Shock Capturing for High-Order Discontinuous Galerkin Simulation of Transient Flow Problems

Per-Olof Persson*

University of California, Berkeley, Berkeley, CA 94720-3840, U.S.A.

We present some recent results for shock capturing using high-order discontinuous Galerkin schemes on fully unstructured meshes. We study the application of sensor-based artificial viscosity to problem with moving shocks, time-stepped using high-order accurate implicit schemes with Jacobian-based Newton-Krylov solvers. We demonstrate that the sensors can be coupled weakly without losing robustness, which simplifies the implementation and reduces the computational times. The weak coupling also allows for non-compact regularization of the artificial viscosity, and we show how different levels of smoothness in the sensor affect the solutions. We present a number of benchmarks and applications of our methods, including transonic, supersonic, and hypersonic flow problems in two and three space dimensions.

I. Introduction

While it is clear that the discontinuous Galerkin (DG) and related high-order methods[1–3] are getting sufficiently mature to handle realistic problems, they are still suffering from the lack of nonlinear stability and their high sensitivity to under-resolved features. This directly affects the solution of important problems involving shocks and turbulence models, but it has also turned out to be a problem for simpler problems such as laminar or inviscid flows, if the meshes are not well adapted to the solution fields.

Several approaches have been proposed for handling shocks. One simple method is to calculate some sort of sensor that identifies the elements in the shock region and reduce the order of interpolating polynomials.[4, 5] This is usually combined with h-adaptivity to better resolve the shocks, and it can be quite satisfactory for in particular steady-state problems. More sophisticated approaches include limiting, for example based on weighted essentially non-oscillatory (WENO) concepts.[1, 6, 7]

In Ref. 8, we proposed a new strategy inspired by the early artificial viscosity methods,[9] which has proved to be effective in the context of high-order DG methods.[10–12] The method combines a highly selective spectral sensor, based on orthogonal polynomials, with a consistently discretized artificial viscosity added to the equations. The goal is to smooth the discontinuities in the solution to a width that is appropriately resolved by the mesh and the polynomial approximations, which means in particular that the method obtains subcell resolution for high-order discretizations. This gives a number of important benefits, and in Ref. 8 we demonstrated fully converged solutions using a consistent Newton method, for both transonic and supersonic flow problems.

In this work, we present our recent development of the artificial viscosity approach for transient problems with supersonic and hypersonic flows. In particular, we have found that for time-accurate simulations the sensor can be decoupled from the implicit solver without losing stability, which allows for a more efficient Newton solution procedure. This also allows us to process the sensor variable in a non-compact way, and we show that continuity or higher levels of smoothness significantly improves the resulting solutions. We show results for the Woodward-Colella problems of a forward-facing step, the double mach reflection, an implosion, as well as transonic flow over a 3-D wing.

*Assistant Professor, Department of Mathematics, University of California, Berkeley, Berkeley CA 94720-3840. E-mail: persson@berkeley.edu. AIAA Member.

II. Governing Equations and Spatial Discretization

We consider the Euler equations of gas dynamics with a simple Laplacian diffusion term added to each equation:

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_i}(\rho u_i) = \frac{\partial}{\partial x_j}(\varepsilon \frac{\partial \rho}{\partial x_j}), \tag{1}$$

$$\frac{\partial}{\partial t}(\rho u_i) + \frac{\partial}{\partial x_i}(\rho u_i u_j + p) = \frac{\partial}{\partial x_j}(\varepsilon \frac{\partial (\rho u_i)}{\partial x_j}), \quad \text{for } i = 1,2,3, \tag{2}$$

$$\frac{\partial}{\partial t}(\rho E) + \frac{\partial}{\partial x_i}\left(u_j(\rho E + p)\right) = \frac{\partial}{\partial x_j}(\varepsilon \frac{\partial (\rho E)}{\partial x_j}), \tag{3}$$

where ρ is the fluid density, u_1, u_2, u_3 are the velocity components, and E is the total energy. For an ideal gas, the pressure p has the form $p = (\gamma - 1)\rho(E - \frac{1}{2}u_k u_k)$, where γ is the adiabatic gas constant. The parameter ε controls the amount of viscosity. We write the system (1)-(3) in a compact form as a system of equations

$$\frac{\partial \boldsymbol{u}}{\partial t} + \nabla \cdot \boldsymbol{F}(\boldsymbol{u}) = \nabla \cdot (\varepsilon \nabla \boldsymbol{u}), \tag{4}$$

with solution vector $\boldsymbol{u} = (\rho, \rho u_1, \rho u_2, \rho u_3, \rho E)^T$ and flux vector $\boldsymbol{F}(\boldsymbol{u})$. We will consider two boundary conditions, standard characteristic far field conditions with prescribed free-stream flow $\boldsymbol{u}_\infty$, and adiabatic wall boundary conditions with prescribed fluxes $\boldsymbol{F}_{\text{wall}}(\boldsymbol{u}) = (0, pn_1, pn_2, pn_3, 0)$.

II.A. Discontinuous Galerkin discretization

For the spatial discretization of the physical domain, we use a standard nodal discontinuous Galerkin method. We denote the elements of the mesh by $\mathcal{T}_h = \{K\}$. Further, we introduce the finite element spaces $\mathcal{V}_h^p$ and Σ_h^p as:

$$V_h^p = \{\boldsymbol{v} \in [L^2(\Omega)]^5 \mid \boldsymbol{v}|_K \in [\mathcal{P}_p(K)]^5 \ \forall K \in \mathcal{T}_h\}, \tag{5}$$

$$\Sigma_h^p = \{\boldsymbol{\tau} \in [L^2(\Omega)]^{5\times 3} \mid \boldsymbol{\tau}|_K \in [\mathcal{P}_p(K)]^{5\times 3} \ \forall K \in \mathcal{T}_h\}, \tag{6}$$

where $\mathcal{P}_p(K)$ is the space of polynomial functions of degree at most $p \geq 1$ on K. To obtain a form that is appropriate for discretization using the CDG method, we multiply the system of equations (4) by test functions $\boldsymbol{v}, \boldsymbol{\tau}$ and integrate by parts. Our semi-discrete DG formulation is then expressed as: find $\boldsymbol{u}_h \in V_h^p$ and $\boldsymbol{q}_h \in \Sigma_h^p$ such that for all $K \in \mathcal{T}_h$, we have

$$\int_K \frac{\partial \boldsymbol{u}_h}{\partial t} \cdot \boldsymbol{v}\, dx + \int_K (\boldsymbol{F}(\boldsymbol{u}_h) - \varepsilon \boldsymbol{q}_h) : \nabla \boldsymbol{v}\, dx - \oint_{\partial K} \left(\widehat{\boldsymbol{F}(\boldsymbol{u}_h)} - \varepsilon \hat{\boldsymbol{q}}_h\right) \cdot \boldsymbol{v}\, ds = \boldsymbol{0}, \qquad \forall \boldsymbol{v} \in [\mathcal{P}_p(K)]^5, \tag{7}$$

$$\int_K \boldsymbol{q}_h : \boldsymbol{\tau}\, dx + \int_K \boldsymbol{u}_h \cdot (\nabla \cdot \boldsymbol{\tau})\, dx - \oint_{\partial K} (\hat{\boldsymbol{u}}_h \otimes \boldsymbol{n}) : \boldsymbol{\tau}\, ds = \boldsymbol{0}, \qquad \forall \boldsymbol{\tau} \in [\mathcal{P}_p(K)]^{5\times 3}. \tag{8}$$

To complete the description we need to specify the numerical fluxes for all element boundaries ∂K. The inviscid fluxes $\widehat{\boldsymbol{F}(\boldsymbol{u}_h)}$ are computed using Roe's method.[13] For the viscous fluxes $\hat{\boldsymbol{q}}_h$, $\hat{\boldsymbol{u}}_h$, we use a formulation based on the CDG method,[14] which is a slight modification of the LDG method[15] to obtain a compact and sparser stencil with improved stability properties. This compactness is important in a practical solver, where for example the parallelization is greatly simplified when elements only communicate with their neighbors. These fluxes also include a parameter C_{11} which often is set to zero (the so-called *minimal dissipation LDG method*,[16]). However, here we use the value $C_{11} = 10/h_{\min}$ where $h_{\min}$ is the height of the element, in order to provide additional nonlinear stabilization. At a boundary face, we impose the appropriate conditions weakly through the fluxes.

II.B. Semi-discrete equations

The actual discretization procedure is carried out using a standard finite element technique. We define a set of equidistributed nodes $\boldsymbol{x}_j$, $j = 1, \ldots, N_p$, within each element K, where for simplex elements $N_p = \binom{p+D}{D}$

in D spatial dimensions, and for block elements $N_p = (p+1)^D$. We then determine the shape functions as the Lagrange interpolation functions $\phi_i(\boldsymbol{x}) \in \mathcal{P}_p(K)$ such that $\phi_i(\boldsymbol{x}_j) = \delta_{ij}$. Using these, the solution in each element can be written in terms of its discrete expansion coefficients $\boldsymbol{u}_i$ as:

$$\boldsymbol{u}_h(\boldsymbol{x}) = \sum_{i=1}^{n} \boldsymbol{u}_i \phi_i(\boldsymbol{x}) \tag{9}$$

and similarly for the auxiliary variable $\boldsymbol{q}_h$, the test functions $\boldsymbol{v}, \boldsymbol{\tau}$, and the time-derivatives $\partial \boldsymbol{u}_h / \partial t$. We evaluate all integrals in (7),(8) using high-order Gaussian quadrature rules, and setting the test function expansion coefficients to the identity matrix we obtain the semi-discrete form of our equations:

$$\boldsymbol{M} \frac{d\boldsymbol{U}}{dt} = \boldsymbol{R}(\boldsymbol{U}), \tag{10}$$

for solution vector $\boldsymbol{U}$, mass matrix $\boldsymbol{M}$, and residual function $\boldsymbol{R}(\boldsymbol{U})$.

III. Stabilization by Artificial Viscosity

The main idea behind the resolution sensor is to determine the decay rate of the expansion coefficients of the solution in an orthogonal basis. For smooth solutions, the coefficients in the expansion are expected to decay very quickly. But when the solution is under-resolved, the strength of the discontinuity will dictate the rate of decay of the expansion coefficients. Our resolution sensor is based on the amount of the highest order coefficients for one of the solution components, within each element.

We first expand the solution within each element in terms of a hierarchical family of orthogonal polynomials. In 1-D we use the standard orthogonal Legendre polynomials, and for block elements in higher dimensions we use outer products of these. For simplex elements, we use the orthonormal Dubiner/Koornwinder basis[17, 18] within each element. In terms of these, we can write a scalar solution component u in terms of the orthogonal basis functions ψ_i as

$$u = \sum_{i=1}^{N_p} u_i \psi_i,$$

where N_p is the dimension of the solution space, as defined above. In all our examples we choose the density ρ as the scalar field, which in our experience this results in a highly sensitive yet selective shock indicator. We express the solution of order p within each element in terms of an orthogonal basis, we consider a truncated expansion of the same solution, and define a resolution indicator:

$$u = \sum_{i=1}^{N(p)} u_i \psi_i, \quad \hat{u} = \sum_{i=1}^{N(p-1)} u_i \psi_i, \quad s_e = \log_{10} \left(\frac{(u - \hat{u}, u - \hat{u})_e}{(u, u)_e} \right)$$

Next we determine an element-wise viscosity over each element e by the following smooth function,

$$\varepsilon_e = \begin{cases} 0 & \text{if} \quad s_e < s_0 - \kappa \ , \\ \frac{\varepsilon_0}{2} \left(1 + \sin \frac{\pi(s_e - s_0)}{2\kappa} \right) & \text{if} \quad s_0 - \kappa \le s_e \le s_0 + \kappa \ , \\ \varepsilon_0 & \text{if} \quad s_e > s_0 + \kappa \ . \end{cases} \tag{11}$$

Here, $s_e = \log_{10} S_e$ and the parameters $\varepsilon_0 \sim h/p$, $s_0 \sim 1/p^4$, and κ are chosen empirically sufficiently large so as to obtain a sharp but smooth shock profile.[8] These element-wise viscosities are used to define a field $\varepsilon(\boldsymbol{x}) = \varepsilon_{e(\boldsymbol{x})}$, where $e(\boldsymbol{x})$ identifies the element to which $\boldsymbol{x}$ belongs.

In addition to $\varepsilon(\boldsymbol{x})$, we will also consider two modifications with higher level of smoothness:

- For C^0-continuity, we form a new viscosity field $\varepsilon_0(\boldsymbol{x})$ by calculating for each mesh vertex the maximum viscosity ε_e at all neighboring elements e. These values are then interpolated linearly within each element.

- For C^2-continuity, we form a new viscosity field $\varepsilon_2(\boldsymbol{x})$ as follows. In 1-D, we fit an interpolating cubic spline to the element-wise viscosities at the center of each element. In 2-D, we use the Loop subdivision scheme[19] to interpolate the node values of $\varepsilon_0(\boldsymbol{x})$ and produce a C^2-continuous function everywhere except at irregular nodes. This procedure can be generalized to 3-D but we have not yet implemented this.

IV. Time Integration and Nonlinear Solvers

IV.A. Temporal Discretization

We integrate (10) in time using L-stable Diagonally Implicit Runge-Kutta schemes of the form:

$$\boldsymbol{K}_i = \boldsymbol{R}\Big(\boldsymbol{U}_n + \Delta t \sum_{j=1}^{s} a_{ij} \boldsymbol{K}_j\Big), \quad i = 1, \dots, s \tag{12}$$

$$\boldsymbol{U}_{n+1} = \boldsymbol{U}_n + \Delta t \sum_{j=1}^{s} b_j \boldsymbol{K}_j, \tag{13}$$

In particular, we use a three-stage, third-order accurate method[20] with $s = 3$ and the coefficients given by the Runge-Kutta tableaux below.

$$\begin{array}{c|c} c & A \\ \hline & b^T \end{array} = \begin{array}{c|ccc} \alpha & \alpha & 0 & 0 \\ \tau_2 & \tau_2 - \alpha & \alpha & 0 \\ 1 & b_1 & b_2 & \alpha \\ \hline & b_1 & b_2 & \alpha \end{array} \qquad \begin{array}{l} \alpha = 0.435866521508459 \\ \tau_2 = (1+\alpha)/2 \\ b_1 = -(6\alpha^2 - 16\alpha + 1)/4 \\ b_2 = (6\alpha^2 - 20\alpha + 5)/4 \end{array}$$

We also use the standard one-stage backward Euler method, for computing steady-state solutions and providing further stabilization in our adaptive solvers.

IV.B. Inexact Newton and Preconditioned Krylov Methods

For the implicit time-stepping, nonlinear systems of equations $\boldsymbol{K}_i = (\boldsymbol{U}_{n,i})$ must be solved. For this, we use an inexact Newton method with a Jacobian matrix $J = M - \alpha \Delta t A$, where Δt is the timestep, α is a parameter of order one, and $A = \partial \boldsymbol{R} / \partial \boldsymbol{U}$ is the Jacobian matrix of the steady-state problem. This matrix is computed and stored explicitly but re-used between the iterations as well as between the stages. This results in only one Jacobian calculation per timestep, which greatly reduces the assembly time without significantly impairing the Newton convergence.

To solve the linear systems of equations involving J, we use a preconditioned Newton-Krylov technique, consisting of an ILU-preconditioned restarted GMRES solver with element ordering by the Minimum Discarded Fill (MDF) algorithm.[21] Since we re-use the Jacobian matrix many times, we can also re-use the incomplete factorization and the total implicit solution time is dominated by the matrix-vector products and the backsolves in the GMRES method.

IV.C. Weakly coupled sensors

Since our viscosity fields depend on the solution in a non-local way, the Jacobian matrix has a wider stencil than the original DG scheme and it is difficult to compute. However, for time-accurate solutions we have found that it is sufficient to couple the sensor weakly to the Navier-Stokes equations. More specifically, for our DIRK time integrator we simply compute a viscosity field at each initial time and keep it constant for the entire timestep. This simplifies the nonlinear solvers significantly, since the solution-dependency of the sensor does not have to be considered in the Jacobian matrices.

V. Results

V.A. Regularity of the sensor

To demonstrate how a constant viscosity field ε will introduce new irregularities in the solution, we show in figure 1 the numerical solution of Burgers' equation using our three different viscosities – piecewise constant, C^0-continuous, and C^2-continuous. The piecewise constant case uses ε_e according to (11) directly, while the continuous versions $\varepsilon_0(\boldsymbol{x})$ and $\varepsilon_2(\boldsymbol{x})$ are calculated as described in Section III.

The plots clearly show that the piecewise constant viscosity introduces undesired oscillations in the solution. The continuous versions give significantly smoother solutions, however with small differences between the C^0 and the C^2 continuity.

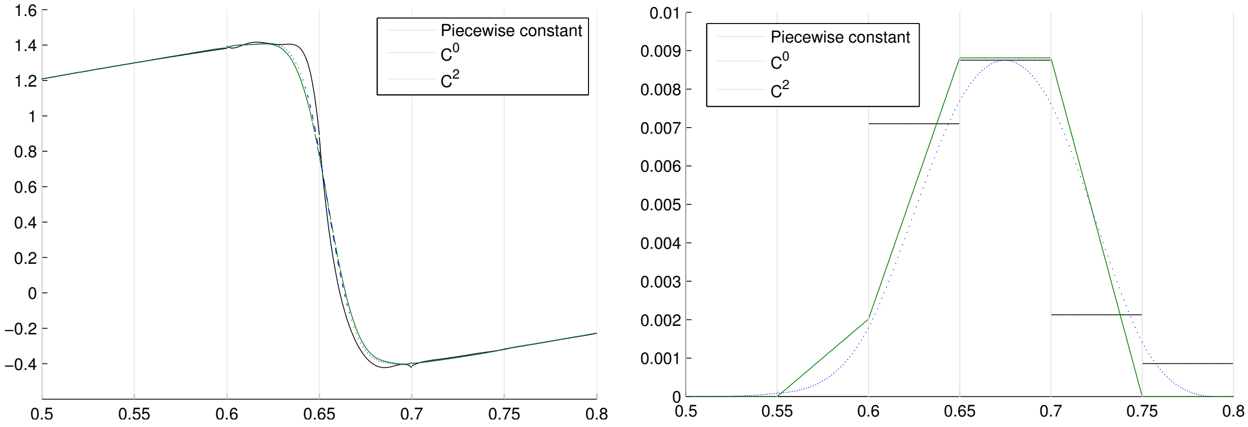
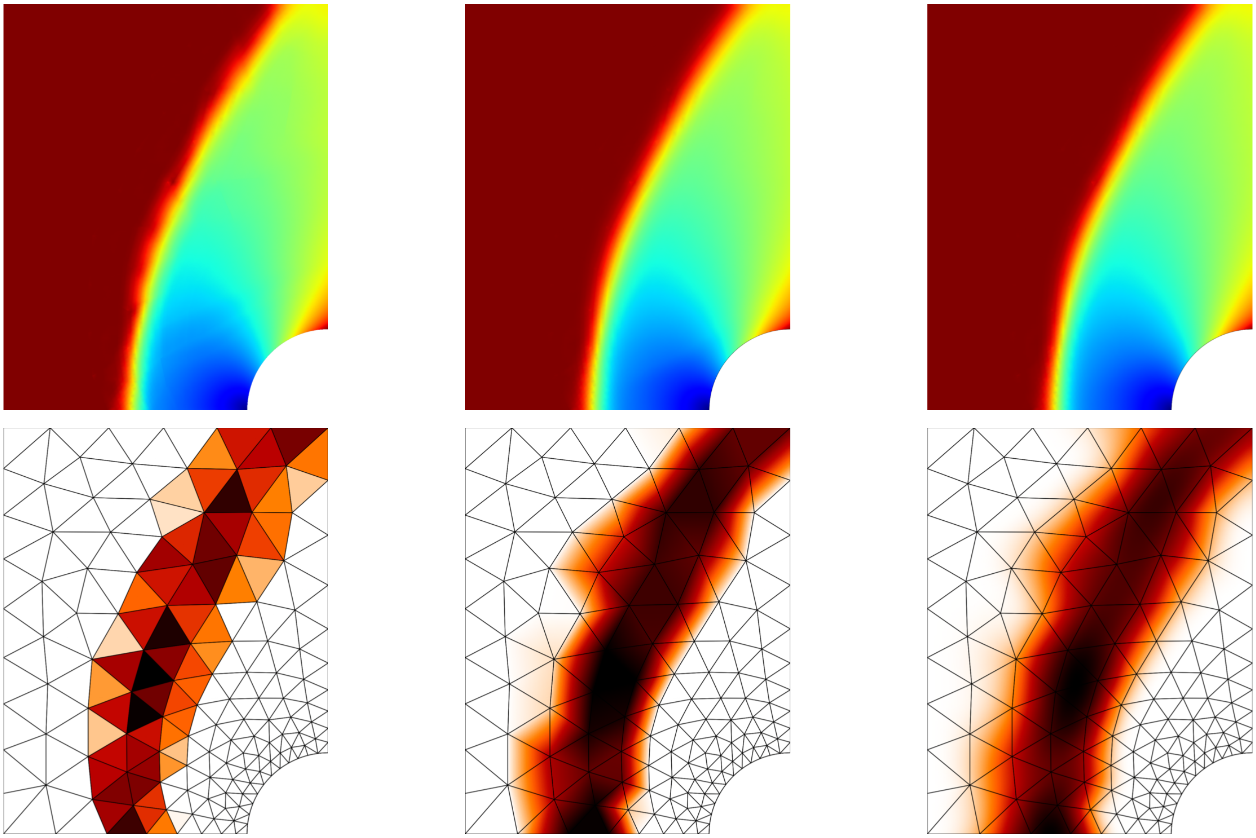


Figure 1. Burgers' equations with three different types of artificial viscosity, solution (left) and viscosity (right).



$\varepsilon(\boldsymbol{x})$ $\varepsilon_0(\boldsymbol{x})$ $\varepsilon_2(\boldsymbol{x})$

Figure 2. Flow around a cylinder at a free-stream Mach number of 2, using three different regularizations of the viscosity. The piecewise constant viscosity $\varepsilon(\boldsymbol{x})$ (left) gives considerably more oscillations than the C^0- and the C^2-continuous viscosities (middle and right).

Our second example shows the supersonic flow around a cylinder at a free-stream Mach number of 2. We use interpolating polynomials of degree $p = 4$, and compare again the 3 viscosities (element-wise constant, continuous piece-wise linear, and subdivision-based C^2-continuity). In figure 2, the Mach number is shown as color plots for the three cases, together with the viscosity fields. The results are similar to the 1-D example, with the element-wise constant viscosity showing clear oscillations in the solution, but small differences between the C^0- and the C^2-continuity.

V.B. The Woodward-Colella forward facing step

Next we show the forward facing step problem of Woodward and Colella.[22] The free-stream Mach number is 3, and we use polynomials of degree $p = 4$ for the space discretization. Figure 3 shows the solution (density), and the C^0-continuous artificial viscosity for two different meshes. We note that the viscosity is highly localized around the shock regions. The benefits of the fully implicit DIRK-scheme is demonstrated by the small elements at the corner, which would severely restrict the timestep for an explicit time integrator. For the finer mesh, we can observe a Kelvin-Helmholtz instability near the top of the domain, which appears to be well resolved by our high-order approximations.

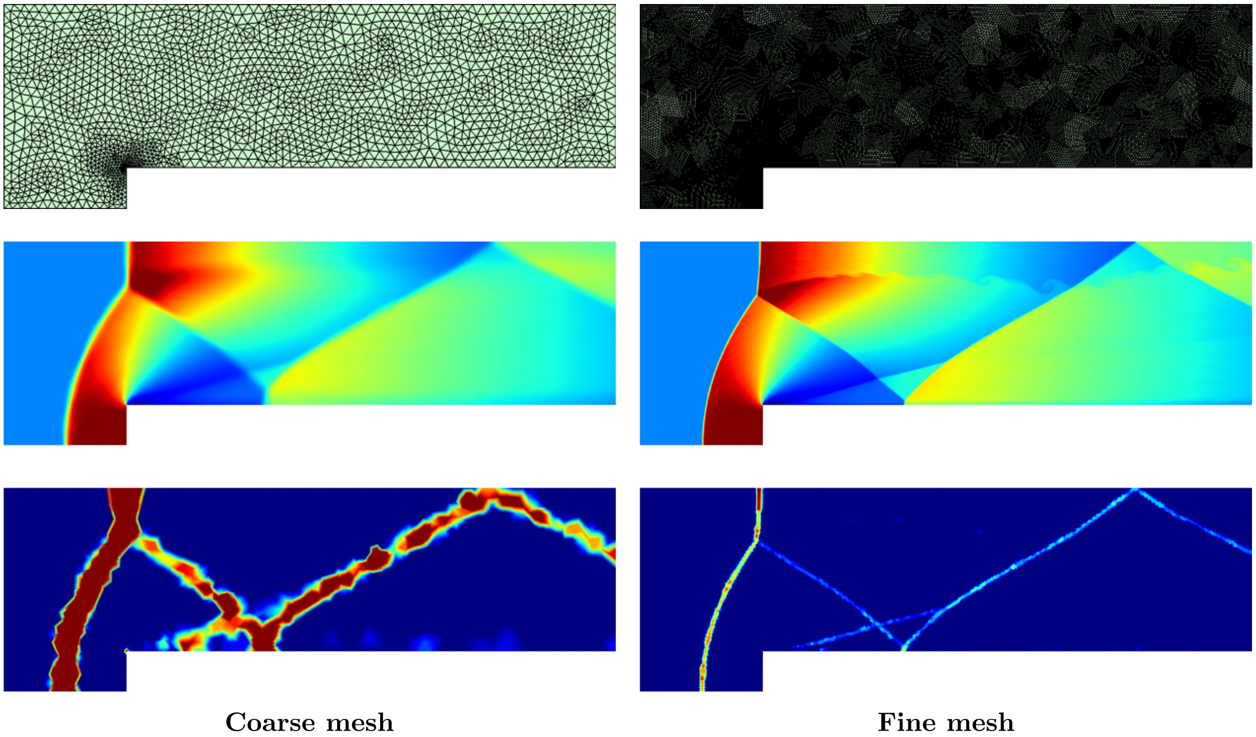


Figure 3. The Woodward-Colella forward facing step problem. The plots show three levels of mesh refinement, and include the mesh, the solution (density) and the C^0-continuous artificial viscosity $\varepsilon_0(\boldsymbol{x})$.

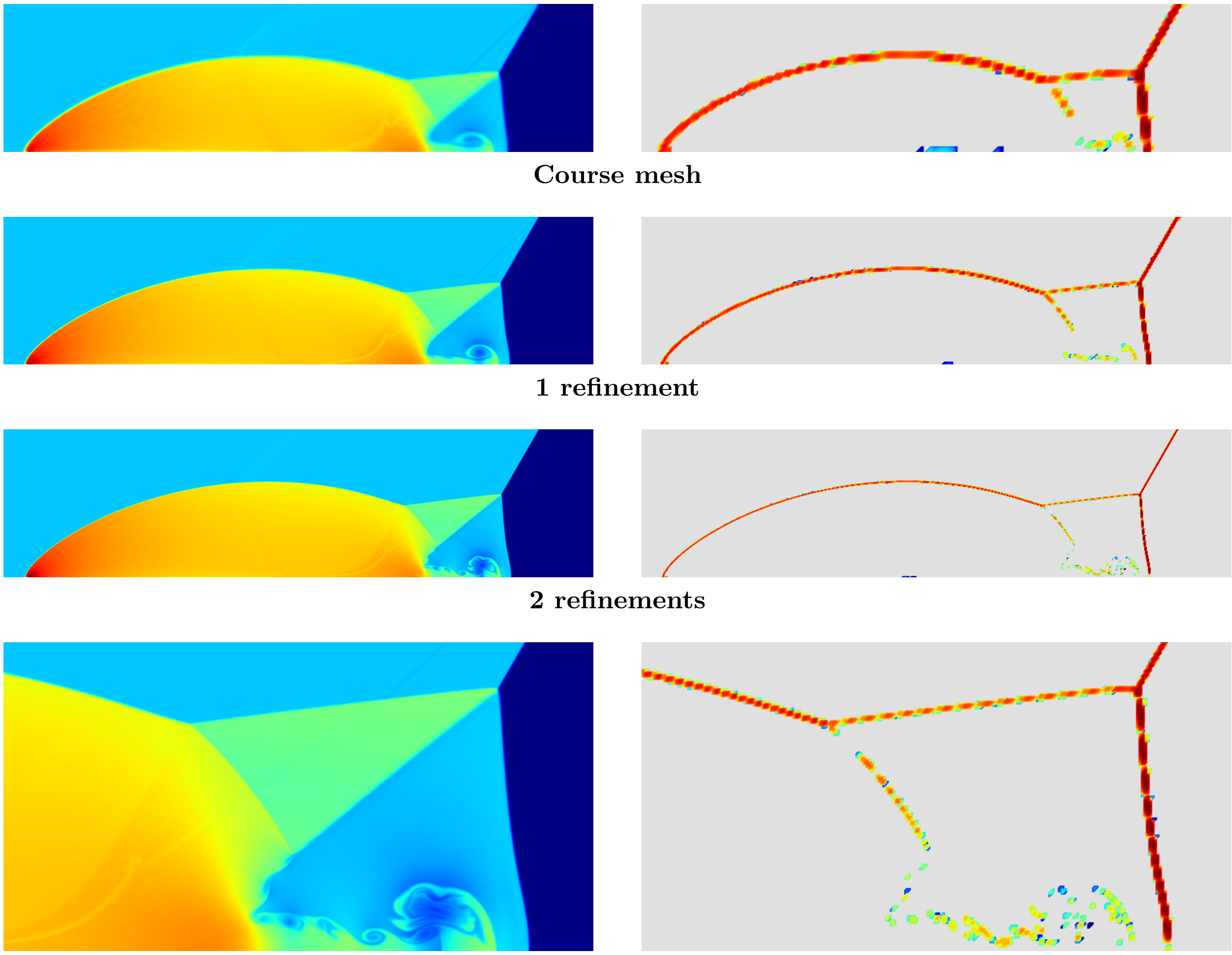
V.C. The Woodward-Colella double mach reflection

We also consider the double mach reflection problem of Woodward and Colella.[22] The hypersonic flow has a Mach number of 10, and a shock wave propagates diagonally into a wall and reflects, forming a jet of denser gas. Figure 4 shows the results for three levels of mesh refinement. The results are similar to the previous problem.

V.D. Implosion

Our next problem involves an implosion,[23] which is a good test of the numerical scheme's ability to maintain symmetry. The problem is solved in a square domain, and initiated by an overpressured region above the line $x + y = 0.5$ which sends a shock wave towards the origin. The shock waves are reflected at the boundaries and create a complex pattern of interacting jets and reflected shocks. The true solution is symmetric across the line $x = y$.

We deliberately use a low-quality, highly non-symmetric triangular mesh (figure 5, left). The solution field (middle) clearly shows the highly symmetric pattern, which is a good indication that the scheme is accurate.



Course mesh

1 refinement

2 refinements

Zoom-in

Figure 4. The Woodward-Colella double mach reflection problem. The plots show three levels of mesh refinement, with the density at the final time (left plots) and the C^0-continuous artificial viscosity $\varepsilon_0(\boldsymbol{x})$ (right plots). The bottom plots show a zoom-in of the lower-right region.

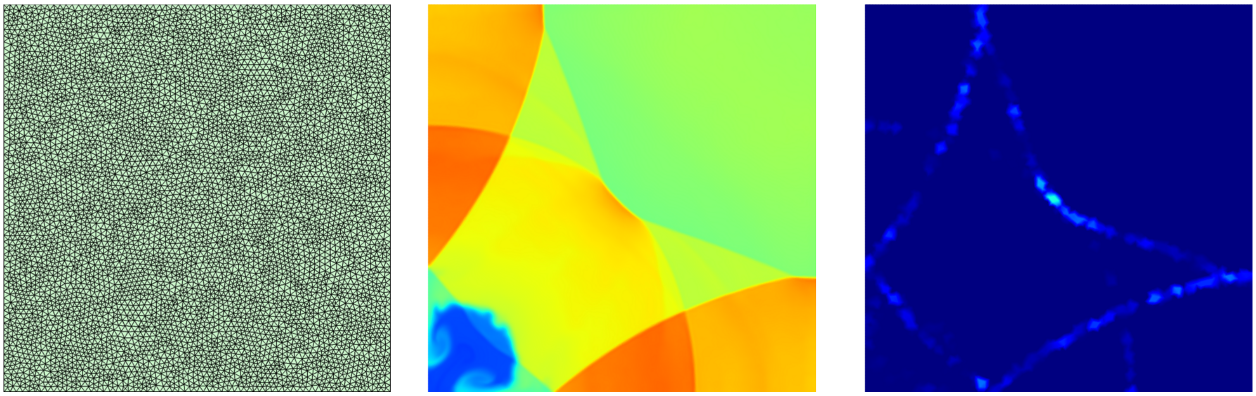


Figure 5. The implosion problem, with unstructured mesh (left), the density (middle), and the C^0-continuous artificial viscosity (right). The symmetry of the numerical solution is a good indication of the accuracy.

V.E. Tapered wing

Our final example is the transonic flow around a 3-D wing section, see figure 6. The free-stream Mach number is 0.85 and the angle of attack 3°. The solution is computed using polynomials of degree $p = 3$ and the C^0-continuous artificial viscosity. The figure shows the pressure at the wing and the symmetry plane, as well as the applied viscosity. We observe similar results as in 2-D, with a narrow band of artificial viscosity and smooth solutions with subcell resolution. The sensor is active in the shock region and along some of the sharp edges around the wing tip, showing how the scheme can be used to stabilize other under-resolved features.

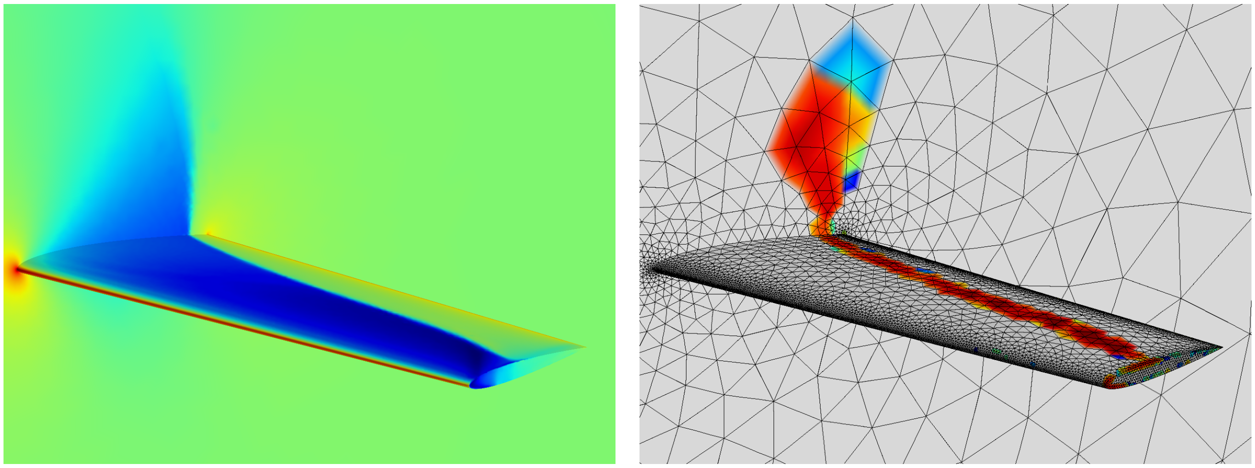


Figure 6. Transonic flow around a tapered wing, pressure (left) and the C^0-continuous artificial viscosity (right).

VI. Conclusions

We have demonstrated a high-order DG solver with implicit time-stepping for transonic, supersonic, and hypersonic flow problems. The artificial viscosity approach is used with a hierarchical sensor, and a simple de-coupled approach is used to integrate the systems in time. We demonstrate the importance of continuity in the sensor, and how the scheme can stabilize problems with strong shocks using only a narrow band of artificial viscosity.

References

[1] Cockburn, B. and Shu, C.-W., "Runge-Kutta discontinuous Galerkin methods for convection-dominated problems," *J. Sci. Comput.*, Vol. 16, No. 3, 2001, pp. 173–261.

[2] Hesthaven, J. S. and Warburton, T., *Nodal discontinuous Galerkin methods*, Vol. 54 of *Texts in Applied Mathematics*, Springer, New York, 2008, Algorithms, analysis, and applications.

[3] Peraire, J. and Persson, P.-O., *Adaptive High-Order Methods in Computational Fluid Dynamics*, Vol. 2 of *Advances in CFD*, chap. 5 – High-Order Discontinuous Galerkin Methods for CFD, World Scientific Publishing Co., 2011.

[4] Baumann, C. E. and Oden, J. T., "A discontinuous *hp* finite element method for the Euler and Navier-Stokes equations," *Int. J. Numer. Methods Fluids*, Vol. 31, No. 1, 1999, pp. 79–95, Tenth International Conference on Finite Elements in Fluids (Tucson, AZ, 1998).

[5] Burbeau, A., Sagaut, P., and Bruneau, C.-H., "A problem-independent limiter for high-order Runge-Kutta discontinuous Galerkin methods," *J. Comput. Phys.*, Vol. 169, No. 1, 2001, pp. 111–150.

[6] Shu, C.-W. and Osher, S., "Efficient implementation of essentially nonoscillatory shock-capturing schemes," *J. Comput. Phys.*, Vol. 77, No. 2, 1988, pp. 439–471.

[7] Shu, C.-W. and Osher, S., "Efficient implementation of essentially nonoscillatory shock-capturing schemes. II," *J. Comput. Phys.*, Vol. 83, No. 1, 1989, pp. 32–78.

[8] Persson, P.-O. and Peraire, J., "Sub-Cell Shock Capturing for Discontinuous Galerkin Methods," *44th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada*, 2006, AIAA-2006-0112.

[9] Von Neumann, J. and Richtmyer, R. D., "A method for the numerical calculation of hydrodynamic shocks," *J. Appl. Phys.*, Vol. 21, 1950, pp. 232–237.

[10] Barter, G. E., *Shock capturing with PDE-based artificial viscosity for an adaptive, higher-order discontinuous Galerkin finite element method*, Ph.D. thesis, M.I.T., June 2008.

[11] Bassi, F., Crivellini, A., Rebay, S., and Savini, M., "Discontinuous Galerkin solution of the Reynolds-averaged Navier-Stokes and $k-\omega$ turbulence model equations," *Computers & Fluids*, Vol. 34, No. 4–5, 2005, pp. 507–540.

[12] Nguyen, N. C., Persson, P.-O., and Peraire, J., "RANS Solutions Using High Order Discontinuous Galerkin Methods," *45th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada*, 2007, AIAA-2007-914.

[13] Roe, P. L., "Approximate Riemann solvers, parameter vectors, and difference schemes," *J. Comput. Phys.*, Vol. 43, No. 2, 1981, pp. 357–372.

[14] Peraire, J. and Persson, P.-O., "The compact discontinuous Galerkin (CDG) method for elliptic problems," *SIAM J. Sci. Comput.*, Vol. 30, No. 4, 2008, pp. 1806–1824.

[15] Cockburn, B. and Shu, C.-W., "The local discontinuous Galerkin method for time-dependent convection-diffusion systems," *SIAM J. Numer. Anal.*, Vol. 35, No. 6, 1998, pp. 2440–2463.

[16] Cockburn, B. and Dong, B., "An analysis of the minimal dissipation local discontinuous Galerkin method for convection-diffusion problems," *J. Sci. Comput.*, Vol. 32, No. 2, 2007, pp. 233–262.

[17] Dubiner, M., "Spectral methods on triangles and other domains," *J. Sci. Comput.*, Vol. 6, No. 4, 1991, pp. 345–390.

[18] Koornwinder, T., "Two-variable analogues of the classical orthogonal polynomials," 1975.

[19] Loop, C. T., *Smooth Subdivision Surfaces Based on Triangles*, Master's thesis, Department of Mathematics, University of Utah, August 1987.

[20] Alexander, R., "Diagonally implicit Runge-Kutta methods for stiff o.d.e.'s," *SIAM J. Numer. Anal.*, Vol. 14, No. 6, 1977, pp. 1006–1021.

[21] Persson, P.-O. and Peraire, J., "Newton-GMRES preconditioning for discontinuous Galerkin discretizations of the Navier-Stokes equations," *SIAM J. Sci. Comput.*, Vol. 30, No. 6, 2008, pp. 2709–2733.

[22] Woodward, P. and Colella, P., "The numerical simulation of two-dimensional fluid flow with strong shocks," *J. Comput. Phys.*, Vol. 54, No. 1, 1984, pp. 115–173.

[23] Liska, R. and Wendroff, B., "Comparison of several difference schemes on 1D and 2D test problems for the Euler equations," *SIAM J. Sci. Comput.*, Vol. 25, No. 3, 2003, pp. 995–1017 (electronic).