Case C1.1: Inviscid Flow through a Channel with a Gaussian Bump

Masayuki Yano* and David L. Darmofal†

Aerospace Computational Design Laboratory, Massachusetts Institute of Technology

I. Code Description

ProjectX is an adaptive discontinuous Galerkin finite element solver. The DG discretization uses Roe's approximate Riemann solver[1] for the inviscid numerical flux. The solution to the discretized system is obtained using a Newton-based nonlinear solver with pseudo-time continuation; however, due to the simplicity of this case, we employ a very high CFL number (10^6) to achieve near-Newton convergence. The linear system arising in each pseudo-time step is solved using GMRES,[2] preconditioned with an in-place block-ILU(0) factorization[3] with minimum discarded fill reordering and $p = 0$ algebraic coarse correction.[4]

An output-based, anisotropic simplex mesh adaptation algorithm is used control the discretization error.[5] The algorithm iterates toward a mesh that minimizes the output error for a given number of degrees of freedom. The anisotropic adaptation decisions are entirely driven by the behavior of an output-based *a posteriori* error estimate; thus, the method handles any discretization order, naturally incorporates both the primal and adjoint solution behaviors, and robustly treats irregular features. The output error estimate uses the dual-weighted residual (DWR) method of Becker and Rannacher.[6] A new mesh that conforms to the metric request is generated using using BAMG (Bidimensional Anisotropic Mesh Generator),[7] and higher-order, globally curved meshes are constructed through linear elasticity.[8]

II. Case Description

II.A. Flow Condition

This case considers inviscid flow through a channel with a Gaussian bump. The freestream Mach number is $M_\infty = 0.5$. The total temperature, total pressure, and the flow angle are specified at the inflow, and the static pressure is specified at the outflow. The inviscid wall condition is specified along the upper and lower walls.

II.B. Convergence Criterion

The ℓ^2 norm of the DG residual of non-dimensionalized Navier-Stokes equations is used to monitor convergence to the steady state. Our solver operates on non-dimensionalized variables

$$\rho^* = \frac{\rho}{\rho_\infty}, \quad u^* = \frac{u}{\|V_\infty\|}, \quad v^* = \frac{u}{\|V_\infty\|}, \quad p^* = \frac{p}{\rho_\infty \|V_\infty\|^2}, \quad e^* = \frac{e}{\|V_\infty\|^2},$$
$$R^* = \frac{R}{c_v}, \quad T^* = \frac{T}{\|V_\infty\|^2/c_v}, \quad \text{and} \quad \mu^* = \frac{\mu}{\rho_\infty L_\infty V_\infty}.$$

The DG residual is computed against the Lagrange test functions with equidistributed nodes, and the ℓ^2 norm of the residual is converged to 1×10^{-9}. (Note that the solver time would not be significantly influenced for any reasonable choice of the tolerance (say $< 1 \times 10^{-7}$), as we achieve Newton convergence in this regime. With the specified non-dimensionalization, the difference between the ℓ^2 residual and the mass residual is well within this offset.)

*Doctoral candidate, 77 Massachusetts Ave. 37-442, Cambridge, MA, 02139, myano@mit.edu
†Professor, 77 Massachusetts Ave. 37-451, Cambridge, MA, 02139, darmofal@mit.edu

II.C. Hardware Specification

All computations are performed in serial on a Linux machine with an Intel i7-2600 processor and 16 Gbytes of RAM. The machine produces a Taubench time of 6.60 seconds.

II.D. Residual Timing

The time for performing a single dof $= 250,000$ residual evaluation, including the full Jacobian construction for the implicit solver, is summarized in Table 1. The residual evaluation is performed on a 10800-element mesh and the times are scaled to 250,000 degrees of freedom.

p	time (work unit)
1	1.02
2	0.78
3	0.88

Table 1. dof $= 250,000$ residual evaluation time (including the full Jacobian construction).

II.E. Initial Mesh

The initial mesh used for this case is shown in Figure 1. The bump geometry is represented using $q = 5$ simplex elements. After each mesh refinement, the geometry representation is refined by re-sampling from the analytical Gaussian geometry.

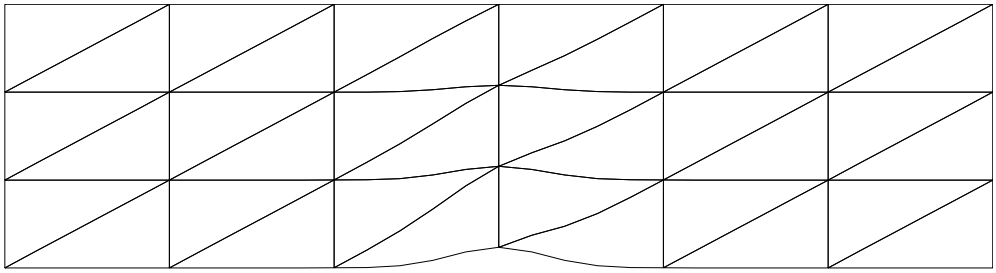


Figure 1. The initial 36-element mesh.

II.F. Adaptation Procedure and Data Reported

For this case, the ranges of the solution orders, p, and the number of degrees of freedom, dof, considered are

$$p \in \{1, 2, 3\} \quad \text{and} \quad \text{dof} \in \{250, 500, 1000, 2000\}.$$

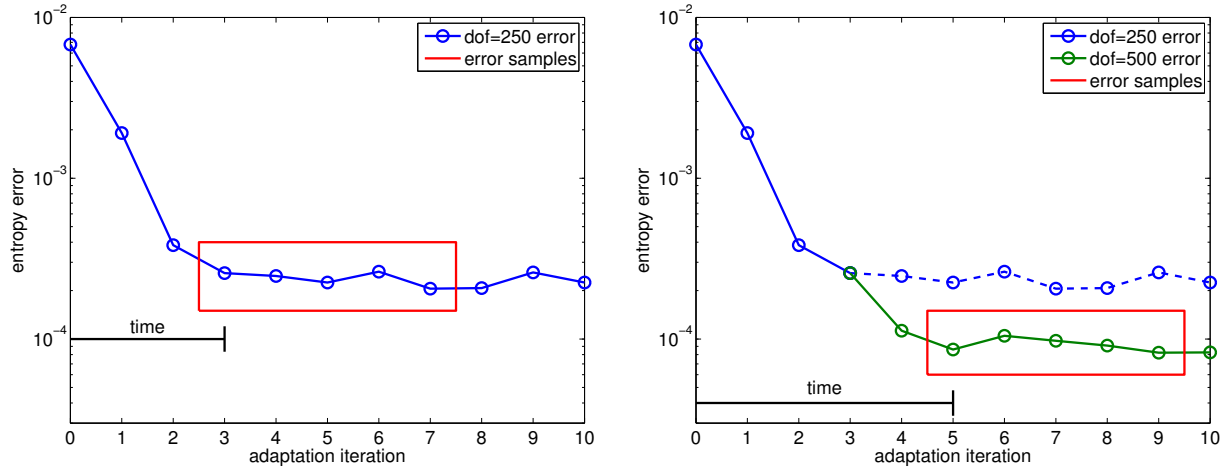
For each p-dof combination, a family of optimized meshes are generated using our anisotropic simplex mesh adaptation algorithm.[5] The output adapted, J, is the difference in the local and freestream entropy measured in the L^2 norm (squared), i.e.

$$J = \int_\Omega \left(\frac{p}{\rho^\gamma} - \frac{p_\infty}{\rho_\infty^\gamma} \right)^2 dx,$$

where p is pressure, ρ is density, p_∞ is the freestream pressure, ρ_∞ is the freestream density, and $\gamma = 1.4$ is the ratio of specific heats.

To illustrate the behavior of the adaptation algorithm for a given p and dof, an example of the entropy-error history obtained for the $p = 1$, dof $= 250$ case is shown in Figure 2(a). Starting from the initial mesh shown in Figure 1, the error decreases for the first three adaptation iterations. The error is stationary after three adaptation iterations, indicating that the mesh has been optimized for this p-dof combination. All meshes generated after the third adaptation iteration belong to a family of optimal meshes, achieving similar error levels but having slightly different triangulations. The "expected" performance of a particular p-dof

combination is quantified by taking the mean of the errors obtained on the five realizations of the meshes in the family. (Note that the mean of the errors, not the error of the mean, is used.) The time reported for the p-dof combination is the time required to reach the first realization of the optimized family. In particular, the timing includes all four flow solves and adaptation overhead required to reach the optimized mesh starting from the initial mesh.



(a) $p = 1$, dof $= 250$ adaptation

(b) $p = 1$, dof $= 500$ adaptation

Figure 2. Illustration of the procedure used to measure the error and the time.

The adaptation history for the next p-dof combination, $p = 1$ and dof $= 500$, is shown in Figure 2(b). Note that the $p = 1$, dof $= 250$-optimized mesh is used as the initial mesh for this new p-dof combination. As a result, the adaptation algorithm produces an optimal mesh in just two more adaptation iterations. As in the $p = 1$, dof $= 250$ case, the next five realizations of the optimal mesh is used to measure the error for this configuration. The time reported for this p-dof combination is the total time required to reach the first optimal mesh starting from the initial mesh shown in Figure 1. Note that this includes the time required to optimize the $p = 1$, dof $= 250$ configuration prior to optimizing the $p = 1$, dof $= 500$ configuration. Following the same procedure, a family of dof $= 1000$ optimized meshes is generated starting from a dof $= 500$ optimized mesh, and a family of dof $= 2000$ optimized meshes is generated starting from a dof $= 1000$ optimized mesh.

The optimal meshes for $p = 2$ and $p = 3$ cases are generated starting from the initial mesh shown in Figure 1 using the same procedure as the $p = 1$ case. In particular, higher p meshes do not use the $p = 1$ meshes as the starting mesh.

III. Results

III.A. Error Convergence

Figure 3(a) shows the convergence of the L^2 norm of the entropy error normalized by the domain area,

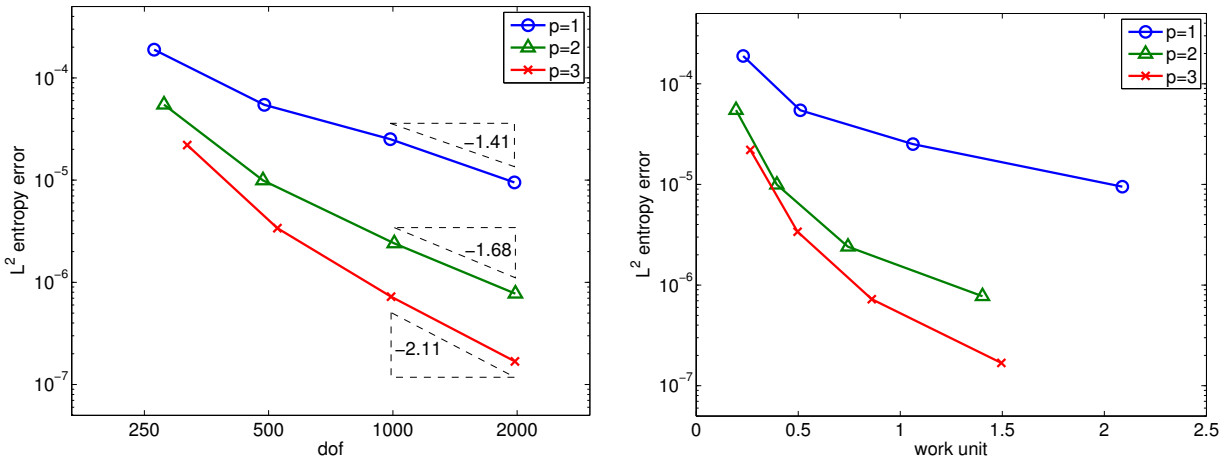
$$\mathcal{E}_{\text{ent}} = \sqrt{\frac{1}{|\Omega|}\int_\Omega \left(\frac{p/\rho^\gamma - p_\infty/\rho_\infty^\gamma}{p_\infty/\rho_\infty^\gamma}\right)^2 dx},$$

against the number of degrees of freedom, dof. For this smooth problem, the expected convergence rate of the L^2 norm of the entropy error is

$$\mathcal{E}_{\text{ent}} = Ch^{p+1} = \tilde{C}(\text{dof})^{-\frac{p+1}{2}}. \tag{1}$$

In particular, for $p = 1$, 2, and 3, the theoretical convergence rate against (dof) is 1, 1.5, and 2, respectively. The figure shows that all discretization orders produce the optimal convergence rate. Note that, because the flow is smooth, adaptive refinement is unnecessary to achieve the optimal convergence rate. However, adaptation does reduce the values of error for a given number of degrees of freedom, i.e. adaptation decreases

the value of the constant C (or $\tilde{C}$) in the error expression Eq. (1), as will be seen in Section III.C. In general, $p > 1$ discretizations are more efficient than the $p = 1$ discretization both in terms of the degrees of freedom (Figure 3(a)) and time (Figure 3(b)).



(a) entropy error vs. dof

(b) entropy error vs. work unit

Figure 3. Area-normalized L^2 entropy error convergence.

III.B. Comparison of Adapted Meshes

Entropy-adapted meshes obtained for select p-dof combinations are shown in Figure 4. In general, the adaptive refinement targets the bump region where the flow undergoes a rapid change; however, the element-size grading is weak as the flow feature is smooth, i.e. non-singular. Furthermore, even though an anisotropic adaptation algorithm is employed, the resulting meshes are mostly isotropic due to the absence of anisotropic flow features.

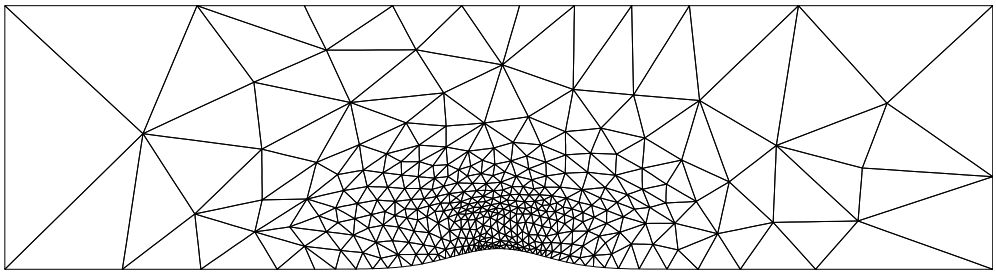
Comparison of the $p = 1$, dof $= 2000$ mesh (Figure 4(a)) and $p = 2$, dof $= 500$ mesh (Figure 4(b)) reveals the difference in the meshes required to achieve the error level of $\mathcal{E} \approx 10^{-5}$ using the $p = 1$ and $p = 2$ discretizations. The $p = 2$ mesh is significantly sparser than the $p = 1$ mesh, requiring eight times fewer elements.

Comparison of the $p = 2$, dof $= 500$ mesh (Figure 4(b)) and the $p = 2$, dof $= 2000$ mesh (Figure 4(c)) shows how the mesh evolves for a given p to achieve a lower error level (or as the number of degrees freedom increases). Once an optimal element size distribution is obtained for a lower number of degrees of freedom, the successive refinement is mostly uniform as the flow is smooth.

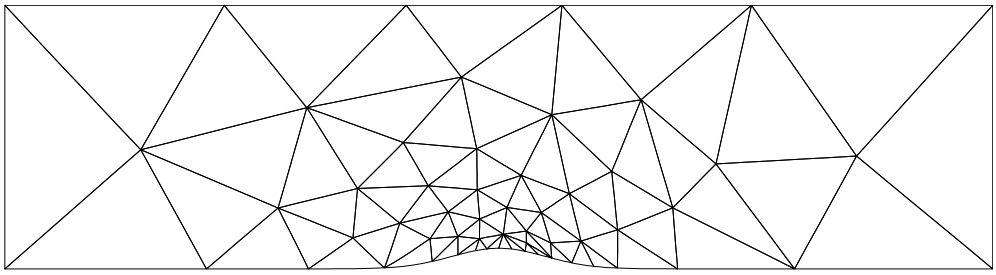
III.C. Uniform vs. Adaptive Refinement

This section compares the effect of adaptation on the accuracy and timing of the solution. Figure 5 shows the convergence results obtained using 1) adaptive refinement, 2) uniform refinement starting from an adapted mesh, and 3) uniform meshes. The adaptive refinement results are obtained using the anisotropic simplex mesh adaptation algorithm. The uniform "refinement" results are obtained by uniformly refining the dof $=$ 500 adapted meshes by subdividing each element into four elements. (Note that the uniform "refinement" meshes are not uniform "meshes" because they result from a step of uniform refinement of an adapted mesh.) The uniform "meshes" have uniform element spacings, resulting from uniformly refining the initial mesh shown in Figure 1.

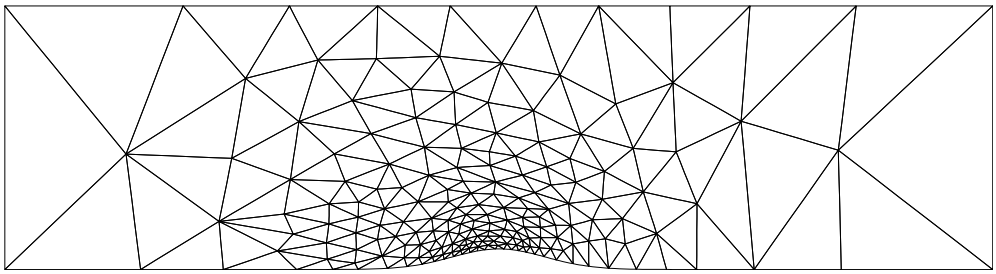
Because the problem is smooth, the optimal convergence rate of $(\text{dof})^{-\frac{p+1}{2}}$ is obtained using either adaptive or uniform refinement. However, even for this simple problem, adaptive refinement produces a significantly more accurate solution than on uniform meshes due to a smaller value of C (or $\tilde{C}$) in Eq. (1). In particular, adaptive refinement requires approximately 30 times fewer degrees of freedom than uniform meshes to achieve a given error level. The reduction is explained by examining the adapted meshes shown in Figure 4; the elements in the vicinity of the bump are significantly smaller than those away from the bump. Thus, even for this smooth, simple problem, adaptation is effective in improving the error-to-dof efficiency.



(a) $p = 1$, dof $= 2000$, $\mathcal{E} = 9.5 \times 10^{-6}$



(b) $p = 2$, dof $= 500$, $\mathcal{E} = 9.9 \times 10^{-6}$



(c) $p = 2$, dof $= 2000$, $\mathcal{E} = 7.7 \times 10^{-7}$

Figure 4. Select entropy-adapted meshes.

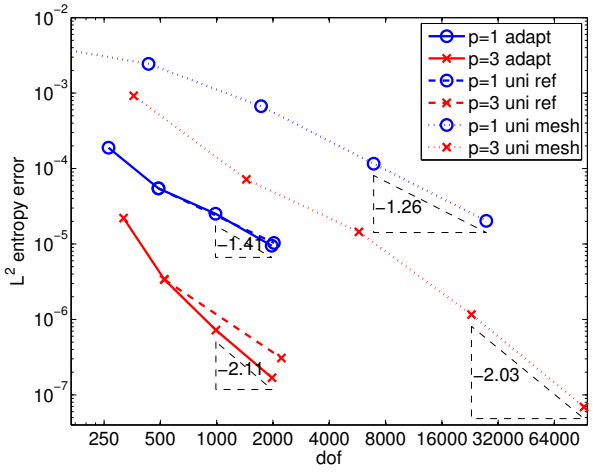
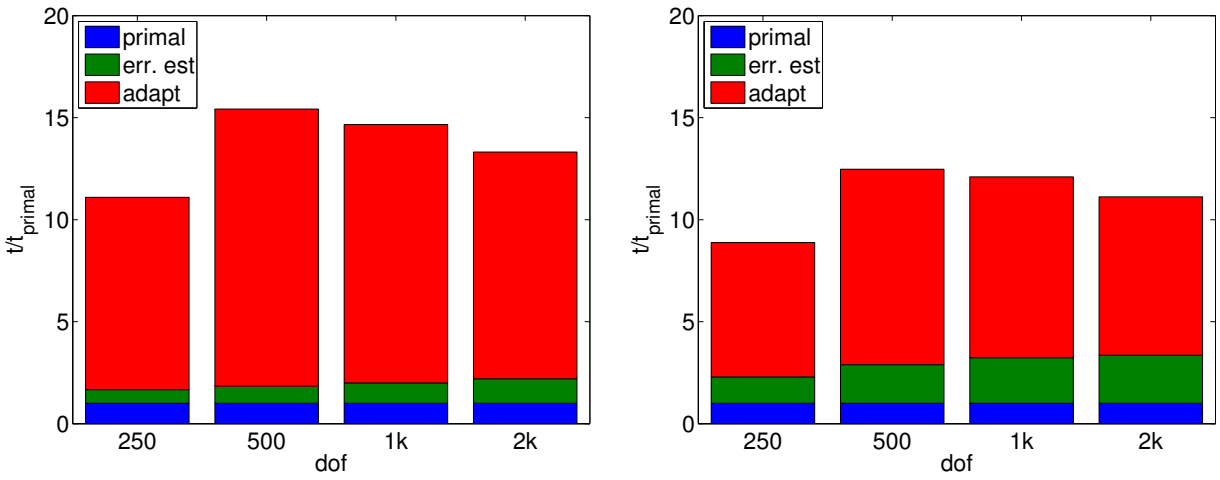


Figure 5. Comparison of adaptive refinement, uniform refinement starting from an adapted mesh, and uniform mesh.

Figure 6 shows the timing breakdown of the adaptation overhead relative to the primal solve (i.e. the flow solve). Due to the small problem size and the ease of converging the primal problem, adaptation overhead is significant for this problem. However, the 30-fold increase in the dof-to-error efficiency warrants the use of the adaptive algorithm even for this simple case.



(a) $p = 1$ (b) $p = 3$

Figure 6. Timing breakdown.

Acknowledgments

The authors would like to thank the entire ProjectX team for the many contributions that enabled this work. This work was supported by the Singapore-MIT Alliance Fellowship in Computational Engineering and The Boeing Company with technical monitor Dr. Mori Mani.

References

[1]P. L. Roe, Approximate Riemann solvers, parameter vectors, and difference schemes, J. Comput. Phys. 43 (2) (1981) 357–372.

[2]Y. Saad, M. H. Schultz, GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM Journal on Scientific and Statistical Computing 7 (3) (1986) 856–869.

[3]L. T. Diosady, D. L. Darmofal, Preconditioning methods for discontinuous Galerkin solutions of the Navier-Stokes equations, J. Comput. Phys. 228 (2009) 3917–3935.

[4]P.-O. Persson, J. Peraire, Newton-GMRES preconditioning for discontinuous Galerkin discretizations of the Navier-Stokes equations, SIAM J. Sci. Comput. 30 (6) (2008) 2709–2722.

[5]M. Yano, D. Darmofal, An optimization framework for anisotropic simplex mesh adaptation: application to aerodynamic flows, AIAA 2012–0079 (Jan. 2012).

[6]R. Becker, R. Rannacher, An optimal control approach to a posteriori error estimation in finite element methods, in: A. Iserles (Ed.), Acta Numerica, Cambridge University Press, 2001.

[7]F. Hecht, Bamg: Bidimensional anisotropic mesh generator, `http://www-rocq1.inria.fr/gamma/cdrom/www/bamg/eng.htm` (1998).

[8]T. A. Oliver, A higher-order, adaptive, discontinuous Galerkin finite element method for the Reynolds-averaged Navier-Stokes equations, PhD thesis, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics (Jun. 2008).