Laminar Flow around a Delta Wing Using Discontinuous Galerkin and Stabilized Petrov-Galerkin Methods

Li Wang, J. Taylor Erwin and W. Kyle Anderson

SimCenter: National Center for Computational Engineering

University of Tennessee at Chattanooga, Chattanooga, TN

1. Code description

The current flow solver makes use of high-order discontinuous Galerkin (DG) discretizations with up to fourth-order accuracy (i.e. p = 3). The convective flux on elemental interfaces is resolved by the HLLC Riemann flux function, while the discretization of the viscous flux is based on implementation of a symmetric interior penalty method. To enhance solution efficiency, the DG solver uses a p-multigrid approach, driven by a linearized element Gauss-Seidel smoother or a Generalized Minimal Residual (GMRES) algorithm. The present high-order discontinuous Galerkin 3D solver implements the standard MPI message-passing library for inter-processor communication, where the mesh is partitioned based on the METIS graph partitioner.

A Petrov-Galerkin code has also been developed with accuracy ranging from second order to third order (p = 2). The solution is obtained using an approximate Newton method where the linear system is solved via a preconditioned GMRES approach with ILU(k) preconditioning. The current 3D PG solver also enables parallel computing using MPI.

2. Case summary

In this benchmark case, the L_2 norm of the density residual is monitored and the residual tolerance is set to be 10^{-11}. The mesh and order refinement studies are performed on multiple processors and the work units are obtained using the formula shown below,

$$\text{Work units} = \frac{\text{wall time} \times \text{number of processors}}{\text{TauBench time}}$$

where the average TauBench CPU time is 13.4394 seconds.

3. Meshes

A sequence of four unstructured meshes, containing 11210, 79738, 600764 and 4671500 tetrahedral elements, are used for the accuracy and efficiency study. The meshes are generated with PointWise software with respective 1.6×10^{-3}, 1.1×10^{-3}, 7.9×10^{-4} and 5.6×10^{-4} viscous spacing normal to the wall. The outer boundary is placed at approximately seven chord lengths away to the delta wing surface. In the DG solver, no-slip and adiabatic wall boundary conditions are prescribed on the wing surface, while isothermal boundary conditions are set in the PG solver. Since no curvature is presented on the delta wing surface, linear meshes containing straight-sided edges and flat faces are employed in the current study. The baseline (i.e. the coarsest) mesh near the wing surface is displayed in Fig. 1 as an example. It should be noted that the surface nodes in the meshes we generate are distributed nearly uniformly, as oppose to the meshes (not shown) provided in the workshop where more clustered elements are placed near the wing tip regions. Tables 1 and 2 list the number of degrees of freedom (nDoFs) per equation and the number of processors used in the DG and PG solvers.

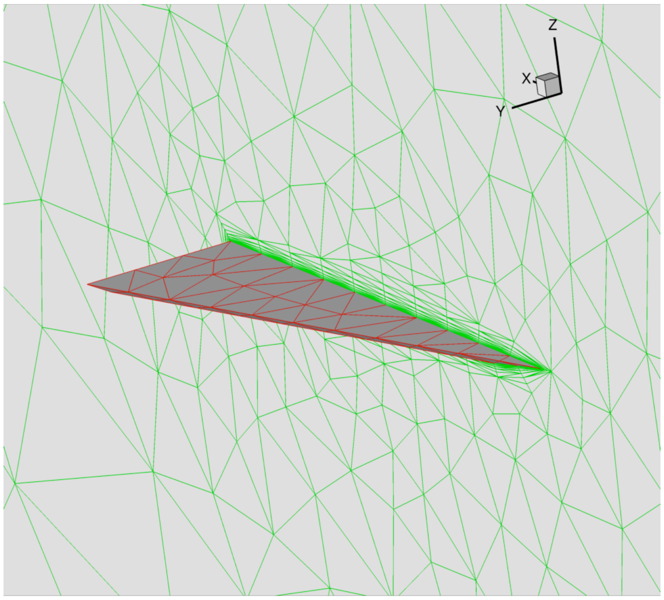


Figure 1. The baseline unstructured mesh (containing 11210 tetrahedrons and 2407 nodes) used in the laminar delta wing benchmark case.

nDoFs	Mesh 0	Mesh 1	Mesh 2	Mesh 3
DG p = 1	44.8×10^{3}	318.9×10^{3}	2.4×10^{6}	18.6×10^{6}
DG p = 2	112.1×10^{3}	797.4×10^{3}	6.0×10^{6}	46.5×10^{6}
PG p = 1	2.4×10^{3}	15.2×10^{3}	107.2×10^{3}	806.2×10^{3}

Table 1. Number of degrees of freedom per equation in the DG and PG schemes.

nprocessors	Mesh 0	Mesh 1	Mesh 2	Mesh 3
DG p = 1	1	8	84	652
DG p = 2	4	28	216	N/A
PG p = 1	1	2	16	100

Table 2. Number of processors used in the DG and PG schemes for the sequence meshes.

4. Results

The simulation starts with uniform flow with Mach number of 0.3 and viscous drag and lift coefficients are computed using the second- and third-order of DG discretizations (p = 1 and p = 2) as the steady state solution is achieved. The reference force coefficients, corresponding to $C_l^{ref} = 0.3455$ and $C_d^{ref} = 0.1605$, are obtained using the fourth-order (p = 3) DG scheme on the mesh with 600764 tetrahedral elements (i.e. containing a total of over 12 million degrees of freedom per equation). As seen, our reference forces agree within two decimal places with the ones provided in the paper by Leicht and Hartmann [JCP 2010] and the slight difference may be caused by the different wall boundary conditions implemented in the current DG code. Fig. 2 displays streamlines originated from the sharp leading edge as well as slices of Mach number contours along the delta wing and the downstream. The generation of vortices near the wing tip region and their pattern behind the wing body can be clearly observed. Fig. 3 shows lift (a) and drag (b) error convergence as a function of length scale ($h = 1/\sqrt[3]{\mathrm{nDoFs}}$) for the DG discretizations, where the third-order DG scheme outperforms the second-order counterpart. For example, with the same length scale or number of degrees of freedom, the third-order DG scheme achieves a higher error level than the second-order DG scheme. Fig. 4 displays a comparison of the force error convergence in terms of work units. It can be observed that the computational cost decreases with increasing discretization order from second order to third order. The outperformance of the higher-order scheme is more evident when a tighter error tolerance is desired.

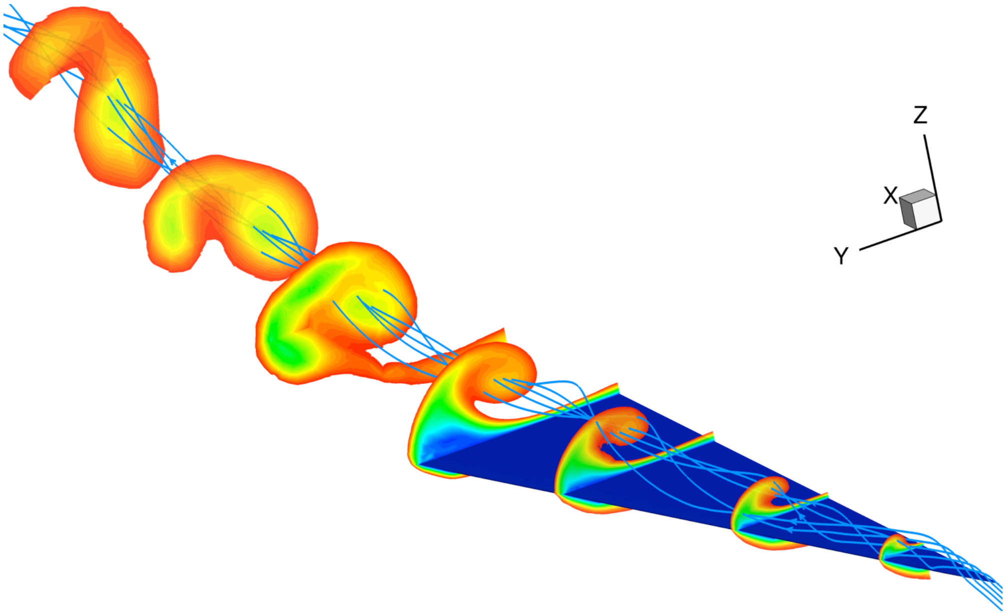
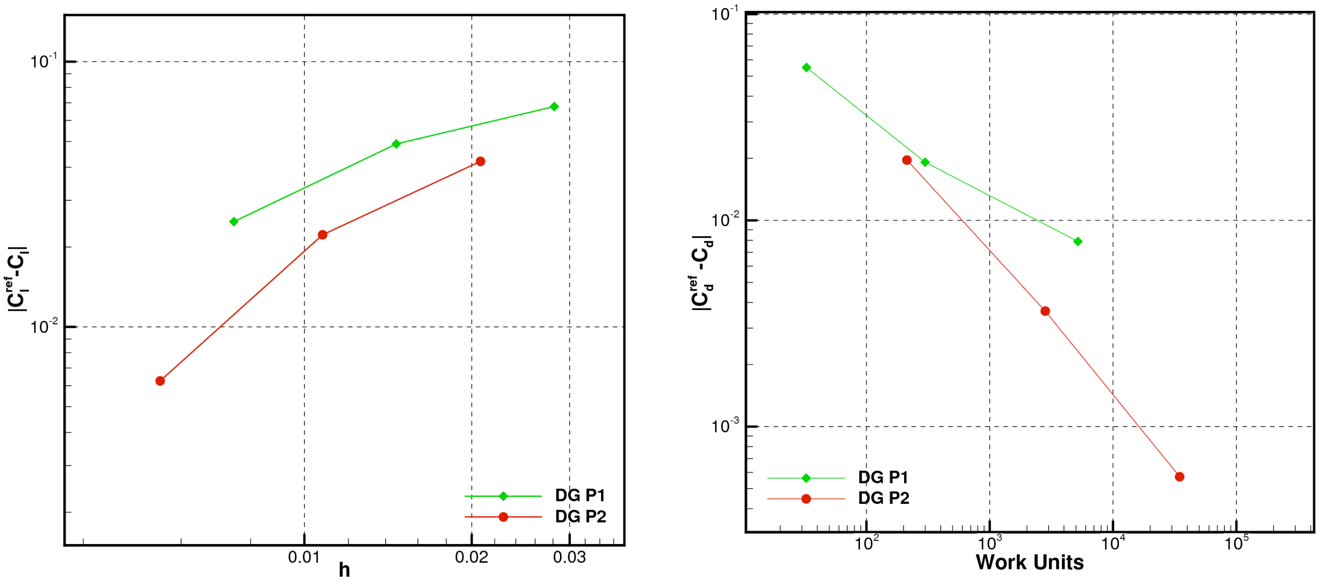


Figure 2. Streamlines and slices of Mach number contours along the delta wing and its downstream using the third-order DG scheme on Mesh 2 (containing approximately 0.6 million tetrahedrons).

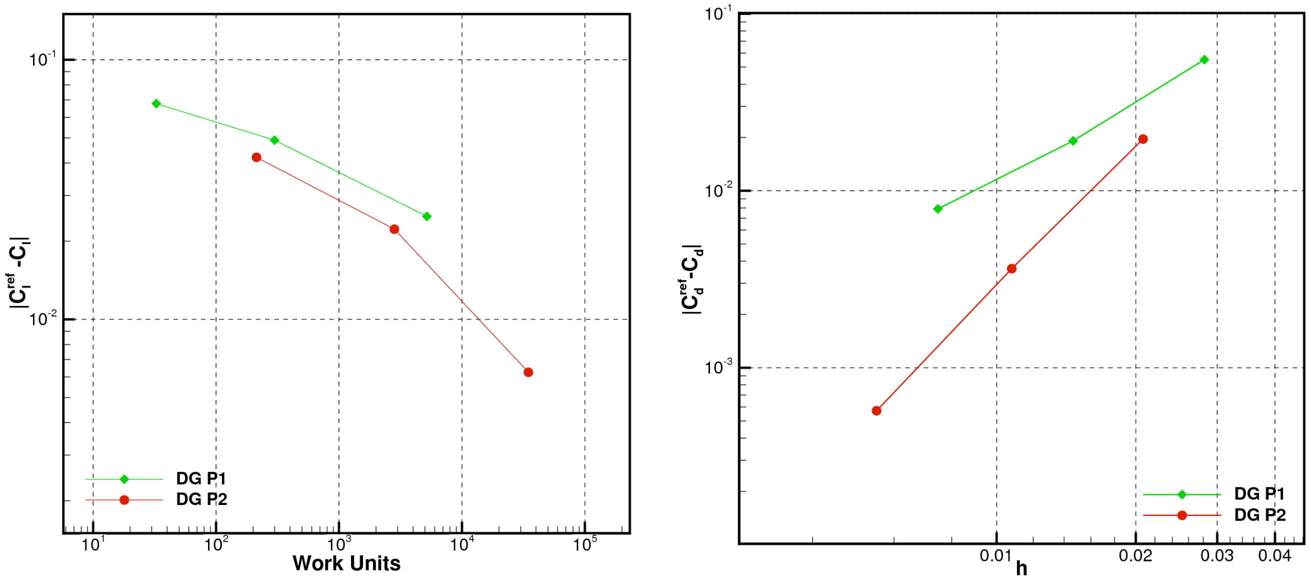
Streamlines and slices of the velocity magnitude near the half delta wing model are shown in Fig. 5 using the second-order accurate Petrov-Galerkin formulation on the finest mesh (Mesh 3). As seen, the solution looks very similar with the one obtained by the DG scheme. Table 3 lists the lift and drag coefficients computed using the second-order PG scheme on the same sequence of meshes used in the DG schemes, where the calculated forces via the second-order PG scheme are slightly different with the second-order DG discretization results. Additionally, some non-monotonic convergence is shown using the stabilized PG scheme. This behavior may be caused by different wall boundary conditions implemented in the two high-order flow solvers. In particular, the current PG formulation does not implement dual-consistent boundary conditions.



(a) C_l error vs. length scale

(b) C_d error vs. length scale

Figure 3. Convergence of lift and drag error as a function of length scale for the 2nd and 3rd accurate discontinuous Galerkin schemes in the laminar delta wing case (M=0.3, Re=4000, α=12.5°).



(a) C_l error vs. work units

(b) C_d error vs. work units

Figure 4. Convergence of lift and drag error as a function of work units for the 2nd and 3rd accurate discontinuous Galerkin schemes in the laminar delta wing case (M=0.3, Re=4000, α=12.5°).

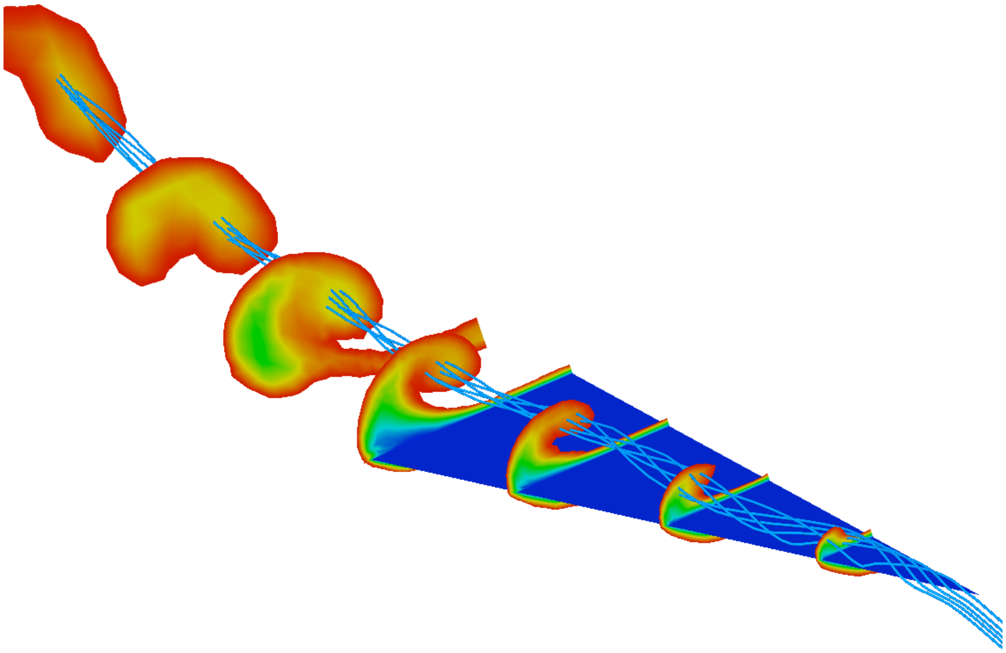


Figure 4. Streamlines and slices of velocity magnitude near the delta wing using the 2nd-order Petrov-Galerkin scheme on Mesh 3 (containing approximately 4.6 million tetrahedrons).

Force coefficients	Mesh 0	Mesh 1	Mesh 2	Mesh 3
C_l	0.4003309	0.3404456	0.3579097	0.3454654
C_d	0.1876298	0.1599375	0.1582711	0.1614232

Table 3. Force coefficients computed by the second-order PG scheme in the laminar delta wing case.