EulFS: a parallel CFD code for the simulation of Euler and Navier-Stokes problems on unstructured grids

Aldo Bonfiglioli[1], Bruno Carpentieri[2], and Masha Sosonkina[3]

[1] Dip.to di Ingegneria e Fisica dell'Ambiente, University of Basilicata, Potenza, Italy, ba001ing@unibas.it
[2] Karl-Franzens University, Institut of Mathematics and Scientific Computing, Graz, Austria, bruno.carpentieri@uni-graz.at
[3] Ames Laboratory/DOE, Iowa State University, Ames, USA, masha@scl.ameslab.gov

Abstract. We present results with a parallel CFD code that computes steady-state solutions of the Reynolds-Favre averaged Navier-Stokes equations for the simulation of the turbulent motion of compressible and incompressible Newtonian fluids. We report on preliminary experiments on 2D and 3D problems, for both internal and external flow configurations.

In this communication we present numerical results with an academic code developed by the first author [3] for simulating the turbulent motion of compressible and incompressible Newtonian fluids on 2D and 3D problems. The dynamic of the fluid is modeled using the Reynolds-Favre averaged Navier-Stokes (RANS) equations. Despite the non-negligible degree of empiricism introduced by turbulence modeling, it is recognized that the solution of the RANS equations still remains the only feasible approach to perform computationally affordable simulations of problems of engineering interest on a routine basis. The code is able to compute steady-state solutions of the RANS equations for both internal [1] and external [5] flow configurations. The computational domain is tesselated using unstructured grids made of triangles and tetrahedra, in 2 and 3 space dimensions, respectively. The integral, conservation-law form of the governing equations is discretized using Fluctuation Splitting (or Residual Distribution) schemes. This discretization technique was introduced in the early eighties by Roe [4] and shares common features with both Finite Element (FE) and Finite Volume (FV) methods. It features linear shape functions and compact stencils that result in more sparse matrices arising from the discretization. Turbulence is modeled using the Boussinesq approximation, and the eddy viscosity is computed by means of the one-equation Spalart and Allmaras turbulence model.

Although the objective here is to calculate steady state solutions, the time-derivative in the governing conservation equations is retained. As explained below, this is done because the integration strategy is partially based on pseudo-time marching. This is achieved by discretizing the time-derivative with an

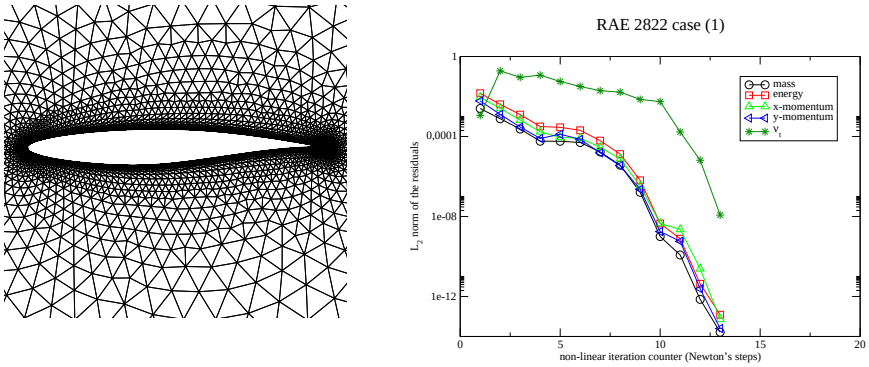
implicit Euler scheme and time-marching the solution towards a stationary state. At each (pseudo) time-step a large nonsymmetric (though structurally symmetric) sparse linear system needs to be solved. The sparse matrix is made up of contributions from the Jacobian of the flow equations and the time derivative term. It has a block structure, whereby each block is associated with a particular grid node. Due to the compact stencil of the schemes, the sparsity pattern of the Jacobian corresponds to the graph of the underlying unstructured mesh. This is because the spatial discretization scheme only involves distance-1 neighbours. On average, the number of non-zero (block) entries per row equals 7 in 2D and 14 in 3D. The analytical evaluation of the Jacobian matrix is rather cumbersome. Therefore, two alternative approaches have been adopted in *EulFS*: one is based on an approximate analytical estimate of the Jacobian, whereas the other uses a numerical approximation of the *true* Jacobian, obtained by means of one-sided finite difference (FD) formulae. In both cases, the elements of the Jacobian matrix are stored in memory. The FD approximation is computationally expensive and therefore its use pays off only if the number of iterations to obtain a converged solution can be significantly reduced with respect to that of simpler iterative schemes. This condition may be achieved by exploiting the quadratic convergence of Newton's method, which is recovered from the time integration scheme as the size of the time-step approaches infinity. On the other hand, the residual reduction of Newton's method is quadratic only if the initial guess lies in a sufficiently small neighbourhood of the steady state. This condition is certainly not met if the numerical simulation has to be started from scratch. Therefore a two-step approach is adopted. The first step is used at the early stages of the calculation, and it consists of solving the equations in an iterative loosely coupled fashion: the turbulent viscosity is kept frozen and the other variables are advanced over a single time-step using an approximate Jacobian; then the turbulent variable is advanced over one or more pseudo time steps using a FD jacobian with frozen mean flow variables. This cycle is applied recursively. However, due to this partial decoupling and also to the use of a finite time-step for stability reasons, this procedure would lead to linear convergence, at most. This is precisely the reason for introducing the second step. Once the solution is sufficiently close to the steady state, a true Newton strategy is adopted: the solution of the mean flow and turbulence transport equation is fully coupled, the Jacobian is computed by FD and the time-step is rapidly increased.

The code is implemented using the PETSc library [2] and has been ported on different parallel computer architectures, including the Linux Beowulf cluster used in the numerical experiments presented herein.

Experiments on the RAE problem

The first test-case that we consider is the compressible, subsonic flow past the two-dimensional RAE2822 profile. Free-stream conditions are as follows: Mach number $M_\infty = 0.676$, Reynolds' number based on chord: $\mathrm{Re}_C = 5.7\,10^6$, angle of attack: $\alpha_\infty = 2.40°$. The computational mesh, which is shown in Figure 1(a),

is made of 10599 meshpoints and 20926 triangles. The simulation is started from



(a) Computational mesh for the RAE2822 aerofoil.

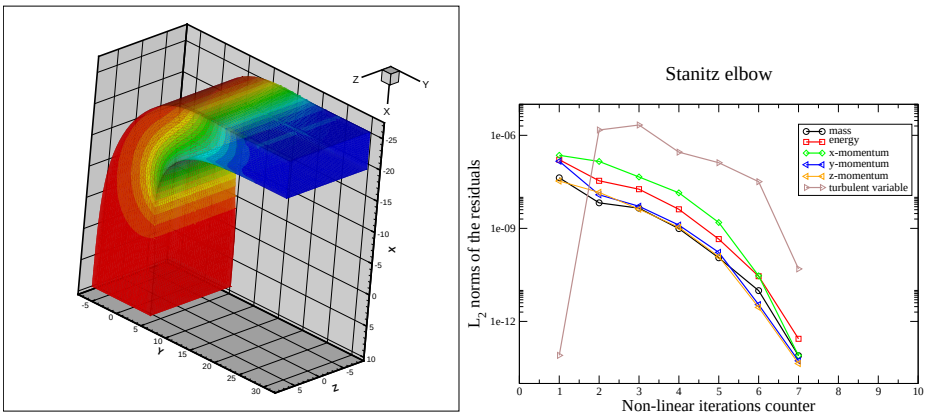
(b) L_2 residual norms versus the Newton's iteration counter.

Fig. 1. RAE2822 airfoil.

uniform flow and the solution advanced in pseudo-time using the approximate linearization. Once the L_2 norm of the residual has been reduced below a preset threshold, the fully coupled approach is put in place. The convergence of Newton's algorithm towards steady-state is shown in Figure 1(b): only thirteen Newton iterations are required to reduce the L_2 norm of the residuals (mass, energy, x and y momentum, turbulent kinematic viscosity ν_t) down to machine zero. For the inner linear solver, we use GMRES(30) [6] preconditioned by block incomplete LU factorization with sparsity strategy based on level of fill (see, e.g., [6]). We take as the initial guess for GMRES $x_0 = 0$ and the stopping criterion consists in reducing the original residual by 10^{-5} that then can be related to a norm-wise backward error.

Experiments on the Stanitz elbow

The three-dimensional test case that we have examined deals with the internal compressible flow through the so-called Stanitz elbow. The simulation reproduces experiments [7] conducted in early 1950's at the National Advisory Committee for Aeronautics (NACA), presently NASA, to study secondary flows in an accelerating, rectangular elbow with 90° of turning. The chosen flow conditions correspond to a Mach number in the outlet section of 0.68 and Reynolds' number $4.3e5$. Figure 2(a) shows the geometry along with the computed static pressure contours. The computational mesh consists of 156065 meshpoints and 884736 tetrahedral cells. The simulation has been run on 16 processors of a Linux Beowulf cluster. Figure 2(b) shows the residual convergence history of the last stages of the iterative process when very large time-step is used so that



(a) Elbow geometry and computed static pressure contours.

(b) Convergence history of Newton's algorithm.

Fig. 2. The "Stanitz" elbow

Newton's method is recovered. In this case we use a block Jacobi preconditioner for GMRES, where the diagonal blocks are approximately inverted using block $ILU(4)$.

The code is still in a development stage. We have shown results of preliminary experiments on 2D and 3D problems. Perspective of future research include enhancing both the performance and the robustness of the code on more difficult configurations, especially in the direction of the design of multilevel preconditioners for solving the inner linear systems.

References

1. Bonfiglioli A., De Palma P., Pascazio G., and Napolitano M. An implicit fluctuation splitting scheme for turbomachinery flows. *Transactions of the ASME Journal of Turbomachinery*, 127:395–401, 2005.
2. Satish Balay, William D. Gropp, Lois Curfman McInnes, and Barry F. Smith. PETSc home page. `http://www.mcs.anl.gov/petsc`, 1998.
3. A. Bonfiglioli. Fluctuation splitting schemes for the compressible and incompressible Euler and Navier-Stokes equations. *IJCFD*, 14:21–39, 2000.
4. H. Deconinck, H. Paillère, R. Struijs, and P.L. Roe. Multidimensional upwind schemes based on fluctuation-splitting for systems of conservation laws. *Journal of Computational Mechanics*, 11(5/6):323–340, 1993.
5. R. Paciorri, A. Bonfiglioli, A. Di Mascio, and B. Favini. Rans simulations of a junction flow. *International Journal of Computational Fluid Dynamics*, 19:179–189, 2005. ISSN: 1061-8562.
6. Y. Saad. *Iterative Methods for Sparse Linear Systems*. SIAM, 2003.
7. John D. Stanitz, Walter M. Osborn, and Joghn Mizisin. An experimental investigation of secondary flow in an accelerating, rectangular elbow with 90° of turning. Technical Note 3015, NACA, 1953.