Case C1.2 - Flow Over a NACA 0012 Airfoil: Subsonic Inviscid, Transonic Inviscid and Subsonic Laminar Flows

Nicolas Ringue*, Brian Vermeire† and Siva Nadarajah‡

Computational Aerodynamics Laboratory, McGill University, Montreal, Quebec, Canada

1 Code Description

The conservation laws are discretized by the correction procedure via reconstruction (CPR) scheme with DG correction functions [1–4] The divergence of the inviscid fluxes are determined either through a chain rule or Lagrange polynomial approach. The Roe flux [5] is employed as the common interface flux and the BR2 scheme [6–8] for the viscous flux. As for boundary conditions, Riemann invariants are used in the far-field, while either slip or adiabatic non-slip on the walls. The dynamic viscosity coefficient is either held constant throughout the computational domain or obtained from the Sutherland's law. Artificial viscosity is used to stabilize the solution in the presence of shocks. A smoothness indicator [9] is used to flag cells where viscosity has to be added. A smoothing algorithm is then used to obtain a continuous (C^0) artificial viscosity field for enhanced robustness. Steady state solutions are obtained using a Newton-Krylov algorithm, which serves as the primary solver. The sparse linear system of equations are solved using GMRES included in the PETSc package version *3.2-p7*, while preconditioning is provided by a block-Jacobi method. Often several block-Jacobi iterations are performed before GMRES is employed. The solver is parallelized using MPI via Open MPI, version *1.4.3* where grid partitioning is achieved through ParMETIS. An implicit-explicit (IMEX) scheme serves as a secondary solver, where a three-stage diagonally implicit Runge-Kutta (DIRK) is used. Each stage is split between an explicit and implicit sub-stage, where the non-stiff regions are solved with an explicit RK, while the stiff portions are solved through the above stated Newton approach. Post-processing is typically performed with Tecplot 360 and/or Gmsh version *2.8.5*.

2 Case Summary

The case considered is the flow over a NACA 0012 airfoil in subsonic inviscid, subsonic laminar and transonic inviscid flow regimes. The flow conditions for the three subcases

*PhD Student, McGill University, Montreal, QC, Canada; nicolas.ringue@mail.mcgill.ca.

†Post-Doctoral Scholar, Imperial College, London, United-Kingdom.

‡Associate Professor, McGill University, Montreal, QC, Canada.

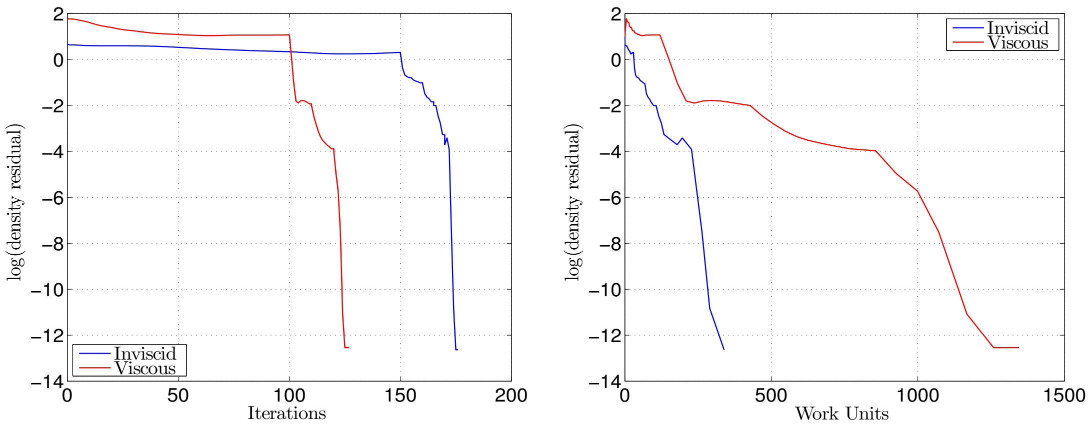


Figure 1: Density residual convergence for P3 solutions of the subsonic inviscid and viscous cases (on mesh ref4, 35840 elements).

are as follows:

1. Subsonic inviscid flow: $\alpha = 2^o$, $M_\infty = 0.5$;

2. Subsonic laminar flow: $\alpha = 1^o$, $M_\infty = 0.5$, $Re = 5000$, $\mu = \text{cst}$;

3. Transonic inviscid flow: $\alpha = 1.25^o$, $M_\infty = 0.8$ (not considered in the present study).

The L_2 norm of the density residuals is monitored to assess the convergence of the iterative solution by ensuring that the initial residuals are reduced by at least 10 orders of magnitude. The residuals are computed following the workshop guidelines. Examples of density residual convergence are shown on Figure 1. As mentionned in the code description, block-Jacobi iterations are used to initiate the computation (150 and 100 block-Jacobi iterations for the inviscid and viscous cases presented on Figure 1 respectively). GMRES iterations then follow to converge the solution to steady state.

The *Guillimin* cluster which belongs to the McGill high performance computing (MHPC) infrastructure and to the Compute Canada HPC network, served for the most intensive computations that used up to 16 cores in parallel on three different architectures: sw/lm, sw2/lm2 and xlm2. The rest of the calculations were performed on a quad-core personal computer (PC). Specifications and Taubench results for these machines are presented in Table 1.

Machine name	Specifications	Taubench CPU time
MHPC–sw/lm	Dual Intel Westmere EP Xeon X5650 6-core, 2.66 GHz, 12MB Cache, 95W	9.5 s
MHPC–sw2/lm2	Dual Intel Sandy Bridge EP E5-2670 8-core, 2.6 GHz, 20MB Cache, 115W	8.1 s
MHPC–xlm2	Quad Intel Sandy Bridge EP E5-4620 8-core, 2.2 GHz, 16MB Cache, 95W	11 s
PC	Intel i7-3770 CPU 4-core, 3.40 GHz, 8MB Cache	5.2 s

Table 1: Computer specifications and TauBench CPU times

3 Meshes

The quadrilateral meshes provided for the workshop have been used in this case. These P4 meshes have been converted to P1 (straight edges). Nodes on the curved boundary were then added based on the analytical definition of the airfoil and cubic spline interpolation. Six grids ranging from 140 (ref0) to 143 360 cells (ref5) with P1 to P9 elements were used to perform *hp*-refinement studies.

4 Results

4.1 Subsonic Inviscid Case

Figures 2 and 3 present the lift and drag coefficients obtained for several meshes and polynomial orders. These plots demonstrate convergence of the aerodynamic coefficients as the discretization is refined. Since no adaptive computation have been carried out for this case, the reference lift and drag coefficients were determined via Richardson extrapolation to a zero mesh size value for each polynomial degree. The reference values obtained in this manner are listed in Table 2. In the absence of an adaptive solution, this process seems to give useful reference values although not thoroughly reliable. Since the theoretical value of the drag coefficient is known to be zero, the lowest extrapolated drag coefficient value (from the P9 solution) is used as a common reference value. Figures 4 and 5 present convergence of the error on lift and drag coefficients based on these reference values.

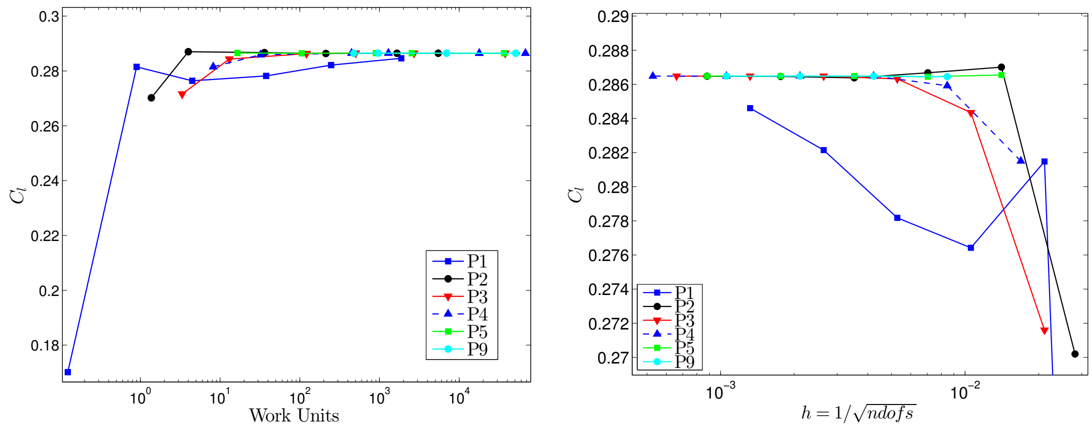


Figure 2: Lift coefficient convergence for the subsonic inviscid case.

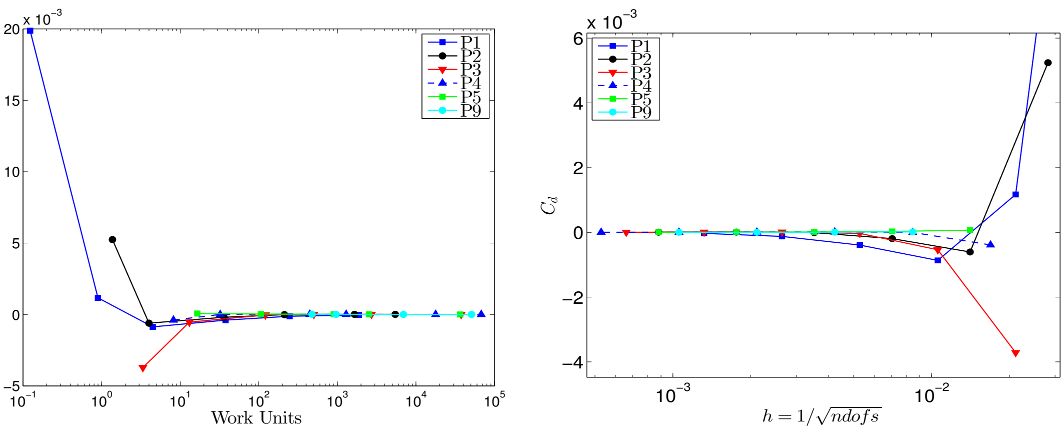


Figure 3: Drag coefficient convergence for the subsonic inviscid case.

Polynomial degree	Lift coefficient C_l	Drag coefficient C_d
P1	0.28860776	2.1066703×10^{-6}
P2	0.28647954	
P3	0.28647845	
P4	0.28647901	
P5	0.28647885	
P9	0.28647948	

Table 2: Reference lift and drag coefficients for the subsonic inviscid case.

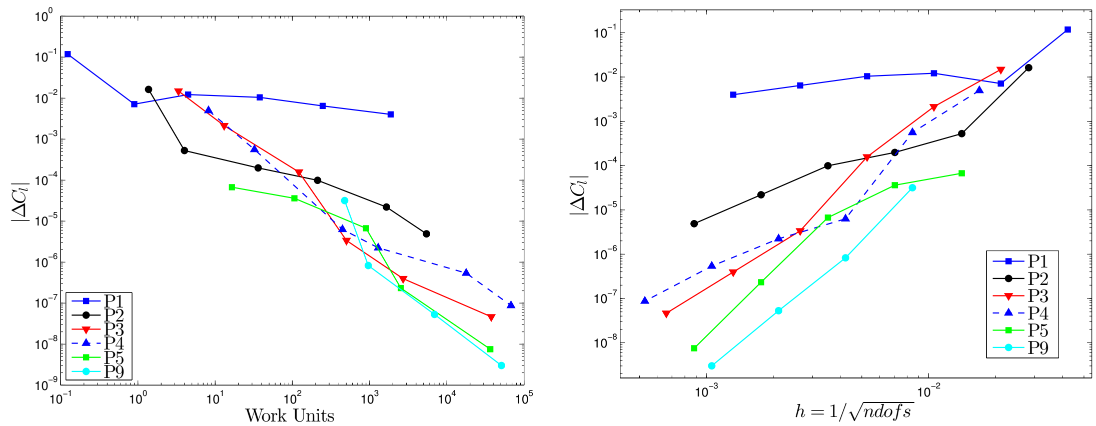


Figure 4: Lift coefficient error convergence for the subsonic inviscid case.

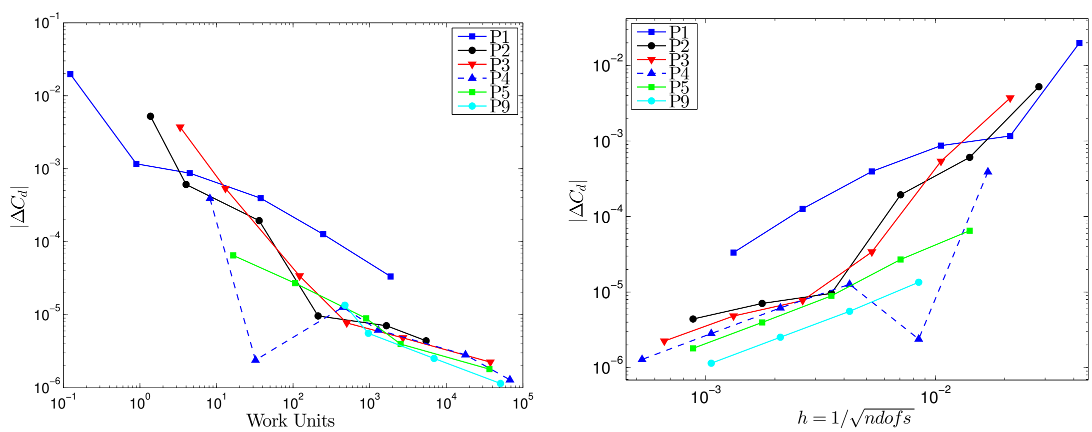


Figure 5: Drag coefficient error convergence for the subsonic inviscid case.

4.2 Subsonic Viscous Case

Figures 6 and 7 present the lift and drag coefficients obtained for several meshes and polynomial orders. These plots demonstrate convergence of these aerodynamic coefficients as the discretization is refined. Similar to the subsonic inviscid case, the reference lift and drag coefficients were determined via Richardson extrapolation to a zero mesh size value for each polynomial degree. The reference values obtained in this manner are given in Table 3. Figures 8 and 9 present convergence of the error on lift and drag coefficients based on these reference values.

Polynomial degree	Lift coefficient C_l	Drag coefficient C_d
P1	0.018710	0.054985
P2	0.018323	0.055311
P3	0.018062	0.055263
P4	0.018469	0.055291
P5	0.018662	0.055362
P6	0.018269	0.055320

Table 3: Reference lift and drag coefficients for the subsonic laminar case.

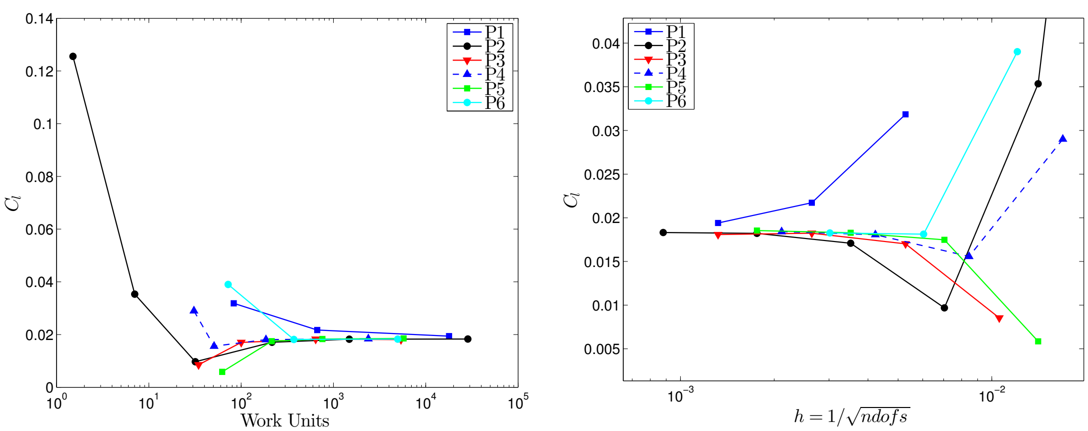


Figure 6: Lift coefficient convergence for the subsonic viscous case.

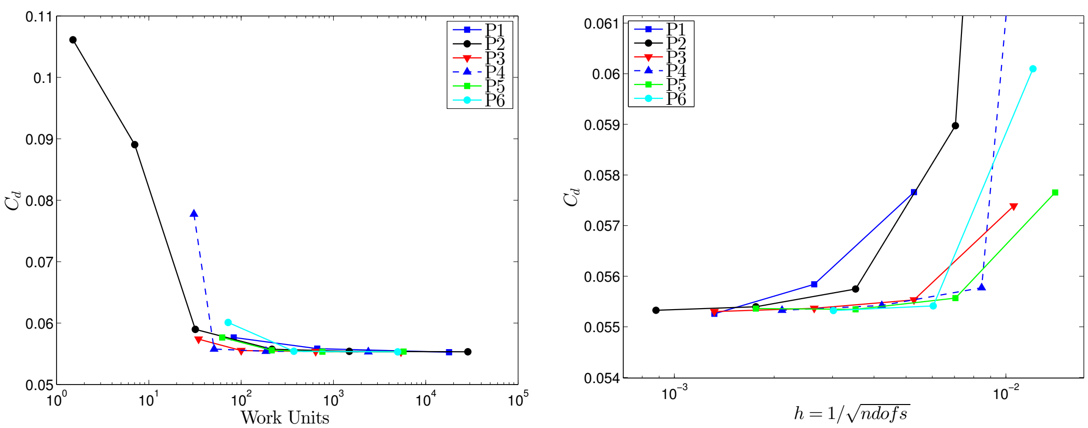


Figure 7: Drag coefficient convergence for the subsonic viscous case.

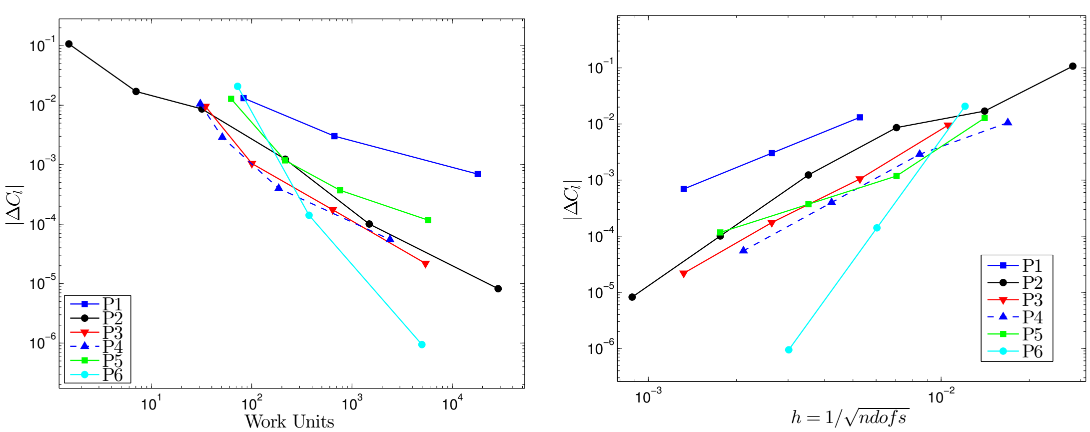
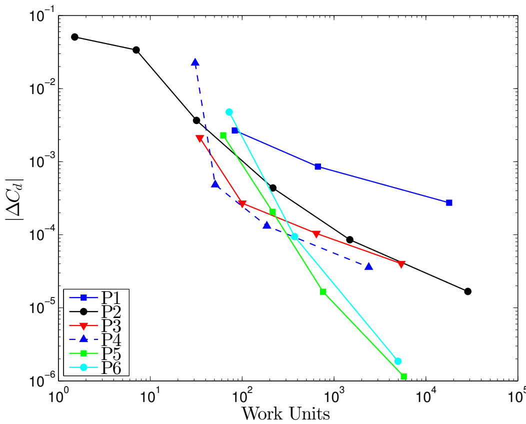


Figure 8: Lift coefficient error convergence for the subsonic viscous case.



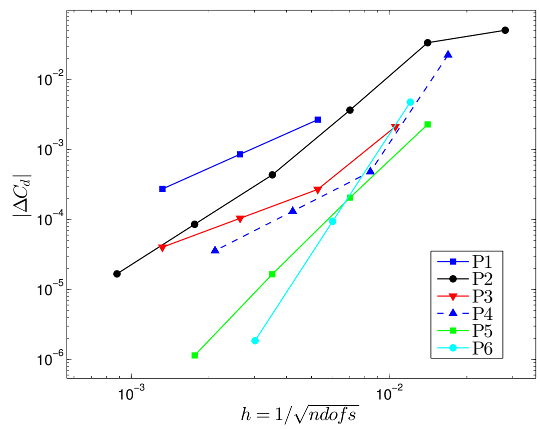


Figure 9: Drag coefficient error convergence for the subsonic viscous case.

References

[1] H. T. Huynh. A flux reconstruction approach to high-order schemes including discontinuous Galerkin methods. In *18th AIAA Computational Fluid Dynamics Conference*, Miami, FL, June 2007. American Institute of Aeronautics and Astronautics, AIAA Paper 2007-4079.

[2] H. T. Huynh. High-order methods including discontinuous Galerkin by reconstructions on triangular meshes. In *49th AIAA Aerospace Sciences Meeting*, Orlando, FL, January 2011. American Institute of Aeronautics and Astronautics, AIAA Paper 2011-44.

[3] Z. J. Wang and H. Gao. A unifying lifting collocation penalty formulation including the discontinuous Galerkin, spectral volume/difference methods for conservation laws on mixed grids. *Journal of Computational Physics*, 228(21):pp. 8161–8186, 2009.

[4] Z. J. Wang. *Adaptive High-Order Methods in Computational Fluid Dynamics*, volume 2 of *Advances in Computational Fluid Dynamics*. World Scientific Pub., 2011.

[5] P. L. Roe. Approximate riemann solvers, parameter vectors, and difference schemes. *Journal of Computational Physics*, 43(2):pp. 357–372, 1981.

[6] F. Bassi, A. Crivellini, S. Rebay, and M. Savini. Discontinuous Galerkin solution of the Reynolds-averaged Navier-Stokes and $k-\omega$ turbulence model equations. *Computers & Fluids*, 34(4):pp. 507–540, 2005.

[7] F. Bassi and S. Rebay. A high order discontinuous Galerkin method for compressible turbulent flows. In *Discontinuous Galerkin Methods*, pages 77–88. Springer, 2000.

[8] F. Bassi and S. Rebay. GMRES discontinuous Galerkin solution of the compressible Navier-Stokes equations. In *Discontinuous Galerkin Methods*, pages 197–208. Springer, 2000.

[9] P.-O. Persson and J. Peraire. Sub-cell shock capturing for discontinuous Galerkin methods. In *44th Aerospace Sciences Meeting and Exhibit*, Reno, NV, January 2006. American Institute of Aeronautics and Astronautics, AIAA Paper 2006-112.