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: ReDAPT

Title: Executive Summary Version 1

Abstract:

This document describes progress made on the Computational Fluid Dynamics models developed to simulate a full marine-current turbine to include turbulence and waves. The work documented describes the results early 3-d simulations of an idealised rotor geometry incorporating an imposed velocity by rotation of a single cylindrical mesh and turbulence models that are based on Reynolds-Averaged Navier–Stokes (RANS) equations. The report concludes by outlining the next steps required in the modelling to improve efficiency and establishing feasibility prior to conducting large scale CFD simulations for a full turbine.

Context:

One of the key developments of the marine energy industry in the UK is the demonstration of near commercial scale devices in real sea conditions and the collection of performance and environmental data to inform permitting and licensing processes. The ETI's ReDAPT (Reliable Data Acquisition Platform for Tidal) project saw an innovative 1MW buoyant tidal generator installed at the European Marine Energy Centre (EMEC) in Orkney in January 2013. With an ETI investment of £12.6m, the project involved Alstom, E.ON, EDF, DNV GL, Plymouth Marine Laboratory (PML), EMEC and the University of Edinburgh. The project demonstrated the performance of the tidal generator in different operational conditions, aiming to increase public and industry confidence in tidal turbine technologies by providing a wide range of environmental impact and performance information, as well as demonstrating a new, reliable turbine design.

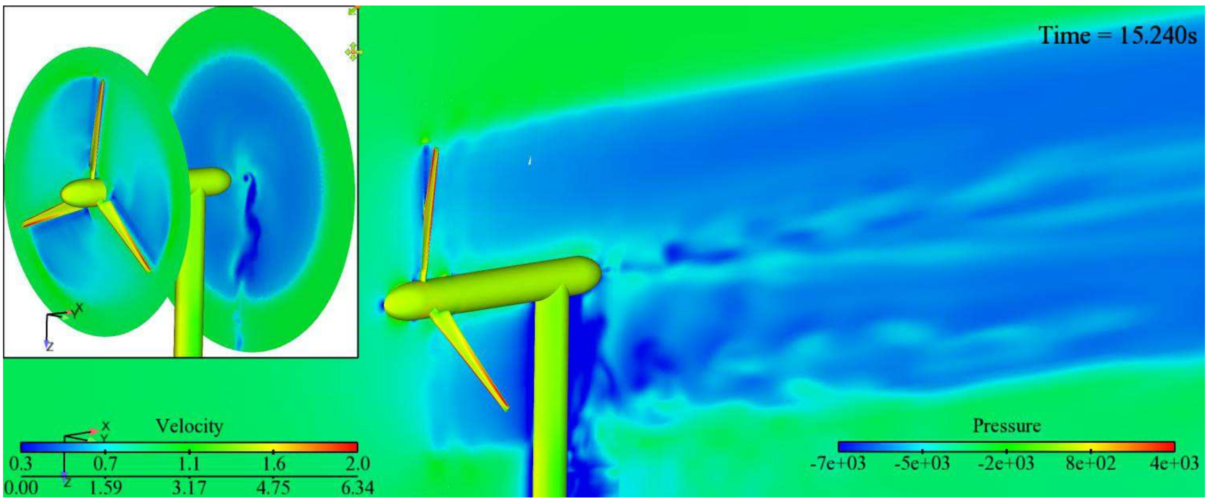
Project	ReDAPT
Deliverable	MD1.1
Responsible author	Dr David Apsley
Circulation	University of Manchester, EDF
To be approved by	Prof. Dominique Laurence
Date	30 March 2011 (revised 29th June and 6th July 2011)
Version	1 (revision 2)

EXECUTIVE SUMMARY

The document describes progress with the application of EDF's open-source CFD solver *Code_Saturne* toward a full marine-current-turbine simulation, including turbulence and waves, as part of the wider ReDAPT project. The CFD component employs two full-time research staff at the University of Manchester: one PhD student (currently 18 months through) and one PDRA. Following the resignation of the original PDRA to take up a lectureship elsewhere, the post has recently (May 2011) been taken by Dr Imran Afgan. In a short time Dr Afgan has already brought his considerable experience with massively parallel computation to bear on this project.

An idealised geometry has been produced, based on the turbine used for flume experiments by Prof. Bahaj's group at the University of Southampton. Single meshes derived from this have been used with *Code_Saturne* for simulation of an isolated turbine using either rotating reference frame (with Coriolis body forces) or a rotating mesh (ALE method).

To simulate turbine loading and wake behaviour for a rotating device influenced by a non-stationary sea surface, nearby sea bed and complex support structure requires a sliding-mesh capability. No such operational capability currently exists within *Code_Saturne* and it was deemed necessary to develop our own. This is implemented as an internal boundary between non-conforming and translating mesh blocks. The method is shown to operate as intended for laminar flow in 2-d and 3-d geometries and work is in progress, particularly on the pressure-velocity coupling and use of implicit timestepping, to ensure its smooth and efficient operation in turbulent flows and on parallel processors. In the most recent developments (after the original submission of this report) calculations have been successfully undertaken with a new sliding-interface method for a rotating turbine, including the support structure. The interface method performed entirely satisfactorily with a mesh of 4 million cells on 2048 processors, albeit for laminar flow only.



1. INTRODUCTION

1.1 Scope of this Document

This report explains the work undertaken to complete milestone MD1.1; specifically:

- 3-d simulation of a rotor, performed with:
  - idealised geometry (blade set and nacelle, but no support mast);
  - imposed velocity by rotation of a single cylindrical mesh;
  - RANS turbulence modelling;
  - no waves.
- Identify the development needed to implement an internal sliding mesh.

1.2 Specific Tasks Associated With This Project

The specific milestones for the CFD work on the ReDAPT project are as follows.

MD1.1 (This report)	Ideal turbine geometry; imposed rotation of a single cylindrical mesh; RANS turbulence; no waves. Report to identify development necessary for sliding mesh.
MD1.2	Ideal turbine geometry; rotation via sliding mesh; LES turbulence; no waves
MD1.3	Real turbine geometry; sliding mesh; RANS turbulence; waves
MD1.4	Real turbine geometry; sliding mesh; LES turbulence; waves Comparison of loads, velocity and near-wake turbulence with field data.

1.3 Staff on the Project

The research staff employed on this project are a PhD student (James McNaughton) and a post-doctoral research associate. The latter post was filled by Yacine Addad from February to September 2010. Following Dr Addad's resignation to take up a lectureship in Abu Dhabi, the position was readvertised and offered to Dr Imran Afgan (who has previously worked with *Code Saturne* for the University of Paris and EDF). Following a delay in getting a work permit, Dr Afgan began work on this project on 6 May 2011. In his short time with us he has already brought considerable expertise on the inner working of *Code_Saturne* and on massively-parallel computing to this project.

James McNaughton started in September 2009 and has PhD supervisors Dr David Apsley and Dr Alistair Revell. Overall responsibility for the University of Manchester's contribution to ReDAPT rests with Prof. Peter Stansby. Other Manchester staff who have contributed informal assistance in the project include Dr Juan Uribe (*Code_Saturne* expertise), Dr Tim Stallard (related PerAWAT project) and Olivier Cozzi (moving-mesh/free-surface methodology).

2. STATUS OF RESEARCH

2.1 Idealised Geometry

The turbine geometry chosen for the initial feasibility study is that of Bahaj et al. at the University of Southampton (Bahaj et al., 2007a,b). This was chosen because the blade geometry and support structure are relatively simple, the flow is well-characterised and there is comprehensive data for the load characteristics in the form of power and thrust coefficients for a range of tip-speed ratios and blade pitch settings. There is no substantive velocity data in the wake nor the pressure distribution along the blades, but some blade-element momentum calculations using the 2-d panel code XFoil have been performed for this geometry by a related group (Batten et al., 2008).

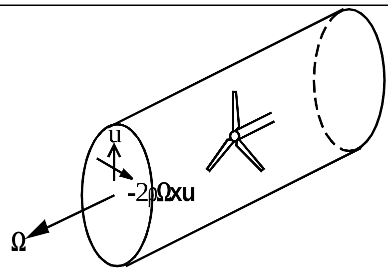
A model geometry and grid topology has been developed using the gambit tool supplied with ANSYS Fluent and the resulting grids exported for use with EDF's open-source CFD solver *Code_Saturne* (Archambeau et al., 2004). The idealised geometry originally consisted only of the turbine blades and the central axis and nacelle alone. (Since the initial draft of this report the complete support structure has been added, and the first flow calculations have shown it to exert considerable influence on the wake.)

The block structure of the mesh enables the turbine flow calculations to be conducted (see Figure 1):

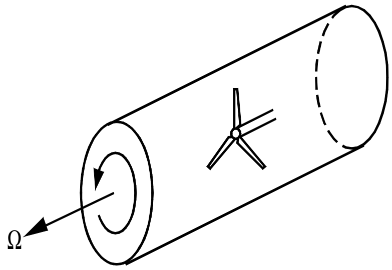
(I) as a stand-alone cylindrical block:
  - (Ia) in the frame of the rotating turbine, using a Coriolis body force;
  - (Ib) in an absolute frame with rotating mesh;

(II) as a cylindrical block rotating within a larger cuboid block, which also contains the support structure and free surface.

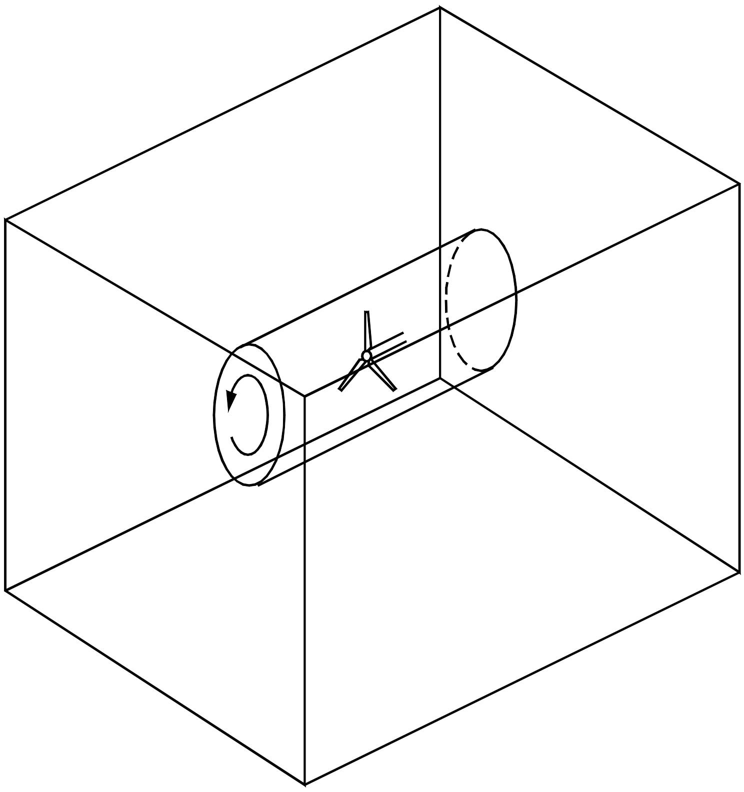
Calculations using methods (Ia,b) are described in subsection 2.3. Neither of these will themselves allow us to compute flow about a rotating marine current turbine with fixed support and interaction with the sea bed and sea surface; rather their purpose is to allow some validation of the techniques developed for (II). A sliding-mesh interface method has been developed and coded to accomplish (II) and some preliminary tests, including a complete turbine/support-structure configuration, are documented in subsection 2.3. The latter permits us to combine a region of the mesh rotating with the turbine to a wave-affected, free-surface-conforming moving-mesh procedure in an outer block.



Ia – single mesh: rotating reference frame, Coriolis body forces



Ib – single mesh: rotating mesh by ALE method



II – cylindrical mesh rotating within a larger stationary mesh.

Figure 1: Mesh configurations for a marine current turbine.

2.2 Single-Mesh Calculations

An isolated turbine, remote from bed, free surface and other structures will experience a uniform free stream.

In the first single-mesh approach (Ia), the CFD calculation is conducted in the reference frame of the turbine and the Coriolis force, which is the apparent force on the observer because of the acceleration of his reference frame, is imposed as a body force (per unit volume):

$$-2\rho\mathbf{\Omega}\wedge\mathbf{u}\,.$$

The other rotation–related body force, namely the centrifugal force

$$-\rho\mathbf{\Omega}\wedge(\mathbf{\Omega}\wedge\mathbf{r})=\rho\Omega^2\mathbf{R}\,,$$

can be written as the gradient of a scalar field, i.e. as $\nabla(\frac{1}{2}\rho\Omega^2 R^2)$, and hence can, at least in principle, be incorporated in a modified pressure. (Note, however, that experience with other codes suggests that, in cases where the "outside world" supplying the boundary conditions is *not* rotating then it is better to include this term explicitly, rather than via a modified pressure force). Additionally, the inflow velocity is modified to become the relative velocity

$$\mathbf{U}\rightarrow\mathbf{U}_\infty-\mathbf{\Omega}\wedge\mathbf{r}\,,$$

where $\mathbf{r}$ is the position vector relative to a point on the axis of rotation. ($\mathbf{R}$ is the part of this normal to the axis of rotation).

In the second single-mesh approach (Ib) velocities are those in an absolute frame, but the mesh itself is rotated. This is a special case of the ALE (Arbitrary Lagrangian-Eulerian) method. The scalar-transport equation for a variable ϕ in a moving control volume V is

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_V \rho\phi\ \mathrm{d}V+\oint_{\partial V}\rho(\mathbf{u}-\mathbf{u}_{grid})\phi\,.\,\mathrm{d}\mathbf{A}+\oint_{\partial V}\mathbf{F}^{(\phi)}\,.\,\mathrm{d}\mathbf{A}=\int_V \rho s\ \mathrm{d}V\,,$$

where $\mathbf{F}^{(\phi)}$ is any non-advective flux and s is the source per unit mass. The only change necessary to a fixed-grid calculation is that the *net* volume flux through a cell face must be modified to account for mesh movement by subtracting

$$\oint_{\partial V}\mathbf{u}_{grid}\,.\,\mathrm{d}\mathbf{A}\,,$$

which is ($1/\Delta t$ times) the volume swept out by a cell face during one timestep. The ALE approach, which is only regularly used for solid tubes vibrating in a fluid, has been adapted to free surfaces by Yacine Addad and Olivier Cozzi (re-activating the early version of Archambeau et al. 1999), and will also be used in conjunction with a free-surface-moving algorithm for the simulation of wave motion. In the latter (essentially the same as that in Apsley and Hu, 2003) the free-surface is tracked by following the motion of control points which are confined to move vertically, regenerating the grid below to retain the same relative mesh sizes in a manner similar to the use of "sigma coordinates", but without the explicit coordinate transformation necessary to use those.

The figures below show calculations of blade pressure distribution and turbulent kinetic energy for single-mesh calculations using the RANS equations and a k-ω turbulence model. Calculations were performed on a parallel computer, confirming the capabilities of *Code_Saturne* for an idealised turbine geometry. These are only very preliminary calculations: in future simulations the side boundaries will need to be moved substantially further from the blades to avoid blockage effects. However, the ability to model this particular geometry and compute the flow with *Code_Saturne* are demonstrated, and one of the assets of the code is its scalability on massively parallel computers. Details and more representative calculations will be given in Report MD1.2.

Note that a detailed grid- and timestep sensitivity analysis was performed by Cozzi (2009) for his moving-mesh simulation of a submerged hydrofoil, but have yet to be undertaken for the present turbine case because of the need to press forward with the development of a sliding-mesh capability.

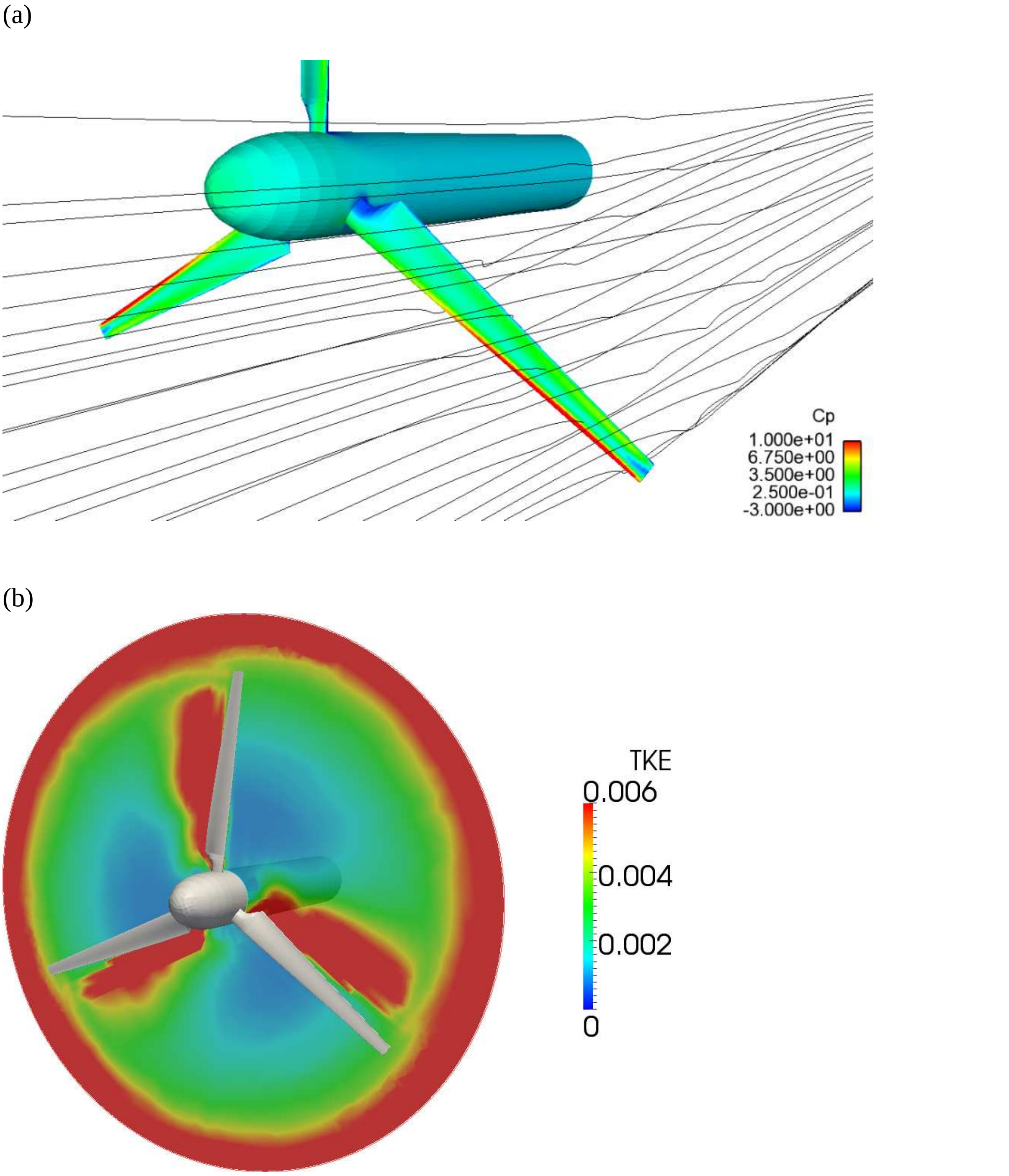


Figure 2: Flow about an isolated marine current turbine; (a) distribution of pressure coefficient on the blades and streamlines (moving-mesh ALE method); (b) turbulent kinetic energy in the wake (rotating reference frame with Coriolis forces).

2.3 Sliding Mesh

The following 3 options were considered.
1. To use a chimera (i.e. overlapping) mesh method.
2. To use the procedure ostensibly available in *Code_Saturne* version 2.0.
3. To develop our own sliding-mesh method via an internal boundary condition.

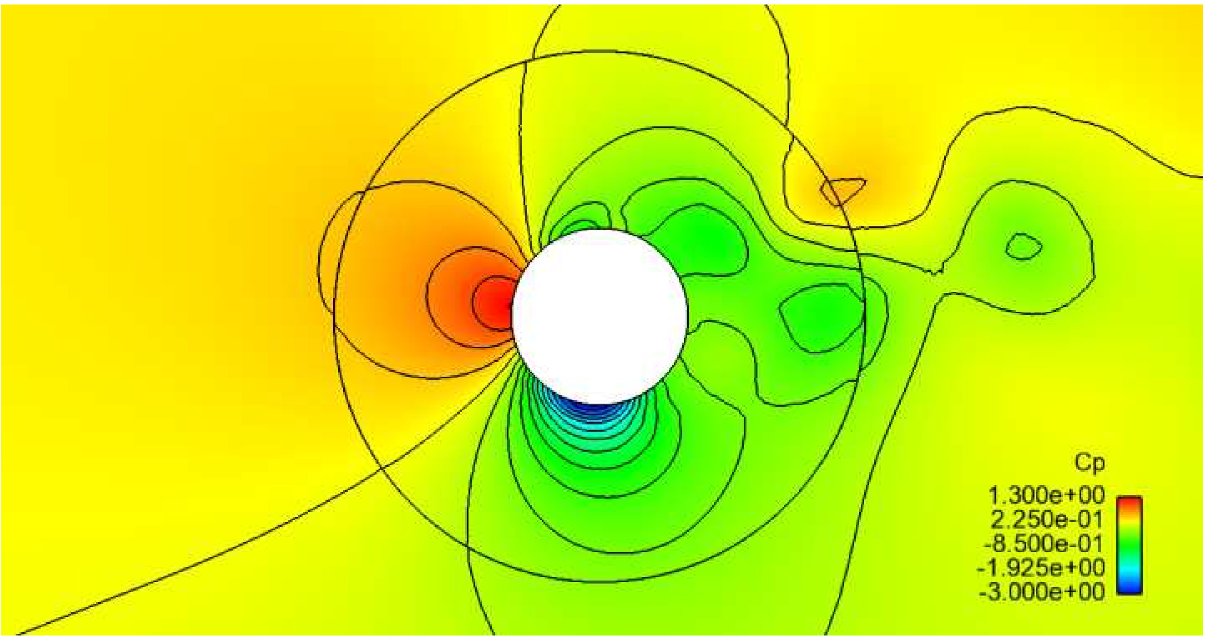
Option 1, using overlapping meshes, was originally suggested as a means of simulating turbine rotation. However, it was not pursued because mobile chimera meshes are not available in official releases of *Code_Saturne* and too much code development would be required. It has been brought to our attention, however, that other researchers with a similar open-source code (*Code_Safari*) from EDF have been using overlapping meshes (Berland et al., 2010), albeit only for a very simple geometry.

We spent a considerable time (circa 6 months) trying to make Option 2 work, both for our geometry and for simpler rotating meshes. The method requires two separate processes of *Code_Saturne* to be running simultaneously and exchanging information; fluxes and pressure need to be transferred across the interface on the basis of "most-overlapping" faces. The two-process regime caused us problems and our calculations with the simplest of rotating geometries showed that pressure was incorrectly transferred across the interface. Option 2, which would have required least in-house code development, was reluctantly dropped.

Option 3 – to develop our own sliding-mesh interface – has, however, shown considerable promise. The methodology was developed first in an in-house structured-mesh code (*STREAM*) and subsequently implemented for fully-unstructured meshes in *Code_Saturne* by the PhD student James McNaughton. The interface is treated as an internal boundary in much the same manner as is routinely done for multi-block structured codes (e.g. Lien et al., 1996) whereby equations are solved separately in adjacent blocks, with boundary conditions at their interface updated explicitly at the end of each inner iteration. The differences here are that the adjacent blocks are non-conforming (and translating), so that (a) there is not perfect flux conservation and (b) some efficient cell-search and interpolation routines have had to be added to *Code_Saturne*. The method is theoretically non-conservative (as, indeed are both the alternatives, Options 1 and 2), but experience with *STREAM* has shown mass losses to be negligible. The method is second-order accurate in space (the same as the underlying advection scheme). The only significant constraints are easily anticipated: that adjacent cells should not slide more than about half a cell width in one timestep and that adjacent cells on either side of the interface should be of comparable depth. Full mathematical details will be given in the next milestone report (MD1.2), but a brief summary as given in the PhD student's latest 3-monthly report can be supplied directly to the industrial partners on request.

Option 3 is being actively taken forward. Figures 3 and 4 below shows the method successfully applied in laminar flow for a simple 2-d rotating cylinder and a 3-d rotating cube respectively. The latter is essentially the same overall mesh topology as is required by ReDAPT – central block spinning within a stationary outer block. In both cases the flow behaviour is in accordance with expectations and, more importantly, the pressure (and other variables) are continuous across the interface. (Any small discrepancies here are an artefact of the post-processing.) In recent developments (after the original submission of this report) calculations have been successfully undertaken with the new interface method for a rotating turbine, including the support structure. The interface method performed satisfactorily with a mesh of 4 million cells on 2048 processors, albeit for laminar flow only. An example of this calculation is shown in Figure 5. Before proceeding to the more computationally-demanding turbulent calculations – both RANS and LES – we propose to spend time on the pressure-velocity solver to make the inner loop of the timestepping scheme fully implicit. However, our initial simulations have shown the new interface routines, and, in particular, the search algorithms for near-interface cells, to perform very well in a parallel environment.

Pressure distribution



Streamlines

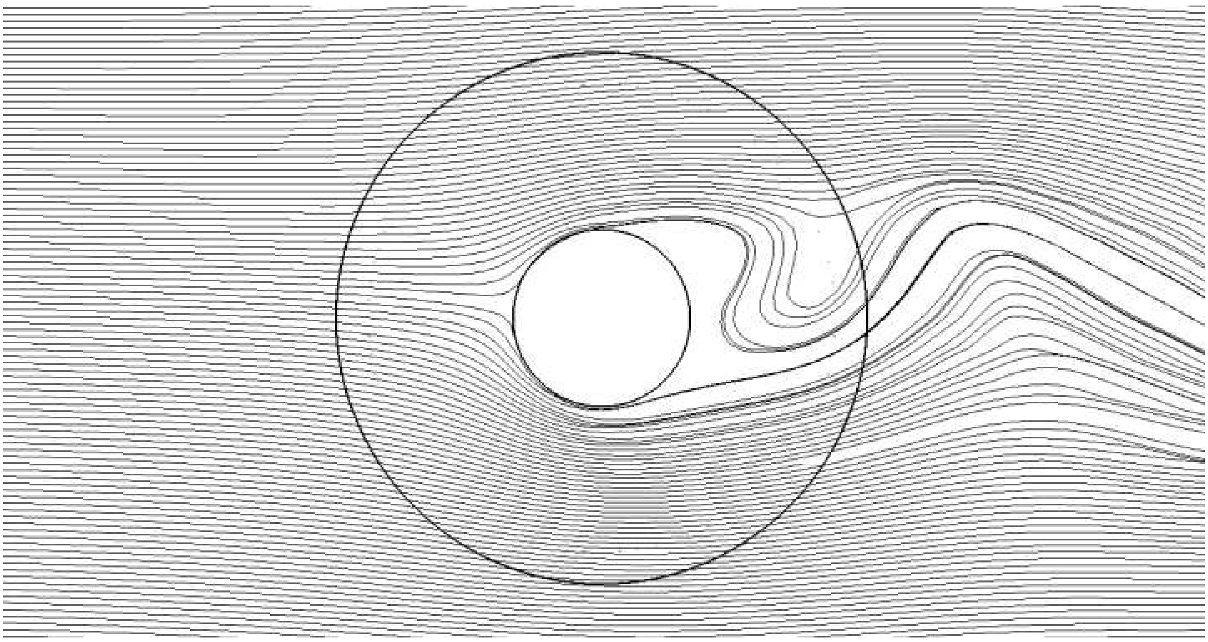
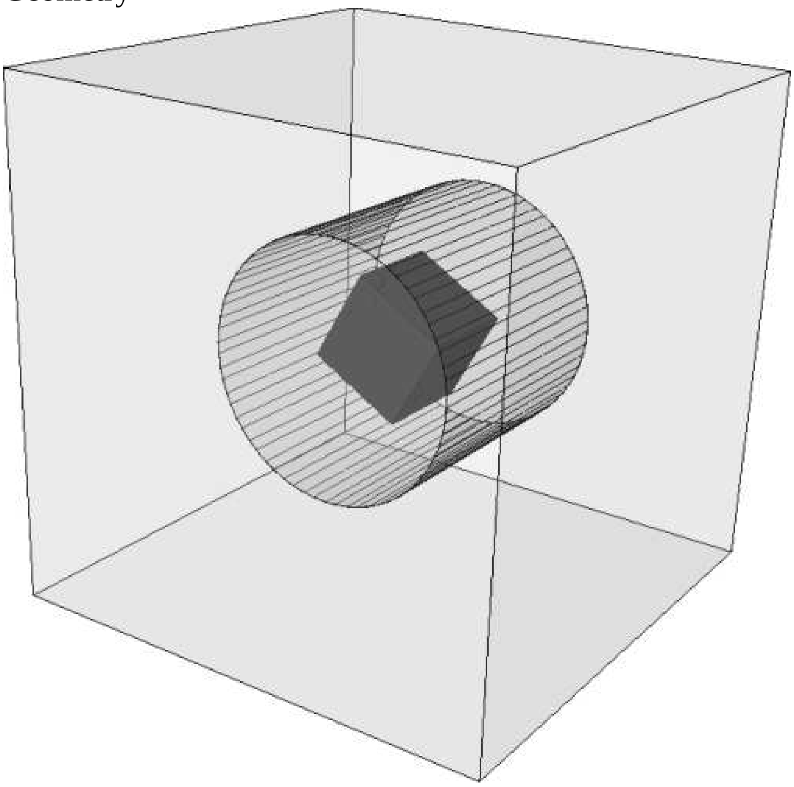


Figure 3: Pressure distribution and streamlines around a 2-d rotating cylinder. The inner cylindrical mesh, whose boundaries are marked, rotates with the cylinder.

Geometry



Instantaneous pressure distribution on a slice through the centre

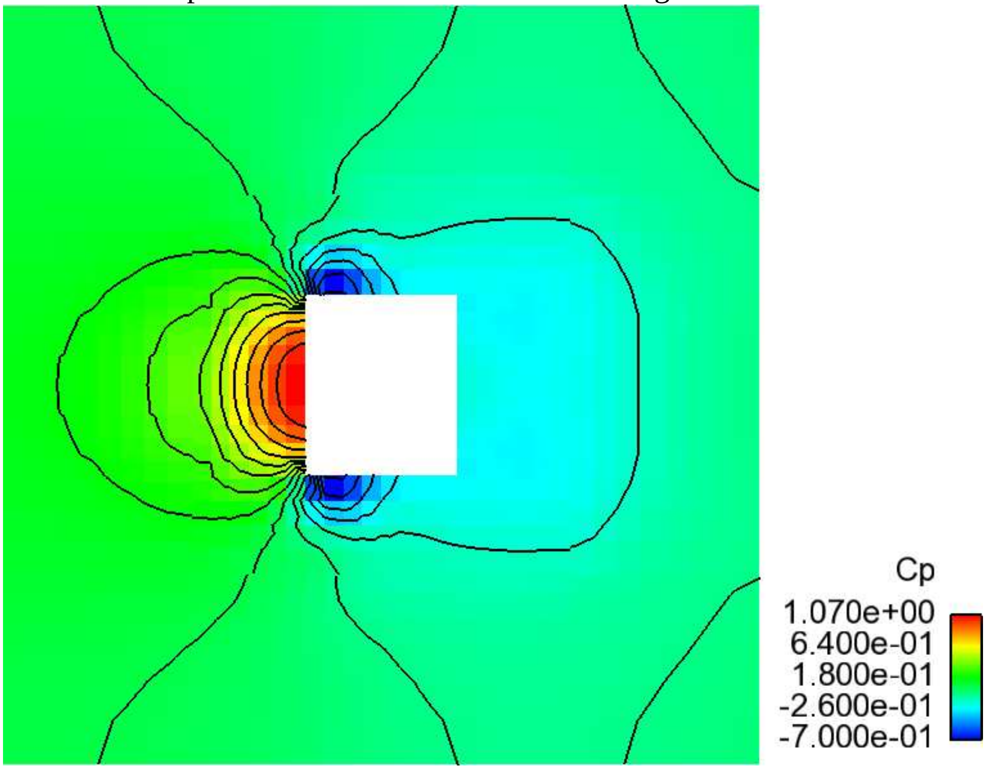


Figure 4: Geometry and pressure distribution for a rotating cuboid. The inner cylindrical mesh rotates with the cube.

(a)

(b)

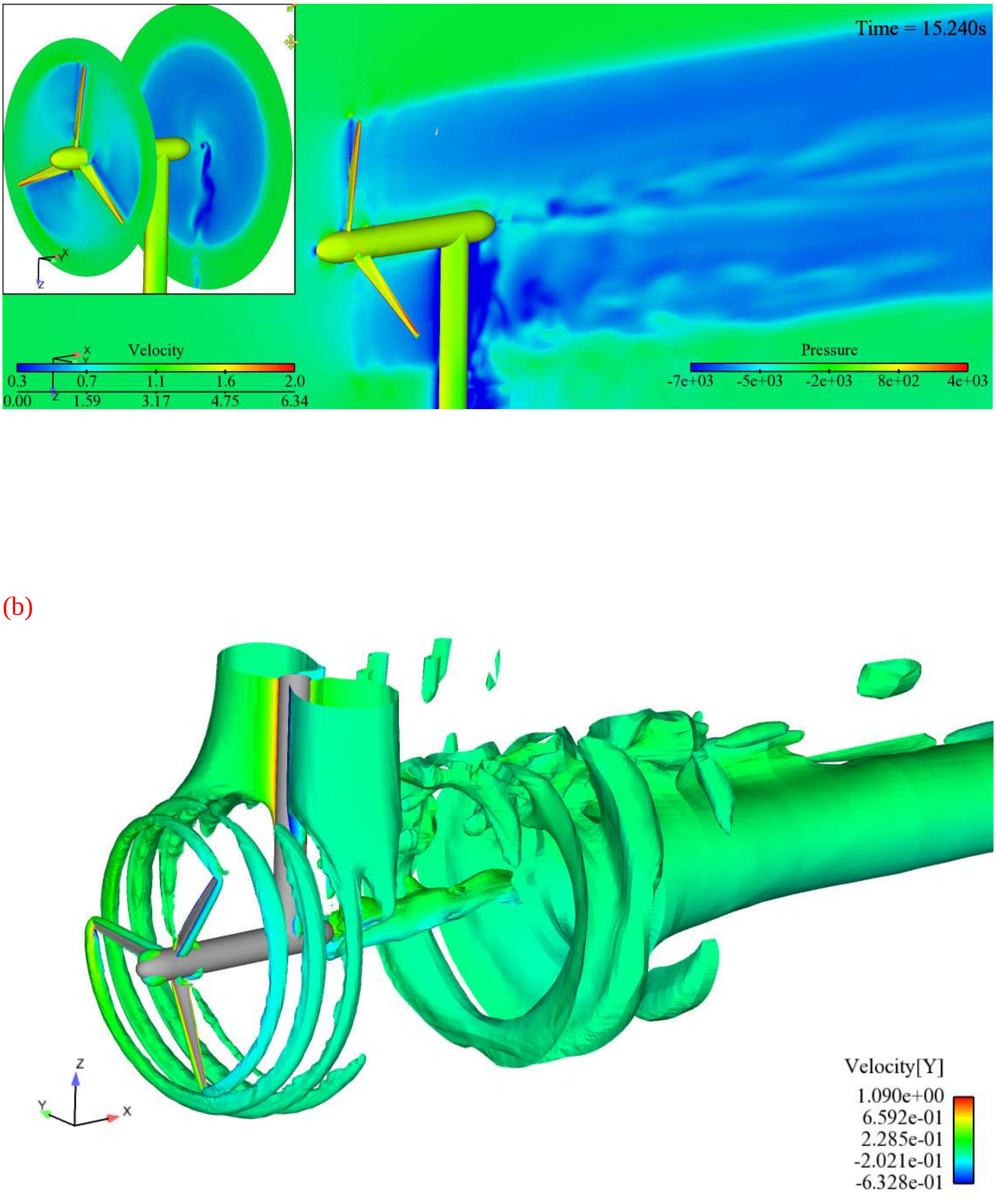


Figure 5: Initial laminar-flow calculations of a rotating turbine, including support structure, using the sliding-mesh interface method; (a) velocity magnitude (in the flow) and pressure (on turbine and support); (b) an isovorticity surface, coloured by spanwise velocity, to illustrate the effects of rotation (blade-tip vortices) and interaction with the mast.

3. NEXT STAGES

To improve efficiency, as well as establishing feasibility, it is desirable to make the following adjustments to *Code_Saturne* before it would be appropriate to conduct large, turbulent, CFD simulations for a full turbine and support.

(i) Nature of the pressure-velocity solver and interaction with the turbulence model.
The inner-iteration loop of the pressure-velocity solver in *Code_Saturne* is a very complex procedure, as in most incompressible flow solvers. In its default setting there is only a single inner-loop iteration, which renders the time-marching algorithm explicit and that, in itself, leads to an excessively small timestep when the mesh is fine. Inspection of the source code shows that multiple inner-loop iterations within a timestep – necessary for implicit timestepping – are possible and have already been envisaged, but some careful re-ordering of code is necessary. New calls to update the turbulence scalars (at least for the k-ε model) within the inner iteration loop have also been implemented since the first issue of this report. To avoid the requirement for prohibitively small timesteps further work must be done here and we are presently working on appropriate inner-loop convergence criteria.

(ii) Parallelisation and testing of the interfacing routines.
Code_Saturne has a very impressive automatic parallelisation routine (METIS) which permits considerable speed-up of calculations when thousands of processors are available (for example, on Daresbury STFC or EDF's Blue Gene facility). However, the user has comparatively little control over the domain decomposition and cannot dictate that all cells on either side of the interface are allocated to the same processor. Memory-management routines must therefore be written to ensure the necessary efficient transfer of information between processors to implement the search algorithms supplying Dirichlet boundary conditions at the interface. (Note: since the first issue of this report and following the first calculations of a full turbine and support geometry we believe that this has now been successfully achieved.) Testing of the interface routines have shown that: (a) with a stationary, non-rotating configuration there is no discernible difference between results with our own interface routines and those obtained by "pasting" adjacent meshes in the usual approach of *Code_Saturne*; (b) with a rotating mesh the position of the interface (a cylinder extending a short distance upstream and downstream and radially to about 10% of a diameter outside the turbine blade tips in Figure 5) is not detectable. The spatial discretisation is formally second-order and centred (not upwinded) at the interface – the same as is commonly done in LES calculations – so we do not anticipate issues with the free passage of eddies here.

(iiii) Sensitivity studies
All CFD work requires comprehensive sensitivity studies with regard to: spatial and temporal discretisation – both step size and numerical scheme; influence of upstream/downstream boundaries; different turbulence models; different codes(!). These issues are not new to the project team and all have been, or will be, examined in depth for simpler geometries (2d rotating cylinder; 3d rotating cube). Only *Code_Saturne* at present offers us the opportunity to perform the sort of calculations required for the full TGL turbine and here spatial and timestep sensitivity, together with a comparison of k-ε, LES and, if possible, a second RANS model will be undertaken.

In our case we need also to examine the treatment of the interface. Detailed experiments will be performed with a 2d rotating cylinder (Re = 200) to compare single, two-part (pasted) and two-part (new-interface) meshes, the last with the inner mesh either stationary or rotating with the cylinder. Comparison between methods is also possible for a 3-d stationary cube in a multi-part mesh, whilst for the rotating cube it is possible, in principle at least, to compare the interface method with calculations on a single mesh but rotating reference frame (Section 2). In all cases, since our interface method is not inherently conservative, the degree of mass-flux inconsistency at the interface must also be checked.

Detailed mesh and timestep convergence studies have been performed with *Code_Saturne* at the University of Manchester for other similar cases (for example, see Cozzi, 2009). Comparable sensitivity studies are presently ongoing for rotating cylinder and rotating cube and are documented in the PhD student's regular progress reports.

Looking further ahead, the other key steps on the way to a full marine-current turbine calculation are the following.

(iv) <u>Full testing of the 3-d free-surface algorithm.</u>
The combination of a free-surface-movement algorithm (via vertically-moving control points) and the ALE moving-mesh procedure are nominally in place following the work of Yacine Addad. However, they remain to be fully tested for a 3-d flow.

(v) Linking the inner rotating turbine mesh with a wave-driven outer block.

(vi) Mesh construction and computations for the actual TGL turbine configuration.

Items (i), (ii) and (iii) are required for milestone MD1.2 and are primarily the work of the PhD student, although the parallel-computing expertise of the PDRA is already proving an asset here. Items (iv), (v) and (vi) are required for the later milestones (MD1.3 and MD1.4), but should not be ignored at this stage. Item (iv) in particular, is an essential task for the PDRA. For item (vi) Dr Stallard has already been discussing with the manufacturers the form in which the geometry data will be supplied, and we have also given some input on desirable data to be obtained from the field experiments.

REFERENCES

Apsley, D.D. and Hu, W., 2003, "CFD simulation of two- and three-dimensional free-surface flow", International Journal for Numerical Methods in Fluids, 42:465–491.

Archambeau, F., Mechitoua, N. and Sakiz, M., 2004, "*Code_Saturne*: a Finite-Volume Code for the Computation of Turbulent Incompressible Flows – Industrial Applications", International Journal on Finite Volumes, 1.

Archambeau, F., Guimet, V. and Bastin, G., "Application du prototype de module ALE du Solveur Commun à des cas de surface libre", Rapport EDF HE-41/99/054/A, 1999.

Bahaj, A.S., Batten, W.M.J. and McCann, G., 2007a, "Experimental verifications of numerical predictions for the hydrodynamic performance of horizontal axis marine current turbines", Renewable Energy, 32, 2479–2490.

Bahaj, A.S., Molland, A.F., Chaplin, J.R. and Batten, W.M.J., 2007b, "Power and thrust measurements of marine current turbines under various hydrodynamic flow conditions in a cavitation tunnel and a towing tank", Renewable Energy, 32, 407–426.

Batten, W.M.J., Bahaj, A.S., Molland, A.F. and Chaplin, J.R., 2008, "The prediction of the hydrodynamic performance of marine current turbines", Renewable Energy, 33, 1085–1096.

Berland, J., Lafon, P., Crouzet, F., Daude, F. and Bailly, C., 2010, "Numerical Insight into Sound Sources of a Rod-Airfoil Flow Configuration Using Direct Noise Calculation", 16th AIAA/CEAS Aeroacoustics Conference.

Cozzi, O., 2009, "Free surface flows in *Code_Saturne*", MPhil Thesis, University of Manchester.

Lien, F.S., Chen, W.L. and Leschziner, M.A., 1996, "A multiblock implementation of a non-orthogonal, collocated finite volume algorithm for complex turbulent flows", International Journal for Numerical Methods in Fluids, 23, 567–588.