3rd International Workshop on High-Order CFD Methods

January 3-4, 2015 at the 53rd AIAA Aerospace Sciences Meeting in Kissimmee, Florida (Orlando)

Abstract available at:

https://www.grc.nasa.gov/hiocfd/workshop-results/

Case C1.2b: Flow over the NACA0012 airfoil - transonic case

Andrea Ferrero*and Francesco Larocca†
Department of Mechanical and Aerospace Engineering
Politecnico di Torino, Italy

1 Code description

Numerical simulations were performed with a discontinuous Galerkin code written in Fortran 90 which is currently under development. The code can solve Euler equations, Navier-Stokes equations or RANS equations with different turbulence models (Spalart-Allmaras, Wilcox k-omega, k-omega+Laminar Kinetic Energy) in 2D.
Several approximate Riemann problem solvers and numerical fluxes (Osher, Roe, AUSM+, Rotated-RHLL, Lax-Friedrichs) are available for the computation of convective fluxes. In particular in this test case the Osher-Solomon flux [1] is used.
Diffusive fluxes are computed by means of a recovery based approach [2]. The implemented method is inspired to the original recovery approach proposed by Nomura and van Leer [3] but it makes use of a different recovery basis and a different boundary procedure.
The numerical solution inside the element is represented through an orthonormal modal basis obtained by the modified Gram-Schmidt procedure. Both physical space defined and element space defined basis functions can be chosen. In the first case a set of monomials defined in the physical space is used to start the orthonormalization procedure, following the approach of [4]. In the second case, the orthonormalization is initialized with a tensor product of Legendre polynomials defined on the reference element. In this test case the first approach is used.
Several shock capturing methods are implemented: limiters, adaptive filters and artificial viscosity. For this test case the artificial viscosity method described by [5] is considered. Curvilinear elements are implemented up to fourth order for quadrilaterals and third order for triangles.
Both explicit (RK-TVD and SSP-RK) and implicit (backward Euler) time integration schemes have been implemented. For this steady test case the implicit

*PhD candidate, email: andrea_ferrero@polito.it
†Professor, email: francesco.larocca@polito.it

backward Euler method is used. The jacobian is evaluated numerically. The implementation of an analytically evaluated jacobian is under development.
As far as parallelization is concerned, the explicit version of the code is fully parallelized through OpenMP directives. In the implicit version of the code, the computation of fluxes and the linear solver are parallelized by OpenMP. In particular, the GMRES method with ILU(0) preconditioner from the library PARALUTION [6] is used. The numerical evaluation of the jacobian is performed in serial.
The code can perform both h-adaptive or hp adaptive strategies. The h-adaptive algorithm is based on isotropic splitting and is controlled by an error indicator based on residuals, according to [7]. At each refinement step the algorithm splits 20 % of the elements which are characterized by the largest values of the error indicator. After that, a further check and splitting is performed in order to avoid too large jumps in the mesh size distribution.
The implemented hp-adaptive algorithm is controlled by the same error indicator of the h-adaptive algorithm. However, the elements flagged for refinement are further analyzed by means of a smoothness test. The test is based on the smoothnees sensor proposed by [8] and applied to pressure. If the element is considered sufficiently smooth, the order of the reconstruction is increased. Otherwise the element is splitted.
As far as postprocessing is concerned, the code can generate output files in which each mesh element is subdivided in several elements depending on the number of degrees of freedom of the reconstruction. This makes it possible to obtain a visualization which takes into account all the information related to the high order reconstruction

2 Case summary

2.1 Computational domain and mesh

The computational domain is reported in Figure 1. The external boundaries are at more than 2000 chords from the airfoil. This distance has been chosen according to the meshes reported on the website of the Workshop. The original mesh has been generated by Gmsh 2.8.5 [9] and a detail is reported in Figure 2. Cubic triangular elements are used in all simulations.

2.2 Time discretization

All simulations were stopped when L_2R/L_2R_0 dropped down 10^{-10} where L_2R and L_2R_0 are the L2-norm density residuals at the current iteration and at the first time step. The residuals refer to the zero order modal coefficient. In order to accelerate convergence, the solution is obtained through a sequence of progressively higher order reconstructions. All simulations are initialized with a uniform flow field and a reconstruction order p=0 on the original mesh of Figure 2. Then the reconstruction order is increased up to the desired value. The

adaptation is performed when the final reconstruction order is reached. The reported computational time refers to the entire process, from p=0 to the final p. It also includes the cost required by the adaptive procedure.

The pseudo-transient continuation technique is employed to accelerate convergence to steady state. The chosen CFL number evolution strategy is a mix between exponential progression and switched evolution relaxation.
The minimum CFL number is 10^1 and the maximum CFL number is 10^{10}. The GMRES iterative solver is stopped when the relative error reaches 10^{-2} or when the number of iterations exceeds 250.

2.3 Hardware specification

A Linux machine with an Intel i7-3930x processor and 32 Gigabytes of RAM was used. The machine produces a Taubench time of 6.5 seconds. All computations were performed in serial. In Table 1 the work units required to perform 100 residual evaluations with 250000 DOFs are reported. The data refer to a convection-diffusion discretization which takes into account the artificial viscosity shock capturing term. The results for the implicit integration are dominated by the cost of the numerical jacobian evaluation.

p	Explicit	Implicit
1	9.40	131
2	16.4	649
3	34.0	2326

Table 1: Work Units for 100 residual evaluations in an inviscid transonic problem (with artificial viscosity) with 250000 DOFs

3 Results

Computations were performed with both the h-adaptive algorithm (p=1 and p=2) and the hp-adaptive algorithms ($1 \leq p \leq 2$). In each case five refinement steps were performed.

The errors on the lift and drag coefficients are computed by taking the RWTH values from a precedent edition of the Workshop ($Cl = 3.529140 * 10^{-1}$;$Cd = 2.274636 * 10^{-2}$) as reference. Figure 4 and 5 show the Cd and Cl errors as a function of the equivalent length scale $h = 1/\sqrt{nDOFs}$. Figure 6 and 7 show the Cd and Cl errors as a function of the work units.

The results show that the hybrid hp-adaptive strategy is more convenient than the h-adaptive strategy for both $p = 1$ and $p = 2$. This is probably due to the fact that the chosen higher order reconstruction does not pay in the shock

region.
Indeed, a comparison between the results for $p = 1$ and $p = 2$ with the h-adaptive algorithm shows that the $p = 1$ reconstruction is more convenient. These results could be related to the use of an isotropic splittig strategy which introduces a lot of degrees of freedom in the shock region. The degrees of freedom related to the high-order modes do not bring a significant benefit in this region because the singularity induces a degradation of the local order of accuracy. However, they increase significantly the computational cost. For this reason, an anisotropic splitting strategy would be a better approach.

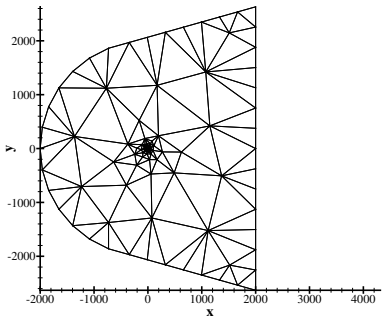

Figure 1: Computational domain

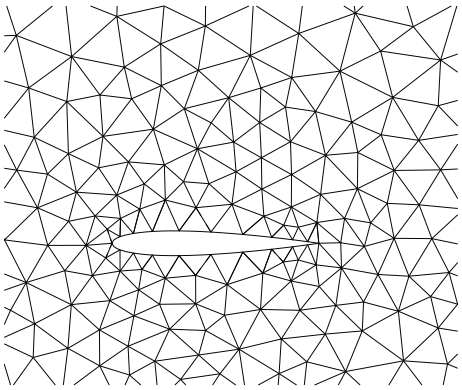

Figure 2: Initial mesh (1060 elements)

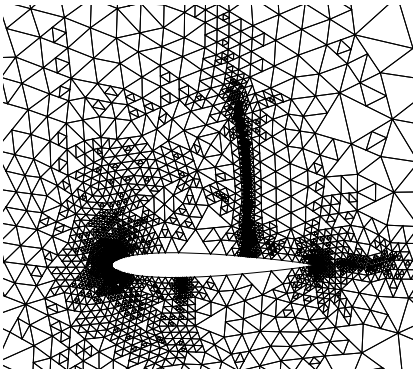

Figure 3: Adapted mesh after 5 refinements (11806 elements)

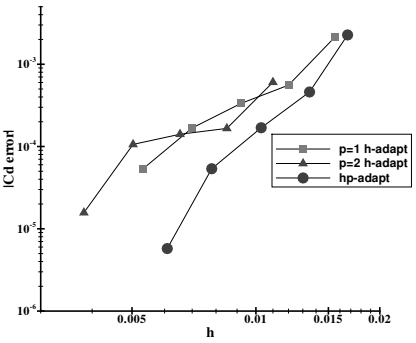

Figure 4: Cd error vs length scale

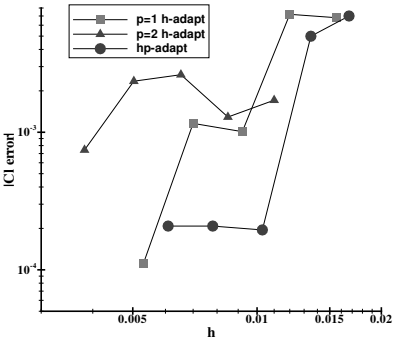

Figure 5: Cl error vs length scale

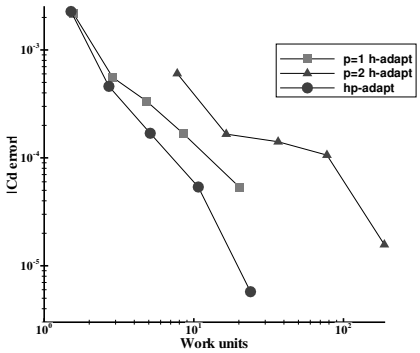

Figure 6: Cd error vs work units

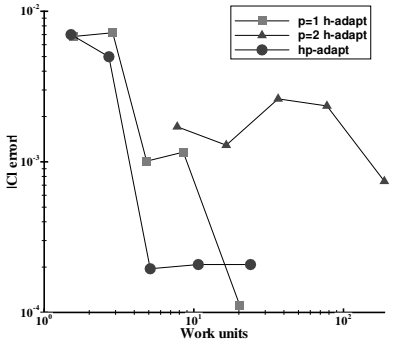

Figure 7: Cl error vs work units

References

[1] Osher S. and Solomon, F. (1982) Upwind difference schemes for hyperbolic systems of conservation laws, Mathematics of Computation, Vol. 38, No. 158, pp.339374.

[2] Ferrero A., Larocca F., Puppo G. A robust and adaptive recovery-based discontinuous Galerkin method for the numerical solution of convection-diffusion equations, *Int. J. Numer. Meth. Fluids*, 77:63-91, 2015.

[3] van Leer B, Nomura S. (2005) 'Discontinuous Galerkin for Diffusion', AIAA paper 2005-5108.

[4] Bassi F, Botti L, Colombo A, Di Pietro DA, Tesini P. (2012) 'On the flexibility of agglomeration based physical space discontinuous Galerkin discretizations', Journal of Computational Physics; 231 : 45-65.

[5] Nguyen N.C. and Peraire, J. (2011) 'An Adaptive Shock-Capturing HDG Method for Compressible Flows', AIAA Paper 2011-3060.

[6] Dimitar Lukarski, Nico Trost, 'PARALUTION Project', http://www.paralution.com/

[7] Leicht T, Hartmann R. (2010) 'Error estimation and anisotropic mesh refinement for 3d laminar aerodynamic flow simulations', J. Comput. Phys. (229), pp. 7344-7360.

[8] Persson, P.-O. and Peraire, J. (2006) 'Sub-Cell Shock Capturing for Discontinuous Galerkin Methods', In 44th AIAA Aerospace Sciences Meeting and Exhibit, no. AIAA-2006-12.

[9] Geuzaine, C. and Remacle, J.-F. (2009) 'Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities', International Journal for Numerical Methods in Engineering 79(11), pp. 1309-1331.