

# An h-adaptive Discontinuous Galerkin Method for Laminar Compressible Navier-Stokes Equations on Curved Mesh

Sun Qiang, Lyu Hongqiang\*, Wu Yizhao

College of Aerospace Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, P. R. China

(Received 14 April 2015; revised 18 August 2015; accepted 5 September 2015)

**Abstract:** An h-adaptive method is developed for high-order discontinuous Galerkin methods (DGM) to solve the laminar compressible Navier-Stokes (N-S) equations on unstructured mesh. The vorticity is regarded as the indicator of adaptivity. The elements where the vorticity is larger than a pre-defined upper limit are refined, and those where the vorticity is smaller than a pre-defined lower limit are coarsened if they have been refined. A high-order geometric approximation of curved boundaries is adopted to ensure the accuracy. Numerical results indicate that highly accurate numerical results can be obtained with the adaptive method at relatively low expense.

**Key words:** h-adaptivity; high-order discontinuous Galerkin methods(DGM); N-S equations; high-order boundary approximation

**CLC number:** V211.3

**Document code:** A

**Article ID:** 1005-1120(2016)05-0566-10

## 0 Introduction

Discontinuous Galerkin (DG) methods<sup>[1-13]</sup> have received increasing attention in computational fluid dynamics in recent years due to various attractive features. Bassi and Rebay<sup>[3-5]</sup> developed a high-order discontinuous finite element method to solve the Euler and Navier-Stokes equations. Cockburn and Shu<sup>[6, 7]</sup> devised a high-order accurate total variation bounded (TVB) Runge-Kutta discontinuous Galerkin (RKDG) method to simulate the nonlinear systems of conservation laws. More recently, high-order DG methods have been applied to solve various engineering problems<sup>[8-18]</sup> on unstructured grid. In fact, DG methods are similar to finite element methods which can achieve higher-order accuracy via using high-order polynomial approximation inside elements. Moreover, upwind scheme can be easily implemented through using appropriate numerical fluxes over element interfaces. In addition, DG methods lead to compact space discretization formulae for both the Euler and the Navier-Stokes (N-S) equations.

The compactness of the methods has advantages for parallel implementation.

Despite these advantages, DG methods still need to be improved in many respects, such as the shock-capturing and the huge computational expense caused by the high-order polynomial approximation<sup>[8-15]</sup>. Usually, high discontinuity only exists locally, e. g. the boundary layer in the flow field. It will cost high expense to capture them by enhancing the order of the polynomials or generating more dense mesh globally. Adaptive DG methods are helpful to solve such problems. Thanks to the simple communication at element interfaces, elements with "hanging nodes" can be treated as elements without hanging nodes, which simplifies local mesh h-refinement. In addition, the communication at element interfaces allows different orders between neighboring elements. Several adaptive DG methods<sup>[19-23]</sup> have been developed to improve the accuracy and reduce the computational expense.

It has been proved that high-order DG methods are inaccurate at curved solid walls if a piece-

\* Corresponding author, E-mail address: hongqiang.lu@nuaa.edu.cn.

wise linear approximation of the geometry of the boundary is employed<sup>[3]</sup>, and a higher-order boundary representation is necessary to ensure the accuracy of the solution. In the paper, an h-adaptive strategy is developed for DG methods to simulate compressible laminar N-S equations on highly accurate boundary. Because of the high intensity of the vorticity in boundary layer and shedding vortex regions, vorticity is used as the sensor of the h-adaptivity. For the steady case, a Newton method<sup>[24]</sup> is employed to solve the nonlinear discrete systems and the Block-Gauss Seidel<sup>[10, 11]</sup> method is used to solve the resulting sparse linear system at each nonlinear iteration. The time integration of the unsteady case presented below can be accomplished by means of an explicit method. The four-stage Runge-Kutta scheme is used in the paper. Since DG methods are relatively sensitive to the initial guess, a hierarchical solution procedure is suggested<sup>[10, 12]</sup>.

## 1 Governing Equations

The two-dimensional laminar N-S equations can be written as follows

$$\partial_t \mathbf{u} + \nabla \cdot \mathbf{F}_e(\mathbf{u}) - \nabla \cdot \mathbf{F}_v(\mathbf{u}, \nabla \mathbf{u}) = 0 \quad (1)$$

where the conservative variables  $\mathbf{u}$  and the cartesian components  $\mathbf{f}_e(\mathbf{u})$  and  $\mathbf{g}_e(\mathbf{u})$  of the inviscid (Euler) flux function  $\mathbf{F}_e(\mathbf{u})$  are given by

$$\mathbf{u} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho e \end{bmatrix}; \mathbf{f}_e(\mathbf{u}) = \begin{bmatrix} \rho u \\ \rho u^2 + P \\ \rho uv \\ \rho hu \end{bmatrix}; \mathbf{g}_e(\mathbf{u}) = \begin{bmatrix} \rho v \\ \rho uv \\ \rho v^2 + P \\ \rho hv \end{bmatrix} \quad (2)$$

where  $\rho$ ,  $P$  and  $e$  denote the density, pressure and the total internal energy per unit mass respectively.  $u$  and  $v$  are the velocity components. The total enthalpy per unit mass  $h$  is defined as  $h = e + \frac{p}{\rho}$ .

By assuming that the fluid obeys to the perfect gas state equation,  $p$  can be calculated as  $P = (\gamma - 1)\rho \left( e - \frac{(u^2 + v^2)}{2} \right)$ , where  $\gamma$  is the ratio of the specific heats.

The cartesian components  $\mathbf{f}_v(\mathbf{u}, \nabla \mathbf{u})$  and  $\mathbf{g}_v(\mathbf{u}, \nabla \mathbf{u})$  of the viscous flux function  $\mathbf{F}_v(\mathbf{u}, \nabla \mathbf{u})$

are given by

$$\mathbf{f}_v(\mathbf{u}, \nabla \mathbf{u}) = \begin{bmatrix} 0 \\ 2\mu_x + \lambda(u_x + v_y) \\ v_x + u_y \\ \mu[2u_x + \lambda(u_x + v_y)] + v(v_x + u_y) + \gamma/Pre_x \end{bmatrix} \quad (3)$$

$$\mathbf{g}_v(\mathbf{u}, \nabla \mathbf{u}) = \begin{bmatrix} 0 \\ v_x + u_y \\ 2\mu_y + \lambda(u_x + v_y) \\ \mu(v_x + u_y) + v[2v_y + \lambda(u_x + v_y)] + \gamma/Pre_y \end{bmatrix} \quad (4)$$

where  $\mu$  is the dynamic viscosity coefficient,  $Pr$  the Prandtl number, and using the Stokes hypothesis,  $\lambda = -\frac{2}{3}\mu$ .

## 2 DG Discretization

The weak formulation of Eq. (1) can be obtained by multiplying a "test function"  $\mathbf{v}$ , integrating over the domain  $\Omega$  and performing integration by parts

$$\int_{\Omega} \mathbf{v} \frac{\partial \mathbf{u}}{\partial t} d\Omega + \oint_{\partial\Omega} \mathbf{v} \mathbf{F}(\mathbf{u}, \nabla \mathbf{u}) \cdot \mathbf{n} d\delta - \int_{\Omega} \nabla \mathbf{v} \cdot \mathbf{F}(\mathbf{u}, \nabla \mathbf{u}) d\Omega = 0 \quad \forall \mathbf{v} \quad (5)$$

where  $\mathbf{F}(\mathbf{u}, \nabla \mathbf{u}) = \mathbf{F}_e(\mathbf{u}) - \mathbf{F}_v(\mathbf{u}, \nabla \mathbf{u})$ ,  $\partial\Omega$  is the boundary of  $\Omega$ .

The integrals over the domain  $\Omega$  can be expanded into the sum of integrals over a collection of non-overlapping triangle elements  $\{E\}$ . The semi-discrete equations for element  $E$  can be written as

$$\frac{\partial}{\partial t} \int_E \mathbf{v}_h \mathbf{u}_h d\Omega + \oint_{\partial E} \mathbf{v}_h \mathbf{F}(\mathbf{u}_h, \nabla \mathbf{u}_h) \cdot \mathbf{n} d\delta - \int_E \nabla \mathbf{v}_h \cdot \mathbf{F}(\mathbf{u}_h, \nabla \mathbf{u}_h) d\Omega = 0 \quad \forall \mathbf{v}_h \quad (6)$$

where  $\partial E$  is the boundary of  $E$ . In each element, the functions  $\mathbf{u}_h$  and  $\mathbf{v}_h$ , which are the approximations to  $\mathbf{u}$  and  $\mathbf{v}$ , are given by

$$\begin{cases} \mathbf{u}_h(\mathbf{x}, t) = \sum_{i=0}^n \mathbf{U}_i(t) \varphi_i(\mathbf{x}) \\ \mathbf{v}_h(\mathbf{x}, t) = \sum_{i=0}^n \mathbf{V}_i(t) \varphi_i(\mathbf{x}) \end{cases} \quad (7)$$

where the expansion coefficients  $\mathbf{U}_i$  and  $\mathbf{V}_i$  denote

the degrees of freedom of the numerical solution and the test function in element  $E$ ,  $\varphi_i$  the  $n$  (shape) basis functions of degree  $p$ . Since Eq. (6) must be satisfied for any element  $E$  and function  $\mathbf{v}_h$  and  $\mathbf{v}_h$  are a linear combination of  $n$  shape functions  $\varphi_i$ , Eq. (6) is equivalent to the following system

$$\frac{\partial}{\partial t} \int_E \varphi_i \mathbf{u}_h d\Omega + \oint_{\partial E} \varphi_i \mathbf{F}(\mathbf{u}_h, \nabla \mathbf{u}_h) \cdot \mathbf{n} d\delta - \int_E \nabla \varphi_i \cdot \mathbf{F}(\mathbf{u}_h, \nabla \mathbf{u}_h) d\Omega = 0 \quad 1 \leq i \leq n \quad (8)$$

Note that, no global continuity is enforced on  $\mathbf{u}$  and  $\varphi_i$ , discontinuities are allowed over element interfaces. Flux terms are not unique at element interfaces due to the discontinuous function approximation. The physical normal flux  $\mathbf{F}(\mathbf{u}_h, \nabla \mathbf{u}_h) \cdot \mathbf{n}$  in Eq. (8) is replaced by a numerical flux  $\mathbf{H}(\mathbf{u}_h^-, \nabla \mathbf{u}_h^-, \mathbf{u}_h^+, \nabla \mathbf{u}_h^+, \mathbf{n})$ , which is calculated using the internal  $\mathbf{u}_h^-$ , external interface state  $\mathbf{u}_h^+$  and the normal vector  $\mathbf{n}$  pointing outward from  $E$ . The numerical flux for the inviscid part of the equations can be analogous to those employed in upwind finite volume methods. In the paper, the LLF scheme is used<sup>[10, 24]</sup>.

In the context of the DG method an auxiliary variable  $\boldsymbol{\theta} = \nabla \mathbf{u}$  is introduced for the treatment of the viscous flux. Since DG methods obey to the hyperbolic systems of conservation laws, the following system of two first-order equations is obtained

$$\begin{cases} \boldsymbol{\theta} = \nabla \mathbf{u} \\ \partial_t \mathbf{u} + \nabla \cdot \mathbf{F}_e(\mathbf{u}) - \nabla \cdot \mathbf{F}_v(\mathbf{u}, \nabla \mathbf{u}) = 0 \end{cases} \quad (9)$$

Similar to the treatment of Eq. (1), the weak formulation of the first equation can be obtained by applying the DG discretization

$$\int_E \varphi_i \boldsymbol{\theta} d\Omega - \oint_{\partial E} \varphi_i (\mathbf{u}_{0h} - \mathbf{u}_h) d\delta - \int_E \varphi_i \nabla \mathbf{u}_h d\Omega = 0 \quad (10)$$

where  $\mathbf{u}_{0h} = \frac{1}{2}(\mathbf{u}_h^- + \mathbf{u}_h^+)$  if it is on the internal interface, and  $\mathbf{u}_{0h} = \mathbf{u}_{bh}$  ( $\mathbf{u}_{bh}$  denotes the boundary data) if it is on the boundary face. A function  $\mathbf{r}_e$  is suggested to be introduced<sup>[5]</sup>

$$\int_E \varphi_i \mathbf{r}_e d\Omega = \int_e \varphi_i (\mathbf{u}_{0h} - \mathbf{u}_h) \mathbf{n} d\delta \quad \forall e \in \partial E \quad (11)$$

$\boldsymbol{\theta}$  can be written as the following formulation

by comparing Eqs. (10), (11)

$$\boldsymbol{\theta} = \nabla \mathbf{u}_h + \sum_{e \in \partial E} \mathbf{r}_e \quad (12)$$

where  $\sum_{e \in \partial E} \mathbf{r}_e$  represents the effect of interface discontinuities, and  $\boldsymbol{\theta}$  can be regarded as a modified gradient of the variables  $\mathbf{u}$ . With the modification, the DG discretization for the second equation of system (9) can be written as

$$\frac{\partial}{\partial t} \int_E \varphi_i \mathbf{u}_h d\Omega + \sum_{e \in \partial E} \int_e \varphi_i \mathbf{H}_h(\mathbf{u}_h^-, \nabla \mathbf{u}_h^-, \mathbf{u}_h^+, \nabla \mathbf{u}_h^+, \mathbf{n}) d\delta - \int_E \nabla \varphi_i \cdot \mathbf{F}(\mathbf{u}, \nabla \mathbf{u} + \sum_{e \in \partial E} \mathbf{r}_e) d\Omega = 0 \quad (13)$$

The numerical flux function  $\mathbf{H}_h$  includes the inviscid numerical flux  $\mathbf{H}(\mathbf{u}_h^-, \mathbf{u}_h^+, \mathbf{n})$  and the viscous numerical flux function  $\mathbf{H}_v$ . In the paper,  $\mathbf{H}_v$  is given by the average value of the viscous fluxes on the interface.

$$\begin{aligned} \mathbf{H}_v(\mathbf{u}_h^-, \nabla \mathbf{u}_h^- + \mathbf{r}_e^-, \mathbf{u}_h^+, \nabla \mathbf{u}_h^+ + \mathbf{r}_e^+, \mathbf{n}) = \\ \frac{1}{2} [\mathbf{F}_v(\mathbf{u}_h^-, \nabla \mathbf{u}_h^- + \mathbf{r}_e^-) + \mathbf{F}_v(\mathbf{u}_h^+, \nabla \mathbf{u}_h^+ + \mathbf{r}_e^+)] \cdot \mathbf{n} \end{aligned} \quad (14)$$

### 3 Relaxation

By assembling together all the elemental contributions, the semidiscrete equations can be written as

$$\mathbf{M} \frac{d\mathbf{U}}{dt} + \mathbf{R}(\mathbf{U}) = 0 \quad (15)$$

where  $\mathbf{M}$  is the mass matrix,  $\mathbf{U}$  the global vector of the degrees of freedom, and  $\mathbf{R}(\mathbf{U})$  the residual vector. Due to the block diagonal structure of  $\mathbf{M}$ , the time integration of the unsteady problem presented below can be accomplished by means of an explicit method. In the paper, the TVB Runge-Kutta schemes is used<sup>[6]</sup>.

For steady problems, the Newton method<sup>[23]</sup> is used to solve the nonlinear system in Eq. (15)

$$\mathbf{U}^{n+1} = \mathbf{U}^n + \omega \Delta \mathbf{U}^n \quad (16)$$

where  $\omega$  is the under-relaxation factor and  $\Delta \mathbf{U}^n$  is obtained by solving the following linear system

$$\frac{\partial \mathbf{R}^n}{\partial \mathbf{U}} \Delta \mathbf{U}^n = -\mathbf{R}^n \quad (17)$$

Note that at each time-step a linear system of algebraic equations must be solved and  $\frac{\partial \mathbf{R}^n}{\partial \mathbf{U}}$  is a

$n \times n$  sparse block matrix, where  $n$  is the number of elements. The number of nonzero blocks of a generic  $i$  is equals to the number of elements around. Only the nonzero blocks need to be stored and the linear system (17) at each Newton iteration is solved by means of the Block Gauss-Seidel method<sup>[23]</sup>.

In order to improve the conditioning of the linear system (17), a pseudo-time derivative is introduced to original discrete system<sup>[25]</sup>

$$\mathbf{R}^n = \mathbf{R}^{n-1} - \left( \frac{\mathbf{U}^n - \mathbf{U}^{n-1}}{\Delta T^n}, \mathbf{W} \right) \quad (18)$$

It can be easily observed from Eqs. (17), (18) that  $\frac{\partial \mathbf{R}^n}{\partial \mathbf{U}}$  will become diagonally dominant with small  $\Delta T^n$ , which makes the Block Gauss-Seidel method easier to converge.

## 4 Adaptive Strategy

The entire computation procedure starts from solving  $p = 0$  ( $p$  is the order of the basic functions) solution on a very coarse initial grid. As the order  $p$  increases, the mesh needs to be refined in the region where the solution is not smooth enough. After a number of iteration steps, the solution in the region mentioned may become smooth enough. Then, the elements which have been refined should be coarsened to reduce the computational expense when the solution becomes smooth enough.

The vorticity  $v$  exists everywhere in the viscous flow. Moreover, due to the great velocity gradient in the shear layer and the vortices, it can be huge in these regions. In the paper, the vorticity  $v$  is used as the adaptivity sensor.

### 4.1 Mesh refinement

During the computational process, elements should be refined when  $v$  is larger than the pre-defined upper limit. The "father" element which needs to be refined is divided into four "child" elements (See Fig. 1).

It has been proved that the high-order DG method is inaccurate at curved solid walls if only a piecewise linear approximation of the boundary is employed and a higher-order boundary representa-

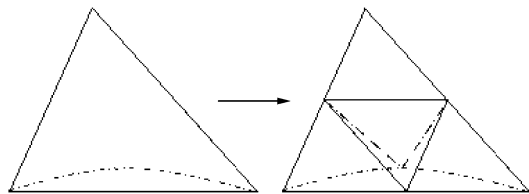


Fig. 1 Element refinement

tion can improve the accuracy of the solution<sup>[3]</sup>. In the paper, the edges on the solid wall of the boundary elements are represented by a high-order polynomial. The designed high-order (Sixth-order) curve can represent the real solid wall precisely (See the dash lines in Fig. 1).

If the boundary elements need to be refined, the mid-point of the boundary edge is found according to the designed curve and the new "child" elements are generated by connecting it with the mid-points of the other two edges as shown in Fig. 1. Two of the new "child" elements are on the solid wall and their boundary edge will also be represented by the high-order polynomial (See the dash lines in Fig. 1).

In order to avoid significant gradient in mesh size between neighboring elements, a smoothing strategy is employed. If element  $e$  in Fig. 2 needs to be refined, the neighboring element  $f$  must also be refined to avoid extreme difference in local mesh size. In another word, the maximum difference between refinement times of neighboring elements is 1. At the same time, a minimum mesh size  $h_{\min}$  is pre-defined and the refinement will stop when the element's genomic size reaches  $h_{\min}$  to avoid the unlimited refinement of the element.

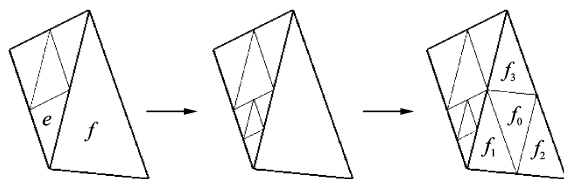


Fig. 2 Smoothing strategy

### 4.2 Mesh coarsening

During the computation, solution in the elements which have been refined may become smooth enough. In order to reduce the expense, these elements can be coarsened. In this case, the four "child" elements will merge into one "father"

element where the vorticity  $v$  is smaller than the pre-defined lower limit.

In Fig. 3, the solid lines indicate the elements which are not on the solid wall. If the "father" element is on the solid wall, its boundary edge will also be represented by a high-order polynomial to approach the real wall (See the dash lines in Fig. 3). Like in the refinement, in order to avoid extreme difference in local mesh size, "child" elements  $f_i (i = 0, 1, 2, 3)$  cannot be merged (See Fig. 2) if  $e$  is coarsened. In another word, mesh coarsening is the inverse operation of the mesh refinement and the max difference between refinement times of neighboring elements is 1. In addition, the initial element cannot be coarsened.

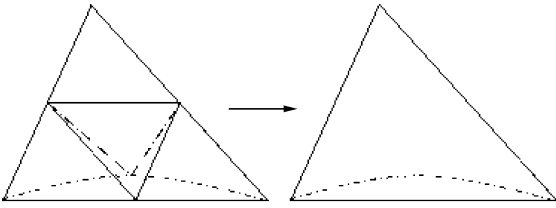


Fig. 3 Element coarsening

### 4.3 Data storage structure of grid

To ensure the high program transportability, the mesh adaptivity works as an independent module. It only changes the mesh and flow field solver module is not impacted. In the paper, all the information of the points, edges and elements is stored in a list structure. Fig. 4 demonstrates the refinement of element  $E$ .

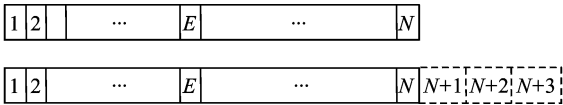


Fig. 4 Element increasing

In Fig. 4,  $E$  denotes the "father" element which needs to be refined. The center "child" element remains the same index as  $E$ , and the other three around the center "child" element (See Fig. 2) range at the end of the list. The same method is applied to the points and edges. Data transmission between grid module and flow field solver module will work well without any influence from mesh adaptivity.

Similarly, Fig. 5 shows that four "child" ele-



Fig. 5 Element decreasing

ments merge into a "father" element.

The three "child" elements  $f_i$  will be merged to the center "child" element  $E$ , and accordingly the index of the elements after  $f_3$  should be subtracted 3.

Two flags are attached to each element to indicate the initial index and refinement times during the mesh refinement. The mesh coarsening will work according to these two flags and the relationship between neighboring elements.

## 5 Numerical Results

### 5.1 Transonic flow around NACA0012 airfoil

Firstly, the subsonic viscous flow around the NACA0012 airfoil ( $Ma = 0.5, \alpha = 0^\circ, Re = 5\ 000$ ) is simulated. The initial mesh contains 478 elements, 260 grid points, and there are 32 grid points on the solid boundary (See Fig. 6).

Fig. 7 demonstrates the Mach contours and the vorticity distribution obtained when  $p = 4$  on the initial grid. It is obvious that the solution is not smooth enough because of the coarse grid in the boundary layer, which suggests smaller mesh size in this region to improve the accuracy of the solution.

In order to improve the accuracy of the solution and reduce the computational expense, the local adaptive method introduced above is applied. Fig. 8(a) shows the final mesh and the vorticity  $v$  distribution, where only the local mesh near the boundary and in wake region is refined. Fig. 8(b) shows the final local mesh after adaption compared with the initial mesh.

Fig. 9 depicts the Mach contours obtained with the adaptive method, where the solution is much smoother than that obtained on the initial mesh in Fig. 7. In addition, Fig. 9(b) shows the streamlines and the two symmetrical vortices<sup>[3]</sup> in the wake region are captured well.

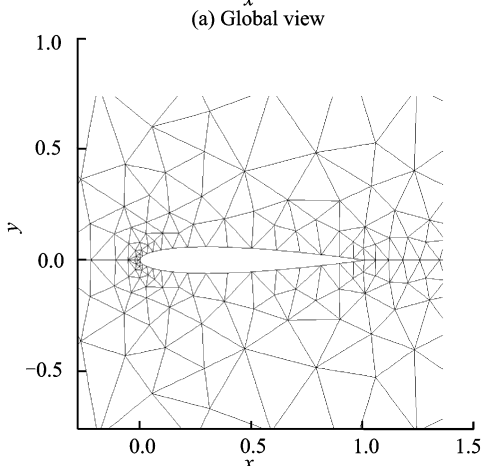
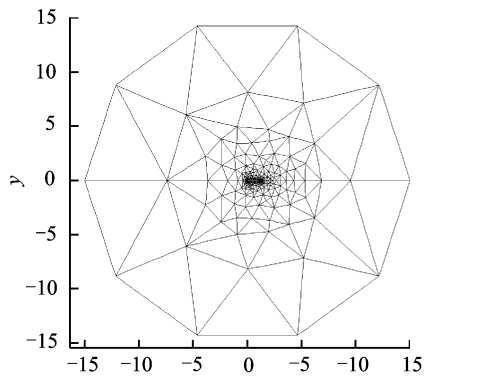


Fig. 6 Initial grid

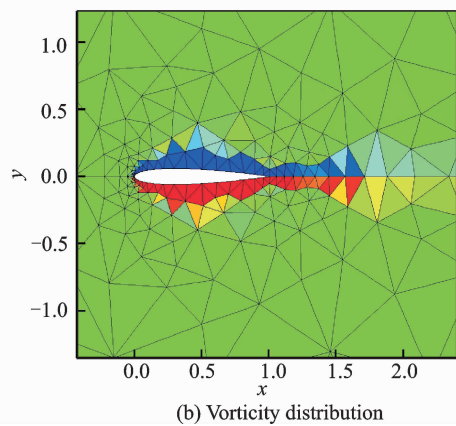
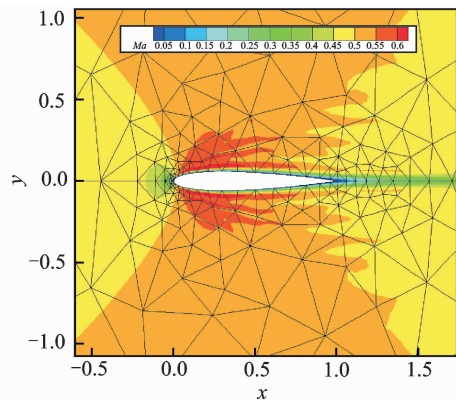


Fig. 7 Mach contours and vorticity  $v$  distribution when  $p=4$

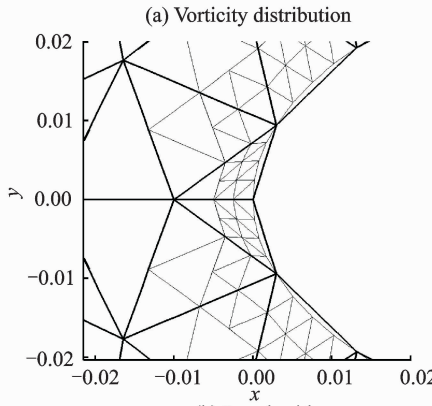
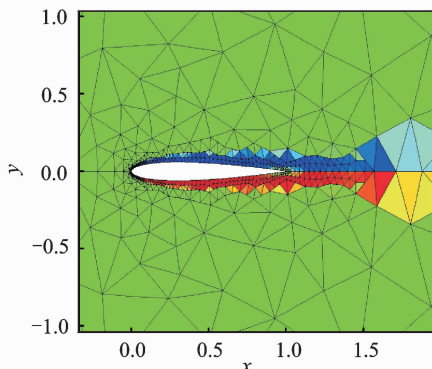


Fig. 8 Vorticity distribution and local grid after adaption

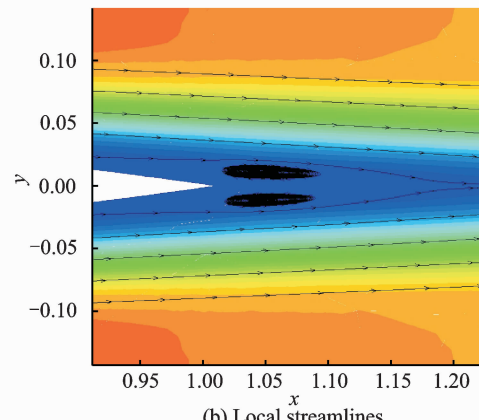
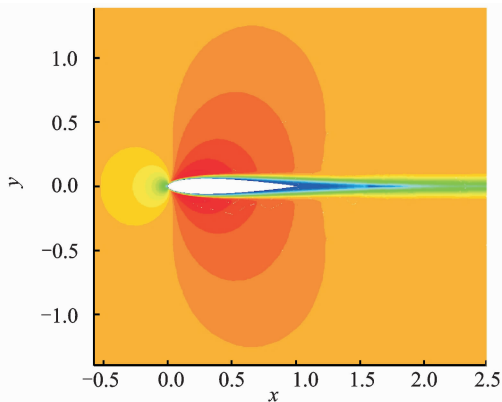


Fig. 9 Mach contours and streamlines after adaption

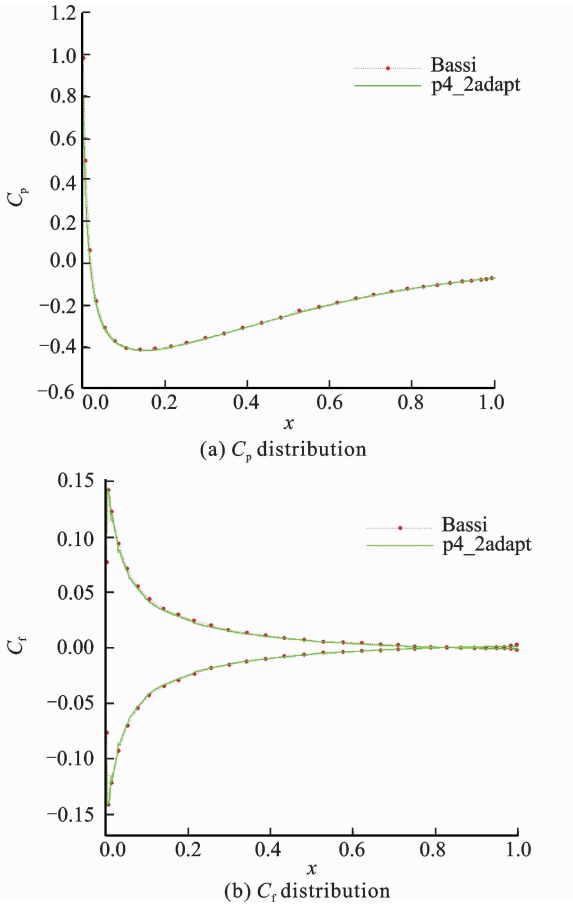


Fig. 10  $C_p$  and  $C_r$  distribution

### 5.2 Von Karman vortex street

The well-known Von Karman vortex street is simulated, where  $Ma = 0.1, \alpha = 0, Re = 150$ . The initial mesh is shown in Fig. 11, which contains 1114 elements and there are only 12 points on the solid boundary (See Fig. 11).

Fig. 12 demonstrates the Von Karman vortex street obtained when  $p = 4$  on the initial grid. The solution is not smooth and the resolution of the vortices is low because the initial mesh is not fine enough. Unfortunately, shedding vortices are not captured precisely on the initial coarse mesh even if the high-order scheme is applied. On the other hand, the intensity of the vortices is low due to the big numerical dissipation which is mainly caused by the large mesh size. It is suggested that the mesh in these regions should be refined.

Adaptive method introduced above is used in this case and the final local grid and Von Karman vortex street are shown in Fig. 13. Due to the smaller mesh size in the boundary layer region and

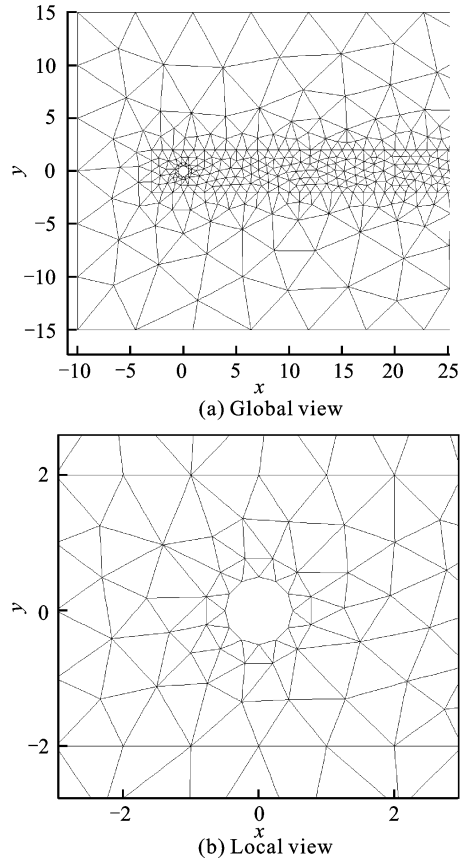


Fig. 11 Initial grid

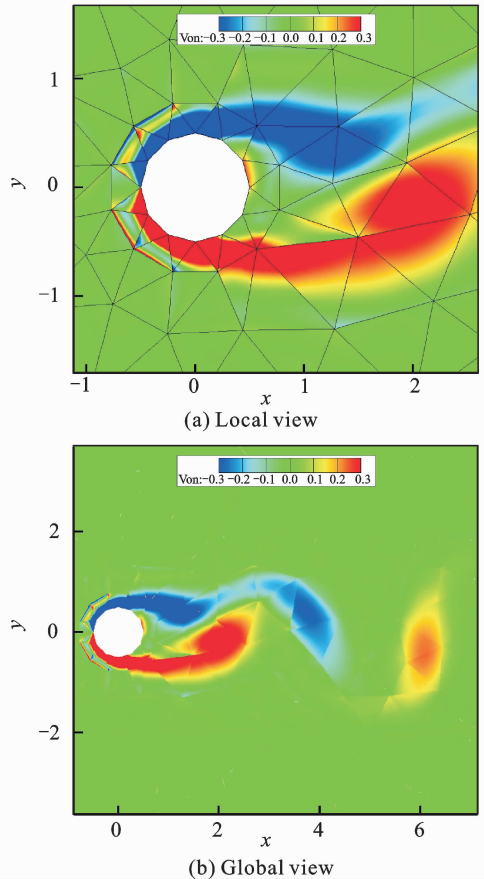


Fig. 12 Von Karman vortex street on initial mesh

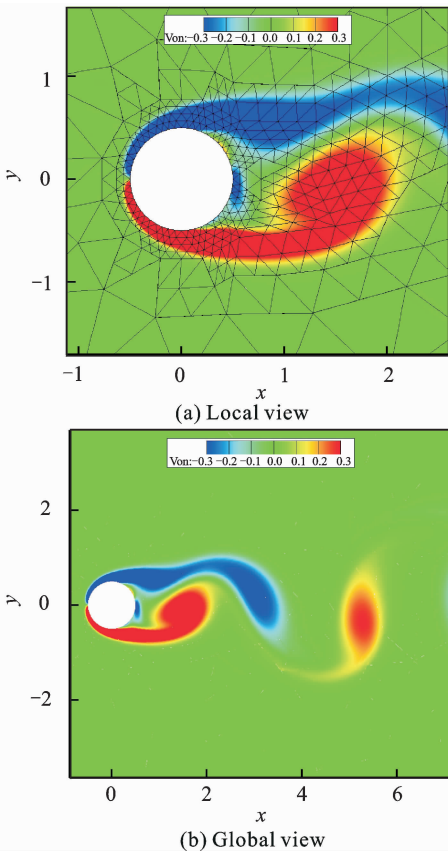


Fig. 13 Von Karman vortex street with adaptive method

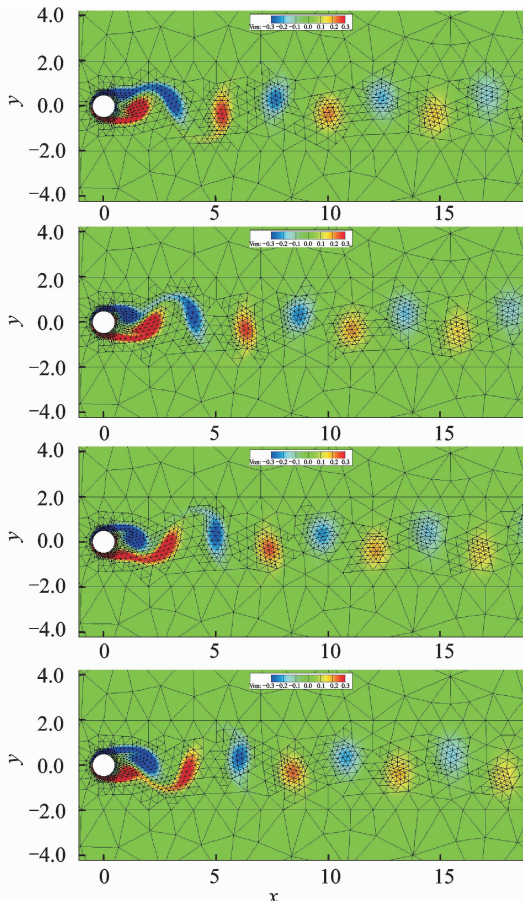


Fig. 14 Von Karman vortex street with mesh adaptivity

wake region after adaption, the solution is much smoother compared with that in Fig. 12 and the intensity of the vortices is enhanced because of the low dissipation.

In the paper, refinement and coarsening always work simultaneously. The elements where the vorticity is greater than the pre-defined upper limit are refined, and those where the vorticity is smaller than the pre-defined lower limit are coarsened to reduce the computation and storage expense if they have been refined. Fig. 14 shows the Von Karman vortex street with the adaptive method introduced above on several time of one period.

The Von Karman vortex street is a quasi-steady case because of its periodicity. The number of elements in the entire domain should vary in a small scale for which it is a good case to test the behavior of the introduced method. Fig. 15 shows the history of the number of the elements over the entire domain, where the element number varies around 2 500.

The evolution of the drag and lift coefficient in time is shown in Fig. 16 while the period of vortex shedding (Strouhal number) is found to be  $St=0.185$ . In Table 1, the variations of lift coefficient  $C_l$  and drag coefficient  $C_d$  are documented with amplitudes and mean values and the results of sixth-order finite difference scheme are also included as a reference<sup>[26]</sup> for comparison. It is clearly that the accuracy of lift and drag coefficient is drastically improved by using the adaptive method.

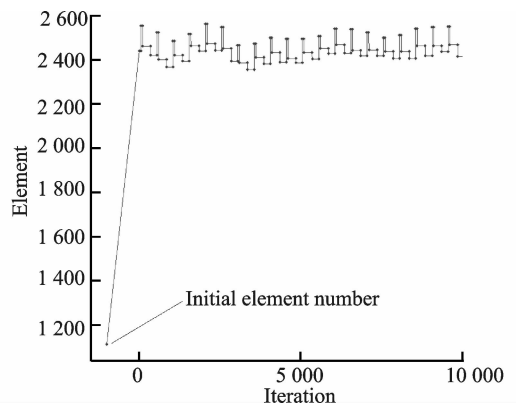


Fig. 15 Number of element during iteration



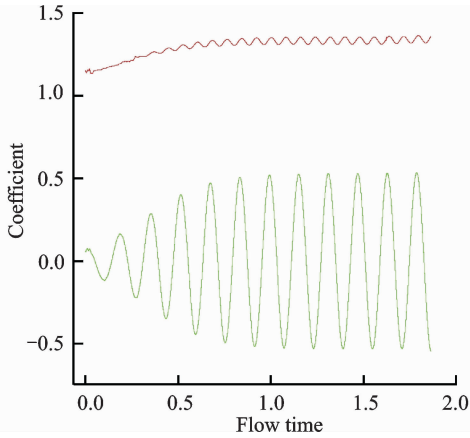


Fig. 16 Variation of lift and drag coefficient

Tab. 1 Comparison between lift and drag coefficient

Mesh	$C_l$ amplitude	$C_d$ amplitude	$C_d$ mean
Initial mesh	0.48	0.020	1.21
Adaptivity	0.53	0.027	1.33
Reference	0.52	0.026	1.32

## 6 Conclusions

An h-adaptive DG method is developed to solve the two-dimensional compressible laminar N-S equations. The vorticity  $v$  works well as the sensor of the mesh adaptivity in the subsonic viscous flow cases. In order to ensure the accuracy of the solution, a high-order approximation boundary is designed to approach the real solid wall during the h-adaptivity. Numerical results show that grid refinement and coarsening only work in the local region and the accuracy of the solution is improved at low expense.

### Acknowledgement

This work was supported by the National Natural Science Foundation of China (11272152).

### References:

- [1] REED W H, HILL T R. Triangular mesh methods for the neutron transport equation; Los Alamos Report LA-UR-73-479[R], 1973.
- [2] LESAIN P, RAVIART P A. On a finite element method for solving the neutron transport equation[J]. Mathematical Aspects of Finite Elements in Partial Differential Equations, 1974(33): 89-123.
- [3] BASSI F, REBAY S. A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier-Stokes equations[J]. Journal of Computational Physics, 1997, 131(2): 267-279.
- [4] BASSI F, REBAY S. High-order accurate discontinuous finite element solution of the 2D Euler equations [J]. Journal of Computational Physics, 1997, 138(2): 251-285.
- [5] BASSI F, REBAY S. A high order discontinuous Galerkin method for compressible turbulent flows [M]. [S. l.]: Springer Berlin Heidelberg, 2000: 77-88.
- [6] COCKBURN B, SHU C W. The Runge-Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems[J]. Journal of Computational Physics, 1998, 141(2): 199-224.
- [7] COCKBURN B, SHU C W. The local discontinuous Galerkin method for time-dependent convection-diffusion systems[J]. SIAM Journal on Numerical Analysis, 1998, 35(6): 2440-2463.
- [8] LU H, BERZINS M, GOODYER C E, et al. Adaptive high-order discontinuous Galerkin solution of elasto-hydrodynamic lubrication point contact problems[J]. Advances in Engineering Software, 2012, 45(1): 313-324.
- [9] LU H, XU Y, GAO Y, et al. A CFD-based high-order discontinuous Galerkin solver for three dimensional electromagnetic scattering problems[J]. Advances in Engineering Software, 2015, 83: 1-10.
- [10] LU H, SUN Q. A straight forward hp-adaptivity strategy for shock-capturing with high-order discontinuous Galerkin methods [J]. Advances in Applied Mathematics and Mechanics, 2014, 6(1): 135-144.
- [11] LU H, SUN Q, QIN W L. 3D numerical solution of aero-noise with high-order discontinuous Galerkin method[J]. Transactions of Nanjing University of Aeronautics & Astronautics, 2013,30(3): 227-231.
- [12] LV Hongqiang, ZHU Guoxiang, SONG Jiangyong, et al. High-order discontinuous Galerkin solution of linearized Euler equations [J]. Chinese Journal of Theoretical and Applied Mechanics, 2011, 43(3): 621-624. (in Chinese)
- [13] LV H Q. High-order discontinuous Galerkin solution of low-Re viscous flows[J]. Modern Physics Letters B, 2009, 23(3): 309-312.
- [14] LV H Q, XU Y, GAO Y, et al. A high-order discontinuous Galerkin method for the two-dimensional time-domain Maxwell's equations on curved mesh [J]. Advances in Applied Mathematics & Mechanics, 2016,8(1):104-116.
- [15] LV H Q, CAO K, WU L B Y, et al. High-order mesh generation for discontinuous Galerkin methods

- based on elastic deformation[J]. *Advances in Applied Mathematics & Mechanics*, 2016, 8(4): 693-702.
- [16] Zhang Laiping, Liu Wei, He Lixin, et al. A class of discontinuous Galerkin/finite volume hybrid schemes based on the "static re-construction" and "dynamic re-construction"[J]. *Chinese Journal of Theoretical and Applied Mechanics*, 2010, 42(6): 1013-1022. (in Chinese)
- [17] YU J, YAN C. An artificial diffusivity discontinuous Galerkin scheme for discontinuous flows[J]. *Computers & Fluids*, 2013, 75: 56-71.
- [18] YU Jian, YAN Chao. Study on discontinuous Galerkin method for Navier-Stokes equations[J]. *Chinese Journal of Theoretical and Applied Mechanics*, 2010, 42(5): 962-969. (in Chinese)
- [19] YANG X, JAMES A J, LOWENGRUB J, et al. An adaptive coupled level-set/volume-of-fluid interface capturing method for unstructured triangular grids [J]. *Journal of Computational Physics*, 2006, 217(2): 364-394.
- [20] REMACLE J F, LI X, SHEPHARD M S, et al. Anisotropic adaptive simulation of transient flows using discontinuous Galerkin methods [J]. *International Journal for Numerical Methods in Engineering*, 2005, 62(7): 899-923.
- [21] XU Yun, YU Xijun. Adaptive discontinuous Galerkin methods for hyperbolic conservation laws[J]. *Chinese Journal of Computational Physics*, 2009, 26(2): 159-168. (in Chinese)
- [22] WU Di, YU Xijun. Adaptive discontinuous Galerkin method for Euler equations[J]. *Chinese Journal of Computational Physics*, 2010, 27(4): 492-500. (in Chinese)
- [23] HARTMANN R, HOUSTON P. Adaptive discontinuous Galerkin finite element methods for the compressible Euler equations[J]. *Journal of Computational Physics*, 2002, 183: 508-532.
- [24] XIA Yidong, WU Yizhao, LV Hongqiang, et al. Parallel computation of a high-order discontinuous Galerkin method on unstructured grids [J]. *Acta Aerodynamica Sinica*, 2011, 29(5): 537-541. (in Chinese)
- [25] HILLEWAERT K, CHEVAUGEON N, GEUZ-AINE P, et al. Hierarchic multigrid iteration strategy for the discontinuous Galerkin solution of the steady Euler equations[J]. *International Journal for Numerical Methods in Fluids*, 2006, 51(9/10): 1157-1176.
- [26] INOUE O, HATAKEYAMA N. Sound generation by a two-dimensional circular cylinder in a uniform flow[J]. *Journal of Fluid Mechanics*, 2002, 471: 285-314.

Mr. **Sun Qiang** received B. S. degree from College of Aerospace Engineering, Nanjing University of Aeronautics and Astronautics in 2011. In September 2011, he became a post-graduate student at the Aerodynamics Department. His research is focused on the discontinuous Galerkin method and mesh adaptivity.

Prof. **Lv Hongqiang** received B. S. degree from Aerodynamics Department of Nanjing University of Aeronautics and Astronautics in 1999 and M. S. degree in 2002 from the same university. He received Ph. D. and Doctor degrees from University of Leeds in 2006. In the same year, he joined in College of Aerospace Engineering of Nanjing University of Aeronautics and Astronautics. His current research interest includes the application of the discontinuous Galerkin methods and numerical simulation of the Maxwell equations.

Prof. **Wu Yizhao** received B. S. degree from University of Science and Technology of China in 1968, and received M. S. degree in 1981 and Ph. D. degree in 1987 from Nanjing Aviation Institute. He is currently a professor and doctoral supervisor at College of Aerospace Engineering of Nanjing University of Aeronautics and Astronautics, and his research interests are computational fluid dynamics and multi-grid.

(Executive Editor: Xu Chengting)

