Oral presentation | Numerical methods

## Numerical methods-VII

Thu. Jul 18, 2024 2:00 PM - 4:00 PM  Room A

# [11-A-04] Exploring Interface Conservation in Computational Fluid Dynamics

*Hong Luo[1], Gianni Absillis[1], Xiaodong Liu[1]  (1. North Carolina State University)

Keywords: Interface Conservation, Discontinuous Galerkin Methods, grid adaptation methods

# Exploring Interface Conservation for Discontinuous Galerkin Methods in Computational Fluid Dynamics

H. Luo, G. Absillis and X. Liu
Corresponding author: hong_luo@ncsu.edu
Department of Mechanical and Aerospace Engineering, North Carolina State University, Raleigh, NC 27695, USA

**Abstract:** The necessity of enforcing conservation in computational elements or cells (element conservation) for discontinuous solutions is well understood and respected for solving conservation laws in computational fluid dynamics (CFD). In contrast, interface conservation, where the conservation across cell interfaces is enforced, is long ignored, and yet is also ruled and required by the underlying physics just like element conservation. The interface conservation is examined and explored thoroughly in this work. A novel error indicator based on the interface conservation is developed as a weight function in a *r*-adaptive grid method and implemented in a discontinuous Galerkin (DG) finite element method. A number of numerical experiments are conducted to assess the effectiveness of the interface conservation-based error indicator and performance of the resulting *r*-adaptive DG method. Numerical results for a variety of flow problems obtained demonstrate that the physics-based error indicator can reliably and efficiently detect and identify all types of discontinuities for inviscid flows and all under-resolved flow features and regions of high gradient solutions for viscous flows. The *r*-adaptive DG method is able to align mesh cell interfaces with discontinuities, and effectively offer anisotropic mesh adaption for both inviscid and viscous shock waves and isotropic mesh adaptation for singularities and under-resolved flow regions.

*Keywords:* Computational Fluid Dynamics, Interface Conservation, Discontinuous Galerkin Methods, *r*-Adaptive Mesh Method.

## 1 Introduction

The discontinuous Galerkin (DG) finite element methods [1-30] have become a popular choice to solve conservation laws with arbitrary order of accuracy. They are widely used in different computation areas including computational fluid dynamics, computational acoustics and computational magnetohydrodynamics. The discontinuous Galerkin methods have many attractive advantages like 1) its ability to achieve high-order (>2nd) accuracy on fully unstructured grids; 2) useful mathematical properties with respect to conservation, stability and convergence; 3) its adjoint consistency to be powerful for adjoint-based optimization. In addition, the methods can also handle non-conforming elements, where the grids are allowed to have hanging nodes. Furthermore, spacetime discontinuous Galerkin methods [31-33] provide discretization of systems of conservation laws by simultaneously discretizing space and time. Like other DG methods, the spacetime DG method also offers the prospect of both arbitrary-order accuracy in space and time and adjoint consistency. However, the DG methods have a number of weaknesses that have not yet be addressed. Besides of computational cost and storage requirement, one aspect is how the properties behave in flows that are not smooth and contain discontinuous interfaces, such as material interface and shocks. Even though DG explores a set of discrete function space with discontinuous, piecewise polynomials and it can represent the discontinuous interfaces in principal, this requires that the interfaces are aligned with the grids. The stability of the DG approach may fail when misaligned grid is used. Indeed, how to control spurious oscillations in the presence of strong discontinuities has been an outstanding issue, whose mathematically sound and numerically effective solution has been an active research subject for many decades.

Moving grid methods [34-37] are a widely used approach for solving a variety of flow problems in computational fluid dynamics. In Lagrangian hydrodynamics, grids are moved to tracking contact discontinuities and material interfaces. Arbitrary Lagrangian-Eulerian (ALE) methods are widely used for moving and deforming boundary problems. In R-adaptation methods, mesh points are moved into regions of needed high resolution, which can significantly increase solution accuracy. In shock-fitting schemes, grids are moved to track shock waves.

Recently, a moving discontinuous Galerkin finite element method with interface condition enforcement, termed MDG-ICE, was formulated for compressible flows with discontinuous interfaces by Corrigan et al. [38-41], where both conservative quantities and discrete grid geometry are treated as independent variables and both conservation laws and interface conservation (IC) are solved simultaneously in the space-time domain. In the MDG-ICE formulation, a space-time DG formulation is used to solve the governing equations in the standard discontinuous solution space, and the geometry variables are determined by enforcing the interface condition in its discontinuous solution trace space. A variant of MDG-ICE [42,43] was introduced by Luo et al., where a different variational formulation is used to enforce the interface conservation. Two attractive features of the MDG-ICE method, among others, are 1) no strategies in the form of a limiter or an artificial viscosity are required to eliminate spurious oscillations in the vicinity of discontinuities and thus maintain the nonlinear stability of the DG methods, as interfaces are detected by the interface condition enforcement, and tracked by the grid movement and the interface condition; and 2) no numerical fluxes in the form of a Riemann solver have to be needed to maintain linear stability of the DG methods. Numerical results obtained indicate that the MDG-ICE methods are able to deliver the designed order of $p$-convergence even for discontinuous solutions, and detect and fit all types of discontinuities and interactions of different discontinuities due to the interface condition enforcement and grid movement.

The objective of the efforts presented in this work is to develop a *r*-adaptive DG method for solving compressible viscous flow problems by exploring the interface conservation. A novel error indicator based on the interface conservation is introduced in the weight function of the *r*-adaptive grid method. A number of numerical experiments for both inviscid and viscous flow problems are conducted to assess the effectiveness of the interface conservation-based error indicator and performance of the resulting *r*-adaptive DG method. Preliminary numerical results for a variety of flow problems are highly promising and encouraging, demonstrating that the physics-based error indicator can reliably and efficiently detect and identify all types of discontinuities for inviscid flows and all under-resolved flow features and regions of high gradient solutions for viscous flows. The *r*-adaptive DG method is able to align mesh cell interfaces with discontinuities, and effectively offer anisotropic mesh adaption for both inviscid and viscous shock waves and isotropic mesh adaptation for singularities and under-resolved flow regions. The remainder of this paper is organized as follows. The governing equations are described in Section 2. The developed *r*-adaptive DG method is presented in Section 3. Numerical experiments are reported in Section 4. Concluding remarks and future work are given in Section 5.

# 2 Governing Equations

The Reynolds-averaged Navier-Stokes equations governing unsteady compressible viscous flows over any region $\Omega$ with boundary $\Gamma=\partial\Omega$ can be expressed as

$$\frac{\partial \mathbf{U}}{\partial t}+\frac{\partial \mathbf{F}_j}{\partial x_j} = \frac{\partial \mathbf{G_j}}{\partial x_j} \tag{2.1}$$

where the summation convention has been used. The conservative variable vector **U,** inviscid flux vector **F**, and viscous flux vector **G**, are defined by

$$\mathbf{U}=\begin{pmatrix}\rho\\ \rho u_i\\ \rho e\end{pmatrix} \qquad \mathbf{F}_{\boldsymbol{j}}=\begin{pmatrix}\rho u_i\\ \rho u_i u_j \quad + p\delta_{ij}\\ u_j(\rho e+p)\end{pmatrix} \qquad \mathbf{G}_{\boldsymbol{j}}=\begin{pmatrix}0\\ \sigma_{ij}\\ u_l(\sigma_{lj}+q_j)\end{pmatrix} \tag{2.2}$$

Here ρ, $p$, and $e$ denote the density, pressure, and specific total energy of the fluid, respectively, and $u_i$ is the velocity of the flow in the coordinate direction $x_i$. The pressure can be computed from the equation of state

$$p = (\gamma - 1)\rho(e - \frac{1}{2}u_j u_j) \tag{2.3}$$

which is valid for perfect gas, where γ is the ratio of the specific heats. The components of the viscous stress tensor $\sigma_{ij}$ and the heat flux vector are given by

$$\sigma_{ij} = \mu(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}) - \frac{2}{3}\mu \frac{\partial u_k}{\partial x_k}\delta_{ij} \qquad q_j = \frac{1}{\gamma - 1}\frac{\mu}{\Pr}\frac{\partial T}{\partial x_j} \tag{2.4}$$

In the above equations, $T$ is the temperature of the fluid, Pr the laminar Prandtl number, which is taken as 0.7 for air. μ represents the molecular viscosity, which can be determined through Sutherland's law

$$\frac{\mu}{\mu_0} = \left(\frac{T}{T_0}\right)^{\frac{3}{2}} \frac{T_0 + S}{T + S} \tag{2.5}$$

$\mu_0$ denotes the viscosity at the reference temperature $T_0$, and $S$ is a constant which for are assumes the value $S$ = 110°K. The temperature of the fluid $T$ is determined by

$$T = \gamma \frac{p}{\rho} \tag{2.6}$$

Neglecting viscous effects, the left-hand side of Eq. (2.1) represents the Euler equations governing unsteady compressible inviscid flows.

# 3 *r*-Adaptive Discontinuous Galerkin Method Based on the Interface Conservation

## 3.1 Discontinuous Galerkin method

The system of the governing Navier-Stokes equations is discretized in space using a DG finite element formulation. In a DG method, the computational domain Ω is divided by a set of non-overlapping control volumes $\Omega_i$. We use $\Gamma_e$ to denote the boundary of $\Omega_e$ and **n** the unit outward normal vector to $\Gamma_e$. We introduce the following broken Sobolev space $\mathbf{V}_h^p$

$$\mathbf{V}_h^p = \{v_h \in \{L_2(\Omega)\}^m:\ v_h|_{\Omega_e} \in\ V_p^m\ \ \forall\Omega_e \in \Omega\}, \tag{3.1}$$

which consists of discontinuous vector-valued polynomial functions of degree $p$, and where m is the dimension of the unknown vector and

$$\mathbf{V}_h^m = span\{\prod x_i^{\alpha_i}:\ \ 0 \le \alpha_i \le\ p, 0 \le i \le d\} \tag{3.2}$$

where α denotes a multi-index and d is the dimension of space. To formulate the discontinuous Galerkin method, we introduce the following weak formulation, which is obtained by multiplying the above Navier-Stokes equations (2.1) by a test function $w_h$, integrating over an element $\Omega_e$, and then performing an integration by parts,

$$
\begin{cases}
\text{Find } \mathbf{U}_h \in \mathbf{V}_h^p \text{ such that} \\
\dfrac{d}{dt}\int\limits_{\Omega_e} \mathbf{U}_h w_h d\Omega - \int_{\Omega_e} \mathbf{F}_{\boldsymbol{k}}(\mathbf{U}_h)\dfrac{\partial w_h}{\partial x_k}\, d\Omega + \int_{\Gamma_e} \mathbf{F}_{\boldsymbol{k}}(\mathbf{U}_h)\mathbf{n}_{\boldsymbol{k}}\, w_h\, d\Gamma = \\
\int_{\Gamma_e} \mathbf{G}_{\boldsymbol{k}}(\mathbf{U}_h)\mathbf{n}_{\boldsymbol{k}}\, w_h\, d\Gamma - \int_{\Omega_e} \mathbf{G}(\mathbf{U}_h)\dfrac{\partial w_h}{\partial x_k}\, d\Omega
\end{cases}
\tag{3.3}
$$

where $\mathbf{U}_h$ and $w_h$ are represented by piecewise-polynomial functions of degrees $p$, which are discontinuous between cell interfaces. Assume that $B_i$ is the basis of polynomial function of degrees $p$, Eq. (3.3) is then equivalent to the following system of N equations,

$$
\begin{cases}
\text{Find } \mathbf{U}_h \in \mathbf{V}_h^p \text{ such that} \\
\dfrac{d}{dt}\int\limits_{\Omega_e} \mathbf{U}_h B_i d\Omega - \int_{\Omega_e} \mathbf{F}_{\boldsymbol{k}}(\mathbf{U}_h)\dfrac{\partial B_i}{\partial x_k}\, d\Omega + \int_{\Gamma_e} \mathbf{F}_{\boldsymbol{k}}(\mathbf{U}_h)\mathbf{n}_{\boldsymbol{k}}\, B_i\, d\Gamma = \qquad , \\
\int_{\Gamma_e} \mathbf{G}_{\boldsymbol{k}}(\mathbf{U}_h)\mathbf{n}_{\boldsymbol{k}}\, B_i\, d\Gamma - \int_{\Omega_e} \mathbf{G}(\mathbf{U}_h)\dfrac{\partial B_i}{\partial x_k}\, d\Omega = 0, \qquad i \le i \le N
\end{cases}
\tag{3.4}
$$

where N is the dimension of the polynomial function space. Since the numerical solution $\mathbf{U}_h$ is discontinuous between element interfaces, the interface fluxes are not uniquely defined, and need to be computed carefully for the consideration of stability. This scheme is called discontinuous Galerkin method of degree p, or in short notation DG(P) method. By simply increasing the degree p of the polynomials, the DG methods of corresponding higher order are obtained. The domain and boundary integrals in Eq. (3.4) are calculated using Gauss quadrature formulas. The number of quadrature points used is chosen to integrate exactly polynomials of order of 2p and 2p+1 for volume and surface inner products in the reference element. In the DG methods, numerical polynomial solutions $\mathbf{U}_h$ in each element are expressed using either standard Lagrange finite element or hierarchical node-based basis as below

$$
\mathbf{U}_\mathrm{h} = \sum_{\mathrm{i}=1}^{\mathrm{N}} \mathbf{U}_\mathrm{i}\, \mathrm{B}_\mathrm{i}(\mathrm{x},\mathrm{t}) \tag{3.5}
$$

where $B_i$ are a set of the finite element basis functions. In the present work, the piecewise polynomial solutions are represented using a linear Taylor series expansion at the cell centroid, which can be expressed as a combination of cell-averaged variables and their gradients at the cell centers regardless of the element shapes. As a result, the very same numerical polynomial solutions are used for arbitrary shapes of elements, which can be triangle, quadrilateral, and polygon in 2D, and tetrahedron, pyramid, prism, and hexahedron in 3D.

## 3.2 *r*-adaptive grid method

Adaptive methods have been extensively developed and used to improve the accuracy of numerical solutions for the last few decades. The basic idea underlying most adaptive methods is to assess the quality of an initial numerical solution by using some form of a posteriori error estimate and then to dynamically change the mesh and/or the solution space, in a systematic manner, to improve the quality of the solution. In the *r*-adaptive mesh methods [46-49], grid points are repositioned in such a way that the grid is dense in regions of large error and coarse in regions of smoother solution. Most *r*-adaptive grid methods are based on the error equidistribution principle. The basic idea behind them is to equidistribute the solution error over mesh edges,

$$
\int\limits_{x_i}^{x_j} w(x)dx = \text{Constant} \tag{3.6}
$$

where $x_i$ and $x_j$ are the position for the two nodes of an edge *ij* and *w* is a positive scalar error function termed weight or monitor function. The solution of Eq. (3.6) is equivalent to solving the Euler-Lagrange equation,

$$\frac{d}{d\xi}\left(w\frac{dx}{d\xi}\right) = 0 \tag{3.7}$$

where *x* and ξ denote the physical and computational coordinates, respectively. When the mesh is viewed as a network of springs whose stiffness constants represent the edge-based weight function *w*, the solution of the Euler-Lagrange equation can be obtained from the solution of an energy minimization problem. For each node *i*,

$$\min_{x_i} P_i = \min_{x_i} \sum_j (x_i - x_j)^2 w_{ij} \tag{3.8}$$

where $P_i$ denotes the potential energy of the all active springs sharing the node i and $w_{ij}$ are their associated stiffness constants. After simplifying the constant and collecting the contributions of each node, Eq. (3.3) is reduced to the system describing the equilibrium state of a spring network,

$$\sum_j (x_i - x_j) w_{ij} = 0 \tag{3.9}$$

which is simply solved using a relaxation Jacobi method in this work.

## 3.3 Error indicator based on the interface conservation

An important step in the *r*-adaptive grid methods is to determine the weigh function *w*. Different error estimation techniques, which are used in a posteriori error estimation, can be classified into three major groups [46]: interpolation methods, post-processing methods, and element residual methods. Although these error estimate techniques are different, all of them are based on some types of mathematical error analysis and rely on certain smoothness of the differential solution, which is not the case for discontinuous flows. In fact, most error estimates found in the literature become singular at discontinuities. To remove this singularity and to make the adaptive grid sufficiently smooth, a grid smoothing procedure must be used. As a result, the grid adaptation near discontinuities is driven by the grid smoothing procedure rather than the error estimate itself. The error indicator we develop in this work is based on the interface conservation, which is ruled and driven by the underlying physics and which can effectively address the singularity issue at discontinuities.

Traditionally, the governing equations for the conservation laws are only solved on elements $\Omega_e$ (computational cells). Similarly, the conservation laws should be enforced on element interfaces $\Gamma_e$ as required by the physics.

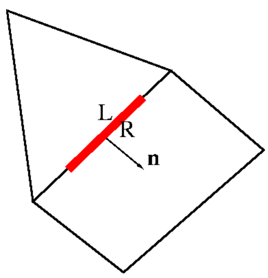


Figure 1. Illustration of a zero-thickness control volume on an interface

Applying the conservation laws on a zero-thickness control volume along an interface $\Gamma_e$ as shown in red in Figure 1 leads to the following interface condition or jump condition for the flux function across the interface

$$(\mathbf{F}(\mathbf{U}_h^R) - \mathbf{G}(\mathbf{U}_h^R)) \cdot \mathbf{n} - (\mathbf{F}(\mathbf{U}_h^L) - \mathbf{G}(\mathbf{U}_h^L)) \cdot \mathbf{n} = \mathbf{0} \quad => \quad [(\mathbf{F}(\mathbf{U}_h) - (\mathbf{F}(\mathbf{U}_h)) \cdot \mathbf{n}] = 0, \tag{3.10}$$

where $\mathbf{U}_h^R$ and $\mathbf{U}_h^L$ are the conservative variable vector on the interface from the left and right elements respectively and the bracket is the so-called the jump operator. This is the so-called interface condition, which is also termed the transmission condition in the hybridized DG [44] or embedded DG [45] formulation. The interface conservation is never considered in all shock-capturing based schemes, because 1) it is automatically satisfied for smooth flows, as long as the flows are fully resolved, which can always be achieved by using high mesh resolution; 2) it can never be satisfied for flows with discontinuities by simply increasingly refined meshes, unless the discontinuities are aligned with mesh interfaces. In other words, the interface conservation can only be achieved, if and only if the mesh interfaces are aligned with discontinuities. Therefore, an error indicator based on the interface conservation can be used to effectively align mesh cell interfaces with discontinuities in the *r*-adaptive grid methods. As in our previous work [43], a continuous variational formulation is used for the interface conservation (3.10) to obtain the following residual vector at grid points,

$$\mathbf{R} = \int_{\Gamma_e} [(\mathbf{F}(u_h) - (\mathbf{F}(u_h)) \cdot \mathbf{n}] \; v_h d\Gamma, \tag{3.11}$$

where $v_h$ is a test function in the continuous solution trace space. In addition, the interface conservation is only considered for the continuity equation, which is simple and valid for all types of flow problems and can detect and identify all types of discontinuities. Clearly, the residual R is zero in fully-resolved flow regions, can never be zero, as long as mesh cell interfaces are not aligned with discontinuities, and therefore serves as a highly reliable, robust, efficient, and simple discontinuity detector. The weigh function on a grid edge *ij* is then defined as

$$w_{ij} = 1 + \mathrm{C}\sqrt{\max(|R_i|, |R_j|)} \tag{3.12}$$

where $R_i$ and $R_j$ are the residual at nodes *i* and *j* of the edge *ij* from Eq. 3.11 and C is a user-specified constant, which can be used to control the clustering of grid points. Note that this error indicator is efficient and simple to compute and is able to align mesh cell interfaces with discontinuities, and effectively offer anisotropic mesh adaption for shock waves and isotropic mesh adaptation for singularities and under-resolved flow regions, as demonstrated in the next section.

# 4 Numerical Examples

The developed *r*-adaptive DG method is used to solve a variety of compressible flow problems. A few examples are presented here to assess the effectiveness of the error indicator based on the interface conservation and to demonstrate the accuracy, robustness, and ability of the *r*-adaptive DG method for both inviscid and viscous compressible flow problems.

**A. Linear advection-diffusion equation**

In this test case [40], the following advection-diffusion equation with Dirichlet boundary conditions

$$\frac{\partial u}{\partial x} = \mu \frac{\partial^2 u}{\partial x^2} \quad \text{in } (0,1)$$
$$u(0) = 0$$
$$u(1) = 1$$

is considered and solved. The exact solution is given by

$$u(x) = \frac{1 - e^{x/\mu}}{1 - e^{1/\mu}}$$

and exhibits a boundary layer like profile at $x$=1, which can be used to qualitatively assess the ability of the mesh adaptation strategy based on the interface conservation to cluster grid points to the boundary layer regions of high gradient solution (boundary layer like solution) and to quantitatively measure the accuracy and convergence of the *r*-adaptive discontinuous Galerkin methods. Numerical solutions on the initial uniform grid of ten cells and on the final *r*-adapted grids obtained by the DG(P1), DG(P2), DG(P3) and DG(P4) methods are compared with the exact solution in Figures 2-5, respectively. One can observe that the *r*-adaptation mesh method based on the interface conservation is extremely effective to cluster grid points towards the regions of high gradient solutions. As expected, the DG methods of all orders and even the fifth order accurate DG(P4) method are unable to produce an oscillation-free and accurate solution on an under-resolved uniform mesh of 10 cells, clearly indicating the motivation and need of having adaptive mesh methods. Au contrary, the DG methods of all orders and even the second order DG(P1) method are able to achieve accurate and oscillation-free solutions on the adapted grids. Figure 6 presents the convergence results for the DG methods of different orders on an initial uniform grid of 10 cells and adapted grids, where one can see that the DG(P1) solution on the adapted mesh is one order of magnitude more accurate than the DG(P4) solution on the uniform grid and the difference is even more profound for the higher DG methods. This example clearly demonstrates that our *r*-adaptive strategy based on the interface conservation is able to relocate grid points to initially under-resolved solution features, therefore allowing the r-adaptive DG methods to achieve highly accurate solutions for these solution features.

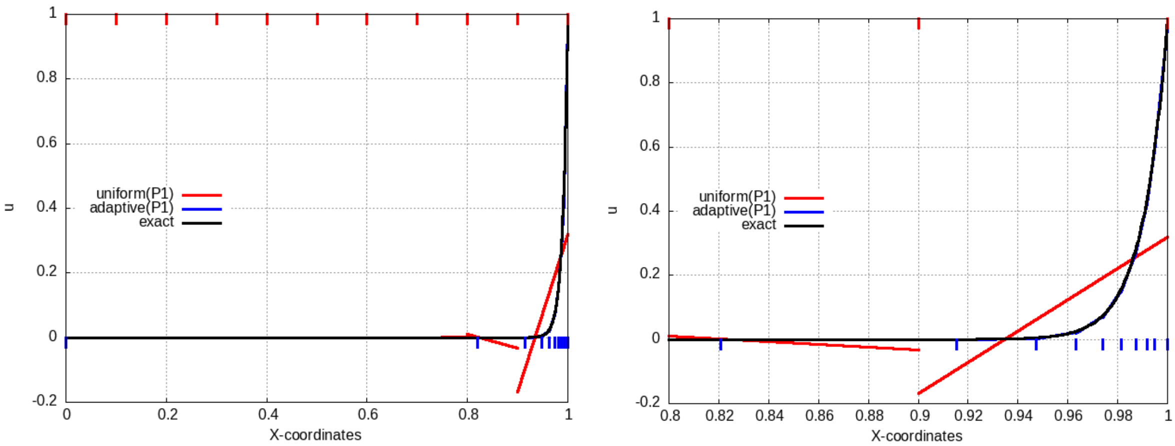


Figure 2. The DG(P1) solutions on an initial uniform grid of 10 cells indicated by the red tick marks and on the *r*-adaptive mesh with blue tick marks are compared with the exact solution.

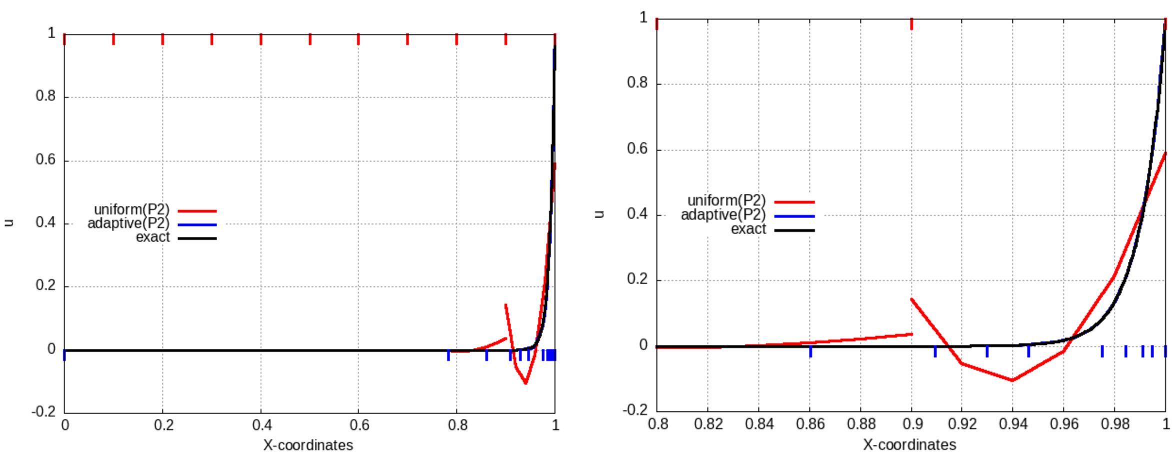


Figure 3. The DG(P2) solutions on an initial uniform grid of 10 cells indicated by the red tick marks and on the *r*-adaptive mesh with blue tick marks are compared with the exact solution.

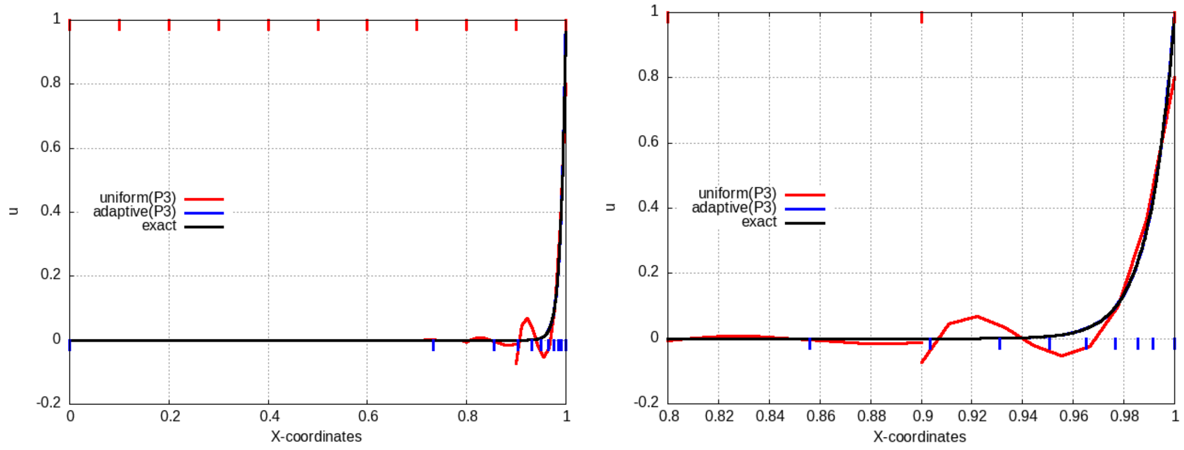


Figure 4. The DG(P3) solutions on an initial uniform grid of 10 cells indicated by the red tick marks and on the $r$-adaptive mesh with blue tick marks are compared with the exact solution.

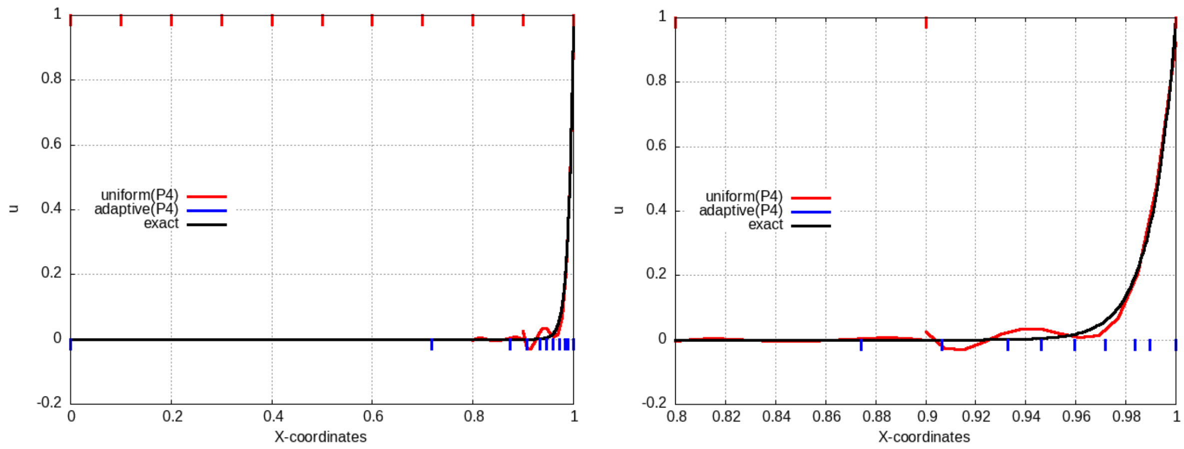


Figure 5. The DG(P4) solutions on an initial uniform grid of 10 cells indicated by the red tick marks and on the $r$-adaptive mesh with blue tick marks are compared with the exact solution.

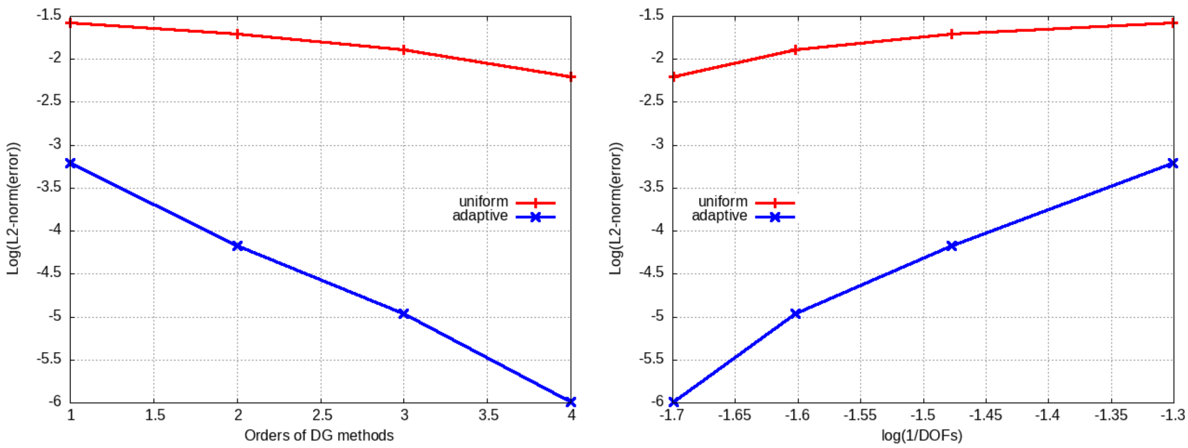


Figure 6. p-convergence of the DG methods on an initial uniform grid of 10 cells and on the $r$-adaptive mesh.

### B. Inviscid transonic flow past a NACA0012 airfoil

Although our main motivation of developing the *r*-adaptive DG methods is for hypersonic viscous flows characterized by a strong bow shock, the *r*-adaptation mesh method based on the interface conservation can be certainly used for inviscid flows and for relatively weak normal shock waves. As an illustrative example, an inviscid transonic flow past a NACA0012 airfoil at a Mach number of 0.8 and an angle of attack of 1.25° is presented using a DG(P1) method in this test case, which exhibits a number of different flow features: both strong and weak normal shocks, singularities (leading and trailing edges), and a trailing edge wake, and therefore poses a great challenge for the interface conservation based r-adaptation method to relocate more grid points to these regions and obtain more accurate solutions to these flow features simultaneously. Figures 7 and 8 show the global view and close-up of the initial and final *r*-adaptive meshes, respectively. The mesh consists of 1,999 elements, 1,048 grid points, and 97 boundary faces with 73 faces on the surface of the airfoil. As one can observe from Figure 7, mesh points in the far field are effectively clustered towards the airfoil and wake region to provide more grid points and more accurate resolution to these regions. Alignments of mesh interfaces with a strong normal shock on the upper surface of the airfoil and a weak normal shock on the lower surface of the airfoil are clearly visible. More grid points are also relocated to the region of both leading and trailing edges. The *r*-adaptive mesh method is able to effectively provide anisotropic mesh adaption for shock waves and isotropic mesh adaptation for singularities like leading and trailing edges, as illustrated in Figure 8. The computed Mach number and pressure contours on the initial grid and the final adapted grid after 5 *r*-adaptations are compared in Figures 9 and 10, respectively. The computed pressure coefficient and Mach number on the surface of the airfoil between the initial and final adapted solutions are shown in Figure 11. As can be seen from these figures, the benefits of the *r*-adaptive mesh method are quite obvious: a more accurate solution near the stagnation point as witnessed by the smallest Mach number of 0.085 on the initial grid and the smallest Mach number of 0.0285 on the adapted mesh and the trailing edge, better resolution for the wake of the trailing edge, and much sharper and almost shock fitting look-like result for shocks.

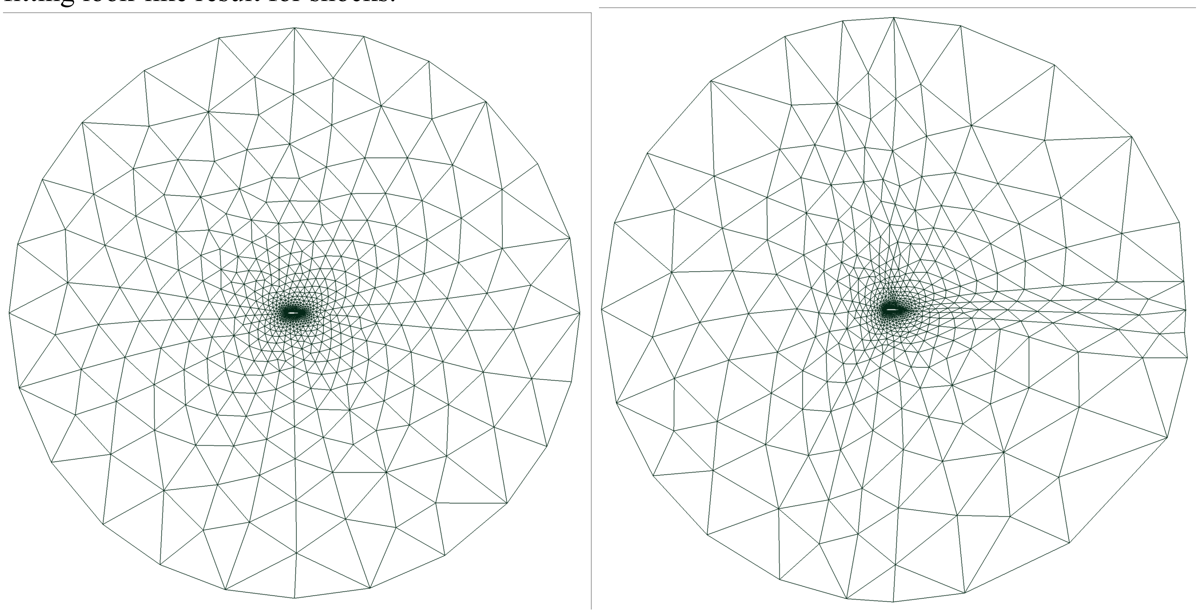

Figure 7. Global view of initial mesh (left) and adapted mesh after 5 *r*-adaptations (right)

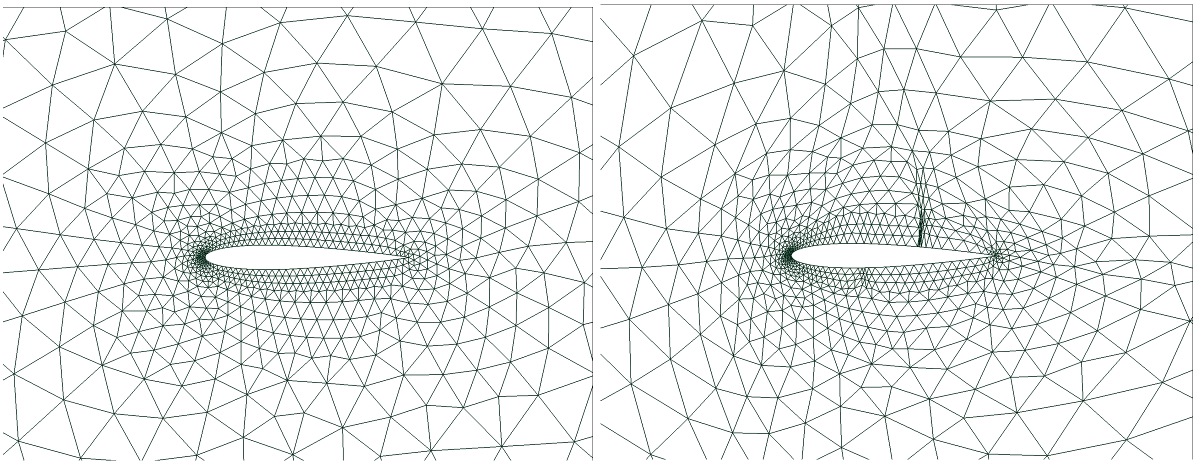

Figure 8. Close-up of the initial mesh (left) and adapted mesh after 5 $r$-adaptations (right)

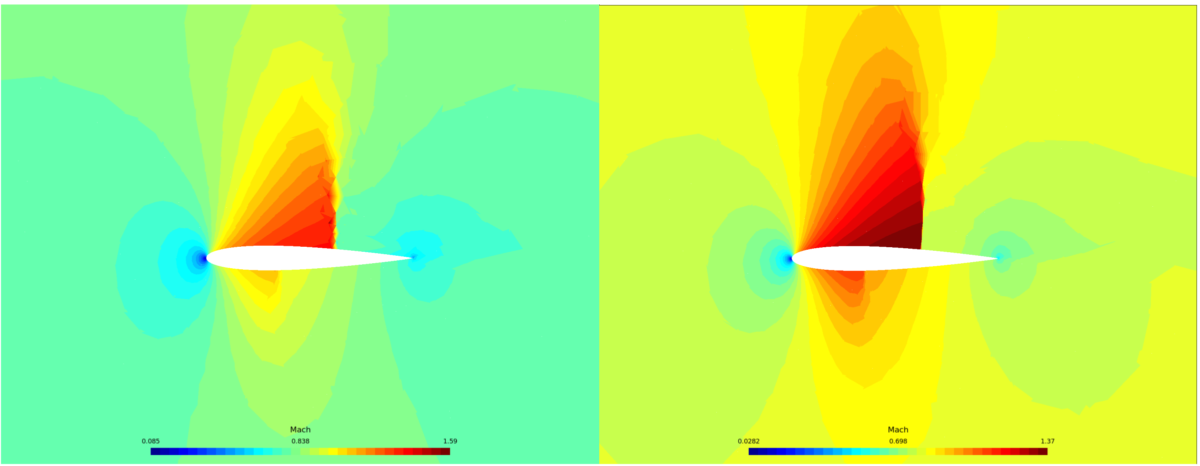


Figure 9. Computed Mach number contours on the initial mesh (left) and adapted mesh after 5 $r$-adaptations (right)

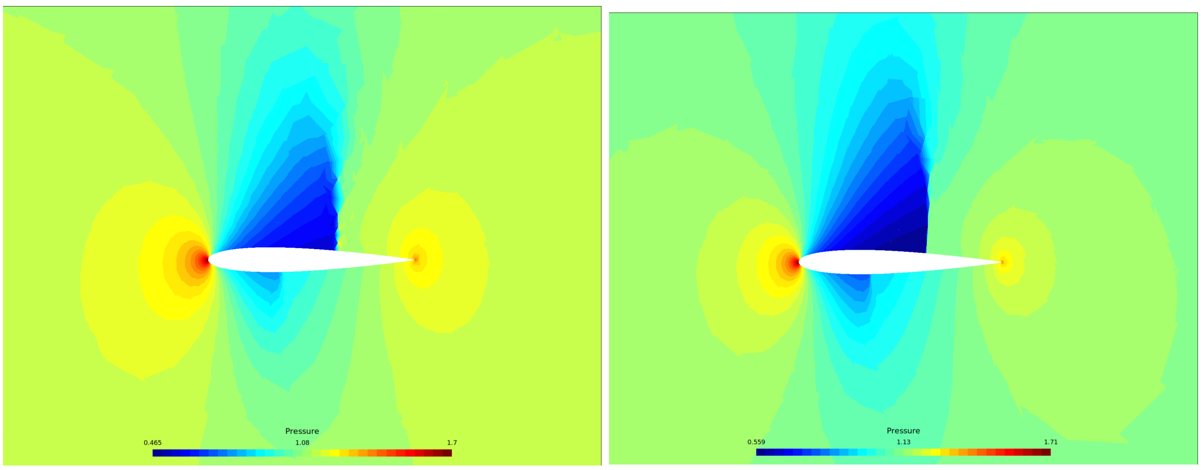


Figure 10. Computed pressure contours on the initial mesh (left) and adapted mesh after 5 $r$-adaptations (right)

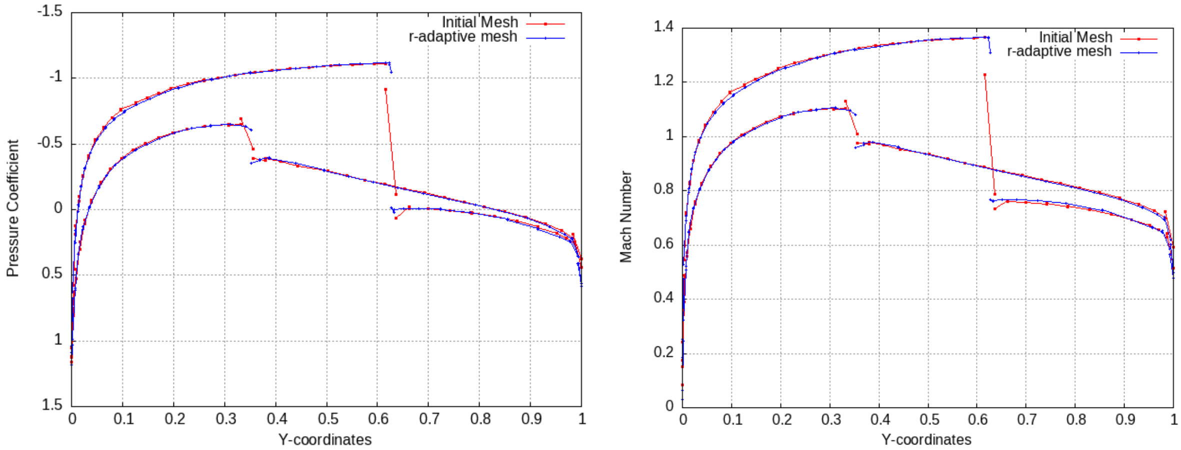


Figure 11. Comparison of the computed pressure coefficient and Mach number on the surface of the airfoil on the initial mesh and the adapted mesh after 5 *r*-adaptations.

### C. Hypersonic viscous flow past a half-circular cylinder on a quadrilateral grid

In this test case [40], a viscous hypersonic flow past a half-circular cylinder at a Mach number of 5 and a Reynolds number of 1,000 with an iso-thermal wall boundary condition is computed using a DG(P1) method on a grid of 40x20 quadrilateral elements with 40 and 20 cells in the circumferential and radial directions, respectively. The temperature at the isothermal wall is given as $T_{wall} = 2.5T_\infty$. where $T_\infty$ is the freestream temperature. Figure 12 presents the grid used in this test case and the corresponding pressure and temperature contours obtained by the DG(P1) solution. Figures 13-15 show the adapted meshes and the corresponding pressure and temperature fields computed by the DG(P1) method after one, three, and five *r*-adaptations, respectively. One can observe that our *r*-adaptive grid method is highly effective to relocate grid points to the bow shock region, provide anisotropic mesh adaptation to and align grid cell interfaces with the bow shock, and consequently significantly improve resolution of the bow shock. Figure 16 compares the computed pressure coefficients and Stanton number, which is a dimensionless coefficient for the normal heat flux, obtained by the DG(P1) method on the initial mesh and the final *r*-adapted mesh. Even though the size of the mesh used in the computation is too small to provide a fully resolved solution, as witnessed by a large jump of pressure across element interfaces, the computed pressure coefficient and even Stanton number which is infamously known to be very difficult to compute accurately, are in good agreement with the reference solutions [40]. Even more surprisingly, the difference between these two solutions is quite small, considering the extremely poor resolution of the bow shock on the initial grid and excellent resolution of the bow shock on the final *r*-adapted grid. One possible explanation might be that the numerical error from the poor resolution of the shock wave is cancelled by the symmetry of the quadrilateral mesh used in the computation.

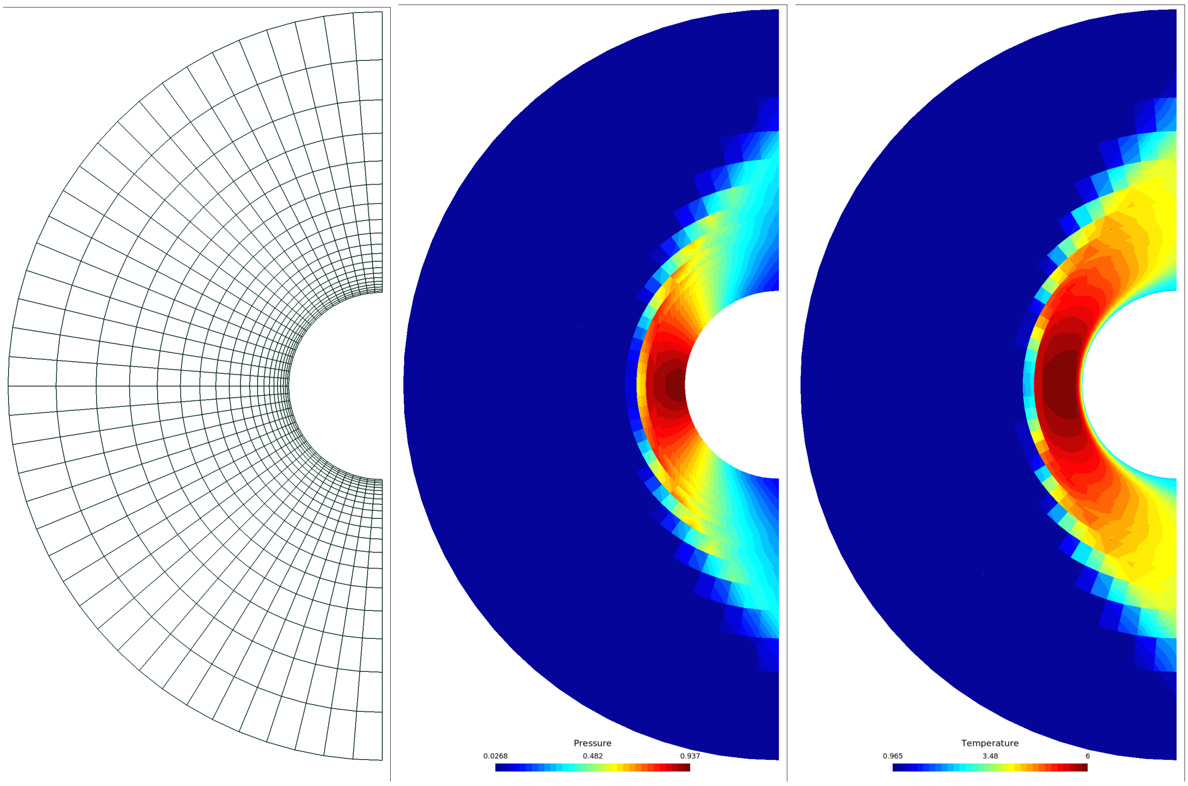


Figure 12. Initial mesh and the corresponding pressure and temperature contours

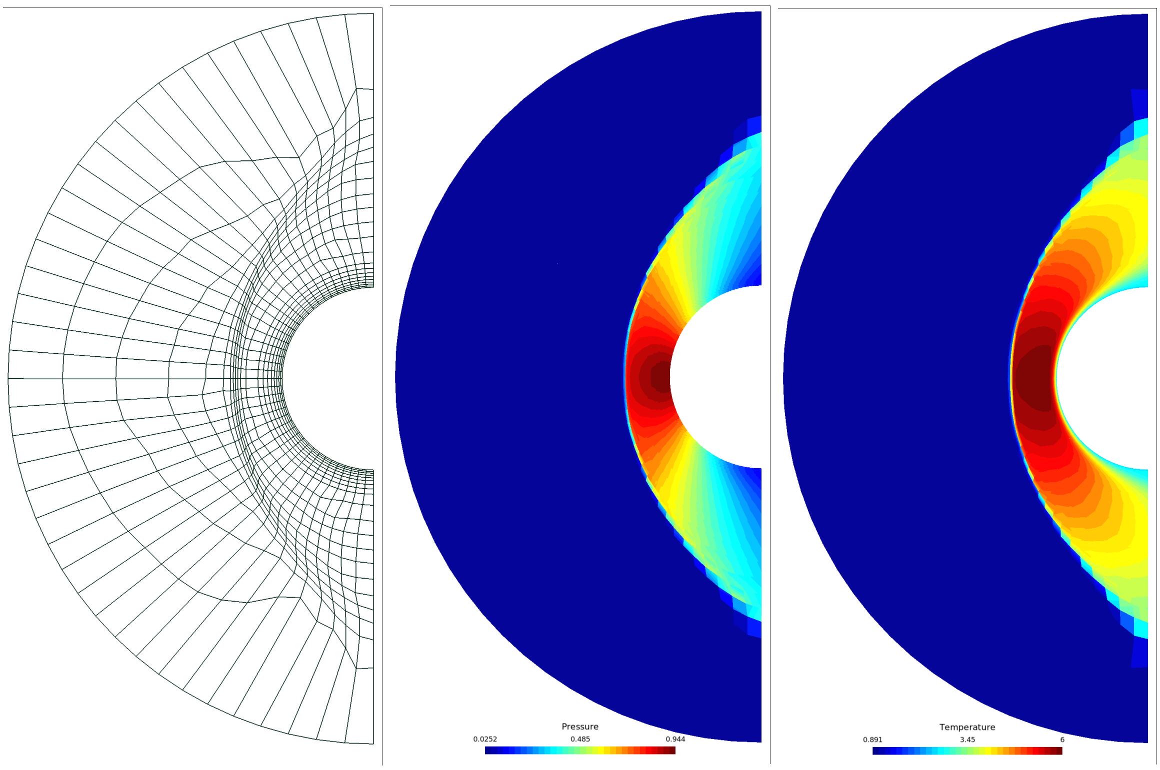


Figure 13. Adapted mesh and the corresponding pressure and temperature contours after one $r$-adaptation

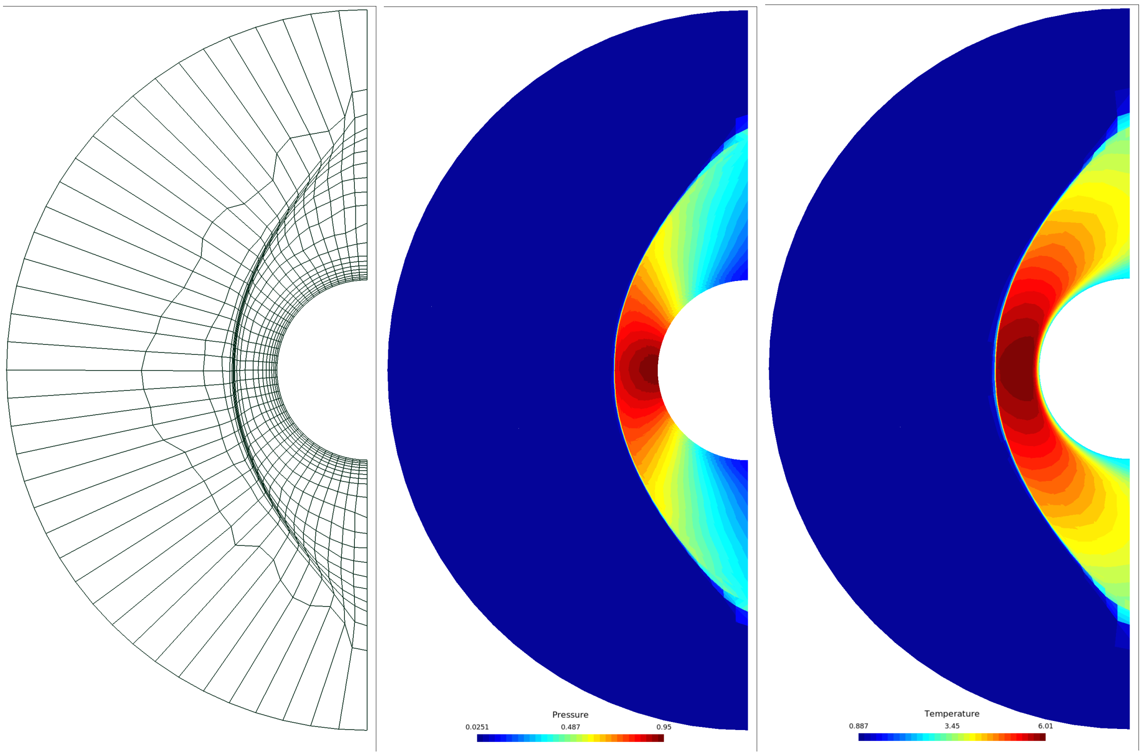


Figure 14. Adapted mesh and the corresponding pressure and temperature contours after three *r*-adaptations

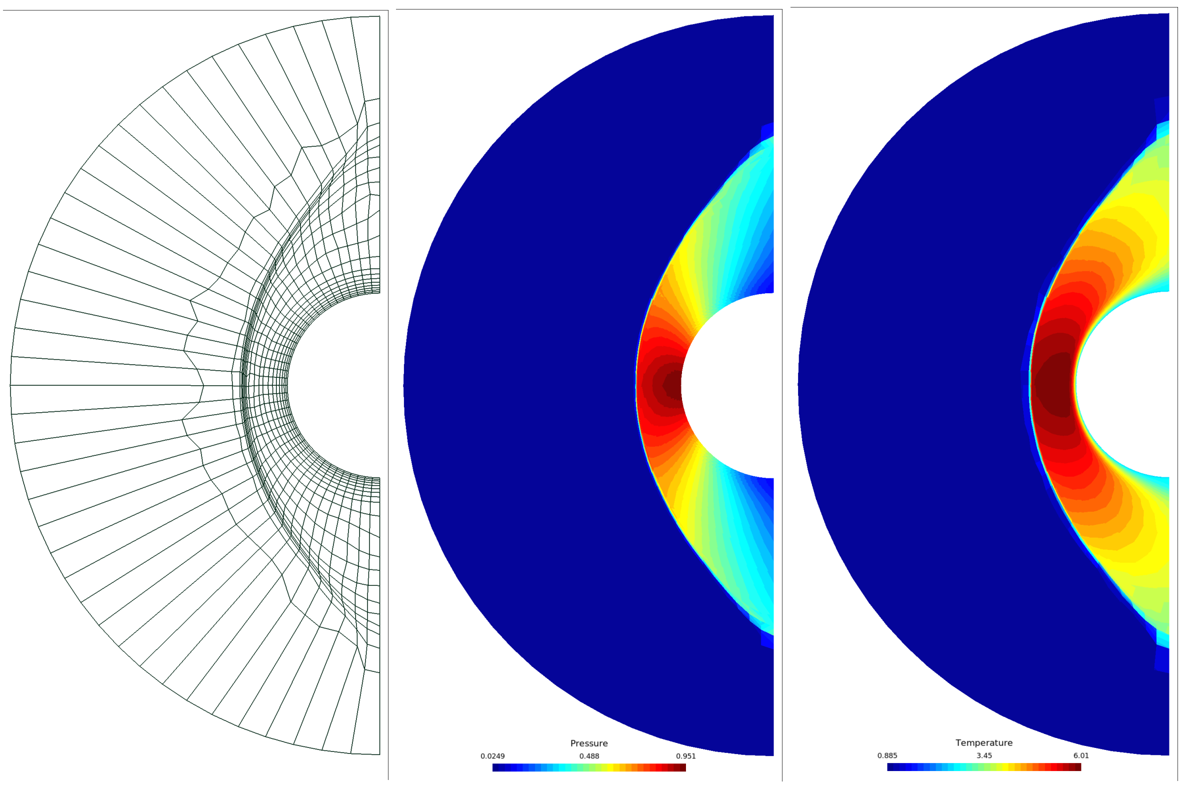


Figure 15. Adapted mesh and the corresponding pressure and temperature contours after five *r*-adaptations

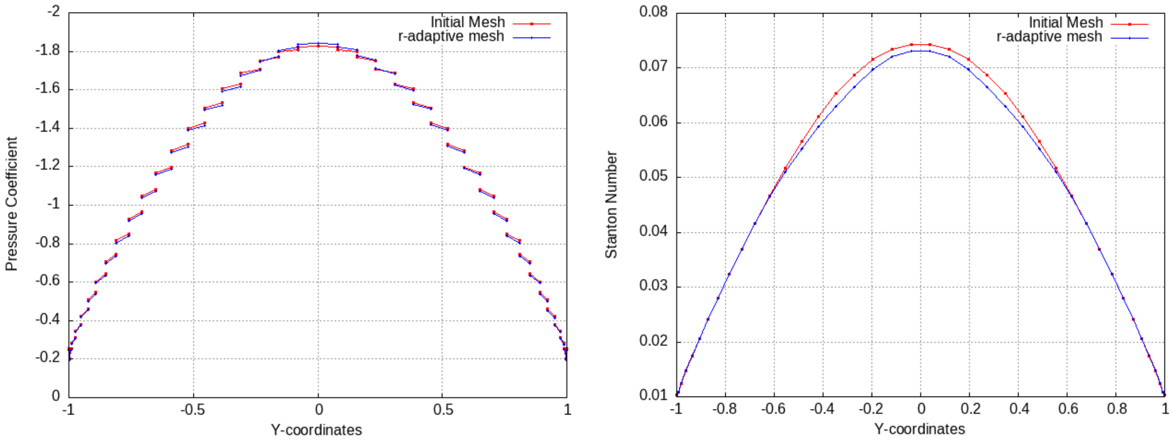


Figure 16. Comparison of the computed pressure coefficients (left) and Stanton number (right) between the initial mesh and the final *r*-adapted mesh.

**D. Hypersonic viscous flow past a half-circular cylinder on a triangular grid**

The same numerical experiment as Test Case C is conducted using a DG(P1) method on a grid of 1,178 triangular elements, 645 grid points, and 110 boundary faces with 39 faces on the surface of the circular cylinder. Figure 17 presents the grid used in this test case and the corresponding pressure and temperature contours obtained by the DG(P1) solution. As expected, the DG(P1) solution is not symmetric due to the very nature of the triangular mesh used in this case and is highly inaccurate due to a lack of mesh resolution in the bow shock region. Figures 18-20 show the adapted meshes and the corresponding pressure and temperature fields computed by the DG(P1) method after one, three, and five *r*-adaptations, respectively. One can observe again that our *r*-adaptive grid method is highly effective to relocate grid points to the bow shock region, provide anisotropic mesh adaptation to and align grid cell interfaces with the bow shock, and consequently significantly improve resolution of the bow shock. Clearly, the DG(P1) solutions become more and more accurate and symmetric with more *r*-adaptations, which is attributed to the highly efficient anisotropic adaptation of the bow shock and relocation of grid points to under-resolved regions. Figure 21 compares the computed pressure coefficients and Stanton number obtained by the DG(P1) method on the initial mesh and the final *r*-adapted mesh. As can be seen, the DG(P1) solution on the initial mesh is unable to obtain accurate pressure and heat fluxes, even though it provides a well resolved solution judging a relatively continuous solution of pressure across element interfaces. However, both pressure coefficient and Stanton number are accurately computed by the *r*-adaptive DG method on the final adapted mesh, producing similar or even more accurate solutions than the one on structured quadrilateral meshes.

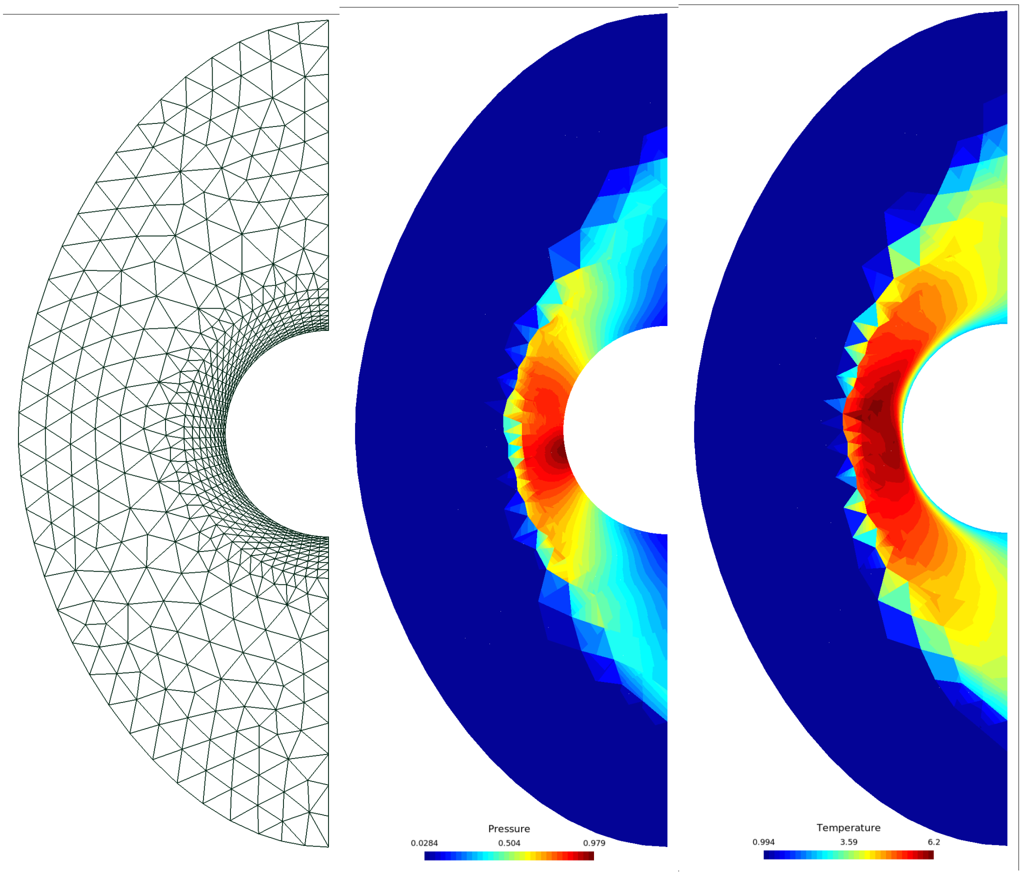

Figure 17. Initial mesh and the corresponding pressure and temperature contours

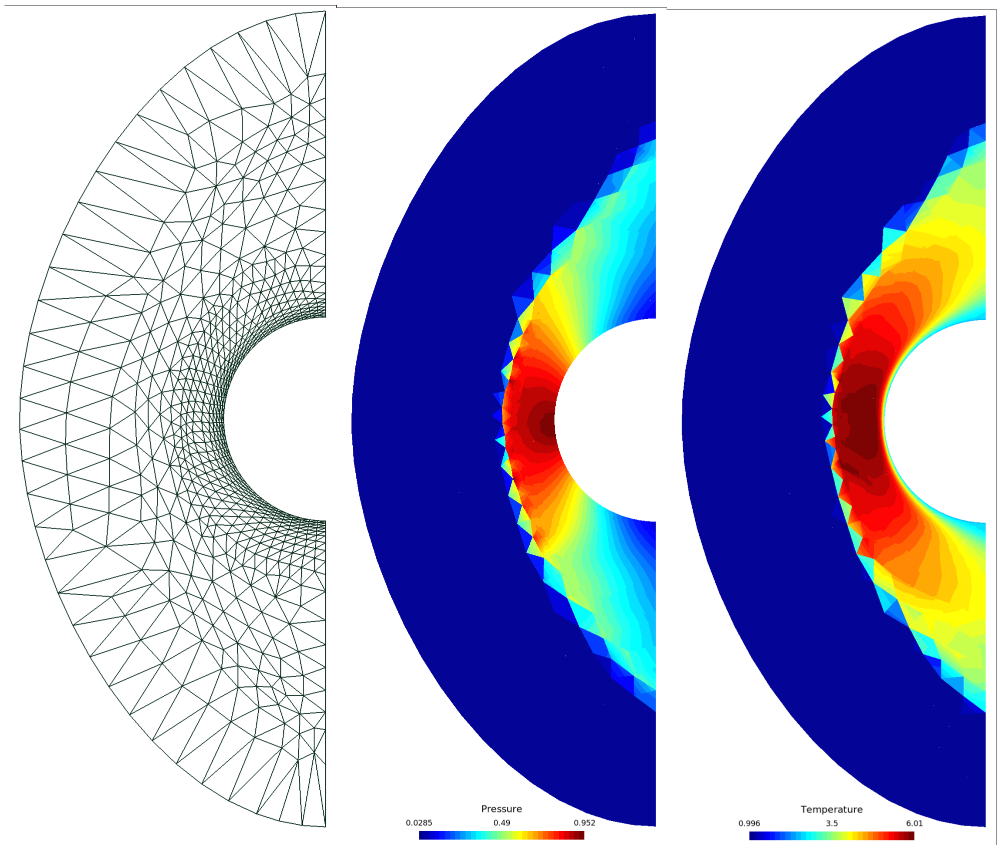

Figure 18. Adapted grid and the corresponding pressure and temperature contours after one *r*-adaptations

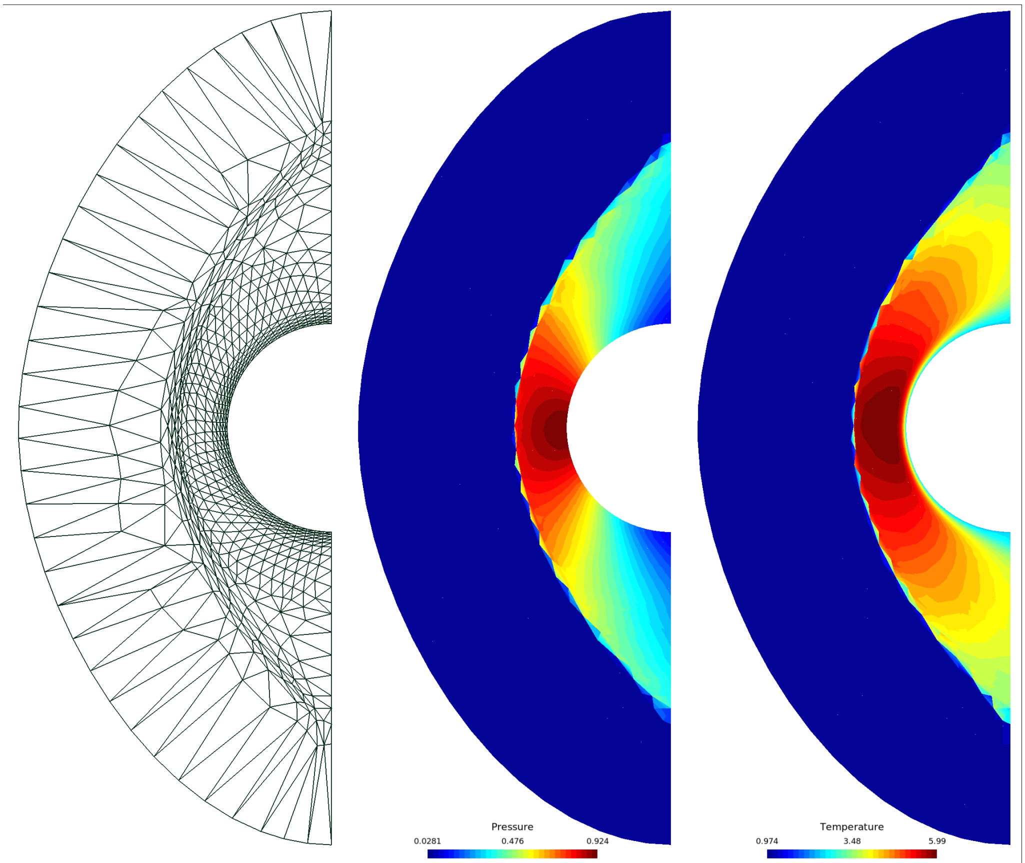


Figure 19. Adapted grid and the corresponding pressure and temperature contours after three *r*-adaptations

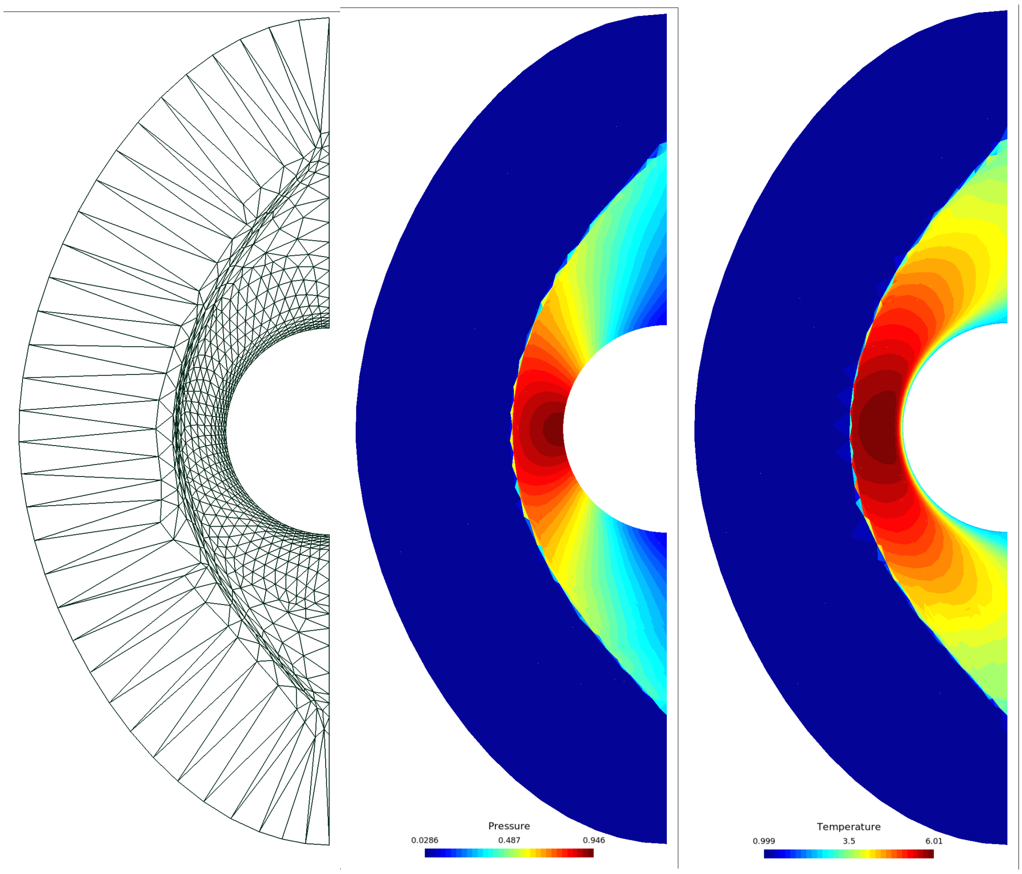


Figure 20. Adapted grid and the corresponding pressure and temperature contours after five *r*-adaptations

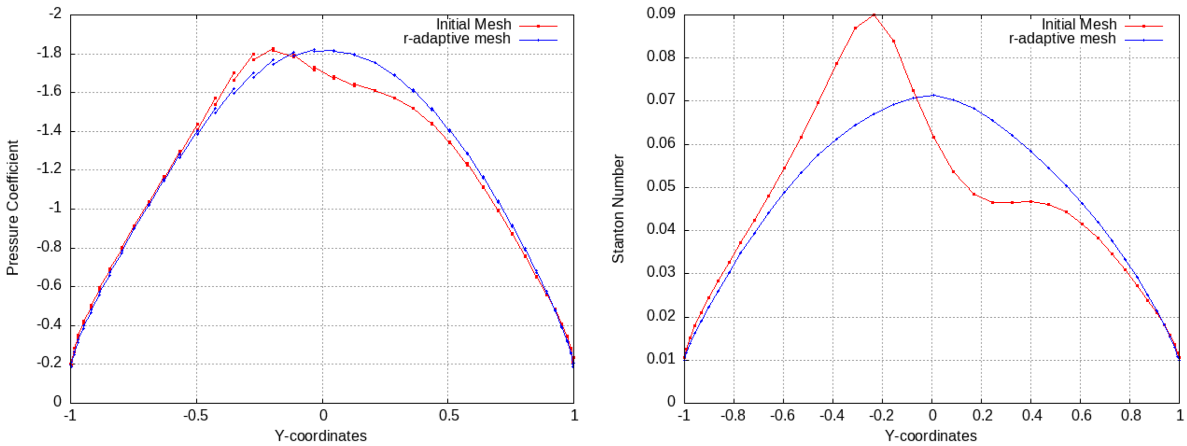

Figure 21. Comparison of the computed pressure coefficients (left) and Stanton number (right) between the initial mesh and the final *r*-adapted mesh.

## 5 Conclusion and Future Work

The interface conservation has been examined and explored thoroughly in this work. A novel error indicator based on the interface conservation has been introduced as a weight function in a *r*-adaptive grid method and implemented in a discontinuous Galerkin (DG) finite element method. A number of numerical experiments have been conducted to assess the effectiveness of the interface conservation-based error indicator and performance of the resulting *r*-adaptive DG method. Preliminary results for a number of benchmark test cases are promising and encouraging, indicating that the physics-based error indicator can reliably and efficiently detect and identify all types of discontinuities for inviscid flows and all under-resolved flow features and regions of high gradient solutions for viscous flows. The *r*-adaptive DG method is able to align mesh cell interfaces with discontinuities, and effectively offer anisotropic mesh adaption for shock waves and isotropic mesh adaptation for singularities and under-resolved flow regions. Ongoing work is focused on extending the *r*-adaptive DG method for chemical- and thermo-nonequilibrium hypersonic viscous flows on 3D hybrid grids.

## References

[1] W.H. Reed and T.R. Hill, Triangular Mesh Methods for the Neutron Transport Equation, Los Alamos Scientific Laboratory Report, LA-UR-73-479, 1973.
[2] B. Cockburn, S. Hou, and C. W. Shu, TVD Runge-Kutta Local Projection Discontinuous Galerkin Finite Element Method for conservation laws IV: the Multidimensional Case, Mathematics of Computation, Vol. 55, pp. 545-581, 1990.
[3] B. Cockburn, and C. W. Shu, The Runge-Kutta Discontinuous Galerkin Method for conservation laws V: Multidimensional System, Journal of Computational Physics, Vol. 141, pp. 199-224, 1998.
[4] B. Cockburn, G. Karniadakis, and C. W. Shu, The Development of Discontinuous Galerkin Method, in Discontinuous Galerkin Methods, Theory, Computation, and Applications, edited by B. Cockburn, G.E. Karniadakis, and C. W. Shu, Lecture Notes in Computational Science and Engineering, Springer-Verlag, New York, 2000, Vol. 11 pp. 5-50, 2000.
[5] F. Bassi and S. Rebay, High-Order Accurate Discontinuous Finite Element Solution of the 2D Euler Equations, Journal of Computational Physics, Vol. 138, pp. 251-285, 1997.
[6] H. L. Atkins and C. W. Shu, Quadrature Free Implementation of Discontinuous Galerkin Method for Hyperbolic Equations, AIAA Journal, Vol. 36, No. 5, 1998.
[7] F. Bassi and S. Rebay, GMRES discontinuous Galerkin solution of the Compressible Navier-Stokes Equations, Discontinuous Galerkin Methods, Theory, Computation, and Applications, edited by B. Cockburn, G.E. Karniadakis, and C. W. Shu, Lecture Notes in Computational Science and Engineering, Springer-Verlag, New York, 2000, Vol. 11 pp. 197-208, 2000.

[8] T. C. Warburton, and G. E. Karniadakis, A Discontinuous Galerkin Method for the Viscous MHD Equations, Journal of Computational Physics, Vol. 152, pp. 608-641, 1999.
[9] J. S. Hesthaven and T. Warburton, Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications, Texts in Applied Mathematics, Vol. 56, 2008.
[10]P. Rasetarinera and M. Y. Hussaini, An Efficient Implicit Discontinuous Spectral Galerkin Method, Journal of Computational Physics, Vol. 172, pp. 718-738, 2001.
[11]B. T. Helenbrook, D. Mavriplis, and H. L. Atkins, Analysis of p-Multigrid for Continuous and Discontinuous Finite Element Discretizations, AIAA Paper 2003-3989, 2003.
[12]K. J. Fidkowski, T. A. Oliver, J. Lu, and D. L. Darmofal, *p*-Multigrid solution of high-order discontinuous Galerkin discretizations of the compressible Navier–Stokes equations, Journal of Computational Physics, Vol. 207, No. 1, pp. 92-113, 2005.
[13]H. Luo, J. D. Baum, and R. Löhner, A Discontinuous Galerkin Method Using Taylor Basis for Compressible Flows on Arbitrary Grids, Journal of Computational Physics, Vol. 227, No 20, pp. 8875-8893, October 2008.
[14]H. Luo, J.D. Baum, and R. Löhner, On the Computation of Steady-State Compressible Flows Using a Discontinuous Galerkin Method, International Journal for Numerical Methods in Engineering, Vol. 73, No. 5, pp. 597-623, 2008.
[15]H. Luo, J. D. Baum, and R. Löhner, A Hermite WENO-based Limiter for Discontinuous Galerkin Method on Unstructured Grids, Journal of Computational Physics, Vol. 225, No. 1, pp. 686-713, 2007.
[16]H. Luo, J.D. Baum, and R. Löhner, A *p*-Multigrid Discontinuous Galerkin Method for the Euler Equations on Unstructured Grids, Journal of Computational Physics, Vol. 211, No. 2, pp. 767-783, 2006.
[17]H. Luo, J.D. Baum, and R. Löhner, Fast, *p*-Multigrid Discontinuous Galerkin Method for Compressible Flows at All Speeds", AIAA Journal , Vol. 46, No. 3, pp.635-652, 2008.
[18] M. Dumbser, D.S. Balsara, E.F. Toro, C.D. Munz. A unified framework for the construction of one-step finite volume and discontinuous Galerkin schemes on unstructured meshes. Journal of Computational Physics, 227:8209-8253, 2008.
[19] M. Dumbser, O. Zanotti, Very high order PNPM schemes on unstructured meshes for the resistive relativistic MHD equations. Journal of Computational Physics, 228:6991-7006, 2009.
[20] M. Dumbser. Arbitrary High Order PNPM Schemes on Unstructured Meshes for the Compressible Navier-Stokes Equations. Computers & Fluids, 39: 60-76. 2010.
[21]F. Bassi and S. Rebay, A High-Order Accurate Discontinuous Finite Element Method for the Numerical Solution of the Compressible Navier-Stokes Equations, Journal of Computational Physics, Vol. 131, pp. 267-279, 1997.
[22]F. Bassi and S. Rebay, Discontinuous Galerkin Solution of the Reynolds-Averaged Navier-Stokes and k-ω Turbulence Model Equations, Journal of Computational Physics, Vol. 34, pp. 507-540, 2005.
[23]H. Luo, L. Luo, and K. Xu, A Discontinuous Galerkin Method Based on a BGK Scheme for the Navier-Stokes Equations on Arbitrary Grids, Advances in Applied Mathematics and Mechanics, Vol. 1, No. 3, pp. 301-318, 2009.
[24]H. Luo, L. Luo, and R. Nourgaliev, A Reconstructed Discontinuous Galerkin Method for the Euler Equations on Arbitrary Grids, Communication in Computational Physics, Vol. 12, No. 5, pp. 1495-1519, 2012.
[25]H. Luo, L. Luo, R. Nourgaliev, V.A. Mousseau, and N. Dinh, A Reconstructed Discontinuous Galerkin Method for the Compressible Navier-Stokes Equations on Arbitrary Grids, Journal of Computational Physics, Vol. 229, pp. 6961-6978, 2010.
[26]H. Luo, L. Luo, A. Ali, R. Nourgaliev, and C. Cai, A Parallel, Reconstructed Discontinuous Galerkin Method for the Compressible Flows on Arbitrary Grids, Communication in Computational Physics, Vol. 9, No. 2, pp. 363-389, 2011.
[27]H. Luo, H., Y. Xia, S. Li, R. Nourgaliev, and C. Cai, A Hermite WENO Reconstruction-Based Discontinuous Galerkin Method for the Euler Equations on Tetrahedral Grids, Journal of Computational Physics, Vol. 231, pp. 5489-5503, 2012.
[28]H. Luo, H., Y. Xia, S. Spiegel, R. Nourgaliev, and Z. Jiang, A Reconstructed Discontinuous Galerkin Method Based on a Hierarchical WENO Reconstruction for Compressible Flows on Tetrahedral Grids, Journal of Computational Physics, DOI: 10.1016/j.jcp.2012.11.026, 2012.
[29]H. Bijl, M.H. Carpenter, V.N. Vasta, and C. A. Kennedy, Implicit Time Integration Schemes for

the Unsteady Compressible Navier-Stokes Equations: Laminar Flow, Journal of Computational Physics, Vol. 179, pp. 313-329, 2002.
[30]L. Wang, and D.J. Mavriplis, Implicit Solution of the Unsteady Euler Equations for High-Order Accurate Discontinuous Galerkin Discretizations. Journal of Computational Physics, Vol. 225, No. 2, pp. 1994–2015, 2007.
[31]Van der Vegt, J., and Van det Ven, H., Space-time Discontinuous Galerkin Finite Element Method with Dynamic Grid Motion for Inviscid Compressible Flows: I. General Formulation, Journal of Computational Physics, Vol. 182, No. 2, pp. 546–585, 2002.
[32]Klaij, C., Van der Vegt, J., and Van det Ven, H., Space-time Discontinuous Galerkin Method for the Compressible Navier-Stokes Equations, Journal of Computational Physics, Vol. 217, No. 2, pp. 589–611, 2006.
[33]Lehrenfeld, C., The Nitsche XFEM-DG Space-time Method and its Implementation in Three Space Dimensions, SIAM Journal on Scientific Computing, Vol. 37, No. 1, pp. A245–270, 2015.
[34]M.D. Salas, Abrief history of shock-fitting. In:Springer 2011(pp.37–53).
[35]A. Bonfiglioli, R. Paciorri, and L. Campoli, Unsteady shock fitting for unstructured grids, International Journal for Numerical Methods in Fluids. 2016;81(4):245–261.
[36]D. Zou, C. Xu, H. Dong, and J. Liu, A shock-fitting technique for cell-centered finite volume methods on unstructured dynamic meshes, Journal of Computational Physics. 2017;345:866–882.
[37]P. Roe and H. Nishikawa, Adaptive grid generation by minimizing residuals, International Journal for Numerical Methods in Fluids. Vol. 40, pp. 121–136, 2002.
[38]A. Corrigan, A. D. Kercher, and D.A. Kessler, A Moving Discontinuous Galerkin Finite Element Method for Flows with Interfaces, International Journal for Numerical Methods in Fluids, DOI: 10.1002/fld.4697, 2019.
[39]A. Corrigan, A. Kercher, D. Kessler, The moving discontinuous Galerkin method with interface condition enforcement for unsteady three-dimensional flows, in: AIAA Scitech 2019 Forum, AIAA-2019-0642.
[40]A. Kercher, A. Corrigan, and D. Kessler, The moving discontinuous Galerkin method with Interface Condition Enforcement for compressible viscous flows, DOI: 10.1002/fld.4939,2020.
[41]A. Corrigan, A. Kercher, D. Kessler, and D. Wood-Thomas, Convergence of the moving discontinuous Galerkin method with interface condition enforcement in the presence of an attached curved shock, in: AIAA Aviation 2019 Forum, AIAA-2019-3207.
[42]H. Luo, Y. Jiang, and R. Nourgaliev, A Moving Discontinuous Galerkin Finite Element Method for Conservation Laws, AIAA Scitech 2020 forum, 6-10 January 2020, Orlando, FL, AIAA-2020-1316, 2020.
[43]H. Luo, Y. G. Absillis, and R. Nourgaliev, A Moving Discontinuous Galerkin Finite Element Method with Interface Condition Enforcement for Compressible Flows, Journal of Computational Physics, 445 (2021) 110618.
[44]B. Cockburn, J. Gopalakrishnan, and R. Lazarov, Unified Hybridization of Discontinuous Galerkin, Mixed, and Continuous Galerkin Methods for Second Order Elliptic Problems, SIAM Journal on Numerical Analysis, Vol. 47, pp.1319-1365, 2009.
[45]N.C. Nguyen, J. Peraire, and B. Cockburn, A class of embedded discontinuous Galerkin methods for computational fluid dynamics, Journal of Computational Physics, Vol. 302, pp. 674-692, 2015.
[46]D. Ait-Ali-Yahia, A Finite Element Segregated Method for Hypersonic Thermo-Chemical Equilibrium and Nonequilibrium Flows Using Adapted Grids, PhD Dissertation, Concordia University, 1996.
[47]F. B. Ameur and A. Lani, r-adaptive algorithms for high-speed flows and plasma simulations, Computer Physics Communications, 261, (2021) 107700.
[48]Tao Tang, Moving mesh methods for computational fluid dynamics, http://dx.doi.org/10.1090/conm/383/07162, 2005.
[49]N. Yamaleev and M. Carpenter, On Accuracy of Adaptive Grid Methods for Captured Shocks, Journal of Computational Physics, Vol. 181, No. 1, pp 280-316, 2002.