



**HAL**  
open science

## Automatic grid adaptation for unstructured finite volumes

Jeroen Wackers, Michel Visonneau, Zaib Ali

► **To cite this version:**

Jeroen Wackers, Michel Visonneau, Zaib Ali. Automatic grid adaptation for unstructured finite volumes. *International Journal of Engineering Systems Modelling and Simulation*, 2010, 2 (1-2), 10.1504/IJESMS.2010.031866 . hal-01203056

**HAL Id: hal-01203056**

**<https://hal.science/hal-01203056>**

Submitted on 26 Jun 2020

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

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

# Automatic grid adaptation for unstructured finite volumes

J. Wackers\*, M. Visonneau and Z. Ali

Laboratoire de Mécanique des Fluides,  
École Centrale de Nantes,  
CNRS-UMR 6598, 44321 Nantes Cedex 3, France  
E-mail: jeroen.wackers@ec-nantes.fr  
E-mail: michel.visonneau@ec-nantes.fr  
E-mail: zaibali@gmail.com  
\*Corresponding author

**Abstract:** An automatic grid refinement method is presented for the simulation of ship flows. It provides directional refinement of unstructured grids and derefinement of refined grids for unsteady simulation, it is fully parallel and includes automatic dynamic load balancing. Different refinement criteria are implemented. Results are presented that confirm the increased accuracy of solutions obtained on refined grids. Refinement around the water surface proves to be very effective for the simulation of strong breaking waves. A pressure gradient criterion is shown to detect the main features of a ship flow and to be able to generate effective fine grids in their entirety.

**Keywords:** automatic mesh refinement; error indicator; ship flows; finite-volume; parallel implementation.

**Biographical notes:** Jeroen Wackers received his PhD in 2007 from Delft University of Technology, on fast solution methods in numerical hydrodynamics. Subsequently, he joined the Fluid Mechanics Laboratory (LMF) at Ecole Centrale de Nantes (ECN), where he is in charge of the development of automatic grid refinement techniques.

Michel Visonneau received his PhD in Fluid Dynamics and Heat Transfer at the University of Nantes in 1985 and entered the Centre National de la Recherche Scientifique (CNRS) as a Research Scientist. He became the Head of the CFD Department of LMF in 1995. In 2001, he received his Research Habilitation Diploma and was promoted as a Research Director within CNRS in 2006. His main research topics are computational fluid dynamics (CFD), ship hydrodynamics and turbulence modelling for high Re flows. In 1991, he received the 2nd Cray Prize for CFD and he has been named 30th Georg Weinblum Memorial Lecturer (2007–2008).

Zaib Ali obtained his Masters in Computational Mechanics from ECN in 2009, on an Erasmus Mundus scholarship. He worked on automatic grid refinement techniques during his Master's research thesis.

---

## 1 Introduction

Adaptive grid refinement is a technique to increase the accuracy and efficiency of numerical simulations for flows with localised flow phenomena. To obtain acceptable numerical errors near these structures, fine grids are needed there; for efficiency, coarse grids are desired in the rest of the domain. Adaptive grid refinement creates these grids by locally dividing the cells of an initial coarse grid into smaller cells.

The water flow around a ship contains many local features. For a surface capturing discretisation, the water

surface appears as a numerically smeared discontinuity in the volume fraction; to capture the surface sharply, very fine cells are needed around this discontinuity. Also, the orbital velocity field of the ship's waves is local. It appears below the water surface, in the wave train. And lastly, the bilge vortices, that play an essential role in the velocity field encountered by the propeller, consist of local very strong gradients.

Thus, to efficiently obtain highly accurate simulations of ship flow, adaptive grid refinement is an excellent choice. This is even more evident for unsteady ship flow, where the position of the local features changes in time. To be

successful for ship flow, the adaptive refinement technique must work for all these different types of flows.

An automatic mesh adaptation method has been developed for ISIS-CFD, the flow solver created by Equipe Modélisation Numérique (EMN, i.e., CFD Department of the Fluid Mechanics Laboratory). The goal of the development was to produce a method that can be used in daily practice for all the applications of this code, notably complex geometries, steady and unsteady flows, free surface capturing, and grid deformation for the imposed or resolved motion of ship hulls. Furthermore, the method needs to be easily maintainable as the code develops over the years. Therefore, the mesh adaptation method has been made general: it allows unstructured grids, directional refinement to keep the size of 3D refined grids low, and derefinement of refined grids to enable unsteady flow simulation. The method is flexible to allow the easy changing of refinement criteria and, like the flow solver, it is completely parallel. Full integration of the refinement method in the flow solver is being completed.

## 2 The ISIS-CFD flow solver

ISIS-CFD, available as a part of the FINE<sup>TM</sup>/Marine computing suite, is an incompressible unsteady Reynolds-averaged Navier-Stokes (URANS) method. The solver is based on the finite volume method to build the spatial discretisation of the transport equations. The unstructured discretisation is face-based, which means that cells with an arbitrary number of arbitrarily shaped faces are accepted. A detailed description of the solver is given by Duvigneau et al. (2003) and Queutey and Visonneau (2007).

The velocity field is obtained from the momentum conservation equations and the pressure field is extracted from the mass conservation constraint, or continuity equation, transformed into a pressure equation. In the case of turbulent flows, transport equations for the variables in the turbulence model are added to the discretisation. Free-surface flow is simulated with a multiphase flow approach: the water surface is captured with a conservation equation for the volume fraction of water, discretised with specific compressive discretisation schemes, see Queutey and Visonneau (2007).

The method features sophisticated turbulence models: apart from the classical two-equation  $k-\varepsilon$  and  $k-\omega$  models, the anisotropic two-equation explicit algebraic stress model (EASM), as well as Reynolds stress transport models, are available, see Deng and Visonneau (1999) and Duvigneau et al. (2003). The technique included for the six degree of freedom simulation of ship motion is described by Leroyer and Visonneau (2005). Time-integration of Newton's laws for the ship motion is combined with analytical weighted or elastic analogy grid deformation to adapt the fluid mesh to the moving ship. Furthermore, the code has the possibility to model more than two phases. For brevity, these options are not further described here.

## 3 Refinement technique

During a flow calculation with adaptive grid refinement, the refinement procedure is called repeatedly to keep the grid permanently adapted to the developing solution. Globally, the method works as follows: the flow solver is run on an initial grid for a limited number of time steps. Then the refinement procedure is called. If a refinement criterion, based on the current flow solution, indicates that parts of the grid are not fine enough, these cells are refined and the solution is copied to the refined grid. On this new grid, the flow solver is restarted. Then the refinement procedure is called again, to further refine or to derefine the mesh. This cycle is repeated several times. When computing steady flow, the procedure eventually converges: once the flow starts to approach a steady state and the grid is correctly adapted to this state according to the refinement criterion, then the refinement procedure keeps being called, but it no longer changes the grids.

### 3.1 Data structure

The refinement is cell-based. At this moment, it is limited to unstructured hexahedral cells, but the code is written such, that other cell types can be easily included. The grid data structure of ISIS-CFD, that uses node coordinates and pointers between cells, faces, and nodes, is used as the basis for the grid refinement. The number of extra pointers is limited; the main addition is a system of cell family ties. These stores the history of the refinement, so refined cells can be derefined again to recover the original grid. When a cell is refined, all the new small cells get a 'mother' pointer to the old big cell and 'sister' pointers to each other. Thus, the group can be found again later and derefined back into the single large cell. The large cell is saved as a 'dead' cell, that has no faces nor a state vector, only family ties. Thus, it conserves its own sisters and mother (probably 'dead' as well), in case it has to be derefined itself, after being restored.

### 3.2 Refinement decision

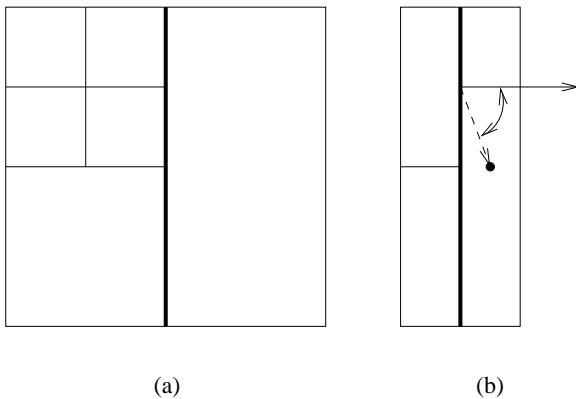
For maximum flexibility, the code is divided in three separate parts: the calculation of the refinement criterion, the refinement decision, and the actual (de)refinement. To permit the user choice of refinement criteria and the easy incorporation of new refinement criteria in the code, the criterion is computed as a field variable (comparable to the velocity or the pressure). It does not depend explicitly on the type or the orientation of the cells.

In the second step, this refinement criterion is transformed into the decision of which cells to refine or to derefine. While this decision may depend on the type of the cells, it does not depend on the specific way in which the refinement criterion is calculated. It remains the same for any refinement criterion.

During the refinement decision step, the decision in each cell is adapted to its neighbour cells. To guarantee the

quality of the mesh, extra cells may need to be refined, or derefinement of cells may be prevented. The most important quality criteria are given in Figure 1. A face of a cell may not be divided twice, which would cause too great differences in the sizes of its neighbour cells. And the angle between face normals and lines cell centre – face centre may not be too great: this reduces the quality of the state reconstruction at the faces, as indicated by Queutey and Visonneau (2007). At the end of the decision step, before a single cell is refined, the refinement of the whole mesh is known; this makes the actual refinement much easier.

**Figure 1** Grid quality criteria. Forbidden are (a) faces that are divided twice (b) too large angles between face normals and lines cell centre – face centre.



Note: The images represent 2D examples.

### 3.3 Refinement

The actual (de)refinement is done cell by cell. Care is taken, after the treatment of each individual cell, to leave a valid mesh with all the pointers between cells, faces, and nodes in place, even if one knows that certain pointers will be changed again when a neighbour cell is refined later on. This guarantees that, when a cell is refined, it does not have to distinguish between neighbour cells that are refined, that will be refined, or that remain unchanged. The added flexibility and robustness of the code are well worth the extra work this represents. Furthermore, in the code, the refinement of cells and faces is completely decoupled. These parts exchange only minimal information; a face does not need to know all the details of the refinement of its neighbour cells. This greatly facilitates unstructured-grid and directional refinement. Finally, the derefinement and refinement procedures are separated. Derefinement is performed first.

### 3.4 Parallel redistribution

For efficient parallel flow computation, automatic redistribution of the cells over the processes is included in the grid adaptation procedure. This redistribution is completely integrated, it is performed between the derefinement and refinement steps; at that moment, the grid is at its smallest. Due to the preceding calculation of the

refinement decision, the final topology of the grid is already known then. Using this information, the grid is first repartitioned in parallel with the ParMeTiS library, see ParMeTiS (2009).

The next step is the actual displacement of cells between the processes. This step is complicated by the strictly local numbering used in each block. To solve this problem, each block is domain-decomposed itself into sub-blocks to be dealt with by specific processes. The data structure for each sub-block is the same as the structure of the blocks and each individual sub-block gets its own local numbering. These sub-blocks are exchanged between the processes by message passing interface (MPI) and then concatenated to produce the new blocks.

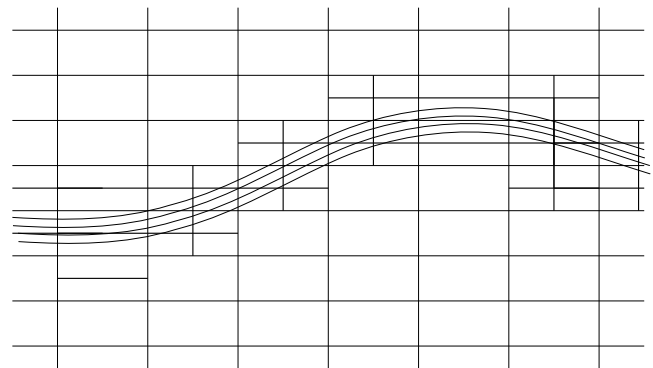
## 4 Refinement criteria

The refinement criterion has to be carefully chosen depending on the flow problem that is simulated. Several criteria, for different ship flow problems, have been developed or are currently under study.

### 4.1 Free-surface criterion

Refinement in the neighbourhood of the water surface, for water-air two-phase flow, has been chosen as a first refinement criterion. Directional refinement is achieved by refining the grid in the direction normal to the water surface only. Where the free surface is diagonal with respect to the grid directions, isotropic refinement is used, but where the surface is horizontal, directional refinement is chosen to keep the total number of cells low. The zone of directional refinement includes the undisturbed water surface, as well as smooth wave crests and troughs. Figure 2 gives an illustration of this refinement principle; an image of a refined ship grid can be found further on in Figure 6.

**Figure 2** Isotropic and directional refinement at the free surface



Note: The curves represent volume fraction isolines.

### 4.2 Gradient criteria

Recently, initial tests have been performed with criteria that are based on the absolute values of the gradients of solution quantities in each cell. These criteria detect the regions

where the flow field changes rapidly; they react to most features of a flow and are thus more general than the free-surface criterion. Initially, these criteria have been applied to single-fluid flows.

Three criteria are chosen, one based on the gradient of the pressure:

$$\text{Crit}_p = \sqrt{\left(\frac{\partial p}{\partial x}\right)^2 + \left(\frac{\partial p}{\partial y}\right)^2 + \left(\frac{\partial p}{\partial z}\right)^2}, \quad (1)$$

one on the gradients of the three velocity components:

$$\text{Crit}_u = \sqrt{\left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y}\right)^2 + \left(\frac{\partial u}{\partial z}\right)^2 + \left(\frac{\partial v}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial y}\right)^2 + \left(\frac{\partial v}{\partial z}\right)^2 + \left(\frac{\partial w}{\partial x}\right)^2 + \left(\frac{\partial w}{\partial y}\right)^2 + \left(\frac{\partial w}{\partial z}\right)^2}, \quad (2)$$

and one on the vorticity:

$$\text{Crit}_\omega = \sqrt{\left(\frac{\partial v}{\partial z} - \frac{\partial w}{\partial y}\right)^2 + \left(\frac{\partial w}{\partial x} - \frac{\partial u}{\partial z}\right)^2 + \left(\frac{\partial u}{\partial y} - \frac{\partial v}{\partial x}\right)^2}. \quad (3)$$

Effectively, the vorticity can also be seen as a norm of the velocity gradients. For each criterion, a cell is refined when the product of the criterion and the cell size exceeds a given threshold.

The main difference between the pressure gradient criterion and the two velocity-gradient based criteria is their effect on boundary layers. There, the pressure varies little, while the velocity gradients are very strong. Thus, the velocity-gradient based criteria refine everywhere in boundary layers, but the pressure gradient criterion only there where the external flow creates a pressure gradient. In those cases where the flow outside the boundary layers is the main interest of the computation, costly and unnecessary refinement in all boundary layer cells can be prevented by using the pressure gradient criterion. An example will be shown in Section 5.4.

### 4.3 Error estimation

Earlier work on grid adaptation and refinement criteria has been performed within EMN by Hay et al. (2004) and Hay and Visonneau (2006). One of the refinement criteria that they studied is an error-indicating criterion based on the second spatial derivatives of flow quantities. Also, they developed a sophisticated estimation of the error production, derived from the residual obtained when a high-order accurate discretisation of the flow equations is applied to the solution. Both of these criteria proved to be highly successful, producing accurate solutions on grids with a small number of cells.

Currently, these techniques are being parallelised and adapted to the type of grids produced by the present grid refinement method.

## 5 Test cases

Four test cases are presented which show that the grid refinement improves free-surface wave and viscous flow computations, and that it is able to handle unsteady flow and interaction with moving bodies.

### 5.1 Virtue Container ship

The first case is the Virtue Container ship, a test case in the European project VIRTUE (VIRtual Tank Utility in Europe) in which EMN participated. The test case concerns a typical modern container ship with a bulbous bow. Model tests for this ship have been performed by the Hamburg ship model basin HSVA. The computations are performed at  $Fr = 0.272$ , both at the model scale  $Re = 1.84 \cdot 10^7$  and at the full ship scale  $Re = 2.89 \cdot 10^9$ . Results on adaptively refined grids are compared with results on unrefined original grids.

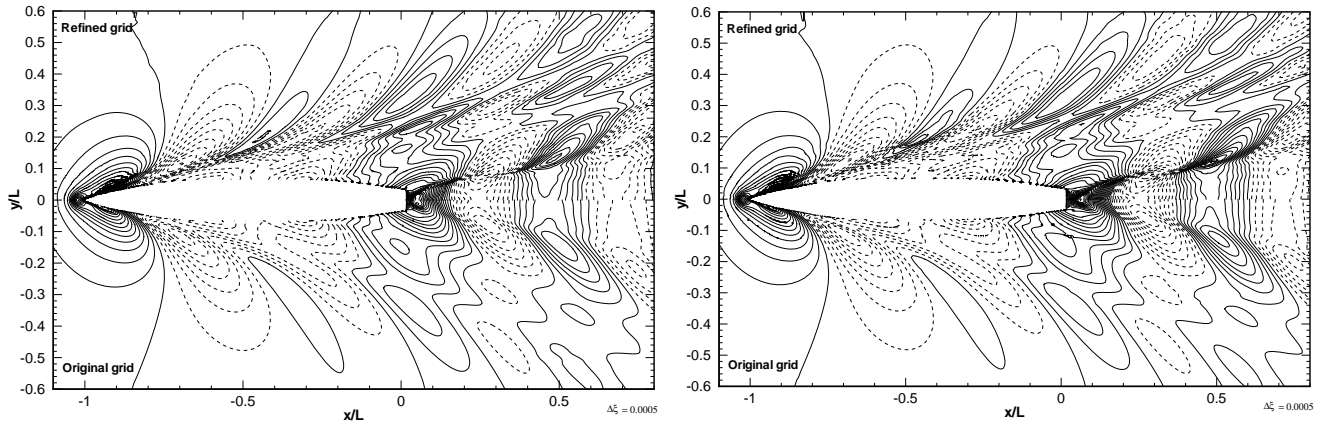
The interest of this test case is the simulation of the water surface and the waves generated by the ship. This particular ship generates a complex wave pattern, consisting of sharp breaking bow and stern waves and many small, secondary waves. Thus, good grid resolution near the water surface is essential for this case.

Therefore, the free-surface refinement criterion is used for the computations. The results in Figure 3 show the great increase in solution accuracy obtained with refined grids, both for the model scale and the full-scale case. The bow wave is higher and more sharply defined; it suffers less from numerical diffusion as it moves away from the ship. The strong breaking stern wave is resolved in much greater detail near the ship hull and, like the bow wave, it is damped out less as it moves out. Between the bow and stern waves, more small flow details can be observed.

For the model scale, a comparison with the HSVA experiments is given in Figure 4. This figure confirms the better resolution of the larger waves and the presence of more small wave details. In most places, the experiments are reproduced well on the refined grid; even in those places where differences remain, the shape of the waves is reproduced better. At the stern, the breaking of the wave system is resolved notably better, which improves the entire wave field behind the ship.

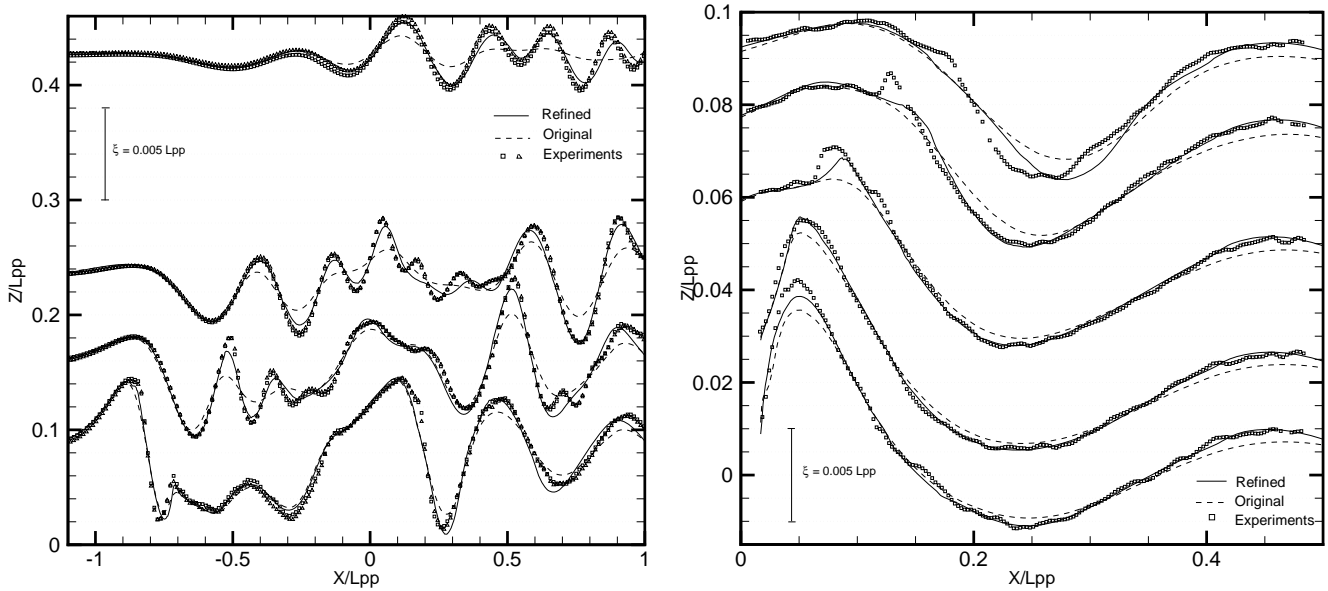
One of the major advantages of CFD for ship flow computations is that full-scale flows can be computed, enabling an evaluation of the difference with the model scale flows and thus a better interpretation of model tests. Due to the greater resolution of the solutions on the refined grids, these scale effects can be observed more precisely there. An example is given in Figure 5 that shows the stern wave of the Virtue Container ship. On the refined grids, a small breaking wave is observed just behind the stern for the model scale computation; this wave disappears at full scale. On the original grids, it is impossible to detect this scale effect.

**Figure 3** Improvement in solution quality on a refined grid, compared with the non-refined original grid, for the Virtue Container ship



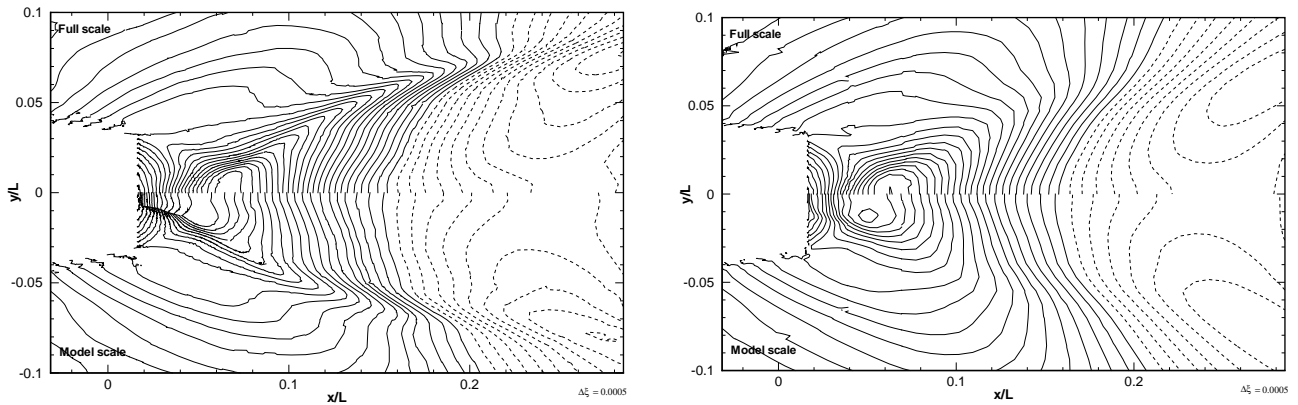
Note: Model scale  $Re = 1.84e7$  (left), full scale  $Re = 2.89e9$  (right).

**Figure 4** Improvement in solution quality for the Virtue Container ship



Notes: Comparison of computations on a refined and a non-refined grid with experiments, model scale.  $y$  cross-sections of the free-surface at the side of the ship (left) and behind the ship (right).

**Figure 5** Scale effects in the breaking waves at the stern, for the Virtue Container ship



Note: The topology difference in the model- and full-scale flows on refined grids (left) cannot be observed on the original grids (right).

The grid refinement requires only a small increase in the number of cells. For the model scale computation, the refined grid has 3.88 M cells and the original grid 3.07 M cells: an increase of 26%. For the full-scale computation with its further developed boundary layers, the refined grid has 4.92 M cells and the original grid 3.87 M cells, which represents an increase of 27%. This good efficiency is mainly due to the use of directional refinement.

## 5.2 Series 60

The second test case is the Series 60 ( $C_b = 0.6$ ) ship at  $Fr = 0.316$  and  $Re = 5.3 \cdot 10^6$ . This is a slender ship that generates a strong but quite regular wave pattern. Measurements for this case have been performed at IHR, University of Iowa, by Longo and Stern (2002). Here, the case is used to test the possibility of accelerating computations by changing the discretisation of the volume fraction equation.

Initially, like for the Virtue Container ship, computations have been performed using the compressible blended reconstructed interface capturing scheme (BRICS) for the volume fraction of water, a further development of the schemes described by Queutey and Visonneau (2007). This scheme keeps the water surface very sharp, but it imposes a limitation on the Courant number that requires the use of small time steps. Then this scheme is replaced by the non-compressive Gamma difference scheme (GDS) of Jasak (1996). While this scheme allows the water-air interface to diffuse more, reducing the accuracy of the solution, it imposes no restriction whatsoever on the time step. Here, we shall see if, using the GDS scheme and more grid refinement, converged steady solutions can be obtained with the same accuracy as the BRICS solutions, in less computation time.

Relatively coarse grids are used, to observe changes in the solution due to the grid refinement as clearly as possible. Computations are started from an original grid with 330 k cells. In all cases, the free-surface refinement criterion is used.

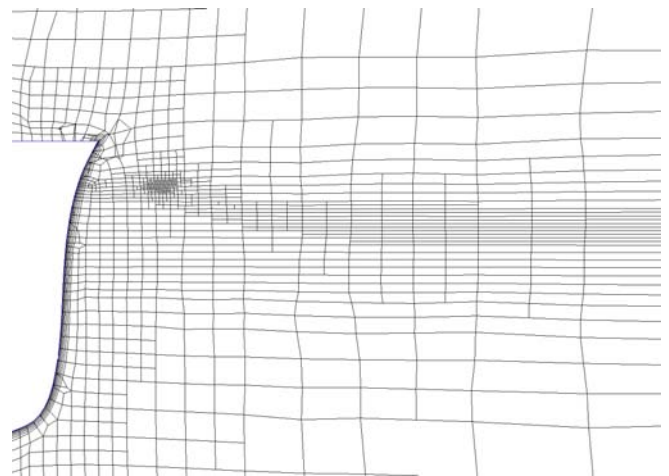
For the BRICS computation, the refinement criterion is set to give a desired cell size normal to the surface of  $d = 0.001 L$ , which is half the cell size of the original grid. Figure 6 gives a cut through this refined grid that shows isotropic refinement in the bow wave region and directional refinement around the undisturbed surface. This grid has 506 k cells. The solution on the refined grid has clearly improved, as shown in Figure 7: the bow and stern wave are sharper, steeper and less damped out far away from the ship. The agreement with measurements is much better.

For the GDS scheme, to compensate for the more diffused interface, the refinement criterion is set to give  $d = 0.0005 L$ , or half the size of the BRICS grid. The resulting grid has 1.15 M cells. As shown in the wave cuts

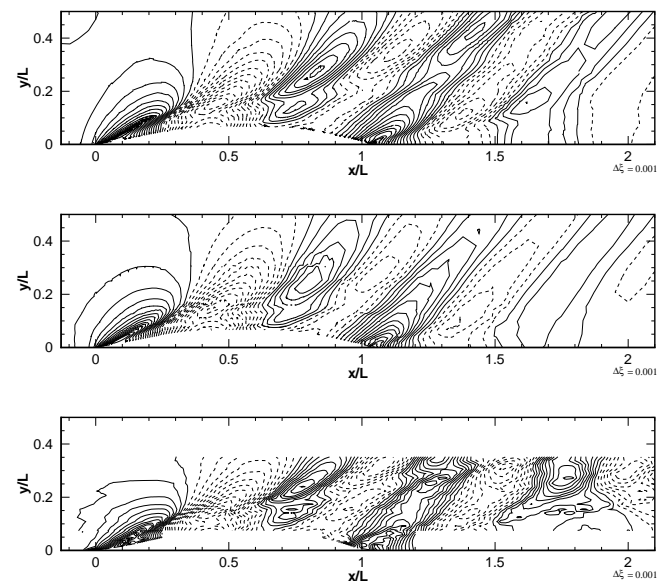
of Figure 8, this solution has about the same accuracy as the BRICS solution: the forms of the waves are reproduced better and the waves are less damped out. The remaining difference with the experiments is mostly due to the coarse initial grid.

The GDS solution, however, is computed with a time step that is ten times larger than the one for the BRICS solution. Thus, even though the refined grid has more cells, the computation time has been reduced by a factor of about four. Therefore, the combination of non-compressive schemes with grid refinement forms an attractive alternative for the usual compressive schemes.

**Figure 6** Cross-section at  $x/L = 0.1$  (i.e., bow-to-stern view) through a refined grid for a Series 60 ship, showing refinement around the bow wave

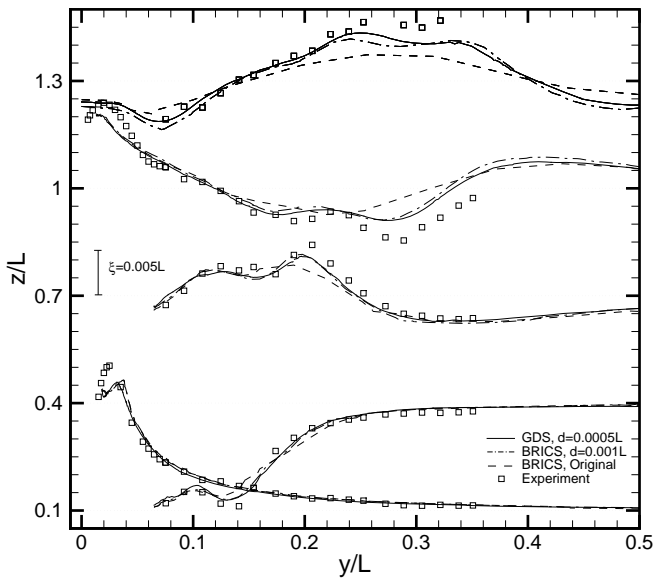


**Figure 7** Wave profiles for the Series 60 ship



Note: Computations using the BRICS scheme on the  $d = 0.001 L$ , refined grid (top) and on the original grid (middle), compared with experiments (bottom).

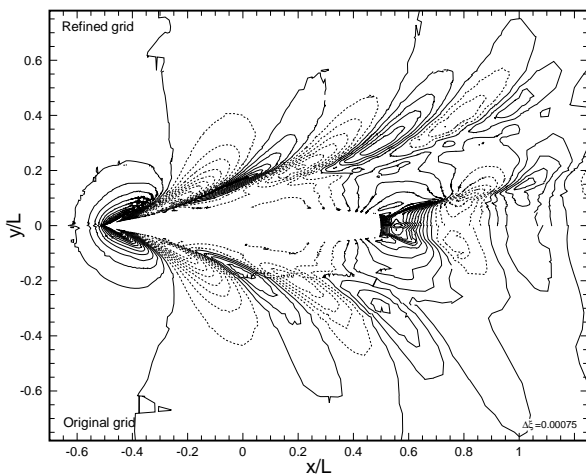
**Figure 8** Series 60,  $x = \text{constant}$  wave cuts for non-compressive (GDS) versus compressive (BRICS) volume fraction discretisations



### 5.3 DTMB 5415

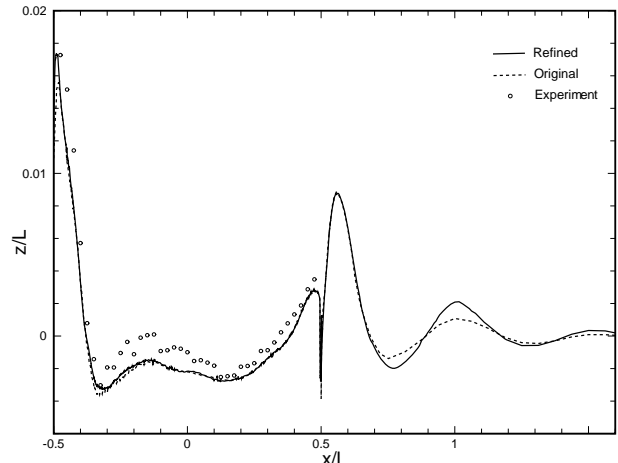
The third test case is the DTMB (David Taylor model basin) 5415 frigate in still water with free trim and sinkage,  $Fr = 0.280$ , the case 1.3 of the CFD Workshop Tokyo 2005, see Hino (2005). The computation is performed to test the combination of the refinement procedure with the mesh deformation capability of ISIS-CFD as described by Leroyer and Visonneau (2005), to resolve the free ship motion. The free-surface refinement criterion is used. Like for the Virtue Container ship, the bow and stern waves are resolved better on the 2.03 M cell refined grid (Figure 9). However, as this wave train is less complex, it is already resolved well on the 570 k cell original grid, especially near the ship hull; therefore, the refinement does not change the solution much on the hull (Figure 10). This is also indicated by the ship's motion (Figure 11), which remains close to the one on the original grid.

**Figure 9** DTMB 5415 with free trim and sinkage

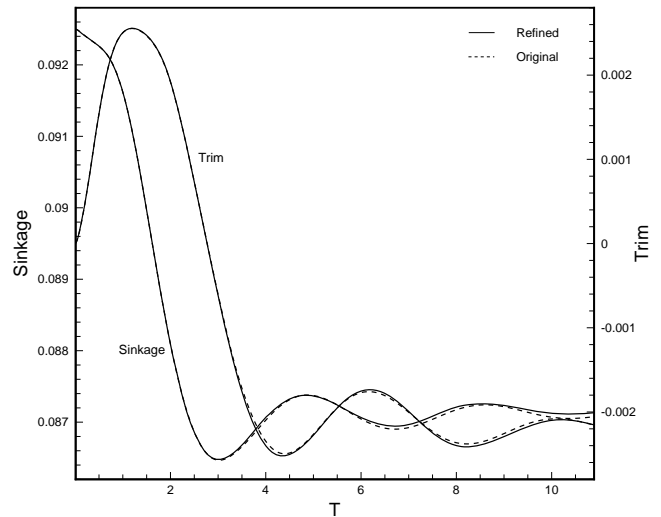


Note: Improvement in the wave pattern on the refined grid versus the original grid.

**Figure 10** DTMB 5415 with free trim and sinkage: wave cut on the hull and the symmetry plane



**Figure 11** Ship motion during computation for the DTMB 5415 with free trim and sinkage



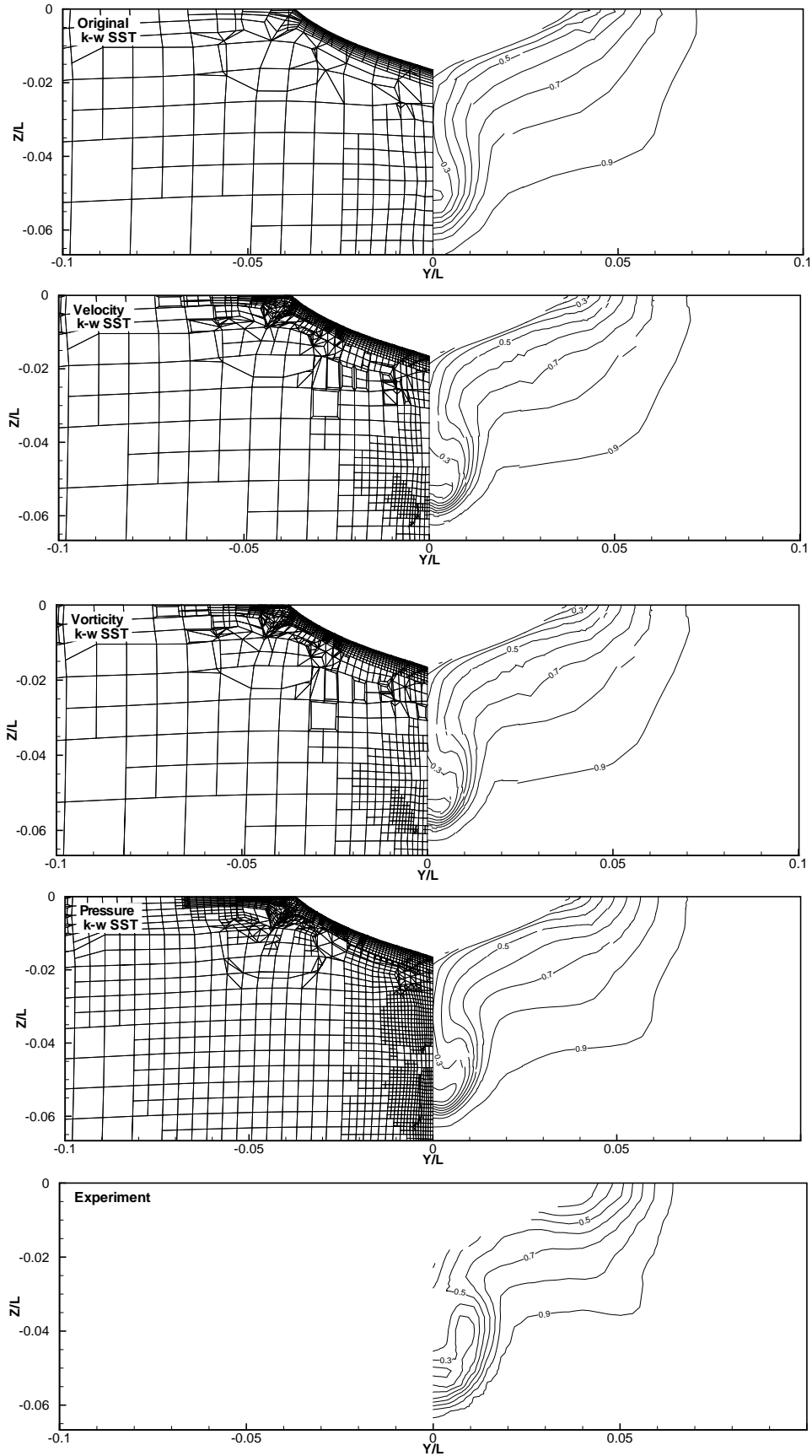
The important point of this test case is that the grid refinement procedure (called every 0.5 s) does not disturb the ship motion. Thus, its combination with the resolved motion capability of ISIS-CFD is successful.

### 5.4 KVLCC2

Finally, the flow around the KRISO very large crude carrier (KVLCC2) tanker is computed. This model scale computation at  $Re = 4.6 \cdot 10^6$  was one of the test cases in the Gothenburg 2000 workshop, see Larsson et al. (2000). It is a double model test, so the water surface motion is not taken into account. The computations are performed with a single fluid (water) and a symmetry boundary condition is imposed at the position of the undisturbed water surface.

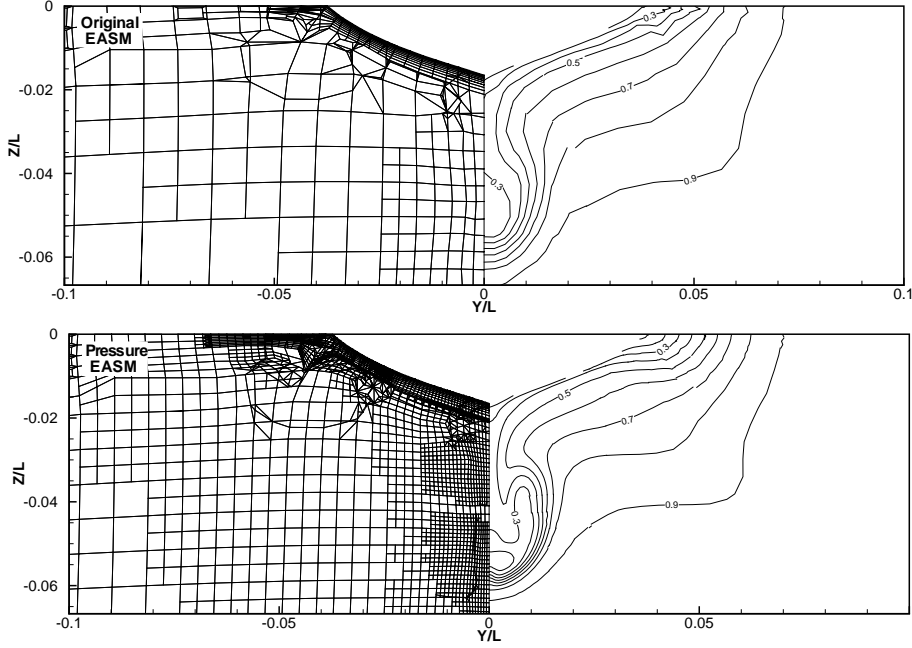


Figure 12 KVLCC2 tanker, cuts in the propeller plane at  $x / L = 0.0175$



Notes: Grid cross-sections and axial velocity  $u / U_\infty$  isolines are shown on the original coarse grid (58 k cells) and the refined grids for the velocity gradient criterion (630 k cells), the vorticity criterion (620 k cells) and the pressure gradient criterion (803 k cells). The isolines are compared with measurements.

**Figure 13** KVLCC2 tanker with the EASM turbulence model



Note: Grid cross-sections and axial velocity  $u/U_\infty$  isolines are shown on the original coarse grid (58 k cells) and the refined grid for the pressure gradient criterion (1.07 M cells).

The main interest of this case is the computation of the flow in the propeller plane at the rear of the ship. The flow in this plane is strongly influenced by the vorticity that is generated on the ship hull and shed into the fluid at the rear of the ship, where the hull tapers into the stern. This vorticity generates a non-uniform flow field. Good prediction of this field is essential for ship design, on one hand, because it must be known as an input for the propeller design, on the other hand, because a more uniform propeller plane flow improves the performance of the propeller and is thus an important target for the optimisation of the ship hull shape.

To accurately compute this flow field, locally refined grids can be generated by hand, using fine cells at the rear of the ship. In practice, it is known where these fine cells must be placed, but generating the grids is a specialist job that requires experience. It is our goal for this test case to start the computation from a very coarse mesh, to see if an effective fine mesh can be created entirely by grid refinement. For the tests, the gradient criteria are used.

Initially, the flow has been computed using the three different criteria. As turbulence model, the  $k-\omega$  SST model of Menter (1993) is used. For each criterion, the computation is started from the converged solution on the original 58 k cells grid. To give a good comparison, the three criteria are then tuned such that the first refinement step, based on this converged solution, gives an equal number of refined cells for each criterion. After these first steps, the computations and refinement steps are continued until the solution and the grid converge. To prevent an excessive number of cells due to the isotropic criteria, each cell is allowed to be refined at most two times. The resulting grid sizes are 630 k cells for the velocity gradient criterion,

620 k cells for the vorticity criterion and 803 k cells for the pressure gradient criterion.

Results in the propeller plane are shown in Figure 12. The ship hull can be seen at the top of these figures, the propeller would be placed below the hull. As seen in the experimental result, the most striking feature of the flow field is the hook-shaped region of low axial velocity near the ship's symmetry plane. On the original grid, this feature cannot be observed at all.

The velocity gradient and vorticity criteria concentrate the refinement in the boundary layer, as already mentioned in Section 4.2. Some refinement can be observed in the high-gradient regions below the hull, but this is not enough to create a solution that is significantly better than the one on the original grid. It is interesting to see that these two criteria, while computed in a different way from the velocity gradients, give nearly identical results.

On the other hand, the pressure criterion reacts much less to the boundary layer and more to the rest of the flow field. As a result, the flow shows the beginning of a hook shape. Also in the region away from the symmetry plane, the solution is smoother and closer to the experimental results.

The remaining differences between the computation and the experiment are mostly due to the turbulence model. When the  $k-\omega$  SST model is replaced by the anisotropic EASM, the computation improves dramatically (Figure 13). For this turbulence model, the solution on the original grid has not improved, but the solution on the 1.07 M cell refined grid created with the pressure gradient criterion clearly shows the hook shape. Generally, it is in good agreement with the experimental results.

Concluding, the simple gradient criteria are indeed able to detect the relevant flow features. For the simulation of ship aft-body flows, the pressure gradient criterion is to be preferred, because it does not produce unnecessary refinement in boundary layers. The flow depends strongly on the turbulence model; however, the solutions for both the  $k-\omega$  SST and the EASM turbulence models are consistent with results obtained on equivalent manually generated locally refined grids. Yet, with the automatic grid adaptation procedure, the expert knowledge needed to manually create these grids is not needed.

## 6 Conclusions

An automatic grid refinement method is presented for ship flows. The method provides directional refinement on unstructured meshes and derefinement of refined meshes for unsteady calculation. Refinement criteria can be easily exchanged. The method is completely parallel and includes automatic dynamic load balancing.

Different refinement criteria are under development. The criteria studied here are the refinement around the free surface and refinement based on the velocity gradient, pressure gradient or vorticity of the flow. The free-surface criterion proves to be very effective for the simulation of strong and breaking waves. A ship test case with a complex wave field demonstrates the increase in wave accuracy and the better resolution of wave details that can be obtained using free-surface refinement. An example is given of a scale-related flow topology change that was only detected with grid refinement.

A second test case demonstrates the possibility to combine free-surface grid refinement with the non-compressive GDS discretisation for the volume fraction, in order to increase the time steps. While leading to finer grids, this combination reduces total computation times significantly.

With the same refinement criterion, a third case with free ship motion shows the successful combination of the refinement with unsteady mesh deformation.

Finally, a double model case is computed with the three gradient-based criteria. The pressure gradient criterion is the most effective, because it refines less in the boundary layer than the other two criteria and more in the flow away from the body. The solutions are highly dependent on the turbulence model used; solutions on refined grids confirm the existing result that the anisotropic EASM model simulates ship aft-body flow better than the  $k-\omega$  SST model. The test proves that the refinement procedure can effectively generate entire fine grids that are well adapted to relevant flow features.

## References

- Deng, G.B. and Visonneau, M. (1999) 'Comparison of explicit algebraic stress models and second-order turbulence closures for steady flows around ships', *Proceedings of the 7th International Conference on Numerical Ship Hydrodynamics, Nantes, France*.
- Duvigneau, R., Visonneau, M. and Deng, G.B. (2003) 'On the role played by turbulence closures in hull shape optimisation at model and full scale', *J. Marine Science and Technology*, Vol. 8, No. 1, pp.1–25.
- Hay, A., Queutey, P. and Visonneau, M. (2004) 'Computation of three-dimensional free-surface flows with an automatic adaptive mesh refinement and coarsening strategy', *Proceedings of the 25th Symposium on Naval Hydrodynamics, St John's, Newfoundland and Labrador*.
- Hay, A. and Visonneau, M. (2006) 'Error estimation using the error transport equation for finite-volume methods and arbitrary meshes', *Int. J. Computational Fluid Dynamics*, Vol. 20, No. 7, pp.463–479.
- Hino, T. (Ed.) (2005) *CFD Workshop Tokyo 2005*, Tokyo, Japan.
- Jasak, H. (1996) 'Error analysis and estimation for the finite volume method with applications to fluid flows', PhD thesis, University of London, United Kingdom.
- Larsson, L., Stern, F. and Bertram, V., (Eds.) (2000) *Gothenburg 2000, A Workshop on Numerical Ship Hydrodynamics*, Göteborg, Sweden.
- Leroyer, A. and Visonneau, M. (2005) 'Numerical methods for RANSE simulations of a self-propelled fish-like body', *J. Fluid & Structures*, Vol. 20, No. 3, pp.975–991.
- Longo, J. and Stern, F. (2002) 'Effects of drift angle on model ship flow', *Experiments in Fluids*, Vol. 32, pp.558–569.
- Menter F.R. (1993) 'Zonal two equation  $k-\omega$  turbulence models for aerodynamic flows', AIAA Paper 93-2906.
- ParMeTiS (2009) Available at <http://glaros.dtc.umn.edu/gkhome/metis/parmetis/overview>.
- Queutey, P. and Visonneau, M. (2007) 'An interface capturing method for free-surface hydrodynamic flows', *Computers & Fluids*, Vol. 36, No. 9, pp.1481–1510.