

Aerodynamic Simulations using Vertically Integrated Open Source Solutions

Eugene de Villiers¹
Andrew Jackson¹
Francisco Campos¹

¹ ICON, Rofel House, Colet Gardens, London, W14 9DH, United Kingdom, www.iconCFD.com

Abstract

The code transparency and “peer review” nature of Open Source software gives clear advantages over proprietary software. Not only does it provide the user with confidence in the implementation, but it also provides significant cost benefits. In recent years Open Source CFD software has reached a level of maturity which allows the whole CFD process, from mesh generation to post processing, to be completed in a fast, robust, accurate and value-added way.

This paper presents a workflow consisting of a combination of new and existing tools that are available for performing external aerodynamic CFD calculations on a distributed architecture. The tools are developed on top of the OpenFOAM (Open Field Operation and Manipulation) Open Source computational library. A new parallel hex/split-hex automatic mesh generator has been developed to provide fast, robust volume meshing capability. The mesh generator is complimented by a user interface for setting up the meshing and initialisation of solution parameters. Coupling these components with a state-of-the-art Detached Eddy Simulation (DES) solver allows exceptional flow accuracy to be achieved within acceptable time frames. The potentially unlimited scalability of these Open Source tools brings DES simulation into the industrial framework for the first time, as is demonstrated on a full-scale TGV train geometry presented herein.

1. Introduction

A set of tools for carrying out a complete CFD simulation have been developed around the Open Source computational library OpenFOAM, an object oriented C++ toolbox for solving systems of partial differential equations in a finite volume framework.

The main components of the CFD process consist of:

- a user interface for setting-up meshing and solution parameters
- an automated parallel hex/split-hex mesh generator
- a boundary condition and field setup and initialisation utility
- a DES flow solver
- and analysis and post-processing tools.

Due to the parallel nature of the mesh generator, setup utility and solver, the entire process can be executed in a distributed fashion. This allows for the simulation of much larger computational meshes than would be the case if the model had to be aggregated into a single domain.

Efficient solution algorithms provide exceptional accuracy when compared to Reynolds averaged simulation (RAS) approaches without sacrificing stability even when the flow exhibits high Courant numbers. Typically, standard DES can cost an order or more of magnitude more than traditional methods, but optimisation and automation of mesh generation, setup and solution algorithms ensures quick turn-around times.

The outlined methodology is highly customisable due to its Open Source nature, which allows for targeted developments to suit an individual user's needs. The scalability of the approach makes optimisation based product design and other demanding applications a real possibility.

In section 2 the CFD process for performing an external aerodynamics simulation are presented. This includes a user interface for controlling the process, a new parallel mesh generation capability and flow solution and post-processing options in OpenFOAM. In section 3 a simulation of a TGV train geometry using the new mesh generation capability and the DES approach are presented.

2. An Open Source CFD Process for External Aerodynamics

The following subsections give details of the key elements in the Open Source stack for performing simulations on realistic external aerodynamic problems.

2.1 User Interface

Closed source proprietary CFD software has often been seen to have an advantage over Open Source in the interface that is provided between the user and the software. An interface provides a display of dynamically modifiable abstract information for setting up and running the problem and has clear advantages over command line in terms of ease of use and accessibility for non-experts. For this reason a Graphical User Interface (GUI) has been developed to allow easy and fast control of the setup and simulation process. A screen-shot of the GUI is shown in Figure 1.

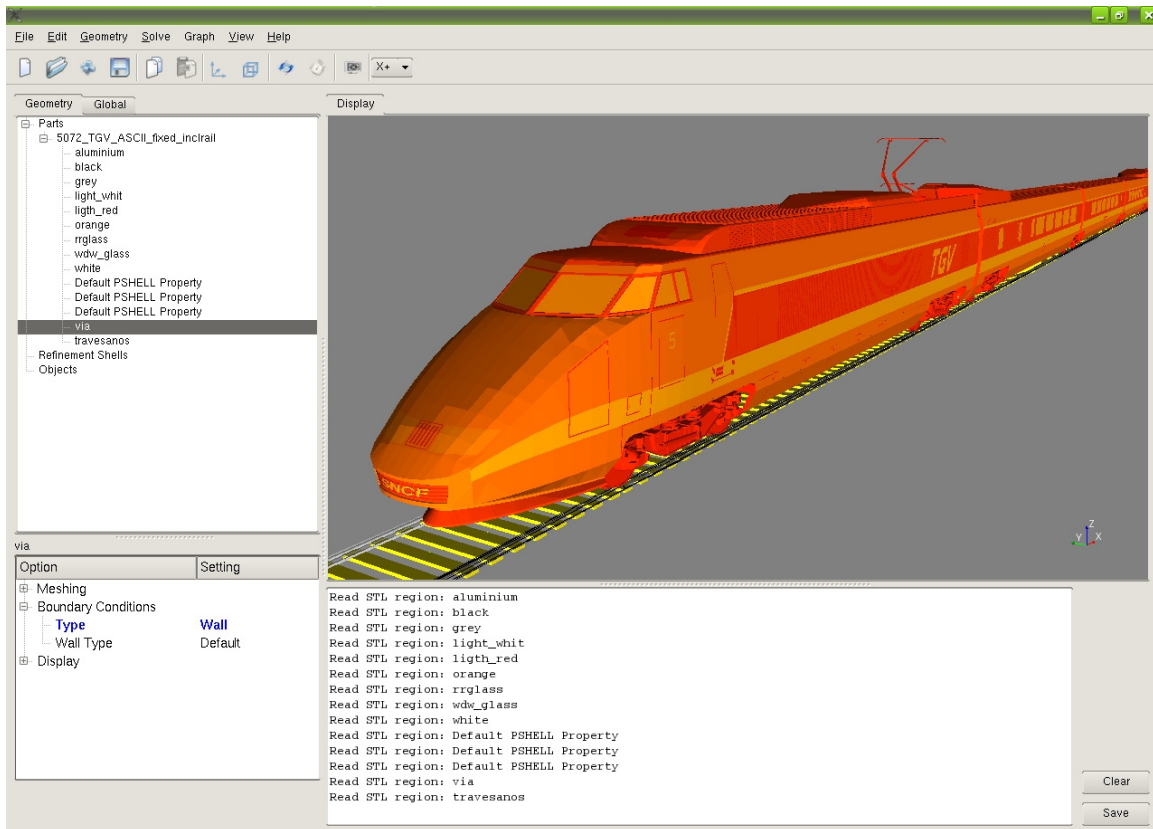


Figure 1 Screen-shot of the GUI used for setting up the meshing parameters, setting boundary conditions and initialisation of the flow-field

The GUI allows the geometry to be imported, viewed and manipulated. The setting of meshing parameters, boundary conditions and flow initialisation methods can also be performed from within the GUI, allowing fast and more importantly error free set-up of cases.

2.2 Mesh Generation

With the need to model ever more complex geometries, the rapid generation of high quality meshes has become a critical stage in the CFD design process. A new automatic parallel mesh generator has been developed inside the OpenFOAM framework. The meshing has been tailored to produce high quality meshes for viscous flow modelling. The key features of the new mesh generation tool are the following:

- Mesh generated in parallel.
- Fully automatic. Only requires a closed geometry. It is not necessary to perform any cleaning up of the CAD.

- Boundary Layer mesh generation on a patch-wise basis.
- Local patch-wise surface (including curvature) refinement and volume refinement away from the surface.
- Surface features in the geometry automatically preserved in the mesh.
- A guarantee that the generated mesh satisfies user defined mesh quality criteria.
- Code transparency

The mesh generation stage has been designed to keep any user interaction in the process to a minimum through the use of sensible defaults and intelligent algorithms. Using the newly developed tools, it is possible to go from CAD to a mesh with tens of millions of cells in hours rather than days. In order to achieve accurate flow solutions, it is essential that the mesh generated can satisfy a set of mesh quality checks. At all stages in the mesh generation process, face and cell based mesh quality is monitored to ensure that the final mesh that is generated satisfies a set of user defined cell and face based quality criteria. Currently the mesh quality checks performed are for orthogonality, face pyramid volumes, face areas, face skewness, face interpolation weights, cell volume ratio, face twist and cell determinant. If at any stage mesh quality cannot be guaranteed the mesh reverts back locally to a previously valid state. The various stages of the mesh generation process are outlined in the following sub-sections.

2.2.1 Stage 1 - Castellated Mesh Generation

In the first stage of the generation an initial uniform Cartesian base mesh is generated extending throughout the solution domain. This Cartesian mesh is refined based on user defined patch-wise surface refinement levels. Additional surface refinement can be performed based on the local surface curvature. As well as surface refinement, volume refinement strategies are available to refine the mesh away from the surface, where mesh resolution is required e.g. wake regions. Figure 2 shows iso-surfaces which have been created to allow refinement of the mesh away from the surface.

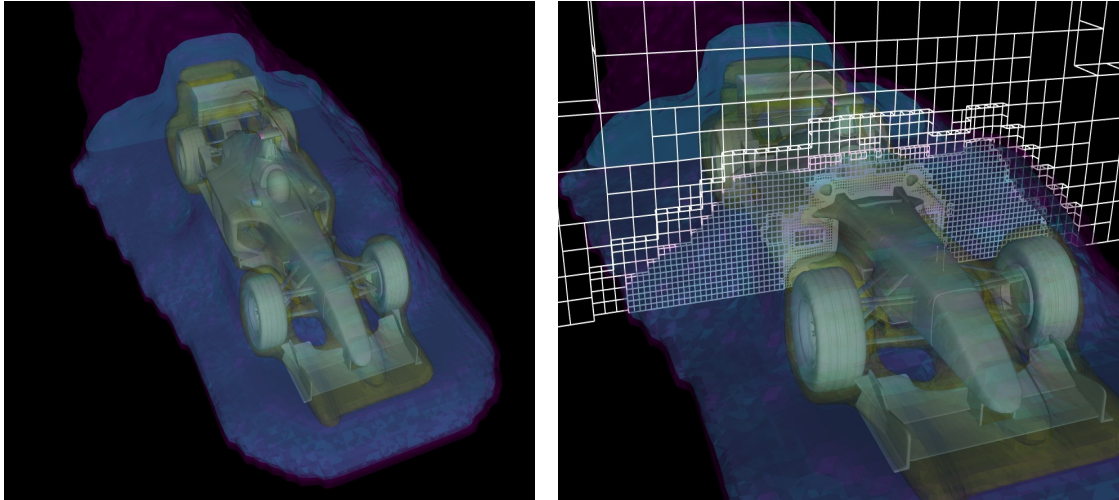


Figure 2 Iso-surfaces (left) used to control volume based refinement. Cut through the mesh (right) showing the volume refinement away from the surface

Once the refinement phase is complete, an intersection check is performed to determine which parts of the mesh should be discarded. This is accomplished by first determining if there is a surface between neighbouring cell centres using an octree-based search. If a surface is detected, the mesh between the neighbouring cells is locally decoupled. As a final stage, a topological walk is performed to agglomerate all connected elements with the remainder being deleted from the mesh. A surface representation of such an intermediate castellated mesh on a generic F1 example test case is shown in Figure 3

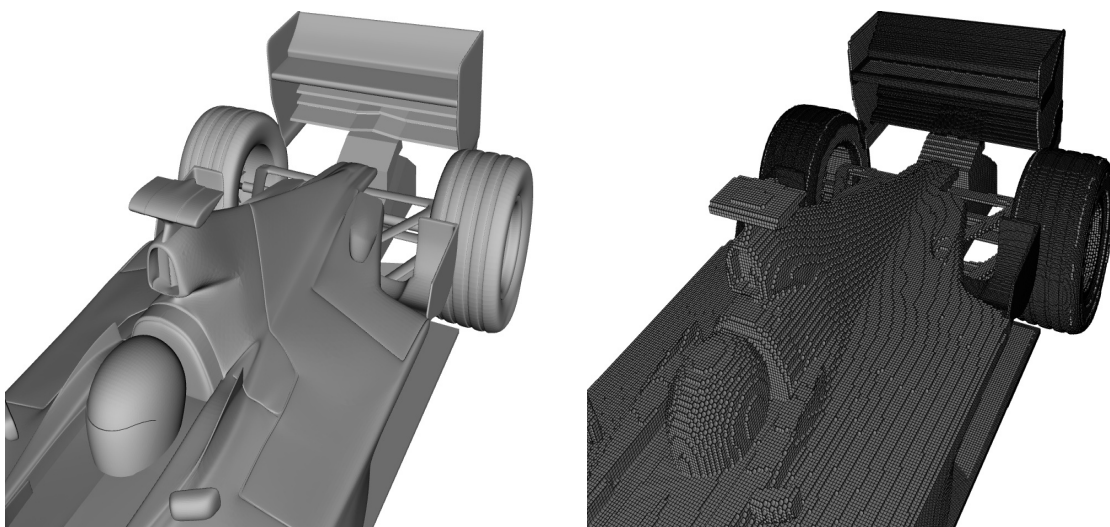


Figure 3 Surface geometry (left) and castellated surface mesh (right)

2.2.2 Stage 2 - Boundary Recovered Mesh

In the second stage of the mesh generation process the surface geometry is recovered. The surface faces of the Castellated mesh, generated at the end of stage 1, are moved (snapped) to the surface. By constraining the boundary faces to be planar during the process, the feature edges present in the geometry are preserved in the surface of the mesh. As an additional step, co-planar boundary faces from the same cell are merged provided this leads to an improvement in the overall mesh quality. The surface of the mesh after these snapping and merging operations is shown in Figure 4.

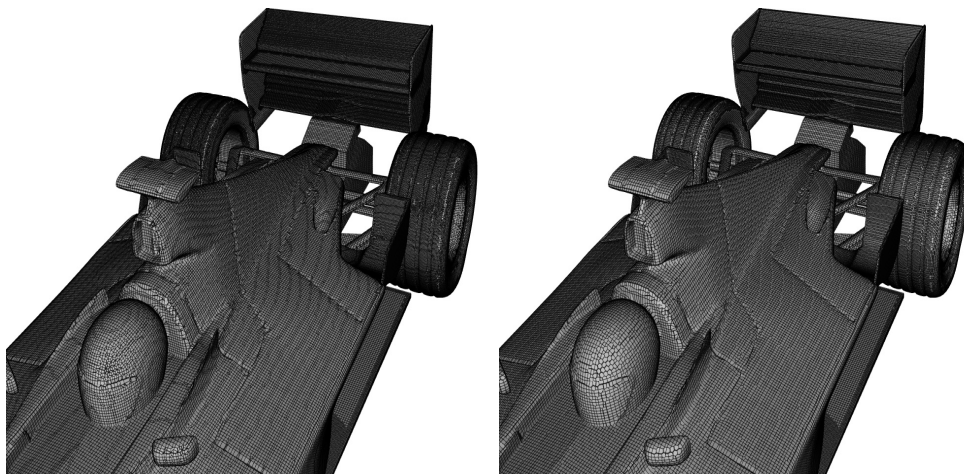


Figure 4 Boundary recovered surface mesh (left) and surface mesh after merging of faces (right)

At every stage during the mesh generation process cell and face based mesh quality are monitored; if a step leads to mesh elements that do not satisfy some user defined quality criteria the mesh is moved back to a previously generated valid state.

2.2.3 Stage 3 - Boundary Layer Addition

With the need for improved viscous flow modelling, an optional final stage in the mesh generation process is to add layers of cells next to the surface. This can be performed on a region-wise basis. The base mesh, generated after stage 2, is projected off the surface and a layer of cells is generated in the void that is created. As in the previous stages of the mesh generation process, if layer addition results in any of the mesh quality criteria being violated, surface layers are not generated in this region.

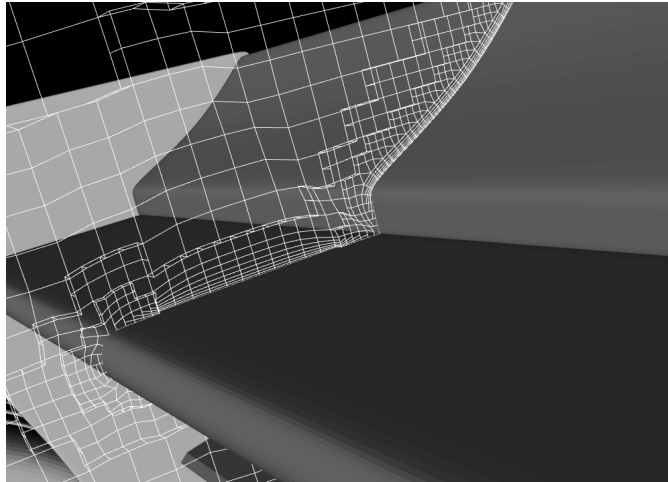


Figure 5 Layer mesh generated on rear wing of F1 car

Figure 5 shows the generated surface layers on the rear wing of a F1 test case. In this figure the layer is terminated in concave and convex regions where mesh quality in the layer mesh cannot be maintained.

Currently there are two layer termination strategies that can be employed; these two approaches are shown in Figure 6. In the first the layer is terminated with a single cell and in the second approach the final terminated cell is merged with the cell directly above. The second approach has benefits when using wall functions by reducing jumps in first cell height in regions where the layer terminates. Termination tends to introduce errors due to the inaccuracy of linear assumptions in the logarithmic layer and should be confined to regions where the equilibrium assumptions do not hold (e.g. sharp corners).

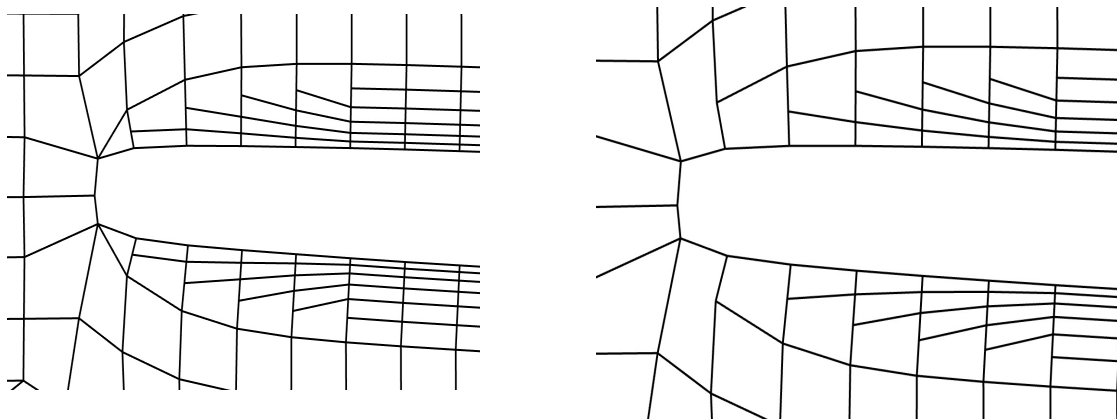


Figure 6 Details of two layer termination strategies employed

2.3 Flow Initialisation and Flow Modelling Using DES

Both setting up the boundary condition and initialisation of the flow field can be performed using the GUI described previously. Most commonly used boundary conditions are supported and flow variables can be initialised using boundary smoothed potential flow, fixed values or length scale and turbulent energy factors in the case of turbulent quantities.

OpenFOAM contains a number of flow solvers and turbulence models for performing CFD on different flow regimes. The emphasis of the methodology described in this paper is on incompressible Detached Eddy Simulation employing a modified hybrid formulation of the Spalart-Allmaras SGS model. Although, DES is considerably more computationally demanding than equivalent Reynolds Averaged methods, the approach is generally more accurate due to the fact that it directly calculates the larger scales of turbulent motion. This improved accuracy is especially important in the wake of bluff body flow as are commonly encountered in aerodynamic applications.

The standard DES solver in OpenFOAM was further modified to improve the stability and range of applicability of operation. Deferred correction and the implementation of non-oscillatory low diffusivity convection schemes have greatly enhanced the quality and stability of the solver. The addition of CFL-adaptive numerical schemes which maintain overall accuracy despite high local Courant numbers and improved parallel communications scheduling and compression have significantly reduced the computational cost of a typical DES simulation.

2.4 Post Processing

Within OpenFOAM there is support for a number of third party post-processing packages. Currently the flow solution can be exported to Fieldview, Ensignt and Paraview formats. Paraview is an Open Source visualisation package which is capable of being run in parallel on distributed and shared memory machines, allowing all CPU intensive stages in the CFD process to be performed in a distributed manner.

3. DES Simulation on a 5 Car TGV (Paris Sud-Est) Train

A realistic simulation using the DES method has been performed on a TGV train geometry. A mesh has been generated using the mesh generation procedure discussed in Section 2.2. The mesh contains a total of 7.8 million cells and was generated in 137 minutes on 8 processors. Details of the parallel partitioned surface mesh are shown in Figure 7.

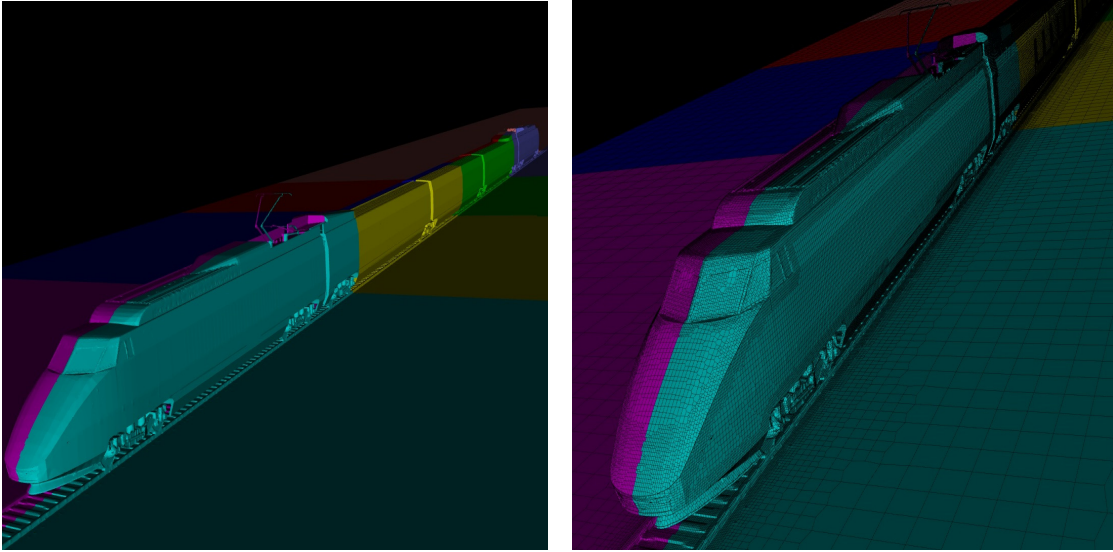
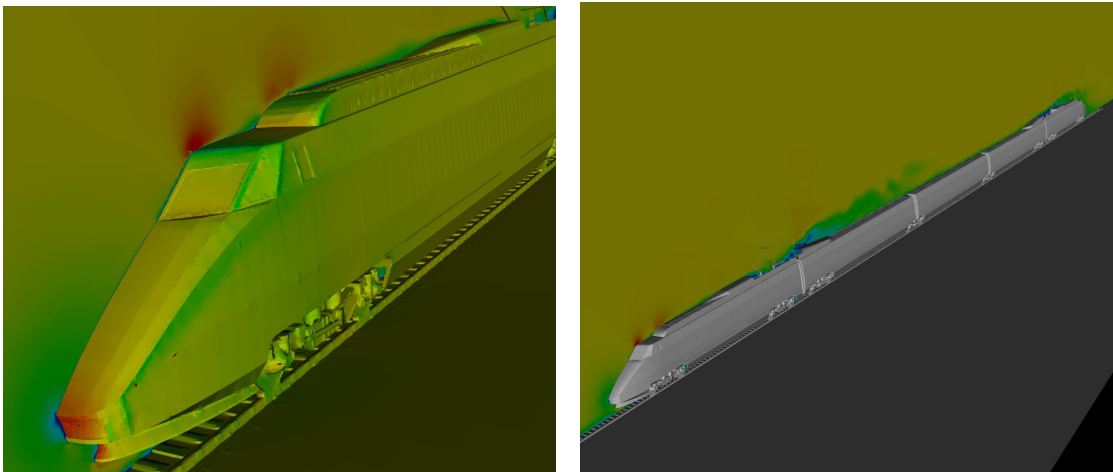


Figure 7 Details of surface mesh generated on TGV for an 8 processor run, colour shaded by processor id

A DES flow solution has been performed on the TGV at its maximum operating speed of 300 Km/hr. In order to speed up convergence a flow solution is first obtained on a coarse mesh and then mapped onto the fine mesh to provide a realistic starting solution for the simulation. The coarse mesh generation, solution and mapping were automated and performed in parallel. A snapshot of the instantaneous flow field obtained during the simulation on the location of the symmetry plane is shown in Figure 8. By directly resolving important turbulent scales, the DES approach not only provides superior accuracy, but also allows the study of flow dynamics and other unsteady phenomena such as flow induced buffeting and aerodynamically generated noise.



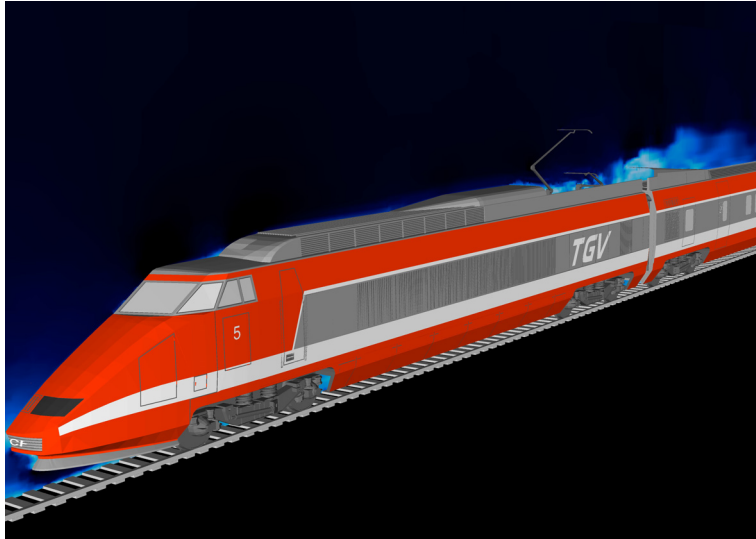


Figure 8 Shaded Instantaneous velocities from a DES simulation of TGV train at 300 Km/hr

4. Conclusions

A highly customised and vertically integrated Open Source application stack for the simulation of external aerodynamic flows has been presented. All stages of the outlined CFD process support distributed computation. Taken in combination with the optimisations performed to improve computational efficiency, this allows highly accurate physical modelling on very large data sets, limited only by hardware resource. The same scalability means that design optimisation becomes an affordable and thus realistic prospect for use in virtual proto-typing.

The Open Source nature of the tools and the generality of the OpenFOAM libraries also allows for a large degree of customisation, so that the same approach presented here could easily be transferred to different application areas like process flows, heating cooling and ventilation and computational aero-acoustics.

5. Acknowledgements

The authors would like thank OpenCFD Ltd. for their collaboration in developing the OpenFOAM meshing and solver utilities discussed in this paper.

For further information regarding this paper, please contact Mr Simon Weston,
s.weston@icon-cg.co.uk



www.iconCFD.com

Icon is an independent provider of confidential CAE simulation services and has had a long association with industry by delivering CFD related projects at both OEM and supplier level since 1992. Close links with academic institutions, and continuous involvement with clients in various industries, has allowed Icon to establish a wide base of knowledge and offer technology transfer to its clients.