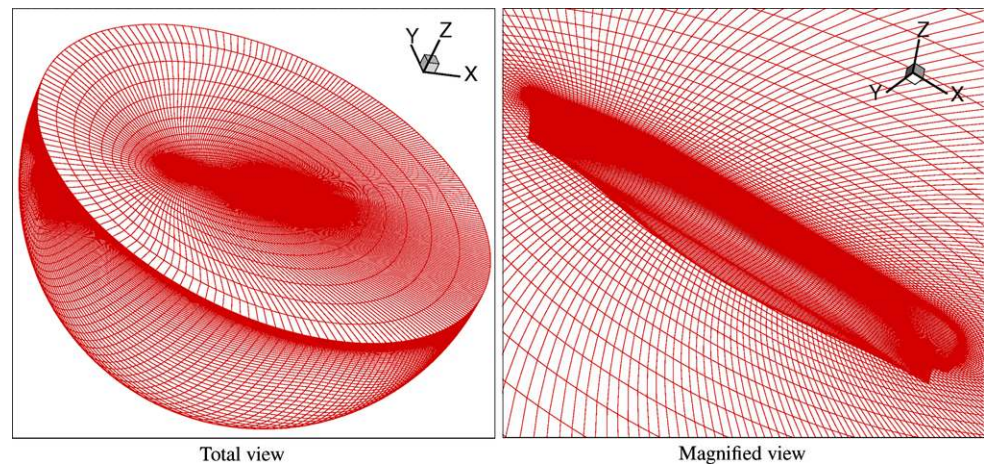
and the averaged minimum spacing near the solid wall is 4.5×10^{-6}. The generated grid is shown in Fig. 42.

Flow Field of Steady Drift Case The computed wave contour in steady drift case is shown in Fig. 43. Asymmetric wave pattern can be observed. The wave profiles along a ship hull are compared between the computation and the measurement [56] in Fig. 44. Although the computed wave peak at the pressure side is slightly under-estimated, both wave profiles agree well with each other.

Figure 45 compares the measured [56] and computed velocity distributions in the cross sections. The starboard shift of low speed region in the downstream direction which corresponds to the vortex movements is reproduced in the computation.

Flow Field of Steady Turning Case The computed wave contour in case of steady turning is shown in Fig. 46. Difference of flow directions between the drift case and the turning case affects the wave pattern significantly. In the turning case, stern waves are larger in the starboard side, whereas the stern waves in the port side are larger in the drift case. The difference can also be observed in the comparison of wave profiles of two cases in Fig. 47.

Figure 48 shows the velocity distribution in various cross sections. Distorted wake pattern is different from the steady drift case in Fig. 45 and the low speed region shifts to the port side. This is again due to the difference of flow directions.

4.3.3 Flows Around a Waterjet Ship

Ship Model and Computational Grid The last example is for a waterjet propelled ship. Since the ship speed is quite high and a waterjet duct and a jet flow behind a hull make a flow field extremely complicated, the flexibility of a level set method is demonstrated in this simulation.

The ship model used is a high-speed boat of 30.5 m long which is equipped with twin waterjet propulsors [36].
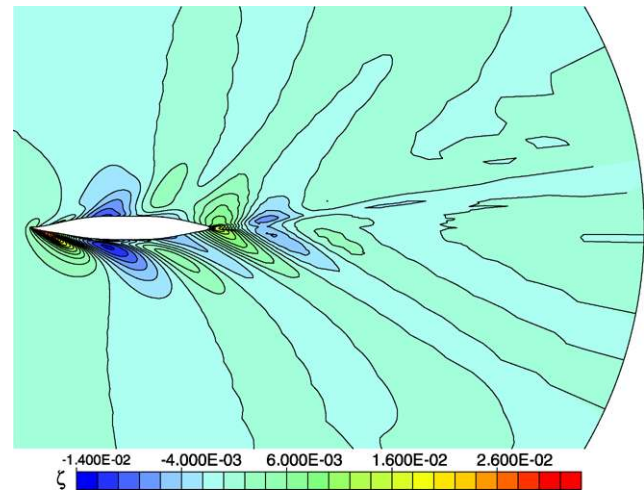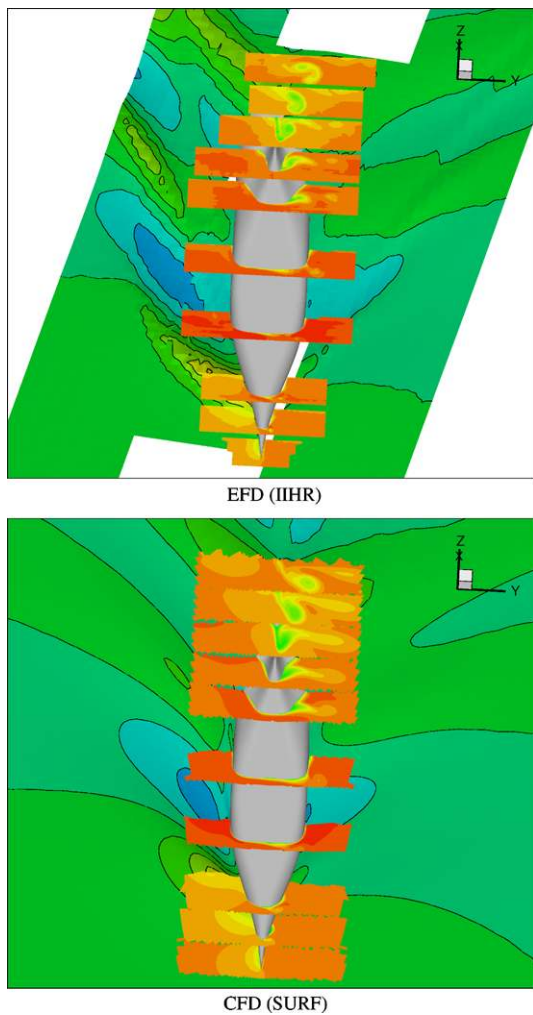

Fig. 43 Computed wave contour of Series 60 ($Cb = 0.60$) in steady drift, $\beta = 10$ deg
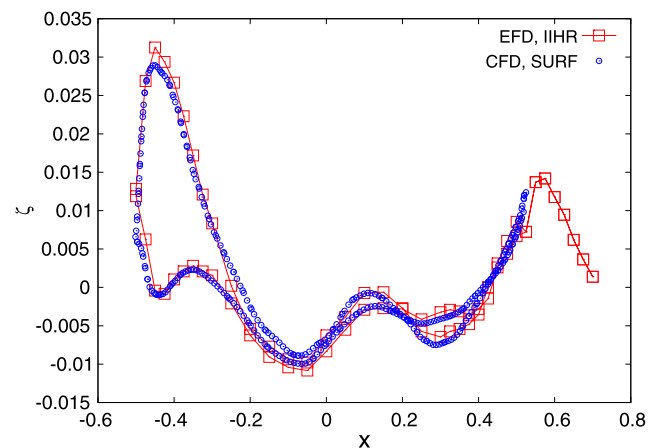
Fig. 44 Comparison of computed and measured wave profiles of Series 60 ($Cb = 0.60$) in steady drift, $\beta = 10$ deg

The Froude number and the Reynolds number of computations are about 1.0 and 10^6, respectively. Although the flow solver has a capability for estimating trim and sink-
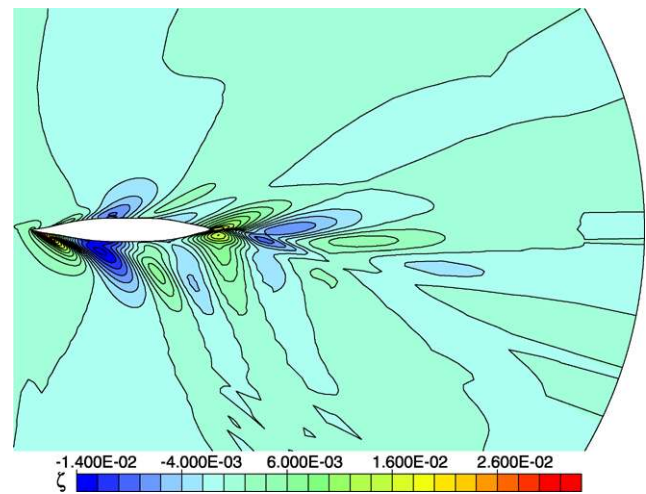
Fig. 45 Comparison of computed and measured velocity distributions of Series 60 ($Cb = 0.60$) in steady drift, $\beta = 10$ deg

Fig. 46 Computed wave contour of Series 60 ($Cb = 0.60$) in steady turning, $r' = 0.4$

Fig. 47 Comparison of computed wave profiles of Series 60 ($Cb = 0.60$) between steady drift and steady turning conditions

age of a ship in a running condition, the attitude of a ship measured in the model test is given and fixed in the present computations.
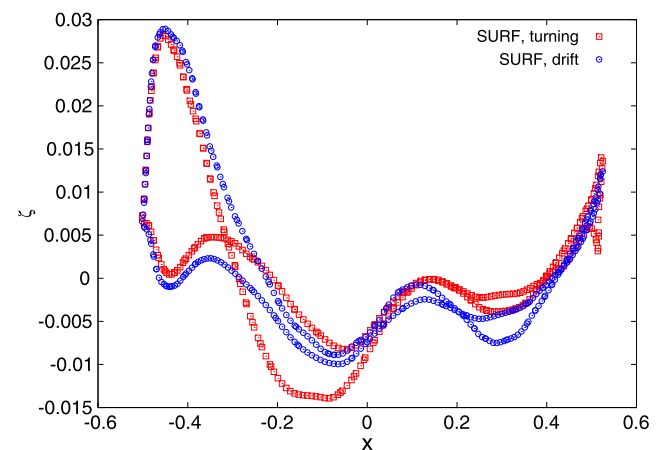
Two grids are prepared: one for the bare hull without a waterjet duct and the other is a hull with a duct. Both grids are based on multiblock structured grids covering only the port side of a domain assuming y-symmetry.

The grid in the case with a duct is generated first and the bare hull grid is obtained by removing a duct block. Thus, the grid blocks around a main hull are identical for both grids. The blocks around a ship consist of $97 \times 81 \times 65$ points and its topology is an O–O type. In the case with a duct, an additional block with $17 \times 17 \times 65$ points is placed inside a duct. Note that a waterjet impeller/stator, shaft and nozzle geometries are not included in the grid for simplicity and the waterjet impeller is modeled by an actuator disk model.

Since the ship speed is quite high and the wave length of generated waves becomes long, the computational domain

is set larger than that of the conventional mid-speed applications. Thus, the domain size is

$$-4 \le x \le 5, \qquad -4.5 \le y \le 0,$$
$$-4.42 \le z \le 0.115,$$

where the ship is placed between $0 \le x \le 1$.

The turbulence model used is the Spalart-Allmaras model and flow fields are assumed to be fully turbulent.

Two computations are carried out. The first one is for a bare hull in a towing condition, the second is for a self-propulsion condition with a hull with a duct configuration.

Flow fields The computed wave patterns are shown in Fig. 49. The Froude number is so high (around 1.0) that the ship is in a planing condition. The so-called "rooster tail" waves are generated behind a transom stern of a ship. The
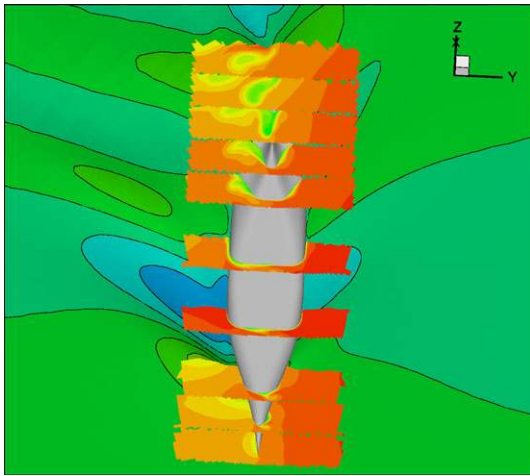
Fig. 48 Computed velocity distribution of Series 60 (*Cb* = 0.60) in steady turning, $r' = 0.4$

Fig. 49 Computed wave patterns: Towed (*top*) and Propelled (*bottom*)

waterjet can be observed in the propelled condition where the actuator disk accelerates the flow inside the duct.

Figure 50 depicts the comparisons of the computed wave contours between towed and propelled conditions. The wave pattern of towed condition is larger than that of propelled condition.

Figure 51 shows the pressure distributions in the duct center plane for the propelled condition. High pressure in the lip of the duct inlet and the low pressure in the fore edge of the inlet can be observed. Also seen is the pressure jump in the impeller position due to the body force. The shape of jet flows is captured by the present level set method quite well.

4.4 Summary and Perspectives

In this section, one of the interface capturing approaches, the level set method adopted in the Navier-Stokes solver *SURF* has been presented.

The key issues of numerical implementation, such as discretization of the convection equation, the procedure for re-initialization and the one-phase flow approach have been described in the framework of an unstructured grid solver.

Various test cases associated with ship hydrodynamics which include ship flows with straight ahead with and without sinkage and trim, flows around a ship in a drift motion or in a turning motion, flow around a ship with a waterjet propulsor, have been shown to demonstrate practical applicability of the present method.

Although the simulations of steady-state flows have been main fields of applications so far, the method can be applied to unsteady free surface flows as well. The flexibility of the present approach will boost the unsteady applications such as seakeeping and dynamic manoeuvring simulations further in the near future.
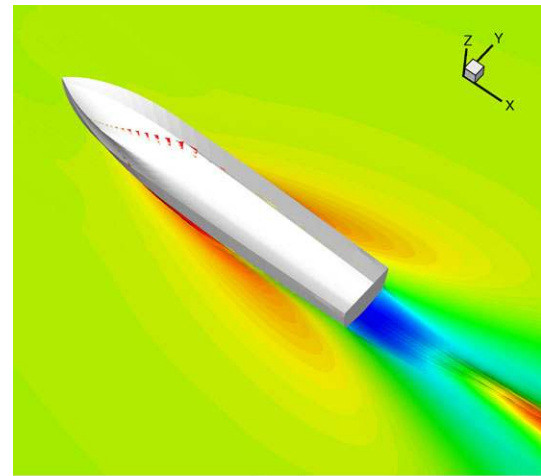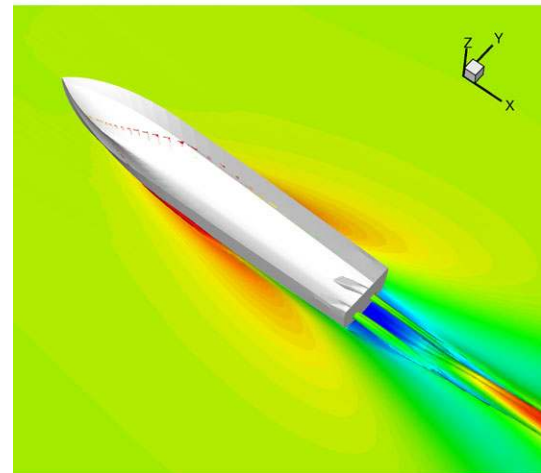
5 Conclusions

In the preceding sections, we have seen that the three methods presented each have their own strong points, which makes them preferable in specific situations. Someone wishing to choose a water surface discretizations method must therefore answer the question, in what situation the method will be used and what will be its requirements, before making a choice. Below, we discuss the factors that may influence such a choice.

Wave Making Prediction In the examples given in the previous sections, we have seen that all three methods presented here are able to accurately predict wave patterns of ships. Thus, the choice of the method must be based on other considerations as well. For those cases where it is applicable, the steady surface fitting method produces a high level of convergence and numerical accuracy quickly and easily. This may be essential for the application envisaged. On the other hand, the capturing methods are more general: they can be
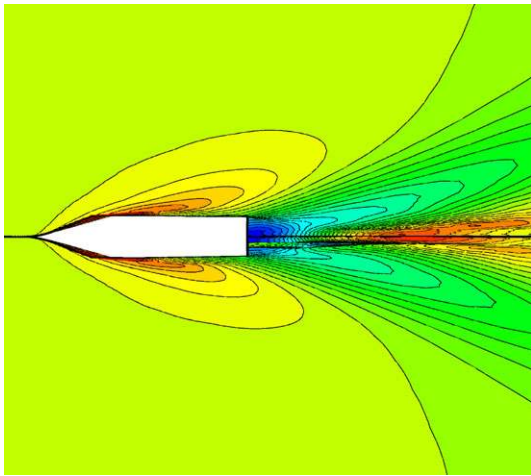
Fig. 50 Comparisons of computed wave contours: Towed (*top*) and Propelled (*bottom*)
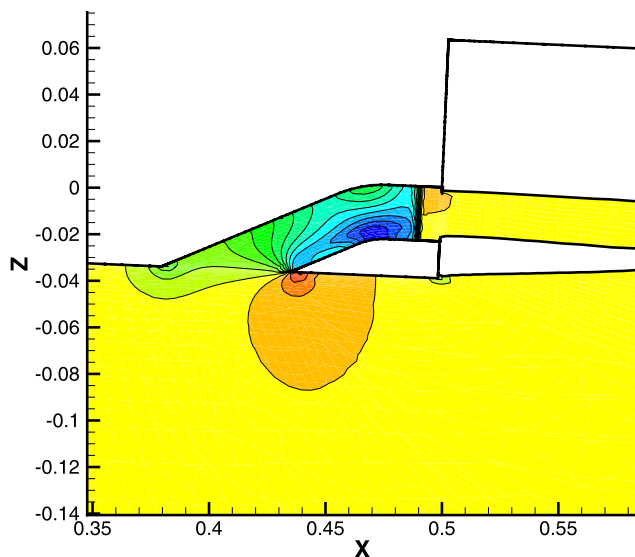
Fig. 51 Computed pressure distributions in the duct center plane

used for more complex problems and the same method is applicable to a wider variety of problems. The influence of several other factors is discussed below.

Complexity of Ship Geometry Some ship geometries are more complex than others. A first example of a complication can be the existence of a bulbous bow or a partially immersed transom stern. Also, appendages can add complications.

For surface fitting, the grid must continue to match the hull surface precisely during the mesh deformation, which in practice may impose limitations on the complexity of the ship geometry near the water surface. An example is the flow over a bulbous bow for which a thin sheet of water passes over the bulb; automatically generating a grid for all flow regimes met in the iterative solution can then be hard. De-

pending on the mesh deformation technique, however, high complexity in regions below the surface is not necessarily a problem.

Capturing techniques can in principle handle arbitrarily complex geometries. An important point for these techniques is to ensure that the water surface can move over the hull surface, despite the no slip condition (see Sect. 4.2.2). Allowing the surface to move, becomes more difficult for complicated surface shapes. However, a good extrapolation approach for the volume fraction or the level set in the boundary layer should take care of this problem.

Solution Speed For those cases where the surface fitting method is applicable, it is very fast. As there is no need to compute and resolve the flow in the air region, and there are no particular requirements for a fine vertical grid resolution in the free-surface region, the method is very economic in the grid points needed. The fitting method used in PARNAS-SOS, owing to a fast convergence of the solution to steady state that is not inhibited by transient wave effects, is remarkably fast even compared to alternative fitting methods.

Compared to the fitting method, both capturing techniques need to resolve an extra flow equation, and require a fine grid around the location of the water surface. The single-fluid level set method has the advantage over the volume-of-fluid method of not needing computation in the air region; however, it requires extra computation for the re-initialization of the level set function. The real computation speed for all methods depends much on the numerical implementation chosen, e.g. the performance of linear solvers.

A case in which solution speed is absolutely essential is the automatic optimization of hull forms, as this requires not one but many different simulations. Surface fitting is a logical choice for this case; capturing methods can probably be used by computing the free-surface shape once and then keeping it in the same place for all subsequent simulations.

Breaking Waves The modeling of breaking waves is signaled as an important point in all three presentations. While none of the methods contains a physically correct model for wave breaking, the volume-of-fluid method comes closest as it approximates the formation of foam and bubbly regions by zones where the volume fraction is dispersed. Numerical dissipation in these zones, while not related to the actual physical dissipation in breaking waves, takes on the same role of dissipating energy, as well as keeping the solution stable. Thus, the method is robust and reasonably accurate even in the presence of strong plunging breakers.

While the level set method always keeps a sharp interface, it can model plunging breakers and flow topology changes. Thus, it is in principle suitable for use in the case of strong breakers.

In surface fitting, the grid must be deformed to match the free surface. Therefore, topology changes, such as in

overturning waves, cannot be well accommodated and the simulation of steep waves might be difficult. However, as Sect. 2.4.3 shows, the method can be made robust with respect to spilling breakers. The need to impose free-surface boundary conditions may be a benefit in certain cases, e.g. as it may allow to add terms in those conditions modelling the effect of spilling breakers. However, the method is most suitable for flows with mild breaking only.

Steady or Unsteady Flows   Of the three methods presented here, ISIS-CFD and SURF are used regularly for unsteady flow simulation. In fact, there is no fundamental difficulty in using either the level set or the volume-of-fluid model for unsteady flows, as these methods, even for steady cases, are usually solved with (pseudo-) time marching. The fitting technique of PARNASSOS is only applicable to steady flows in the present formulation, it would lose its particular efficiency if it was reformulated for unsteady flow.

Ship Movement   The simulation of ship movement is needed in two cases, for the steady resistance problem with free hull position and for the unsteady manoeuvring and seakeeping problems.

The resistance problem with the hull free in trim and sinkage is considered very important for RANS methods and is treated in all three contributions. Only ISIS-CFD, with its 6-DOF motion capacity, is able to simulate the dynamic ship movement to its equilibrium position. Nevertheless, the iterative solving—remeshing techniques of the two other codes constitute a straightforward and attractive alternative. The choice between these two ship motion techniques is in principle independent of the water surface model, except that one must choose an unsteady surface model to perform resolved unsteady motion.

While manoeuvring is in principle an unsteady process requiring ship motion, results for elementary manoeuvres (drift, turning) can be obtained with a steady computation, as demonstrated for SURF in Sect. 4.3.2. An example of a seakeeping simulation is found in Sect. 3.7.3.

Summing up this final discussion, it can be seen that for the whole spectrum of ship flow problems as presented in the Introduction, at least one RANS method is well suited. Thus, due to the greater accuracy and physical realism that these methods offer over older techniques, the coupled simulation of wavemaking and viscous effects deserves its current use for practical ship design, and it is highly probable that this use will strongly increase in the coming years.

References

1. Abgrall R, Karni S (2001) Computations of compressible multi-fluids. J Comput Phys 169:594–623
2. Alessandrini B, Delhommeau G (1994) Simulation of three-dimensional unsteady viscous free-surface flow around a ship model. Int J Numer Methods Fluids 19:321–342
3. Alessandrini B, Delhommeau G (1996) A multigrid velocity-pressure-free surface elevation fully coupled solver for calculation of turbulent incompressible flow around a hull. In: Proceedings of the 21st symposium on naval hydrodynamics, Trondheim
4. Ashgriz N, Poo JY (1991) FLAIR: Flux line-segment model for advection and interface reconstruction. J Comput Phys 93:449–468
5. Burg COE (2002) A robust unstructured grid movement strategy using three-dimensional torsional springs
6. Burg COE (2005) Single-phase level set simulations for unstructured incompressible flows. In: AIAA 2005-5350, proceedings of 17th AIAA computational fluid dynamics conference, Toronto, Canada
7. Burg COE, Sreenivas K, Hyams DG, Mitchell B (2002) Unstructured nonlinear free surface simulations for the fully-appended DTMB Model 5415 series hull including rotating propulsors. In: Proceedings of the 24th symposium on naval hydrodynamics, Fukuoka, Japan
8. Carrica PM, Wilson RV, Stern F (2006) Unsteady RANS simulation of the ship forward speed diffraction problem. Comput Fluids 35(6):545–570
9. Carrica PM, Wilson RV, Noack RW, Stern F (2007) Ship motions using single-phase level set with dynamic overset grids. Comput Fluids 36(9):1415–1433
10. Carrica PM, Wilson RV, Stern F (2007) An unsteady single-phase level set method for viscous free surface flows. Int J Numer Methods Fluids 53:229–256
11. Coleman RM (1981) Nonlinear flow about a three-dimensional transom stern. In: Proceedings of the 4th international conference on numerical ship hydrodynamics, Washington, USA
12. Coleman RM, Haussling HJ (1981) Nonlinear waves behind an accelerated transom stern. In: Proceedings of the 3rd international conference on numerical ship hydrodynamics, Paris, France
13. Cura Hochbaum A, Shumann C (1999) Free surface viscous flow around ship models. In: Proceedings of the 7th international symposium on numerical ship hydrodynamics, Nantes, France
14. Daly BJ (1967) Numerical study of two fluid Rayleigh-Taylor instability. Phys Fluids 10(2):297–307
15. Daly BJ (1969) A technique for including surface tension effects in hydrodynamic calculations. J Comput Phys 4:97–117
16. Darwish M, Moukalled F (1994) Normalized variable and space formulation methodology for high-resolution schemes. Numer Heat Transf 26(1):79–96
17. Darwish M, Moukalled F (2003) The x-schemes: a new consistent high-resolution formulation based on the normalized variable methodology. Comput Methods Appl Mech Eng 192:1711–1730
18. Davis SF (1994) Flux difference splittings and limiters for the resolution of contact discontinuities. Appl Math Comput 65:3–18
19. Degand C, Farhat C (2002) A three-dimensional torsional spring analogy method for unstructured dynamic meshes. Comput Struct 80:305–316
20. Deng GB, Queutey P, Visonneau M (2005) Three-dimensional flow computation with Reynolds stress and algebraic stress models. Eng Turbul Model Exp 6(6):389–398
21. Di Mascio A, Muscari R, Broglia R (2000) Computation of the flow past the US Navy Combatant DTMB5415 by a Godunov-type scheme. In: Gothenburg 2000—a workshop on numerical ship hydrodynamics, Gothenburg, Sweden
22. Di Mascio A, Muscari R, Broglia R (2003) Computation of free surface flows around ship hulls by a level-set approach. In: Proceedings of the 8th international symposium on numerical ship hydrodynamics, Busan, Korea

23. Dommermuth DG, Sussman M, Beck RF, O'Shea TT, Wyatt DC, Olson K, MacNeice P (2004) The numerical simulation of ship waves using Cartesian grid methods with adaptive mesh refinement. In: Proceedings of the 25th symposium on naval hydrodynamics, New Foundland, Canada

24. Dubuc L, Cantariti F, Woodgate M, Gribben B, Badcock KJ, Richards BE (2000) A grid deformation technique for unsteady flow computations. Int J Numer Methods Fluids 32:285–311

25. Duvigneau R, Visonneau M (2003) On the role played by turbulence closures in hull shape optimization at model and full scale. J Mar Sci Technol 8:11–25

26. Farhat C, Degand C, Koobus B, Lesoinne M (1998) Torsional springs for two-dimensional dynamic unstructured fluid meshes. Comput Methods Appl Mech Eng 163:231–245

27. Farmer J, Martinelli L, Jameson A (1993) A fast multigrid method for solving the nonlinear ship wave problem with a free surface. In: Proceedings of the 6th international conference on numerical ship hydrodynamics, Iowa, USA

28. Ferziger JH, Perić M (1996) Computational methods for fluid dynamics. Springer, Berlin

29. Gaskell PH, Lau AKC (1988) Curvature compensated convective transport: SMART, a new boundedness preserving transport algorithm. Int J Numer Methods Fluids 8:617–641

30. Harlow FH, Welch JE (1965) Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. Phys Fluids 8(12):2182–2189

31. Haussling HJ, Miller RW, Coleman RM (1997) Computation of high-speed turbulent flow about a ship model with a transom stern. In: ASME fluids engineering division summer meeting.

32. Hino T (1997) A 3D unstructured grid method for incompressible viscous flows. J Soc Nav Archit Jpn 182:9–15

33. Hino T (1998) Navier-Stokes computations of ship flows on unstructured grids. In: Proceedings of the 22nd symposium on naval hydrodynamics, Washington DC, USA

34. Hino T (1999) An interface capturing method for free surface flow computations on unstructured grids. J Soc Nav Archit Jpn 186:177–183

35. Hino T (2006) CFD-based estimation of propulsive performance in ship design. In: Proceedings of the 26th symposium on naval hydrodynamics, Rome, Italy

36. Hino T, Ohashi K (2009) Numerical simulation of flow around a waterjet propelled ship. In: Proceedings of the 1st international symposium on marine propulsors. Trondheim, Norway

37. Hino T, Jameson A, Martinelli L (1993) A finite-volume method with unstructured grid for free surface flow simulations. In: Proceedings of the 6th international symposium on numerical ship hydrodynamics, Iowa City, USA, pp 173–193

38. Hirata N, Hino T (1999) An efficient algorithm for simulating free-surface turbulent flows around an advancing ship. J Soc Nav Archit Jpn 185:1–8

39. Hirata N, Hino T (2000) A comparative study of zero- and one-equation turbulence models for ship flows. J Kansai Soc Nav Archit Jpn 234:1–8

40. Hirt CW, Nichols BD (1981) Volume of fluid (VOF) method for the dynamics of free boundaries. J Comput Phys 39:201–225

41. Hirt CW, Shannon JP (1968) Free-surface stress conditions for incompressible flow calculations. J Comput Phys 2:403–411

42. Hoekstra M (1999) Numerical simulation of ship stern flows with a space-marching Navier-Stokes method. PhD thesis, Delft University of Technology

43. Hoekstra M, Eça L (1998) PARNASSOS: An efficient method for ship stern flow calculation. In: Proceedings of the 3rd Osaka colloquium on advanced CFD applications to ship flow and hull form design, Osaka

44. Hyams DG, Sreenivas K, Sheng C, Nichols S, Taylor LK, Briley WR, Marcum DL, Whitfield DL (2002) An unstructured multi-element solution algorithm for complex geometry hydrodynamic simulations. In: Proceedings of the 24th symposium on naval hydrodynamics, Fukuoka, Japan

45. Jasak H (1996) Error analysis and estimation for the finite volume method with applications to fluid flows. PhD thesis, University of London

46. Jasak H, Weller HG (1995) Interface tracking capabilities of the inter-gamma differencing scheme. Internal report, Mechanical Engineering Department, Imperial College of Science, London, February 1995

47. Jasak H, Weller HG, Gosman AD (1999) High resolution NVD differencing scheme for arbitrarily unstructured meshes. Int J Numer Methods Fluids 31:431–449

48. Karypis G, Kumar V (1995) A fast and high quality multilevel scheme for partitioning irregular graphs. Technical report 95-035, University of Minnesota, Department of Computer Science, Minneapolis, MN 55455, 1995. Last updated on March 27 1998

49. Kume K, Ukon Y, Takeshi H (2000) Measurements of surface pressure and local velocity field around a KCS model and uncertainty analysis. In: Gothenburg 2000—a workshop on numerical ship hydrodynamics, Gothenburg, Sweden

50. Lafaurie B, Nardone C, Scardovelli R, Zaleski S, Zanetti G (1994) Modelling merging and fragmentation in multiphase flows with SURFER. J Comput Phys 113:134–147

51. Leonard BP (1988) Simple high-accuracy resolution program for convective modelling of discontinuities. Int J Numer Methods Fluids 8:1291–1318

52. Leonard BP (1991) The ULTIMATE conservative difference scheme applied to unsteady one-dimensional advection. Comput Methods Appl Mech Eng 88:17–74

53. Leroyer A, Visonneau M (2005) Numerical methods for RANSE simulations of a self-propelled fish-like body. J Fluids Struct 20(3):975–991

54. Lewis MR (2004) Numerical methods for water flows with free-surface gravity waves. PhD thesis, Delft University of Technology, June 2004

55. Löhner R, Yang C, Oñate E (1998) Viscous free surface hydrodynamics using unstructured grids. In: Proceedings of the 22nd symposium on naval hydrodynamics, Washington DC, USA

56. Longo J, Stern F (2002) Effects of drift angle on model ship flow. Exp Fluids 32:558–569

57. Lopez J, Hernandez J, Gourez P, Faura F (2004) A volume of fluid method based on multidimensional advection and spline interface reconstruction. J Comput Phys 195:718–742

58. Lötstedt P (1982) A front tracking method applied to Burgers' equation and two-phase porous flow. J Comput Phys 47:211–228

59. Maki KJ (2006) Transom stern hydrodynamics. PhD thesis, University of Michigan

60. MARIN (2009) CRS forward speed benchmark, comparison report. Technical report 21447-5-RD, The Netherlands

61. Menter FR (1993) Zonal two-equations $k-\omega$ turbulence models for aerodynamic flows. AIAA paper, 93-290

62. MeTiS. Family of Multilevel partitionning algorithms. http://www-users.cs.umn.edu/karypis/metis/

63. MPI. The message passing interface (MPI) standard. http://www-unix.mcs.anl.gov/mpi

64. Muzaferija S, Peric M (1998) Computation of free surface flows using interface-Tracking and interface-capturing methods. Computational mechanics publications. WIT Press, Southampton, nonlinear water wave interaction edition

65. Nichols BD, Hirt CW (1973) Calculating three-dimensional free surface flows in the vicinity of submerged and exposed structures. J Comput Phys 12:234–246

66. Noh WF, Woodward P (1976) SLIC (simple line interface calculations). Lecture notes in physics, vol 59, pp 330–340

67. Osher S, Fedkiw RP (2001) Level set methods: an overview and some recent results. J Comput Phys 169:463–502

68. Osher S, Sethian JA (1988) Fronts propagating with curvature dependent speed: algorithms based on Hamilton-Jacobi formulations. J Comput Phys 79:12–49

69. Paterson EG, Wilson RV, Stern F (2003) General-purpose parallel unsteady RANS ship hydrodynamics code: CFDShip-Iowa. Technical report 432, Iowa Institute of Hydraulic Research, The University of Iowa

70. Peng D, Merriman B, Osher S, Zhao H, Kang M (1999) A PDE-based fast localized level set method. J Comput Phys 155:410–438

71. Pillod JE, Puckett EG (2004) Second order accurate volume-of-fluid algorithms for tracking material interfaces. J Comput Phys 199:718–742

72. Pržulj V, Basara B (2001) Bounded convection schemes for unstructured grids. In: 15th AIAA computational fluid dynamics conference, AIAA paper 2001–2593, Anaheim, CA, 11–14 June 2001

73. Queutey P, Visonneau M (2007) An interface capturing method for free-surface hydrodynamic flows. Comput Fluids 36(9):1481–1510

74. Ramshaw JD, Trapp JA (1976) A numerical technique for low-speed homogeneous two-phase flow with sharp interfaces. J Comput Phys 21:438–453

75. Raven HC (1996) A solution method for the nonlinear ship wave resistance problem. PhD thesis, MARIN/Delft University of Technology

76. Raven HC (1998) Inviscid calculations of ship wavemaking—capabilities, limitations and prospects. In: Proceedings of the 22nd symposium on naval hydrodynamics, Washington, DC

77. Raven HC, Starke AR (2002) Efficient methods to compute steady ship viscous flow with free surface. In: Proceedings of the 24th symposium on naval hydrodynamics, Fukuoka, Japan

78. Raven HC, van Brummelen H (1999) A new approach to computing steady free-surface viscous flow problems. In: Proceedings of the 1st MARNET-CFD workshop, Barcelona

79. Raven HC, van der Ploeg A, Starke AR (2004) Computation of free-surface viscous flows at model and full scale by a steady iterative approach. In: Proceedings of the 25th symposium on naval hydrodynamics, St. John's, Canada

80. Raven HC, van der Ploeg A, Starke AR, Eça L (2008) Towards a CFD-based prediction of ship performance—progress in predicting full-scale resistance and scale effects. Int J Marit Eng, RINA Trans 150(A4)

81. Rhee SH, Hino T (2002) Numerical simulation of unsteady turbulent flow around maneuvering prolate spheroid. AIAA J 40(10):2017–2026

82. Rhie CM, Chow WL (1983) A numerical study of the turbulent flow past an isolated airfoil with trailing edge separation. AIAA J 17:1525–1532

83. Roe PL (1986) Characteristic-based schemes for the Euler equations. Annu Rev Fluid Mech 18:337–365

84. Sethian JA (1996) Level set methods: evolving interfaces in geometry, fluid mechanics, computer vision and material sciences. Cambridge University Press, Cambridge

85. Sethian JA, Smereka P (2003) Level set methods for fluid interfaces. Annu Rev Fluid Mech 35:341–372

86. Spalart PR, Allmaras SR (1991) A one-equation turbulence model for aerodynamic flows. AIAA paper 92-0439

87. Spalart PR, Allmaras SR (1994) A one-equation turbulence model for aerodynamic flows. Rech Aérosp 1:5–21

88. Starke AR, Raven HC, van der Ploeg A (2007) Computation of transom-stern flows using a steady free-surface fitting RANS method. In: Proceedings of the 9th international conference on numerical ship hydrodynamics, Ann Arbor, Michigan

89. Sundell T (1997) Computations of the free-surface flow around a ship using NS solver FINFLO. Technical report M206, VALB279, VTT Manufacturing Technology, Finland

90. Sussman M, Smereka P, Osher S (1994) A level set approach for computing solutions to incompressible two-phase flow. J Comput Phys 114:146–159

91. Toda Y, Stern F, Longo J (1991) Mean-flow measurements in the boundary layer and wake field of a Series 60 $c_b = .6$ ship model for Froude numbers .16 and .316. Technical report 352, Iowa Institute of Hydraulic Research, August 1991

92. Tzabiras GD (1997) A numerical investigation of 2D steady free surface flows. Int J Numer Methods Fluids 25:567–598

93. Ubbink O (1997) Numerical predictions of two fluid systems with sharp interfaces. PhD thesis, Imperial College of Science, Technology & Medicine. University of London, January 1997

94. van Brummelen EH (2000) Numerical solution of steady free-surface Navier-Stokes flow. Technical report MAS-R0018, CWI, Amsterdam, June 2000

95. van Brummelen EH (2002) Numerical methods for steady viscous free-surface flows. PhD thesis, University of Amsterdam

96. van Brummelen H, Raven HC (2000) Numerical solution of steady free-surface Navier-Stokes flow. In: Proceedings of the 15th int workshop on water waves and floating bodies, Caesarea, Israel

97. van Brummelen EH, Raven HC, Koren B (2001) Efficient numerical solution of steady free-surface Navier-Stokes flow. J Comput Phys 174:120–137

98. van der Ploeg A, Hoekstra M, Eça L (2000) Combining accuracy and efficiency with robustness in ship stern flow computation. In: Proceedings of the 23rd symposium on naval hydrodynamics, Val de Rueil, France

99. van der Ploeg A, Raven HC, Windt JW, Leroyer A, Queutey P, Deng GB, Visonneau M (2008) Computations of free-surface viscous flows at model and full scale—a comparison of two different approaches. In: Proceedings of the 27th symposium on naval hydrodynamics, Seoul, October 2008

100. Van SH, Kim WJ, Yim GT, Kim DH, Lee CJ (1998) Experimental investigation of the flow characteristics around practical hull forms. In: Proceedings of the 3rd Osaka colloquium on advanced CFD applications to ship flow and hull form design, Osaka, Japan, pp 215–227

101. Visonneau M, Queutey P, Deng GB, Leroyer A, Guilmineau E (2008) Computations for a US Navy frigate advancing in head waves in fixed and free conditions. In: Schrefler BA, Perego U (eds) Proceedings of IACM/ECCOMAS 2008, Venice, Italy

102. Vogt M, Larsson L (1999) Level set methods for predicting viscous free surface flows. In: Proceedings of the 7th international symposium on numerical ship hydrodynamics, Nantes, France

103. Wackers J (2007) Surface capturing and multigrid for steady free-surface water flows. PhD thesis, Delft University of Technology, November 2007

104. Wilson RV, Carrica PM, Stern F (2004) Steady and unsteady single-phase level-set method for large amplitude ship motions and maneuvering. In: Proceedings of the 25th symposium on naval hydrodynamics, New Foundland, Canada

105. Yang J, Sakamoto N, Wang Z, Carrica P, Stern F (2007) Two phase level-set/immersed-boundary Cartesian grid method for ship hydrodynamics. In: Proceedings of the 9th international symposium on numerical ship hydrodynamics, Ann Arbor, USA

106. Youngs DL (1982) Time dependent multi-material flow with large fluid distortion. Numerical methods for fluid dynamics. Academic Press, London, pp 273–285