This document was prepared for the ETI by third parties under contract to the ETI. The ETI is making these documents and data available to the public to inform the debate on low carbon energy innovation and deployment.

Programme Area: Marine

Project: PerAWAT

Title: Numerical Modelling of Tidal Turbine Arrays Involving Interactions within an Array: Implementation of the Zero Tangential Shear Condition

Abstract:

This report includes an analysis of numerical modelling of tidal turbine arrays involving interactions within an array. Implementation of the zero tangential shear condition is included.

Context:

The Performance Assessment of Wave and Tidal Array Systems (PerAWaT) project, launched in October 2009 with £8m of ETI investment. The project delivered validated, commercial software tools capable of significantly reducing the levels of uncertainty associated with predicting the energy yield of major wave and tidal stream energy arrays. It also produced information that will help reduce commercial risk of future large scale wave and tidal array developments.

PerAWaT (MA 1003) Report – (WG3 WP2 D2)
Numerical modelling of Tidal Turbine arrays involving interactions within an array:
Implementation of the Zero Tangential shear condition

Participant lead on the deliverable:- **D.M. Ingram**
Other participant and author of report:- **D.A. Olivieri**
Institute Energy Systems, School of Engineering, University of Edinburgh, The King's Buildings, Mayfield Road, Edinburgh EH9 3JL UK
Version v.2
Date of submission:- 24/09/11

Nomenclature

$\vec{A}$, $\vec{B}$	Vector lines formed between the centres of a cell iel and its neighbour cell with index number
C	Constant that preserves the consevation of φ globally
con(iel,i)	Connectivity function giving the index number of the cell face i associated with cell index number iel
C_I	C_I is a set of nodes that are connected I_h but I_h^{th} node is not included in the C_I set of nodes
$\mathrm{H}(s)$	Heaviside function such that $\mathrm{H}(s)$ = 1 if s > 0, otherwise $\mathrm{H}(s)$ = 0 for s ≤ 0
iface(i,ifac)	Code Saturne's connectivity function giving the index number of the ith neighbour cell about cell face index number ifac
I_h	I_h^{th} node in the halo region of the narrow band region
κ	Set of simplexes in a region of the narrow band such that it surrounds an isosurface
nbcell(iel,i)	Cell connectivity function giving the index number of a neighbour cell associated with a central cell with an index number of iel
η_K	Simplex wise correction function associated with the Kth local grid mesh simplex volume function
N_I	Number of simplexes that contain the node I which the K^{th} simplex is a part of
$\hat{n}$	Unit normal to the free surface
nfac	Total number of cell faces from all field cells(code Saturne parameter)
P	Set of nodes with a narrow band region with a set of κ simplexes.
$\Re$	Set of halo nodes of the narrow band region in which I_h^{th} node is a part of $\Re$
S_h	Zero Level Set surface
Simplex	Any triangle or tetrahedron formed from the cell centres of existing mesh cells which around an isosurface region for the purpose of geometric based re-distancing of that region of the isosurface
$\hat{s}$	Tangential unit vector to the free surface position normal to $\hat{n}$ and $\hat{t}$
$\hat{t}$	Tangential unit vector to the free surface position
t	Model time in seconds
$\vec{u}$	Interface velocity of the isocontour, wave.
u_x	x component of $\vec{u}$
u_y	y component of $\vec{u}$
u_z	z component of $\vec{u}$
$\vec{u}_{nearest}$	The nearest velocity to the free surface position
$\vec{X}_J$	Position vector of a the Jth node such that J belongs to the C_I set of nodes in the narrow band region
$\vec{X}_{I_h}$	Position vector of a the I_h^{th} node such that I_h belongs to the $\Re$ set of nodes in the narrow band region
$\vec{x}$	Position vector in the flow field
$\vec{y}$	Position vector on isocontour S_h
φ	Level Set scalar
φ^*	Signed distance function from a flow field point $\vec{x}$ the closest point on the iscontour S_h
ψ	Correction function to correct or redistance φ
ΔV_K	Local grid mesh simplex volume function for simplex K in the narrow band

ΔV	Global volume function for a set κ of simplexes
ξ_{I}	Node wise correction function at node I for a number of simplexes that contain node I
Ω	Flow field
θ	The angle between two vectors $\vec{A}$ and $\vec{B}$
μ_s	Dynamic viscosity at the free surface
σ	Surface tension at the free at the free surface

Table of contents

1 Executive summary

This report, together with the accompanying FORTRAN subroutines, forms deliverable D2 of work package WG3WP2. The WG3WP2 objectives achieved so far are:

- Development of the level-set free surface model - WG3 WP2 D1
- Implementation of the Zero Tangential shear condition - WG3 WP2 D2

The first objective was completed previously in D1 which covered the Level Set model development. The second objective is that for D2 and the free-surface or upstream boundary conditions using a Ghost Fluid Method implemented within Code_Saturne modules. The validation of the above objectives forms the framework of deliverable D3.

To achieve these objectives, the PerAWaT project will make use of the existing parallel, high accuracy open source, CFD code Code_Saturne (developed by EDF) and will extend the model to provide a mechanism for modelling the performance of a small array of marine current turbines at the meso-scale. The content of the deliverable is:

- A review of the experienced gained from the previous deliverable in regards to Level Set modelling involving connectivity and re-distancing.
- The description of the theory to achieve a free surface modelling using a Ghost Fluid Method.
- An in-depth discussion on the current modelling strategy which aims for: emulation of the free surface conditions between the air and regions associated with tidal flow, and avoidance of numerical instabilities.
- Details of the necessary test case verification of the model, (the results of which will be published in D3).

Implementation of the free surface model requires modification to the user routines to enable the location of the free surface to be determined accurately and maintained, and to enforce the required boundary conditions across the free surface. The chosen approach requires the fluid flow equations to be solved only in the water. The implementation described in this report involves the use of a new Fortran-90 module to maintain the required connectivity information on an unstructured grid, as well as modifications to the user routines USCLIM.f90, USINIV.f90, USINI1.f90 and USPROJ.f90.

This work will follow a process of development with the existing flow solver:
(1) The implementation of a free surface model which satisfies the zero-tangential shear boundary condition,
(2) The implementation of an unsteady upstream boundary condition to introduce large scale, synthetic, turbulent eddy structures into the domain,
(3) The development of a parameterised actuator disk model of a horizontal-axis marine current turbine.

The introduction covers the description of the Level Set method, and explains why it was chosen as the best approach to free surface modelling, with reference to the most recent improvements in the numerical scheme, namely improved mass conservation features.

The methodology section gives a detailed description of the Level Set method used, with particular reference to the author's implementation of an advanced mass preserving re-initialisation method and its application to structured or unstructured grid systems.

This deliverable meets the acceptance criteria of D2, which states in WG3 WP2 D3 that

(1) Modules produced are capable of operating within Code Saturne. Source code sufficiently commented such that it can be logically followed by a third party.
(2) Report describes the assumptions and algorithms behind the additional modules and current applications and limitations. The report will verify the functionality of the modules

2 Introduction

The application of Code_Saturne to the simulation of small farms of tidal turbines requires three major developments of the existing flow solver:

1. The implementation of a free surface model which satisfies the zero-tangential shear boundary condition;
2. The implementation of an unsteady upstream boundary condition to introduce large scale, synthetic, turbulent eddy structures into the domain; and,
3. The development of a parameterised actuator disk model of a horizontal-axis marine current turbine.

This report, together with D1 and D3, addresses the point (1), the implementation of a free surface model, which satisfies the zero-tangential shear boundary condition.

The remaining two points will be dealt with in subsequent deliverables. The inclusion of a free surface in flow simulations is thought to be important to the accurate modelling of the flow field in, and around, an array of turbines, as energy extraction leads to deformation of the free surface and subsequent changes in the pressure and velocity distributions around the turbines.

Such a boundary condition is also critical if the influence of surface waves on the turbines is to be investigated. If the simplest approach of using a "rigid lid" is to be avoided a number of approaches are available, including: Surface Fitting [1], Density Function [2], Front-Tracking [3], Smoothed Particle Hydrodynamics [4], Volume of Fluid (VOF) [5], Free Surface Capturing [6] and Level Set Method [7,8]. Most of these approaches simplify the flow problem and the computational requirements by considering only the liquid-component (modelling the gas component by a numerical vacuum).

Modelling methods involving surface-fitting [1] have advantages of speed but lack complexity since the free surface is treated as a moving upper boundary. It therefore has limited application to simple free surface modelling with little skewness[1]. Re-gridding and interpolation methods can avoid the effects associated with grid skewness at the price of accuracy. Other effects of wave breaking[2] cannot be modelled with this method due its inability to specify an explicit boundary at the interface of a breaking wave [11].

The use of marker functions like a volume or signed distance function with surface-tracking schemes such as VoF or Level Set may be employed to define the location of the free surface. As result these methods can model wave breaking but each method has its limitations. With VoF, the location of the free surface will involve reconstruction from the volume fraction. Unfortunately, VoF needs to employ a specific reconstruction algorithm for a particular application, such open channel flow or mould filling. There is also the extra problem of a large number of grid cells needed to accurately calculate the free surface curvature.

[1] The term skewness refers to angle between the vector normal of the cell face between two neighbouring cells and the line vector connecting their centres [10].
[2] Water-air interactions are of significance in wave modelling particularly with overtopping [11].

Lagrangian Grid Methods [15] which are imbedded in the moving surface are an efficient solution to effective free surface definition but are unable to track surfaces that break up or intersect. Other approaches such as Smoothed Particle Hydrodynamics (SPH) involving a large number of Lagrangian particles suffer poor free surface definition unless employing a large number of particles, which makes the method computationally inefficient.

The approach adopted here is to employ a Level Set approach to free surface calculations adopting the boundary condition as laid out in Watanabe et al [16]. The reasons for this choice are born out of numerical efficiency in free surface calculation of curvature and location without resorting parameterising these objects. With the Level Set method, the location of the free surface is defined by the zero contour of a signed distance function. Using this approach has the advantage of representing the discontinuous properties such as density in terms of a continuous function. The Level Set (LS) method main advantages over VoF are better surface definition without the empiricism of a specific reconstruction algorithm for a particular application. Mass conservation with LS, which is automatically met with VoF, has to be achieved by re-distancing techniques, discussed later.

2.1 The Level Set Method

The Level Set Method is a surface-tracking scheme which uses a marker function φ known as signed distanced from the interface S_h whereby $\varphi = 0$. The evolution of marker function, also known as a Level Set given as

$$\frac{\partial \varphi}{\partial t} + \nabla \cdot (\vec{u}\varphi) = 0 , \qquad (1)$$

where $\vec{u} = (u_x, u_y, u_z)$ is the interface velocity in the three dimensional flow field Ω. Both the level set function and the velocity field are functions of $(\vec{x}, t), \vec{x} \in \Omega, t > 0$. In this project the Code_Saturne solver is employed and the Navier-Stokes equations operating in the kernel of the programme will supply the updated interface velocity at each time step. Unfortunately φ is not maintained with reference to conservation laws and therefore the zero φ iso-surface will propagate at the incorrect speed, which means that a mass conservation error will occur at the moving interface. To correct this fault re-distancing methods have been introduced with some as elaborate as level-set- based adaptive Characteristics-Based Matching method [17], which involves the use Adaptive Mesh Refinement applied in the main flow transport solver and the Level Set Method combined. The usual approach to perform re-distancing would involve high-order finite difference schemes, such as high-order essentially non-oscillatory (ENO) schemes like Hamiltonian-Jacobi Partial Differential equations [7, 8, 19]. Unfortunately the method of re-distancing φ is not local enough to each cell centre area within the flow field. For this reason the author selected the approach of Ausas et al [18] which lends itself well to unstructured grids as is case with Code_Saturne.

If the Level Set φ met the conservation mass requirement it would be equal to the sign distance function φ^*. Unfortunately, a correction ψ to the Level Set is needed[3]

[3] The explanation as to why re-distancing is needed, which is indicated in Equation (2), is the most generally adopted description of the re-distancing process see references [7,8,16,18]

$$\varphi = \varphi^* + \psi \qquad (2)$$

In Figure 1 the signed distanced function φ^* is defined, which is the closest signed distance to the zero isocontour S_h, in which $\vec{y}$ belongs to the zero isocontour for a given point in space $\vec{x}$

$$\varphi(\vec{x})^* = \text{sign}[\varphi(\vec{x})] \min_{\vec{y} \in S_h} \|\vec{x} - \vec{y}\| \qquad (3)$$

2.2 The Ghost Fluid Method

Single phase Eulerian schemes perform adequately for most situations with gases involving large deformations. With multi-phase Eulerian schemes, the region near each interface will experience non-physical oscillations in the Navier Stokes calculations, due to the density profile being smeared out and the radical change in equation of state across this region. With Lagrangian schemes these oscillations will not occur, since density profiles are not smeared out and equations of state are still valid over each point. The Ghost Fluid method (GFM) proposed by Fedkiw et al [19, 20] both Eulerian and Langrangian schemes are combined. By considering the water-air interface region, we can solve for water by replacing the air region near the interface with ghost water that acts like the air in every way (same pressure and velocity as air) but appears to be water (same entropy as water). As a result, boundary conditions are captured appropriately in the region narrow band region as seen in Figure 10. The GFM therefore becomes a one phase problem, because the ghost fluids behave consistently with the real fluids they are replacing, and also have the same entropy as the real fluid that is not replaced. For this reason a GFM approach was applied to the free surface boundary problem discussed in this report. The free interface between fluids is tracked with the Level Set approach, which was discussed earlier. The jump conditions at the interface for pressure and velocity are imposed by GFM. The re-distancing discussed in 2.1 would allow GFM to work properly with the Level Set method in Code Saturne as long as the air region above ghost fluid region is not allowed to update itself in terms of pressure and velocity, i.e. $\partial\vec{u}/\partial t = 0$ and $\partial P/\partial t = 0$, for the air region indicated as region B in Figure 10. In Figure 8, the ghost cells are selected a distance $\varphi = |l|$ above the free surface shown as white triangles. The jump conditions for GFM were developed in Kang et al. [20]. The discontinuous derivatives for velocity and pressure across a sharp interface between regions A and B are defined discretely in the narrow band shown in figure 10 as

$$[P] - \hat{n}.[\mu(\nabla\vec{u} + (\nabla\vec{u})^t)].\hat{n} = \sigma\kappa' \qquad (4)$$

$$\hat{t}.\left[\mu_s(\nabla\vec{u} + (\nabla\vec{u})^t)\right]\hat{n} = 0 \qquad (5)$$

$$[\vec{u}] = 0, \qquad (6)$$

where **[.]** = $(.)_G - (.)_R$ represents the jump in general quantity across the free surface in the narrow band region indicated in Figure 10. Quantity $(\psi)_G$ represent the ghost cell quantity ψ from a point in the red band region of the narrow band as indicated in figure 10. In figure 8, the ghost quantity $(\psi)_G$ is associated with the white triangles while image quantity $(\psi)_R$ is associated with the blue triangles.

Pressure defined in equation (4) needs to be accounted for as when solving the Poisson equation. This involves $[p^*]$ across the free surface as

$$[\,p^*\,] - 2\Delta t\hat{n}.[\,\mu\,(\nabla\vec{u} + (\nabla\vec{u})^t)].\,\hat{n} = \Delta t\sigma\kappa' \qquad (7)$$

For the purpose of deliverable D2, the above model has been simplified to an inviscid situation whereby for the free surface, μ_s and σ are treated as negligible quantities. As a result equations (4) becomes

$$[P]=0, \qquad (8)$$

with equations (5) and (7) not used. This indicates that pressure and velocities are continuous across the free surface such that for pressure $(P)_G = (P)_R$ and $(\vec{u})_G = (\vec{u})_R$

3 Methodology

3.1 Introduction

In this section the methodology behind free surface modelling is presented involving the implementation of an unsteady upstream boundary condition. As discussed previously in section 2.1, the preference for a re-distancing method involving a geometric mass preserving approach for level set function is born out the fact that correction to the LS, expressed in equation (2), is more localized to cell centres as compared to the usual Partial Differential Equation-based approach for re-distancing [7,8,18]. Section 3.2 gives a detailed account of the method.

Once re-distancing is completed the free surface, which is also the isocontour of the now corrected LS at $\varphi = 0$, will now be used in the calculation of the surface normal $\hat{n}$ and curvature $\kappa,'$ as given in equations (9) and (10).

$$\hat{n} = -\frac{\nabla\varphi}{|\nabla\varphi|} \qquad (9)$$

$$\kappa'_{v} = \nabla.\hat{n} \qquad (10)$$

These are important geometric features of the free surface which play an important part in the Ghost Fluid Method (see section 2.2). Equations (4), (5) and (7) for the jump conditions developed in Kang et al[20]. In section 3.3 the numerical methodology is discussed in more detail. Further to this, we can also find the unit tangential vector to the free surface $\hat{t}$ in 2D flow. Again this is important in Equation (5) as we are considering a non-zero shear stress at the free surface. This can be found simply by the cross product of the normal to the free surface and the normal to the 2D flow field (Equation (11))

$$\hat{t} = \hat{n} \wedge \hat{k}, \qquad (11)$$

where $\hat{k}$ is the unit normal to the 2D flow field. In the case of 3D flow the nearest flow velocity to the free surface is taken for $\hat{t}$, while the second unit vector $\hat{s}$ is the cross product of $\hat{t}$ and $\hat{n}$, that is

$$\hat{t} = \frac{\vec{u}_{nearest} - (\hat{n}.\vec{u}_{nearest}).\hat{n}}{\hat{n}.\vec{u}_{nearest}} \quad (12)$$

$$\hat{s} = \hat{t} \wedge \hat{n} \quad (13)$$

From the GFM described later in section 3.4 the assumption at the free surface of zero shear stress μ_s = 0 and zero surface tension σ = 0 renders tangential quantities $\hat{t}$ and $\hat{s}$ redundant for D2 cases . In later deliverables, the shear stress at the free surface will be non-zero which will involve $\hat{t}$ in the more advanced GFM, along with the second free surface tangential unit vector $\hat{s}$ in the case of 3D flow. The method by which the slope $\nabla\varphi$ has been calculated in the author's modules is based on gradient smoothing method,(GSM) [22,23]. GSM is based on divergence theorem and is discussed in more detail in section 3.3.

After the calculation of slope and curvature at free surface, the Ghost Fluid Method can be applied at the particular time step Δt. The GFM is applied to the narrow band region, (shown as red in figure 10). In the narrow band region values of pressure and velocity calculated from the image points, as shown in blue triangles in figure 8, are simply passed over to the respective ghost points, shown as white triangles in figure 8. The ghost points populate the red band region of the narrow band region as shown in figure 10, while the image points holding the real values populate the green band region of the narrow band. Finally, implementing an unsteady upstream boundary condition is possible through module `USCLIM.f90`. In this module different velocity and pressure profiles can be entered through the inlet boundary at different positions along the boundary at each time step Δt, since `USCLIM.f90` is referred to at the beginning of every time step. (See section 3.5 for details).

To recap on the entire process of implementing a free-surface boundary condition using a Level Set method with a GFM solver, the following is implemented for each time step Δt

- A single scalar selected from the Code Saturne GUI is used represent the sign distance function φ^*, which is set up across the inlet boundary with `USCLIM.f90` and throughout the flow field initially with `USINV.f90` and latterly updated at end of each time set with `USPROF.f90`.
- The initial surface reconstructions based φ^*on require modification to meet mass conservation through the process of re-distancing. Again this is achieved initially with `USINV.f90` and subsequently with `USPROJ.f90`.
- After re-distancing, geometric attributes of the wave surface are calculated, i.e. $\hat{n}$ etc.
- Finally, the values of ghost values of velocity and presure are updated ready for the next time step Δt.

3.2 Calculation of Level set with re-distancing

In the implementation of the geometric mass-preserving re-distancing approach [18], adjustments to φ are confined to a narrow band of primary neighbour cells about the computed zero isocontour. These are shown in Figure 4 as red cells, which amounts to one cell either side of the isocontour. There are also a set of secondary, or 'halo' cells, shown in pink that surround the primary cells. These provide a means of primary- to- secondary cell exchange as the isocontour changes position at each time step.

In order for Code Saturne to achieve this degree of control over each cell in the flow field the authors have had to develop a method to determine the connectivity of a cell with the locally surrounding neighbour cells. This information is not stored internally in Code Saturne which represents the grid using an edge and volume list. A Fortran-90 module has been developed which derives the connectivity information from the edge list in the narrow band of cells surrounding the zero-isocontour. Based on this derived connectivity information, a set of primary and secondary cells is generated forming the narrow band object seen in Figure 3. The light and dark blue cells form the set of primary cells with the isocontour at the interface between light and dark cells. The light green cells are the halo cells. This forms the basis of each indexed narrow band object that finally forms a spine of narrow band objects as seen in Figure 4.

As the simulated time advances each narrow band object will inhabit different neighbouring cells as the scalar values given at the cell centres change. This alters the position of the narrow band tracking the interface.

As stated previously the change in φ at each cell centre will arise both from advection of $\vec{u}$ and from the secondary influence from re-distancing of φ. The process by which φ is corrected can be outlined briefly if we consider any cluster of narrow band object nodes such that part of the isocontour S_h involved will be a subset of the set of κ simplexes involved where their φ values change sign.

1 The centres of the Finite Volume cells used in the transportation algorithm to calculate φ form the vertices of the triangulation required for the re-distancing algorithm. In Figure 5, the reconstruction of isocontour S_h from calculated values φ will provide the means to calculate the signed distance function φ^* at each node in the narrow band. The disagreement between these values form the basis to calculate the mass correction function ψ as given equation (2)

2 To obtain the mass correction, a piecewise-constant function η_K is found for each simplex formed out of first neighbour cell centres of the narrow band objects. The difference in volumes defined by φ and φ^* would be corrected at each simplex K such that

$$\Delta V_K(\varphi, \varphi^* + \eta_K) = 0 \,. \qquad (11)$$

where ΔV_K is defined as follows

$$\Delta V_K\left(\varphi(\vec{x}),\varphi^*(\vec{x})+\eta_K\right)=\int_K\left[\mathrm{H}(\varphi(\vec{x}))-\mathrm{H}\left(\varphi^*(\vec{x})+\eta_K\right)\right]\cdot d\vec{x} \qquad (12)$$

where $\mathrm{H}(s)$ is the Heaviside function in order that the function is unity if $s > 0$ and zero otherwise. The value of η_K is then determined by a false position method to obtain η_K for that particular simplex. Ultimately, considering ΔV for the set of κ simplices in this region would suggest a set of simplex-wide solutions of η_K

$$\Delta V\left(\varphi,\varphi^*\right)=\sum_{K\in\mathrm{K}}\Delta V_K\left(\varphi,\varphi^*\right) \qquad (13)$$

In view of the discontinuous nature of η_K a node wide solution ξ_I is sought at each node I

3 The simplex wise contributions η_K at each node of the first neighbour cells, which are a part of the simplexes in the narrow band, are then averaged at each node to provide a node-wise correction such that

$$\xi_I=\frac{1}{N_I}\sum\eta_K \ , \qquad (14)$$

$$\psi_I = C\xi_I \qquad (15)$$

where N_I is the number of simplexes that contain the node I which the K^{th} simplex is a part of, and C the constant that preserves the mass conservation of φ globally such that

$$\Delta V\left(\varphi,\varphi^*+C\xi\right)=0 \qquad (16)$$

The solution of equation (16) to find C again involves a false position method.

4 The mass corrected values of φ amongst all nodes of the first neighbour cells will now provide a boundary condition for the re-initialisation of φ on the rest of the mesh points in the halo cells. If we consider just the halo cells with φ positive, for say a $\Re$ set of halo nodes in which $I_h\in\Re$, we use a distance along an edge approximation such for I_h such that

$$\varphi(X_{I_h}) = \min_{J\in C_I}\left[\varphi(\vec{X}_J)+|\vec{X}_{I_h}+\vec{X}_J|\right] \qquad (17)$$

where C_I is a set of nodes that are connected I_h but I_h^{th} node is not included in the C_I set of nodes but $C_I\subset(\mathrm{P}\cup\Re)$, P being the set of nodes associated with κ simplexes. Equation (17) provides an edge distance approximation. For each simplex, and for each node J of the simplex at position $\vec{X}_J$, φ is interpolated linearly on the opposite face FJ (see Figure 6), using the current values at the nodes generated by the earlier stage involving the first neighbour cells. Then, a tentative new value η_J at node J is calculated such that

$$\eta_J = \min_{\vec{x}\in F_J}\left[\varphi(\vec{x})+|\vec{X}_{I_h}+\vec{x}|\right] \qquad (18)$$

$$\varphi(\vec{X}_J) = \eta_J \text{ if } \varphi(\vec{X}_J) > \eta_J \quad (19)$$

5 As a result of the re-distancing routine given above, some of the primary and secondary neighbour cells of the narrow band have altered. New cells will have to be inhabited and old cells evacuated as a result of the change in position of the isocontour. The result of re-distancing can be seen in figures 7(a) and 7(b) showing a map of $|\nabla\varphi|$ over the flow field from the submerged 2D cylinder. In the narrow band region surrounding the isocontour for $\varphi = 0$, indicated as a black wavy line, mass conservation is maintained. This indicated by $|\nabla\varphi| = 1$, seen as light green regions in figure 7(b). In the case where no re-distancing is done, as shown in Figure 7(a) , the isocontour is not surrounded by the light green mapping as much as is the case with Figure 7(b).

3.3 Calculation of slope and the curvature at the free surface

If we consider the particular position on the free surface indicated by a green dot in Figure 8, the process to calculate the normal at this point involves two sets of N_s stencils formed above and below the green dot. If we define the slope at the green dot, which is seen on Figure 8, indicated by $\hat{n}$ its value will be the mean of $\hat{n}_j$ calculated from each of these two N_s stencils. (Note in figure 9 shows an N_s stencil for $\hat{n}_j$ above the free surface). Hence for $\hat{n}$ we get

$$\hat{n} = \left((\hat{n}_j)_1 + (\hat{n}_j)_2\right)/2 \quad (18)$$

To calculate the gradient $\nabla\varphi$ at the j^{th} node, which is the parent cell of one the set of N_s stencils near a free surface position, has used a Gradient Smoothing Method (GSM) approach as outlined by Barth et al and Liu et al [22-23] is used. The GSM approach used by the author involves using neighbouring nodes F, (where F = F_1, or F_2, …), as shown in Figure 9. This requires a 9 cell stencil for 2D flow, or a 27 cell stencil for 3D flow. Applying Divergence Theorem to the 9 or 27 cell stencil scheme we get for $\nabla\varphi_j$

$$\nabla\varphi_j = \underbrace{\lim_{\Delta Q \to 0}} \frac{1}{\Delta Q} \oint_{\Delta s} ds\ \varphi \approx \frac{1}{\Delta A} \sum_{i=1}^{i=Ns-1} \hat{n}_j^i \Delta s_j^i \left(\varphi_j^i + \varphi_j^{i+1}\right)/2, \quad (19)$$

where ΔQ is the volume of fluid contained in the pink region of the N_s cell stencil, ΔA is the total outer surface area of this region, $\hat{n}_j^i$ is the surface normal of a discretized surface region, $\Delta s_j^i \in \Delta A$. (Note scalar φ_j^i at node i is part of the N_s-1 nodes of the N_s cell stencil, which forms part of in the trapezoidal integration method approximating line-integration).

The slope calculated in Equation (14) suffers numerical oscillation effects in an unstructured mesh scheme [22, 23]. To prevent these oscillations, the largest allowable value of limiter Ψ_j is sought, which imposes a "monotonicity principle" on the calculated slope values of φ at the j^{th} free surface position. This means Ψ_j must not exceed the extrema of the φ_j^i from the neighbouring nodes which includes the parent node N_s. To find this we start by calculating φ_j^{min} and φ_j^{max} from each of the neighbouring nodes scalar associated with ith node of the N_s stencil i.e

$$\varphi_j^{min} = min(\varphi_j^{Ns}, \varphi_j^i) \text{ of } i = 1, N_s \quad (20)$$

$$\varphi_j^{max} = max(\varphi_j^{Ns}, \varphi_j^i) \text{ of } i = 1, N_s \quad (21)$$

The next stage is to consider the value of φ_j^i at each mid point on the line connecting a neighbouring node the central node N_s which has value of $\varphi_j^i = \varphi_j^{Ns}$, i.e. $\varphi_j^{mid(i)} = (\varphi_j^i + \varphi_j^{Ns})/2$. Now if the solution of limiter is found at these $(N_s - 1)$ mid-points on the parent cell boundary we get

$$\Psi_j^i = \begin{cases} \min\left(1, \dfrac{\varphi_j^{\max} - \varphi_j^{mid(i)}}{\varphi_j^{mid(i)} - \varphi_j^{Ns}}\right) & \text{if } (\varphi_j^{mid} - \varphi_j^{Ns}) > 0 \\ \min\left(1, \dfrac{\varphi_j^{\min} - \varphi_j^{mid(i)}}{\varphi_j^{mid(i)} - \varphi_j^{Ns}}\right) & \text{if } (\varphi_j^{mid} - \varphi_j^{Ns}) < 0 \\ 1 & \text{if } (\varphi_j^{mid} - \varphi_j^{Ns}) = 0 \end{cases} \quad j = 1, \ldots, N_s \quad (22)$$

Finally the limiter for the j^{th} free surface position becomes

$$\Psi_j = min(\varphi_j^{mid(1)}, \varphi_j^{mid(2)}, \ldots, \varphi_j^{mid(i)}, \ldots, \varphi_j^{mid(Ns-1)}) \quad (23)$$

Therefore the modification to the expression for the slope in equation (14) now becomes

$$\nabla \varphi \approx \frac{(\psi_j)}{\Delta A} \sum_{i=1}^{i=Ns-1} \hat{n}_j^i \Delta s_j^i (\varphi_j^i + \varphi_j^{i+1})/2 \quad (24)$$

The above GSM is similar to what is available in Code saturne in the form of the function *grdcel* . The *grdcel* function can offer either an iterative GSM or Least Square method to calculating the gradient. This is achieved by setting the appropriate value for the variable *IMGRA*. Unfortunately *grdcel* does not guarantee the speed of calculation or numerical stability that has been described above involving GSM. Only further work will determine if *grdcel* can effectively replace the author's initial approach to calculating $(\hat{n}_j)_{1or2}$ near the free surface.

3.4 Calculation methodology for the Ghost Fluid Method

In figure 8, the free surface normals $\hat{n}$ are known at certain boundary points indicated by green dots. Fortunately $\hat{n}$ from a given green dot can also apply approximately to the actual normal at a free surface position indicated by black dot. (The black dot free surface position is formed by the normal line through the cell centre position above the free surface, which indicated by a white triangle and called a ghost point). These ghost points will receive pressure and velocity values from the image positions. From section 2.2 the simplified GFM has been described which states that these interpolated values found at the image positions are simply carried over to the ghost positions indicated as white triangles. These image values are calculated from a suitable bilinear interpolation method applied to the nodes surrounding each image point indicated by a blue triangle in Figure 8.
In Figure 8, nodes 1 to 4 that surround a particular image point are shown. The grid shown as red grid lines will generally be an unstructured grid. For this purpose, the author has used a geometrically isotropic bilinear interpolation scheme based on a bi-cubic interpolation scheme proposed by Ii et al [25] for each image point. This means that three out of the four nodes which surround the image point are selected. The bilinear interpolation scheme will allow velocity or pressure values to be found using the linear polynomial

$$\psi(x,y) = C_{00} + C_{01}\, x + C_{10}\, y\,, \tag{25}$$

where C_{00} , C_{01} and C_{10} are the coefficients of interpolation triangle found from three of the four points as seen in Figure 8 such that

$$\begin{bmatrix} C_{00} \\ C_{01} \\ C_{10} \end{bmatrix} = \begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_4 & y_3 \end{bmatrix}^{-1} \begin{bmatrix} \psi_1 \\ \psi_2 \\ \psi_4 \end{bmatrix} \quad , \tag{26}$$

where ψ may be a velocity or pressure. For 3D flow this would involve trilinear interpolation scheme similar to what has been described above, except that a tetrahedral interpolation box would be selected out of a hexahedral capture box containing the image point for $\varphi(x, y, z)$. The eight corner points of the hexahedral capture box are made up from the cell centres of cells surrounding the image point, such that

$$\psi(x,y,z) = C_{00} + C_{01}\, x + C_{10}\, y + C_{11}\, z \tag{27}$$

From the above interpolation scheme, the pressure $(P)_R$ and velocity $(\bar{u})_R$ found at the image points below the free surface will allow the ghost pressure and velocity values to be found at the ghost points from $(P)_G = (P)_R$ and $(\bar{u})_G = (\bar{u})_R$. These new values will populate those ghost cell points covering the red band region (Figure 10) of the narrow band strip straddling the newly calculated free surface, shown as a black line. The green strip is the region populated with the image points necessary from the GFM. The pressures $(P)_R$ are only the reduced pressure values and not the total pressures.

3.5 Implementing unsteady upstream boundary condition

In the USCLIM.f90 module, the velocity profiles can implemented over the prescribed inlet face. The standard boundary condition is usually a zero-flux condition for pressure and a Dirichlet condition for all other variables. To set a Dirichlet condition for the velocity over the inlet, the functions icodcl(ifac,ivar)=1 and rcodcl(ifac, ivar, 1) are used. Where ivar is equal to either iu(iphase), iv(iphase) and iw(iphase) for the respective 3D velocity components that have been specified at inlet face ifac at time t for a given phase iphase. In case of the investigation iphase =1 since the GFM has reduced the two water air interface problem to a single phase problem.

4 Methodology

4.1 A review on previous connectivity modeling

The internal representation of the grid within Code_Saturne has presented some difficulties. The solver represents the unstructured grid using a list of faces with pointers to the left and right control volumes, rather than providing a list of control volumes together with their connectivity information as required by the re-distancing algorithm. The first stage of implementation has therefore required the development of a set of library routines to compute the connectivity information for the halo cells surrounding the interface from the face connectivity list. These routines have been implemented within the CONNECTIVITY Fortran-90 module coded into the preamble of the user routines. Face connectivity information provided by the NBCELL data structure is processed to find adjacent control volumes and from these, a set of simplexes is constructed. Since the grid used is fully unstructured, the list of control volumes computed has no implicit order and so must be sorted before re-distancing can be performed.

Below is a sample of code taken from the file `USINIV.f90`, which constructs the connectivity between a given cell of index `iel` and in this case 1 to six neighbour cells associated with each 3D hexahedra grid,

```
!===============================================================================
!Building connectivity between particular indexed cell in the flow field and
!its local neighbour cells. For a hexahedral cell this involves 6 neighbours
!===============================================================================
     do ifac = 1, nfac
        iel1 = ifacel(1, ifac)
        iel2 = ifacel(2,ifac)
        switch1 = .true.
        switch2 = .true.
        do i = 1, 6
           if ((switch1).and.(con(iel1,i).eq.(-ihuge) )) then
              con(iel1,i) = ifac
              switch1 = .false.
           endif
           if ((switch2).and.(con(iel2,i).eq.(-ihuge) )) then
```

```
            con(iel2,i) = ifac
            switch2 = .false.
          endif
        enddo
      enddo
```

The connectivity function `con(iel,i)` developed by the author provides the necessary process of indexed cell → local face connectivity, for a given cell index `iel` and a randomly selected cell face `i`. To complete the process of indexed cell → local face → local neighbour cell connectivity an extra piece of coding was needed in the form of another connectivity function developed by the author known as `nbcell(iel,i)`. In this initial coding exercise, it has been set up specifically to deal with only three dimensional hexahedral meshing by having the number of cell faces restricted to six. In the next stage of development the code will be upgraded to meet any general unstructured polyhedral mesh.

4.2 A review on previous Re-Distancing modeling

The re-distancing algorithm has been implemented in several of the user routines, which provide an interface to the main Code_Saturne solver. The routine `USCLIM.f90`, which sets up the boundary conditions for the flow, has been modified to allow properly-distanced values of φ to be introduced through an inflow boundary. This routine does not need to make use of connectivity information since it is applied only to boundary faces. The routine `USINV.f90` sets up the initial values of the level-set across the entire fluid domain and contains the `CONECTIVITY` module described above. `USINV.f90`, also sets up the initial data structure containing the narrow band cells surrounding the zero-contour of the scalar variable. The initial configuration of the narrow band is set up about the specified isocontour position for time zero by using a set of narrow band node elements as seen in Figure 3. These elements finally form a spine of narrow band elements as shown in Figure 4a. Each narrow band element consists of two primary cells, which are positioned either side of the particular part of the isocontour. In addition there are also halo cells either side of the primary. As model time advances, the value of Level Set φ will change in accordance with the transport model and the re-distancing algorithm (section 3.2). At the end of each time step `USPROJ.f90` is called. In this subroutine, the re-distancing algorithm is given, and implements the geometric mass preserving algorithm [1]. This algorithm monitors and updates the narrow band data structure as the free surface moves across the background mesh. The narrow band elements will adapt themselves inline with these prevailing conditions as shown Figures 4b and 7b.

The `USCLIM`, `USINV` and `USPROJ` routines, containing self-documented source code, are contained in the attached files.

4.3 Current programming strategy

Setting up the free surface boundary condition in the narrow band region, as seen in Figures 8 and 10, will allow the GFM to reduce the two phase problem of wind and water interaction to a one phase problem, by the method outlined in section 2.2. This means the red line region of the narrow band

region shown in Figure 10 will be filled with water and given the ghost values of pressure and velocity of $(P)_G = (P)_R$ and $(\vec{u})_G = (\vec{u})_R$, where $(P)_R$ and $(\vec{u})_R$ values are taken from the appropriate image points in the green line region of the narrow band region, (Figures 8 and 10). Usually the region above narrow band region, seen as region B in Figure 10, is not allowed to be updated by the solver, otherwise the solver will interfere with the GFM process and may cause numerical instabilities. This approach is widely used in free surface flow solvers for segregated heterogeneous fluids for reasons of computational efficiency.

The implementation of this approach has proven difficult due to the nature of Code Saturne's kernel and the lack of available detailed documentation on the kernel's structure. To implement the "numerical vacuum" approach the solver either has to be prevented from updating the flow solution for cells containing air, where φ <0, or the momentum and pressure equations in this region must be modified to

$$\begin{aligned} \frac{\partial u_i}{\partial t} &= 0 \\ \frac{\partial p}{\partial t} &= 0 \end{aligned} \quad . \qquad (28)$$

As an alternative to this approach a stratified flow method will be considered making the field density a function of the prevailing Level Set scalar φ. This should allow Code_Saturne to process the entire flow field in a stable manner, without numerical instability issues occurring in the calculation. It should also allow the required GFM process to be performed in the narrow band region, as required for correct free surface modelling. Testing of this approach is still on-going, and results will be presented in D3. The geometrically isotropic bilinear interpolation scheme has involved significant coding with a kd-tree routine to select the appropriate four nodes, as seen in figure 8, with a matrix inversion routine required for equation (26). So far, this approach to coding the interpolation process required for the image point seen in figure 8 has been stable.

5. New Modules

To address the connectivity issues highlighted in Section 4 a set of new library routines has been implemented in the `CONNECTIVITY` module, which is held within the existing Code_Saturne routine `usiniv.f90`. The Level set method using the geometric mass-preserving, re-distancing algorithm has been implemented via developments to the existing Code_Saturne routines of `USCLIM.f90`, `USINV.f90`, `USPROF.f90` and `USINI1.f90`. The source code containing the `USCLIM`, `USINV` and `USPROJ` routines is included on the accompanying CD with the modifications made under D2 highlighted.

6. Results

The re-distancing method has consistently between reliable with all 2D runs without causing failure. A qualitative assessment of the re-distancing is presented in figures 7a and 7b, which maintains the $|\nabla \varphi|$ = 1 requirement near the isocontour. This is an essential requirement before any GFM is conducted.

Test work with the new code associated with the Ghost Fluid Method has been stable, particularly with the geometrically isotropic bilinear interpolation scheme, but verification with experimental data has not yet been performed.

7. Conclusions

The work package objective of WG3 WP2 D2 was to implement a Level-Set free surface model capable of performing a free-surface boundary condition and unsteady upstream boundary condition with Code_Saturne. So far, significant progress has been made to achieve this, but further work is needed on the stratified field approach as a work-around to the obstacles associated with the inaccessibility of the kernel of Code_Saturne. The success of this will be covered in the next deliverable of D3 through verification with experimental data.

In the last deliverable D1, cell connectivity and re-distancing were achieved with its success seen in figure 7. The major developments since D1 have been in coding the Gradient Smoothing Method, which is necessary for the calculation of geometric features of the free surface, and coding the Ghost Fluid Method. The GFM has proven difficult to achieve in view of the lack of advice available to reprogram the kernel of Code Saturne in the way highlighted in section 4.3. The geometrically isotropic, bilinear interpolation scheme employed in the GFM has shown itself to be a stable method. The stratified flow approach will need some care to avoid issues with numerical stability near the interface region of the flow field.
To demonstrate the required functionality of the modifications featured in this deliverable, a set of tests is planned as indicated in Figure 11. The results of these tests form the main content of deliverable D3.

8. References

[1]Farm, J., Martinelli, L., Jameson, A. Fast multigrid method for solving incompressible hydrodynamic problems with free surfaces. AIAA Journal 1994. 32, 1175–1182.

[2] Park, J., Kim, M., Miyata, H.. *Full non-linear free-surface simulations by a 3D viscous numerical wave tank.* International Journal for Numerical Methods in Fluids.1999. 29 (6), 658–703

[3]Unverdi, S., Tryggvason, G.. *A front-tracking method for viscous, incompressible multifluid flow*. Journal of Computational Physics 1992. 100, 25–37.

[4] Monaghan, J. *Simulating free surface flows with SPH*. Journal of Computational Physics.1994. 110, 399–406.

[5] Hirt, C., Nichols, B. *Volume of fluid (VoF) methods for dynamics of free boundaries.* Journal of Computational Physics.1981. 39, 201–225.

[6] Kelecy, F., Pletcher, R. *The development of a free surface capturing approach for multidimensional free surface flows in closed containers.* Journal of Computational Physics.1997. 138, 939–980.

[7] Osher S, Fedkiw R. *Level Set Methods and Dynamic Implicit Surfaces*. Publisher: Springer, Applied Mathematical Sciences. 2003: ISBN 978-0-387-95482-0.

[8] Sussman M, Fatemi E. *An efficient, interface-preserving level set redistancing algorithm and its application to interfacial incompressible fluid flow*. SIAM J. Sci.Comput. 1999; 20:1165–1191.

[9] Youngs, D. *Time dependent multi-material flow with large fluid distortion.* In: Morton, K., Baines, M. (Eds.), Numerical Methods for Fluid Dynamics. Academic Press, London. 1982. 237–285.

[10]Online details of *"Meshing_CompressibleModels.pdf"* http://www.ara.bme.hu/oktatas/tantargy/NEPTUN/BMEGEATAM05/2009-2010II/ea_lecture/Meshing_CompressibleModels.pdf

[11] Ingram, D.M., Gao, F., Causon, D.M., Mingham, C.G., Troch, P. *Numerical investigations of wave overtopping at coastal structures*. Coastal Engineering. 2009. 56, 190-202

[12] Lafaurie, B., Nardone, C., Scardovelli, R., Zaleski, S., Zanetti, G. *Modelling merging and fragmentation in multiphase flows with SURFER.* Journal of Computational Physics. 1994.113, 134–147.

[13] Ubbink, O., Issa, R. *A method for capturing sharp fluid interfaces on arbitrary meshes.* Journal of Computational Physics. 1999. 153, 26–50.

[14] Troch, P., Li, T., De Rouke, J., Ingram, D. *Wave interactionwith a sea dike using a VOF finite-volume method*. In: Chung, J., Prevosto, M.,Mizutani, N., Kim, G., Grilli, S. (Eds.), Proceedings of the13th International Offshore and Polar EngineeringConference, vol.3. International Society of Offshore and Polar Engineering. 2003. 325–332.

[15] Hirt, C.W., Amsden, A.A., Cook, J.L. *An Arbitrary Lagrangian-Eulerian Computing Method for all Flow Speeds.* Journal of Computational Physics. 1997. 135,203-216

[16] Watanabe,Y., Saruwatari, A., Ingram, D.M. *Free-surface flows under impacting droplets* .Journal of Computational Physics.2008. 227:2344-2365

[17] Norgaliev, R.R., Dinh,T.N., Theofanous, T.G. *Sharp Treatment of Surface Tension and Viscous Stresses in Multifluid Dynamics*. Proceedings of the 17th American Institute of Aeronautics and Astronautics Computational Fluid Dynamics Conference. 2005.(Paper 2005-5349).1-20

[18] Roberto F. Ausas, Enzo A. Dari and Gustavo C. Buscaglia. *A geometric mass preserving redistancing scheme for level set function*. International Journal for Numerical Methods in Fluids. Published online : 9 FEB 2010, DOI: 10.1002/fld.2227
[19] Fedkiw R., Aslam, T., Merriman, B., and Osher, S., *A Non-Oscillatory Eulerian Approach to Interfaces in Multimaterial Flows (The Ghost Fluid Method)*, J. Computational Physics, vol. 152, n. 2, 457-492 (1999).
www.it.uu.se/research/publications/reports/2009-026/2009-026-nc.pdf
[20] Kang M., Fedkiw R., Liu X-D., *A boundary condition capturing method for multiphase incompressible flow, J. Sci. Comput.* **15**, 323 (2000).
[21] Sethian J.A,. *Level Set Methods and Fast Marching Methods.* Publisher: Cambridge University Press. 2000: ISBN 0 521 64557 3.
[22] Liu G.R, Xu G.X. *A gradient smoothing method (GSM) for fluid dynamics problems*. International Journal for Numerical Methods in Fluids.2008; **58**:1101-1133
[23] Barth TJ, Jespersen DC. *The design and application of upwind schemes on unstructured grids*. AIAA Paper 0366,1989.
[24] Liu G.R, Xu G.X. *A gradient smoothing method (GSM) for fluid dynamics problems*. International Journal for Numerical Methods in Fluids.2008; **58**:1101-1133
[25] Ii S., Shimuta M., Xiao F. *A 4th-order and single-cell-based advection scheme on unstructured grids using multi-moments*. Computer Physics Communications. 2005;**173**:17–33

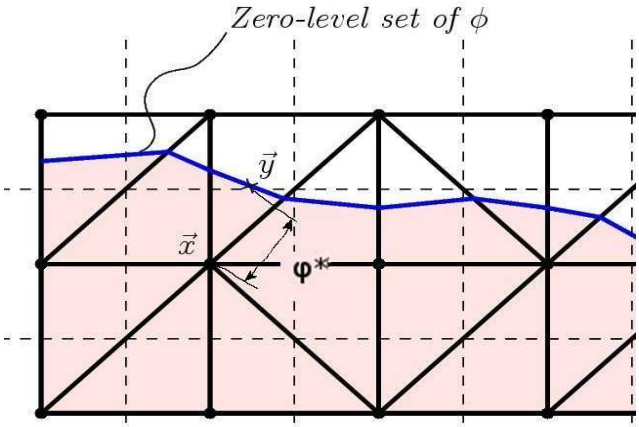


Figure 1: The definition of the signed distance function φ* (Image based on reference [15])

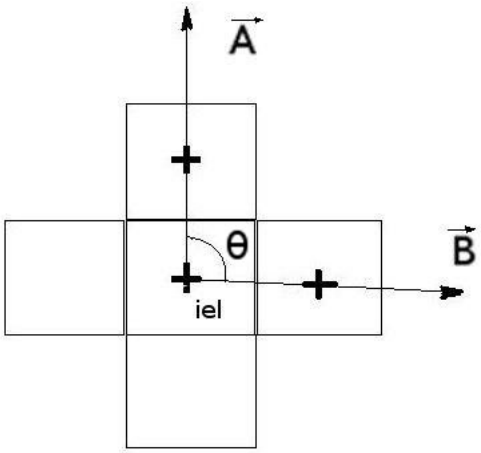


Figure 2: Sorting neighbour cells surrounding centre cell "iel" with attribute θ angle between vectors $\vec{A}$ and $\vec{B}$ which pass though the centre of these neighbour cells.

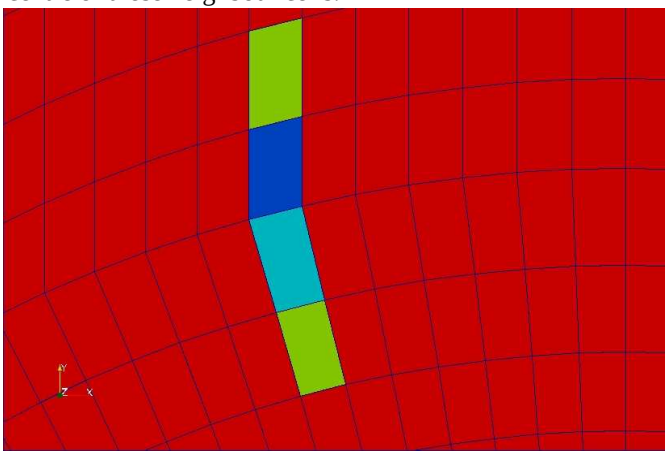
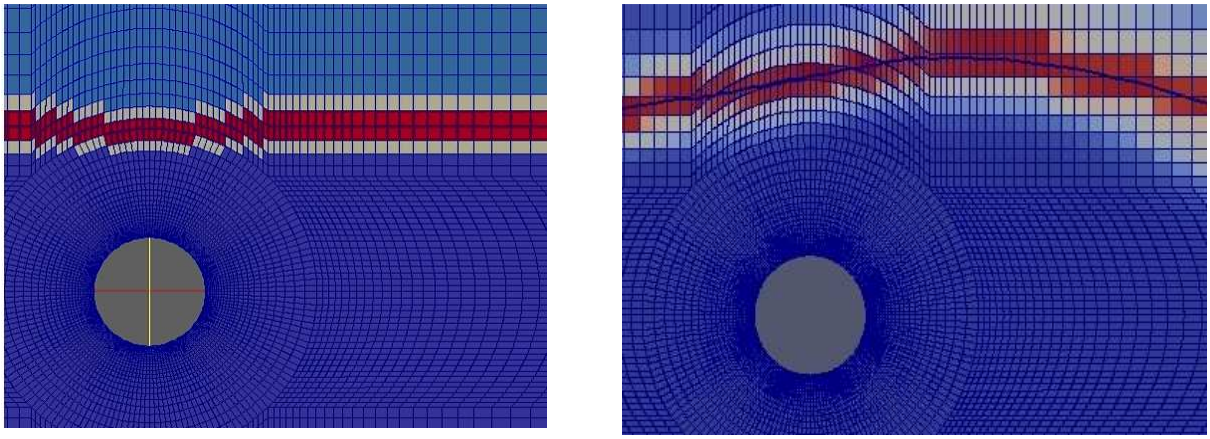


Figure 3: Schematic drawing of the finite volume discretization cells, (shown in dotted lines), and the associated triangulation, (shown in black lines lines), for the re-distancing algorithm [18]



(a) At time t = 0 (b) t >0

Figure 4: A narrow band evolution with time without GFM involved: The dark red cells form the set of primary cells, with the isocontour between the two horizontal bands of dark red cells. The light red being the halo cells with the blue line being the isocontour tracking a streamline at t = 0 and t > 0.

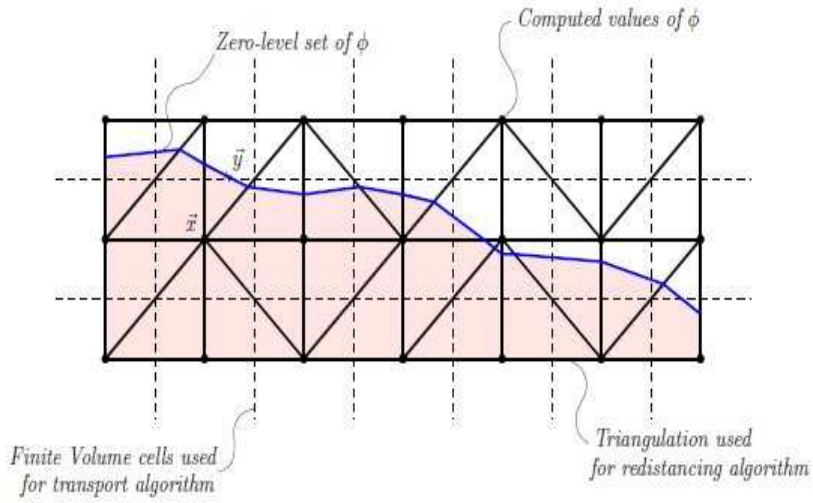
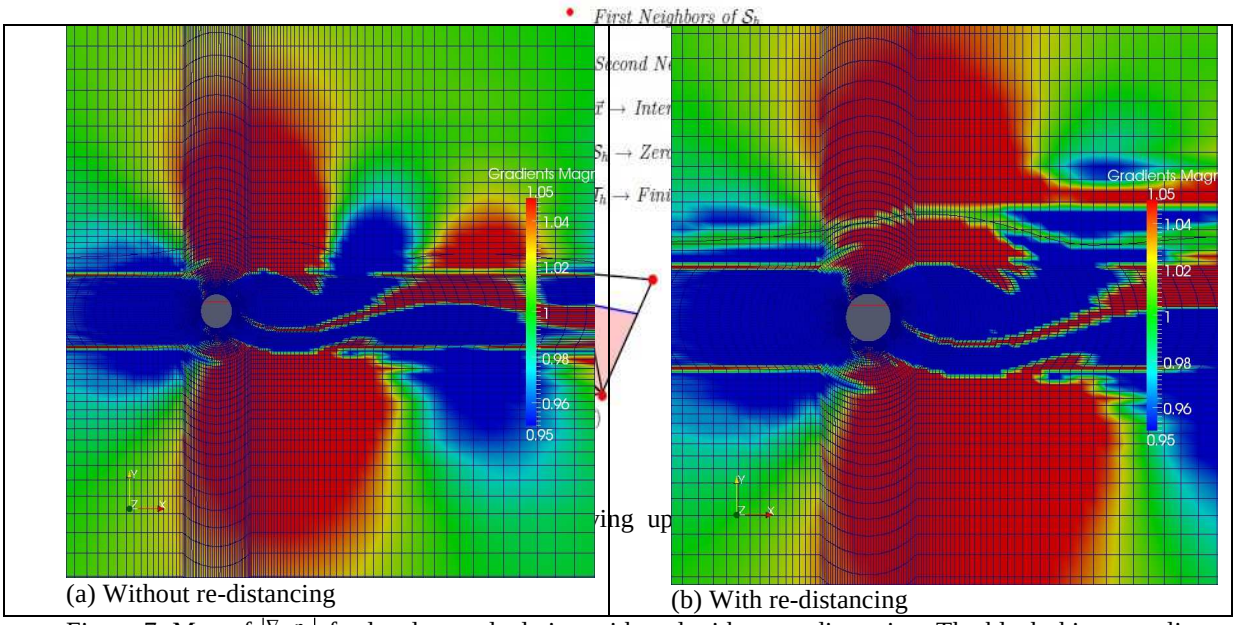


Figure 5: Schematic drawing of the finite volume discretization cells, (shown in dotted lines), and the associated triangulation, (shown in black lines lines), for the re-distancing algorithm [18]



(a) Without re-distancing (b) With re-distancing

Figure 7: Map of $|\nabla \varphi|$ for level set calculation with and without re-distancing. The black thin wavy lines shown in (a) and (b) are the iso-contours at time step 400. The colour range on each graph is from $|\nabla \varphi| = 0.9$ to $|\nabla \varphi| = 1.05$.

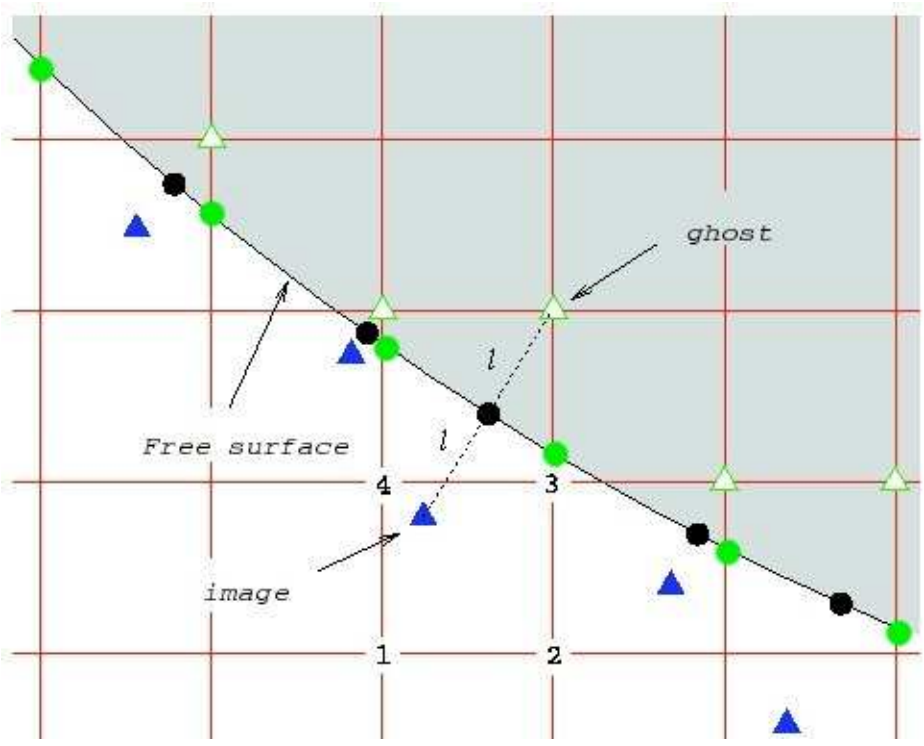


Figure 8. Ghost-cell scheme. (Black circle • represents the free-surface boundary point while the white-green and blue triangles represent the ghost and image points respectively.)

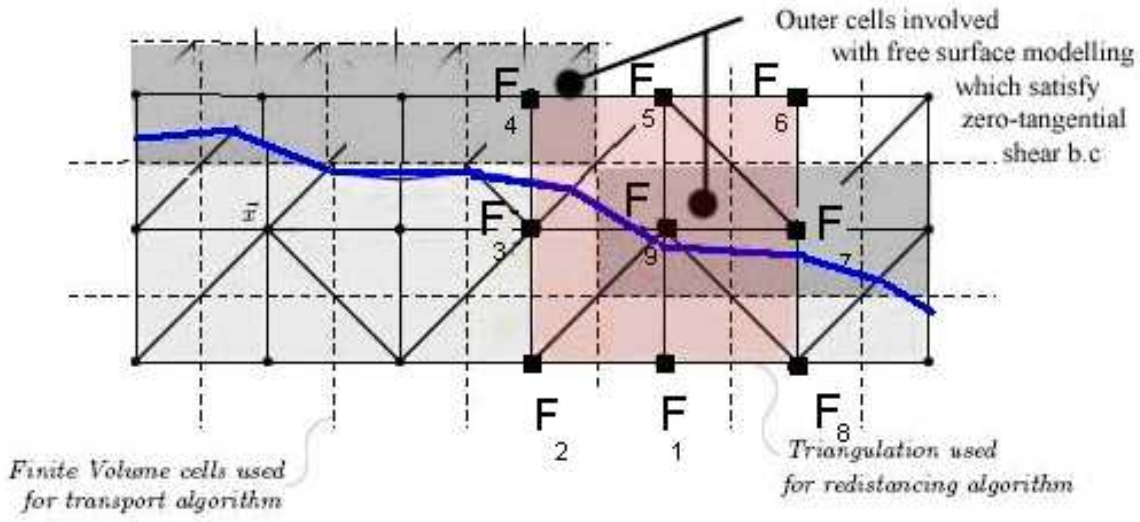


Figure 9: Nine cell stencil indicated in pink for the calculation of $\nabla\varphi$ using a Gradient Smoothing Method at point F_9 on the free surface shown as a blue line.

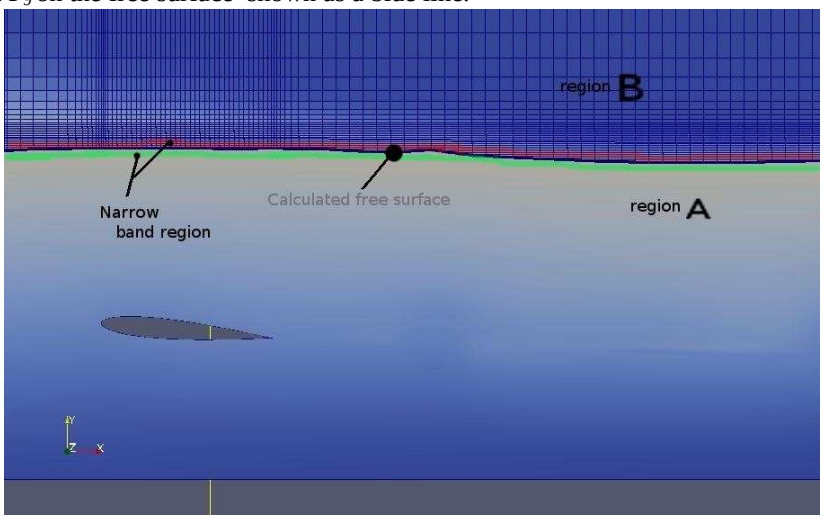


Figure 10: Regions in the water/air flow field: - Region A (water), narrow band region (red and green lines), calculated free surface (black line between the read and green lines) and region B (air).

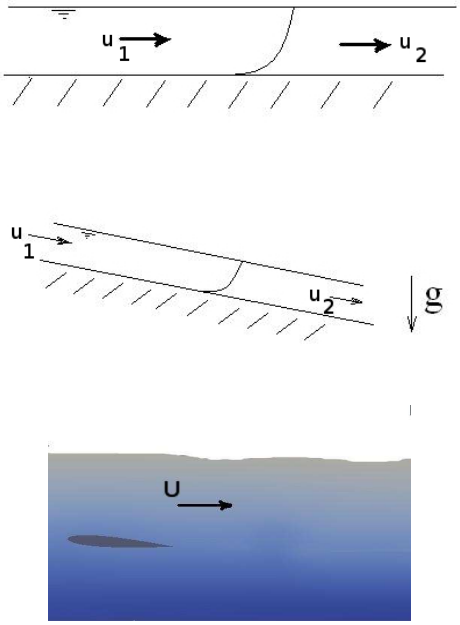


Figure 11 Details of verifications tests for D3