9 Method Used and Highlight Results achieved with the *code_Saturne* Software at EDF

D. Laurence, S. Benhamadouche, and J. Uribe, Electricite de France

Abstract

The numerical method used in EDF's unstructured finite volume code is described, with an emphasis on boundary conditions. Through close collaboration with FLOMANA partners, UMIST in particular, implementation of SSG, SST, V2F models and scalable wall functions could be finalised. The wing-tip vortex and the 3D hill cases were computed in URANS mode, the latter also with LES using a synthetic turbulence method.

9.1 *Code_Saturne*: EDF's unstructured Finite Volume solver

9.1.1 Numerical method

The development of *Code_Saturne* was initiated in 1996 at EDF R&D to gradually replace the block-structured solver ESTET and the Finite Element code N3S. The new solver merges advantages of both methods: simplicity for coding complex models and ability to deal with complex geometries. The numerical method is described in detail in Archambeau et al. (2003). *Code_Saturne* is a collocated finite volume solver, all variables are collocated at the centres of gravity of the cells, which can be of any shape. Hanging nodes are treated as high order polygons. Gradient reconstruction methods as described in Ferziger & Perić, (1999) are used for non-orthogonal cells. Gradients at the cell centres are defined from the Gauss theorem. This requires interpolation of the variables on the cell faces. For structured grids the resulting scheme is similar to finite differencing along lines connecting cell centres. However, on non-rectangular grids, these lines are not orthogonal to cell faces and do not intersect the cell face centres. This interpolation is then corrected using 3D Taylor expansions which in turn involve the gradients of the variables. As these are not yet known, the deferred correction of Ferziger and Peric is introduced in the time-scheme: the implicit part of the fluxes (convection or diffusion) is written "as if" the cells were orthogonal, while "older", known values, of the gradients are used as a correction in the Taylor expansion correction.

The momentum equations are solved by considering an explicit mass flux. Velocity and pressure coupling is insured by the SIMPLEC algorithm. The Poisson equation is solved with a conjugate gradient method. The collocated discretisation requires a Rhie and Chow (1982) interpolation in the correction step to avoid oscillatory solutions, but is not essential for unstructured meshes.

9.1.2 Turbulence Models

Version 1.0 was released early 2001, with a standard k-epsilon and LRR differential stress model as basic turbulence models. FLOMANIA provided an opportunity to develop more advanced models, and all the while LES was developed on own resources. Switching from the LRR to the SSG has been a painless development, but the benefits of FLOMANIA lie in the demonstration of the superiority of the SSG model on a number of test cases. This model will become the default DSM at EDF. Furthermore, a number of near-wall models have been developed by the UMIST partner in EDF's code, and presented in the relevant section. However the Finite Volume numerical issues are presented here. Stability and positivity of turbulence variables are ensured by a combination of a fractional times-step method for and deferred correction. The latter allows balancing explicit source terms with convection-diffusion contributions.

9.1.3 Wall Functions

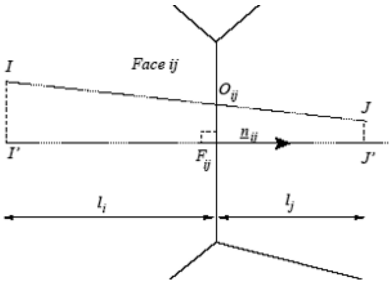

Figure 1 Cell face centre value obtained from cell centre values.

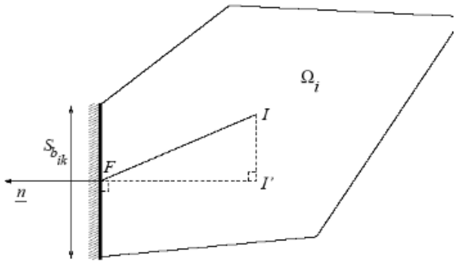

Figure 2 Face value on a solid boundary

Throughout the solver, gradients and cell face values are linked. Cell face values define the cell centre gradients through the Gauss theorem and gradients are used in turn for interpolations from centres to the faces, to correct the fact that the cell centre connecting lines do not intersect the faces at their centre:

$$\phi_{Fij} = \phi_{Oij} + O_{ij}F_{ij}(grad(\phi))_{Oij}$$

A value on a boundary cell face, F, can similarly be obtained by:

$$\phi_f = \phi_{I'} + I'F(grad(\phi).n)_f$$

Hence there is equivalence between specifying a Neumann condition or a face value on the border. Either way, both conditions are needed at some point in the calculation. In the standard wall function approach, the velocity at the border face Uf is calculated by equating the theoretical (or log-law) value of the velocity gradient and the calculated value obtained by the code:

$$G_{theo} = \frac{\partial U}{\partial y} = \frac{u*}{\kappa y}$$

$$G_{calc} = \frac{U_g - U_f}{2d} = \frac{(U_i + U_j)/2 - U_f}{2d}$$

A Cartesian mesh is assumed above for the purpose of simplicity: Ug , is the velocity at the upper face of the boundary cell, obtained by interpolating from *Ui* the velocity at the centre of the boundary element, and *Uj* is the velocity at the centre of the element above the boundary.

When introducing the scalable wall function approach recommended in FLOMANIA, the theoretical gradient is written as:

$$G_{theo} = \frac{\partial U}{\partial y} = \frac{u_*}{\kappa y} = \frac{u_*^{\,2}}{\kappa \tilde{y}^* \nu}$$

$$\tilde{y}^* = \min(y^*, Y^*); \qquad Y^* = 11.06$$

Previously, whenever the first cell fell below the buffer layer, G_{theo} was allowed to transition to the linear, viscous, velocity profile, while the turbulent variables retained their high-Re Neumann conditions. Although more accurate as concerns the velocity, this led to severe overestimation of the production of *k*, and too high friction. With the scalable wall functions, results were dramatically improved on the diffuser test case. The SWF reduces the velocity gradient at the wall, thus alleviating the need for damping functions.

9.1.4 Near wall models

As explained above, a segregated approach is used whereby convection-diffusion is solved successively for each variable, as a series of scalars (with the exception of destruction source terms inter-coupling). This works well when production and dissipation are in reasonable balance, but is more problematic with Near-Wall models where flow-physics dictates a balance between viscous diffusion and sink terms. Moreover, most applications are treated as time-dependant, which excludes stabilising measures such as under-relaxation and local time-stepping. Hence, early attempts with LRN models and necessarily very small CFL numbers had proved discouragingly expensive. On the other hand, EDF had supported early on the development of Durbin's brand of elliptic relaxation LRN models (Manceau et al 2001). During FLOMANIA, the V2F development was thus handed over to J. Uribe at UMIST who very successfully developed a code friendly version of the V2F model into *Code_Saturne* (Cf chapter II-17), by a consistent reformulation that decouples the boundary conditions so that they are compatible with the algorithm described above.

9.2 Highlight Results

9.2.1 Asymmetric Diffuser test case:

Results of case 3 are compiled in Chapter IV. What is analyzed here is the effect of reducing the tensorial diffusion (Daly Harlow) of stresses to an isotropic eddy viscosity, as suggested within the consortium to simplify the DSM model. Indeed divergence of a 3rd rank tensor is need in the full model. In a general purpose unstructured FV software which may include (as in *Code_Saturne*) angular periodicity, this may seem a daunting task. No differences between stress diffusion formulations were noticeable on the channel flow, but on the asymmetric diffuser, the simplification was observed to degrade somewhat the predictions. Nevertheless, the difference needs to be balanced against the coding complexity.

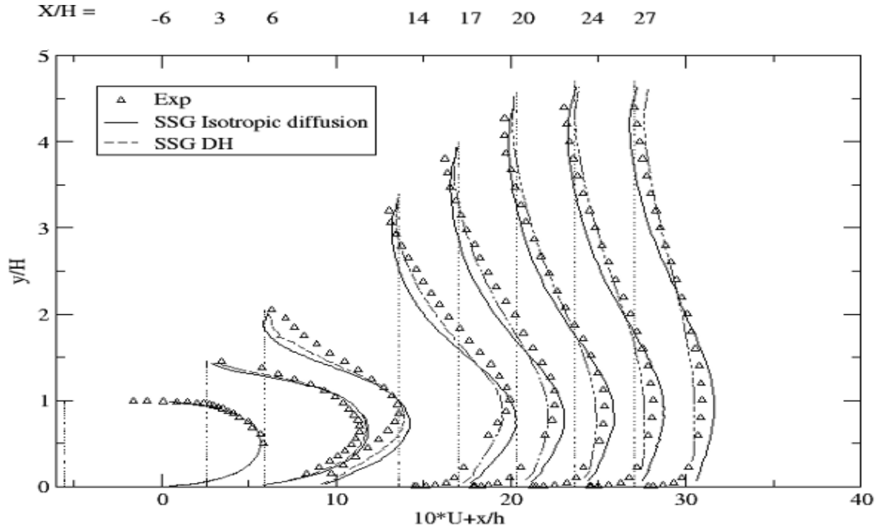

Figure 3 Effect of isotropic vs. tensorial diffusion in DSM model. Diffuser case.

9.2.2 Wing-tip

The wing-tip vortex has been computed with the DSM model and the initial mandatory grid. The initial stage of the formation of the vortex, with high momentum fluid (U/U_∞ =1.7) entrapped in the vortex core and surrounded by low momentum boundary layer fluid was well reproduced, but in later downstream stages, the too coarse grid lead to numerical diffusion and mixing of the 2 streams, as shown by subsequent simulations by C. Robinson on grids of up to 4 Million nodes.

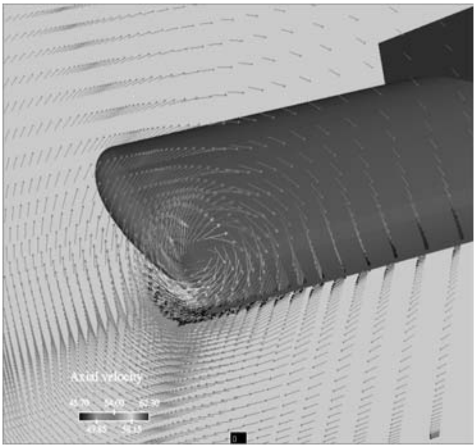

Figure 4 Wing-tip vortex formation, DSM.

9.2.3 *The 3D hill*

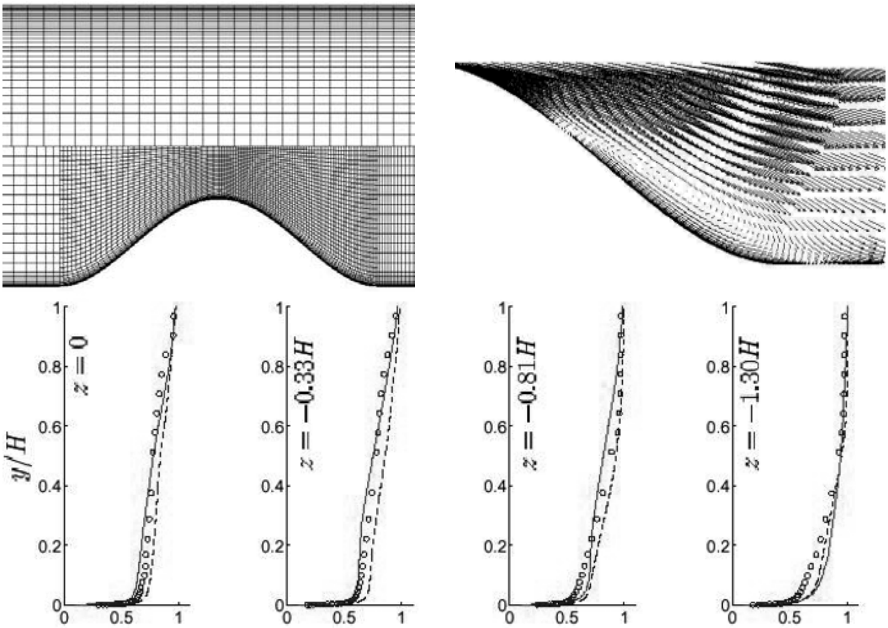

Figure 5 Flow over a 3D hill. LES results. Top: unstructured grid in symmetry plane, shallow separation. Bottom: Mean axial velocity in wake at x/H=3.69; dashed line - LES EDF, solid lines - LES Chalmers, symbols - experiments.

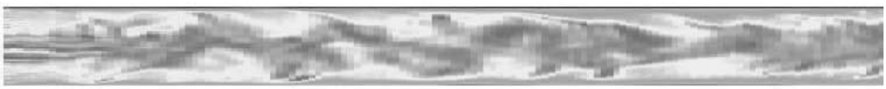

Figure 6 Non periodic channel flow with synthetic vortex inlet

The DSM model has been used to compute the 3D hill flow with a locally refined grid of 0.8 M cells (no symmetry plane), in time dependant mode, and produced an unsteady wake (although inlet was steady-state). These URANS results seem to be among the more realistic in the RANS group, yet not totally satisfactory.

An LES computation was run on the same grid, using the synthetic vortex method developed by Jarrin et al. (2003). This method generates realistic eddies which correspond to a prescribed distribution of Re stress (obtained e.g. from RANS) and are shown to be highly sustainable in channel and pipe flow tests without periodicity (Fig. 6). The hill-flow LES produced results in much better agreement with the experiments and refined RANS-LES of Chalmers. Therefore, this case shows no advantage of U-RANS compared to full LES which are able to reproduce accurately the experimental data in the wake, Fig 6. It can be concluded that none the various RANS or URANS approaches can reproduce this flow.

9.2.4 Near Wall models

Finally, the outcome from the FLOMANIA collaboration for EDF have been mainly new near wall modeling procedures: the scalable wall functions have been developed for two eqn. and DSM models and demonstrated indeed mesh independent behaviour as announced by F. Menter (despite the surprising simplicity of the suggested method). The Analytical Wall Functions of Craft, Gerasimov et al. have also been implemented, and mainly proved advantageous on buoyancy affected flows (as Fig 7), but not so much on Aerodynamic flows (hence not described herein). The SST model newly implemented has proved robust and more accurate that the standard Launder-Sharma model. In particular the code-friendly V2F model suggested by Uribe et al. (Cf. Chap II-17) also implemented in *Code_Saturne* was found to converge much faster than the original V2F model.

It is well known from V2F model publications that this model exhibits best its advantage in heat transfer cases, as illustrated figure 7 for natural convection in a tall cavity. The TU Delft group independently arrived at a similar formulation of the model and a joint publication highlights further benefits for heat transfer applications (Hanjalic et al. 2005).

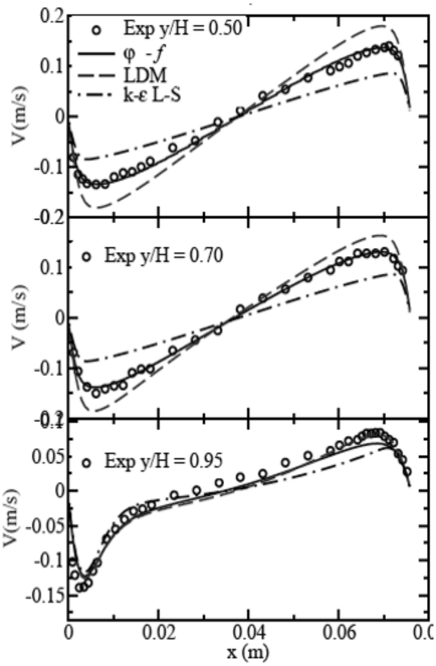

Figure 7 Natural convection in a tall cavity. Lines: new V2F model, dashed lines reference LDM (Lien-Durbin), dot-dashed Launder-Sharma k-epsilon. Symbols: exp. by Betts and Bokhari. 1995.