ICAS 2021 SHANGHAI
International Council of the Aeronautical Sciences
32nd Congress of the International Council of the Aeronautical Sciences
September 6-10, 2021
Pudong Shangri-La, Shanghai, China

Stall prediction of aircraft with adaptive hybrid mesh[1]

Jing Tang[1,2], Jian Zhang[1], Naichun Zhou[1,*], Bin Li[1] & Pengcheng Cui[1]

[1]China Aerodynamics Research and Development Center Computational Aerodynamics Institute, Mianyang 621000, China
[2]Northwestern Polytechnical University, Xi'an 710072, PR China

Abstract

Aerodynamic stall which can cause crash of aircraft is a vital phenomenon in aviation scopes. Besides wind tunnel experiment, numerical simulation with computational fluid dynamics (CFD) is commonly used to predict the aerodynamics of stall. Not only the adopted numerical schemes but also the mesh that represents the discrete space can affect the accuracy of numerical simulation. The geometrical quality of mesh is a primary demand for CFD, a higher requirement to guarantee simulation precision is to dynamically optimize the distribution of mesh elements according to the flow features. In this paper, a robust adaptation method for hybrid unstructured mesh was developed for mesh optimization. Both the tetrahedral elements away from the surface of aircraft and the prism elements adjacent to the surface can be refined during that adaptation procedure. Based on a vortices identification method to drive the mesh adaptation, the flow-fields over a fly-wing aircraft at large angles of attack were simulated. The aerodynamic forces and moments and the stall angle of attack determined by CFD were compared with experimental data in wind tunnel. The results show that a significant improvement has achieved after mesh adaptation, which increases the mesh resolution in the vicinity of vortices separated from the leading edge of the fly-wing.

Keywords: stall prediction; fly-wing; mesh adaptation; hybrid mesh; vortices

1. Introduction

Aerodynamic stall is a great threat to the flight security of aircraft. Stall prediction is a vital aspect for aircraft design and aerodynamics analysis in aviation industry. Nowadays, wind tunnel experiment and numerical simulation with computational fluid dynamics (CFD) are two major ways to predict aerodynamics of stall. Both numerical schemes and computational mesh can affect the accuracy of the numerical result of CFD. The space schemes of order 2 with turbulence models are main stream to simulate general flow field of aircraft in industry [1], and their numerical results are widely accepted by aviation engineers and technicians. However, stall aerodynamics prediction is still a challenge for both wind tunnel experiment and numerical simulation. Besides turbulence model, the mesh is an essential ingredient to obtain a feasible aerodynamic data.

Compared with structured mesh, unstructured mesh is much more popular for engineering cases with complicated shape of geometry. What' more, generation of unstructured mesh is more efficient in the term of less topology consideration and time costing. In the last decade, a considerable body of research has been conducted in exploring the methods to automatically generate unstructured mesh [2-3]. Nevertheless, unstructured mesh generation is still a fairly user-intensive procedure that involves user's experience.

Geometrical quality of unstructured mesh, such as skewness equiangle and volume ratio, is just a primary demand for CFD. Much higher requirement to guarantee simulation precision is to optimize the distribution of mesh elements according to the features of flow filed, such as shock waves and separated vortices. Mesh adaptation is just such a class of method to dynamically adjust the mesh

1 Supported by the National Numerical Windtunnel project of China.
* Corresponding author.
E-mail address: tangjingn@foxmail.com (Tang J), zhccxl@qq.com (Zhou N)

distribution during the numerical iteration based on flow features [4].

Error estimation of mesh elements, mesh editing and parallel algorithm are three key parts of unstructured mesh adaptation. Among kinds of error estimation methods, feature based and output-based error estimation are widely used [5-6]. Mesh editing, including isotropic and anisotropic methods, are used to optimize the elements distribution. Recent progress about anisotropic mesh adaptation was given by Alauzet and Loseille [7]. Compared with anisotropic methods, isotropic methods have the typical advantage of robustness. Parallel algorithms about mesh adaptation have been investigated in recent years with the main focus on load rebalance after mesh adaptation [8-9].

In this study, the stall aerodynamics of a flying wing was simulated with mesh adaptation. A robust adaptation method was developed for unstructured hybrid mesh. The mesh elements both adjacent to the surface and away from the surface of aircraft can be refined in a similar procedure. The vortices were recognized with a feature detector based on the shear-stress ratio method. The precision of stall prediction of the flying wing can be improved significantly with mesh adaptation.

2. Numerical methods

2.1 Flow simulation method

The flow field should be solved on the original mesh prior to conducting mesh adaptation. The flow filed is simulated by the software NNW-FlowStar/MFlow© [10] (MFlow for short). This software is based on the cell-centered finite volume method, which can support polyhedral elements.

The governing equations of fluid adopted here is the Reynolds averaged Navier-Stokes(RANS) equations, whose conservative form can be expressed as

$$\frac{\partial}{\partial \tau}\iiint_V \boldsymbol{W} dV + \oiint_\Omega \left(\boldsymbol{H}_\mathrm{c}\cdot\boldsymbol{n} - \boldsymbol{H}_\mathrm{v}\cdot\boldsymbol{n}\right) dS = 0 \tag{1}$$

where $\boldsymbol{W} = \left[\rho \quad \rho u \quad \rho v \quad \rho w \quad \rho E\right]^\mathrm{T}$ is the conservative flow variables, in which ρ is the flow density, *u*, *v* and *w* is the three components of the velocity vector $\boldsymbol{v}$, and *E* is the energy; $\boldsymbol{H}_\mathrm{c}$ is the convective flux, $\boldsymbol{H}_\mathrm{v}$ is the viscous flux, $\boldsymbol{n}$ is the unit normal vector at the fluid boundary Ω pointing outward from the fluid domain.

The fluid domain discretization, which is also called mesh generation, is the basis of numerical simulation for spatial discretization of the governing equations. The discrete form of those governing equations can be derived by integrating the spatial items of equation (1). For a mesh element *i*, the discrete equation is given as

$$\frac{\partial(\mathcal{V}_i \boldsymbol{W}_i)}{\partial \tau} + \sum_{k=1}^{N}\left(\boldsymbol{H}_{ck}\cdot\boldsymbol{n}_k - \boldsymbol{H}_{vk}\cdot\boldsymbol{n}_k\right) S_k = 0 \tag{2}$$

where $\mathcal{V}_i$ is the volume of mesh element *i*, *N* is the total number of all the adjacent elements that share a mesh face. The flow variable of every element is represented by the value at the center of that element, which is the reason why this method is called cell-centered.

The flux computation in equation (2) is the essential part of simulation, and generally, the flux through one face *k* can be expressed as a function of the flow variables on both sides,

$$\boldsymbol{H}_k = f\left(\boldsymbol{Q}_{k,L},\ \boldsymbol{Q}_{k,R}\right) \tag{3}$$

where $\boldsymbol{Q}_{k,L}$ and $\boldsymbol{Q}_{k,R}$ are the flow variables on the left and right side of that face *k*, respectively. The precise expression of function f depends on the flux scheme. The variable on one side, for example the left side, can be reconstructed with the variable and its gradient at the center of the left-sided element. The gradient of flow variables can be reconstructed by the variables at the elements center.

In order to support polyhedral mesh elements, face based mesh connectivity, such as face-to-nodes and face-to-elements, was devised to compute the flux of all mesh elements in MFlow. The flux through each face is firstly calculated with specific flux scheme. Then the flux is added to its right element and subtracted from its left element. As long as all the faces are traversed, finally, the total flux of every element will be accumulated.

There is on right element for these faces on the boundary of fluid domain. A ghost element is added

on the right side with the aim to deal with boundary faces in a similar way as inner faces. Another benefit of ghost element is that we can apply boundary conditions to boundary faces through setting ghost element with proper flow variables.

More details about the numerical methods, such as temporal discretization and turbulence modelling, can be found in reference [1]. In this paper, Roe flux scheme [11] was used for flux calculation and SA turbulence model [12] was adopted to simulate separated flow. Parallel implementation in MFlow can be found in reference [13].

2.2 Error estimation for mesh adaptation

The flow separation is the essential reason of aircraft stall. Hence, the main aim of mesh adaptation here is to capture the vortices of separated flow. The shear-stress ratio method [14] is adopted here to recognize the zones of vortices located in. For a mesh element *i*, the shear-stress ratio $\tilde{Q}_i$ can be calculated as

$$\tilde{Q}_i = \frac{1}{2}\left(\frac{\omega_i^2}{S_i^2} - 1\right) \tag{4}$$

where ω and S are the shear and stress, respectively, which are defined as

$$\begin{aligned} \omega_i &= \sqrt{2\omega_{kj}\omega_{kj}} = \sqrt{\frac{1}{2}\left(\frac{\partial u_k}{\partial x_j} - \frac{\partial u_j}{\partial x_k}\right)^2} \quad k = 1,2,3 \\ S_i &= \sqrt{2S_{kj}S_{kj}} = \sqrt{\frac{1}{2}\left(\frac{\partial u_k}{\partial x_j} + \frac{\partial u_j}{\partial x_k}\right)^2} \quad j = 1,2,3 \end{aligned} \tag{5}$$

where u_1, u_2 and u_3 are the three components of flow velocity vector in three dimensional space. The estimated error e_i for element *i* is assumed to be proportional to the shear-stress ratio here,

$$e_i = C \cdot \tilde{Q}_i \tag{6}$$

where *C* can be set to 1.0. Given the prescribed critical error e_c, then the element *i* will be marked to be refined if its error satisfy the condition $e_i > e_c$.

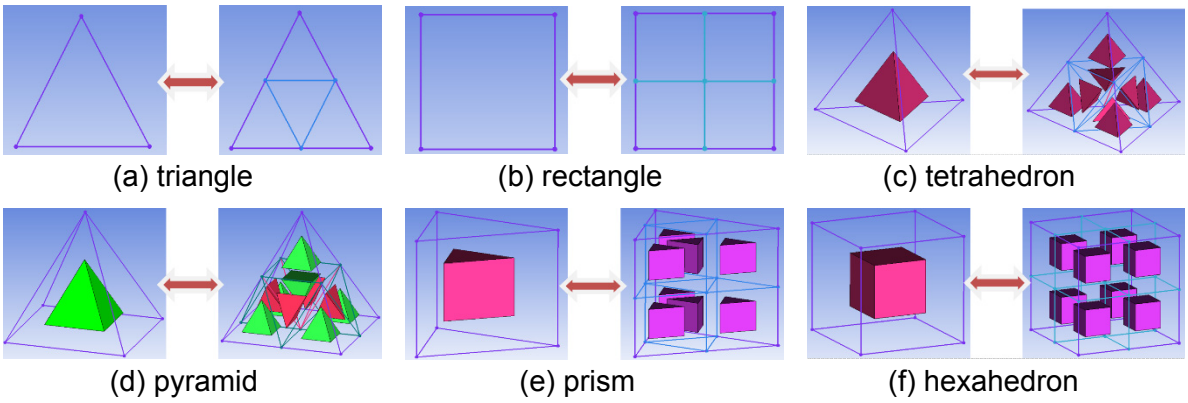
2.3 Mesh refinement method

A robust mesh adaptation method is indispensable for real engineering applications, because several low quality elements are inevitable as a result of their complicated geometry configuration. The distribution of mesh elements is a crucial aspect for a reliable numerical result in computational fluid dynamics. Mesh refinement is used to increase the mesh resolution in some sensitive zones.

Two-dimensional triangle and rectangle faces and three-dimensional hexahedron, prism, pyramid and tetrahedron cells are the standard elements for a regular hybrid unstructured mesh in three-dimensional space. In order to refine the mesh, every element marked to be refined is subdivided isotropically into several small child-elements. The subdivision patterns for standard types of elements are presented in Figure 1. One triangle or rectangle is subdivided to four similar child-elements, while one tetrahedron or prism or hexahedron cell is subdivided to eight similar children. And the element types of their children are preserved. One special case is the pyramid, as six pyramids and four tetrahedrons are generated after refinement.

The suspending points, illustrated in figure 2, will appear on the face and edges between one refined element and one of its unrefined neighbor elements. The mesh with suspending points, unfortunately, is not supported by general CFD software. One solution is to furtherly split the unrefined element. However, new points should be inserted at element's gravity center, or/and face center, or/and edge middle point. There are a great lot of topology types to split that unrefined element, which makes the adaptation system complex. Our solution is to convert these unrefined elements to polyhedrons, which can be supported by these CFD software whose data structure bases on mesh faces, such as MFlow. Taking the unrefined element in figure 2 for instance, the polyhedron consists of nine faces, and five of them are rectangle faces whilst the others are polygon faces consisting of five points. Polygon face cannot be dealt with by general software, so here we split it to standard faces, including triangle face and rectangle face. There are three conversion types for a polygon from a triangle and

five types for a polygon from a rectangle, as seen in figure 3. Compared with tens of types to split volume mesh elements to eliminate suspending nodes, it is fairly simple and compact, which can remarkably reduce the complexity of mesh adaptation system.



(a) triangle (b) rectangle (c) tetrahedron

(d) pyramid (e) prism (f) hexahedron

Figure 1 – Refinement patterns for 6 types of regular mesh elements.

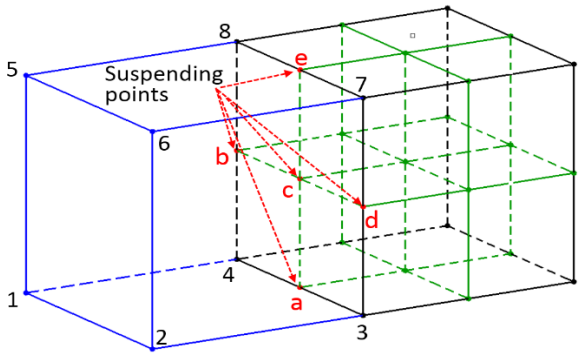
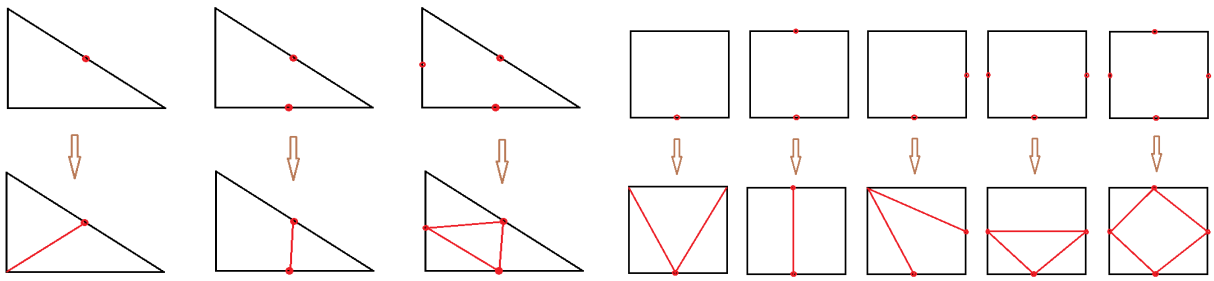


Figure 2 – Schematic diagram of suspending node (in red color).



(a) polygon from a triangle (b) polygon from a rectangle

Figure 3 – Conversion patterns from a polygon to standard faces.

For a conformal mesh after refinement, the elements which have children are inactive, in other words, these elements replaced by its children will not be used for numerical simulation. The remaining mesh elements are active, which will be used to construct the mesh connectivity for subsequent flow simulation.

2.4 Load rebalance algorithms

The parallelization implementation is based on the distributed parallel computers system, for which both the hardware and software are fairly mature and widely used for real engineering applications. Parallel computing for a numerical simulation is started with mesh partition with a graph algorithm here provided by Metis [15]. Then each processor accesses its own portion of the mesh and carries out calculations simultaneously in most of time. The message passing interface (MPI) [16] is adopted here for flow data communication among different processors when data exchange is required. This

parallel strategy is called data parallelism.

The ghost elements on the physical boundaries are usually used to apply boundary conditions. The ghost elements on parallel interfaces, illustrated in figure 4, are set as the backup of the real physical elements on other processors (remote elements) and used to save the data of flow variables received from other processors by MPI.

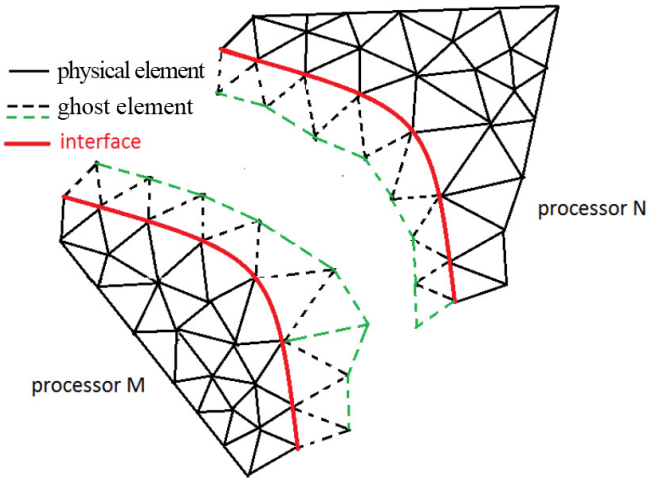


Figure 4 – The physical elements and ghost elements on parallel interface for parallel computing.

Mesh refinement commonly occurs only on parts of processors, which will break the load balance for the subsequent flow simulation. The two steps method of repartition-migration [17] is used here to dynamically rebalance the computing load.

The mesh after adaptation is repartitioned basing on original mesh elements in parallel mode with ParMetis [18], and the weight of each original element is set to the number of its active children. After repartition step, a new processor id may be assigned to these original elements. Then the processor id is passed from original element to all its children.

In the migration step, the elements whose processor id have been changed and their neighbour elements sharing mesh node are transferred to their new ascription processor by MPI. The geometry data and flow data of all these elements are assembled together in a contiguous memory and then sent to ascription processor in order to improve the parallel efficiency by reducing starting MPI communication.

3. Numerical results

The stall aerodynamics of a flying wing of low aspect ratio was studied here. The flying wing is designed as a standard model for verification and validation of wind tunnel and numerical methods, leaded by China Aerodynamics research and development center with the corporation of many other research organizations in China. The geometry configuration is depicted in figure 5(a). Here we focused on the longitudinal stall aerodynamics, so we can use half model to reduce the computing cost. The original base mesh on the surface is shown in figure 5(b). There is about 49 thousand triangles on the wall of flying wing and about 3.65 million three-dimensional elements, including tetrahedrons, prisms and pyramids. The height of first layer of prism is 0.02 mm and the height increasing ratio is 1.25.

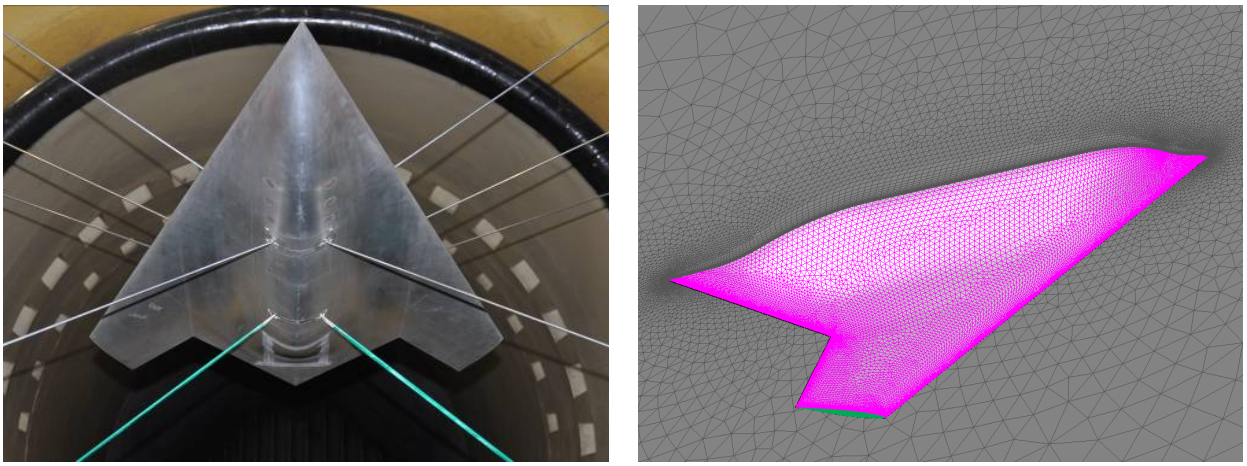
The separation flow field around the flying wing was simulated with mesh adaptation and 256 computer cores were used. The incoming air condition is: the static temperature is 101325 Pa, the static temperature is 288.15K, the Mach number is 0.2, the slide angle is zero. And a set of angles of attack are 20^o, 24^o, 28^o, 32^o, 36^o, 40^o, 42^o, 44^o, 48^o and 52^o. The critical error for the shear-stress ratio method for mesh adaptation is 0.2, and three iterations were conducted for each angle of attack. The number of mesh elements is about 80 million after mesh adaptation.

The lift coefficient predicted by CFD and by the wind tunnel experiment [19] are presented in figure 6(a). The stall angle of attack predicted by CFD on the original base mesh is about 36^o while 40^o by

wind tunnel experiment. Compared with data from wind tunnel, the deviation of lift coefficient by CFD on the original mesh starts at about 24° angle of attack. Similarly, a salient deviation of pitch moment coefficient by CFD on the original mesh can be observed in figure 6(b).

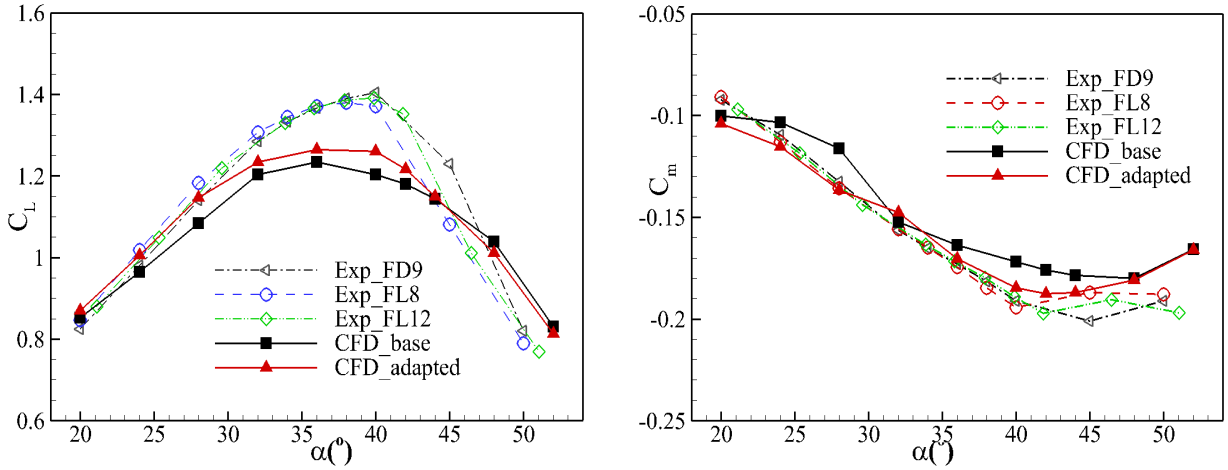
An overall improvement of aerodynamics prediction with mesh adaptation can be seen from the both plots in figure 6. As shown in figure 6(a), the stall angle of attack after mesh adaptation is bout 40°, which agrees much better with experiment data. Moreover, the figure 6(b) suggests that a significant improvement of predicted moment with mesh adaptation has been obtained. As we can see in the figure 7(a), the vortex above the aircraft breaks down, whereas it preserves after adaptation and so the lift is greater than that on initial mesh. The refined mesh elements cluster in the vortex zone according to the right picture in figure 7(b). As a result, the resolution of the vortex can be increased, and the stall aerodynamics can be predicted better.

The pressure distribution on the surface of the flying wing at four typical spanwise sections are shown in figure 8 at 40° angle of attack. The pressure on the upper surface of flying wing is much lower after mesh adaptation, which makes an increment of lift force. The main reason is that the resolution of mesh in the zone where separated vortices located in is increased after mesh adaptation, which decrease the dissipation of the strength of vortices. As a result, the stronger vortices generates much lower pressure on the upper surface.



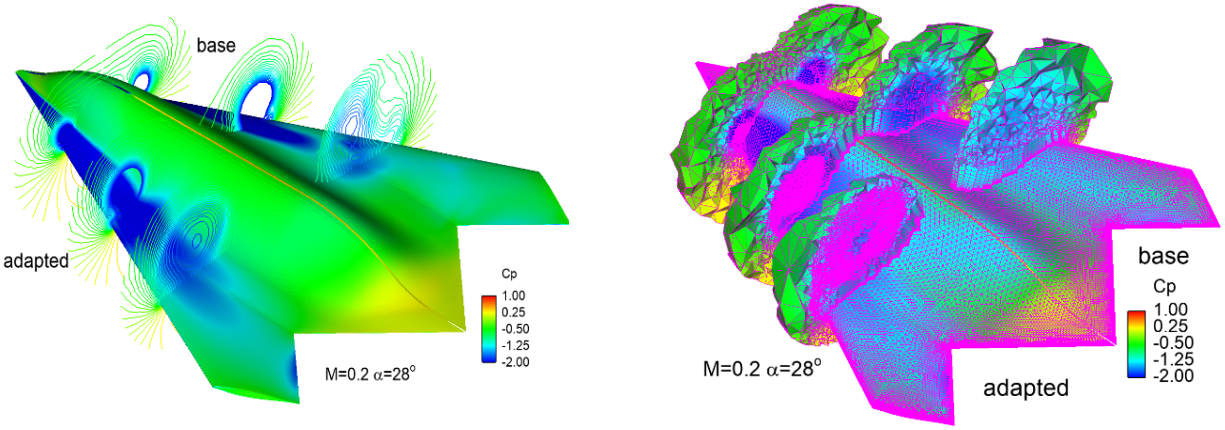
(a) geometry configuration (b) original surface mesh

Figure 5 – The geometry and original base mesh (right) of the flying wing.



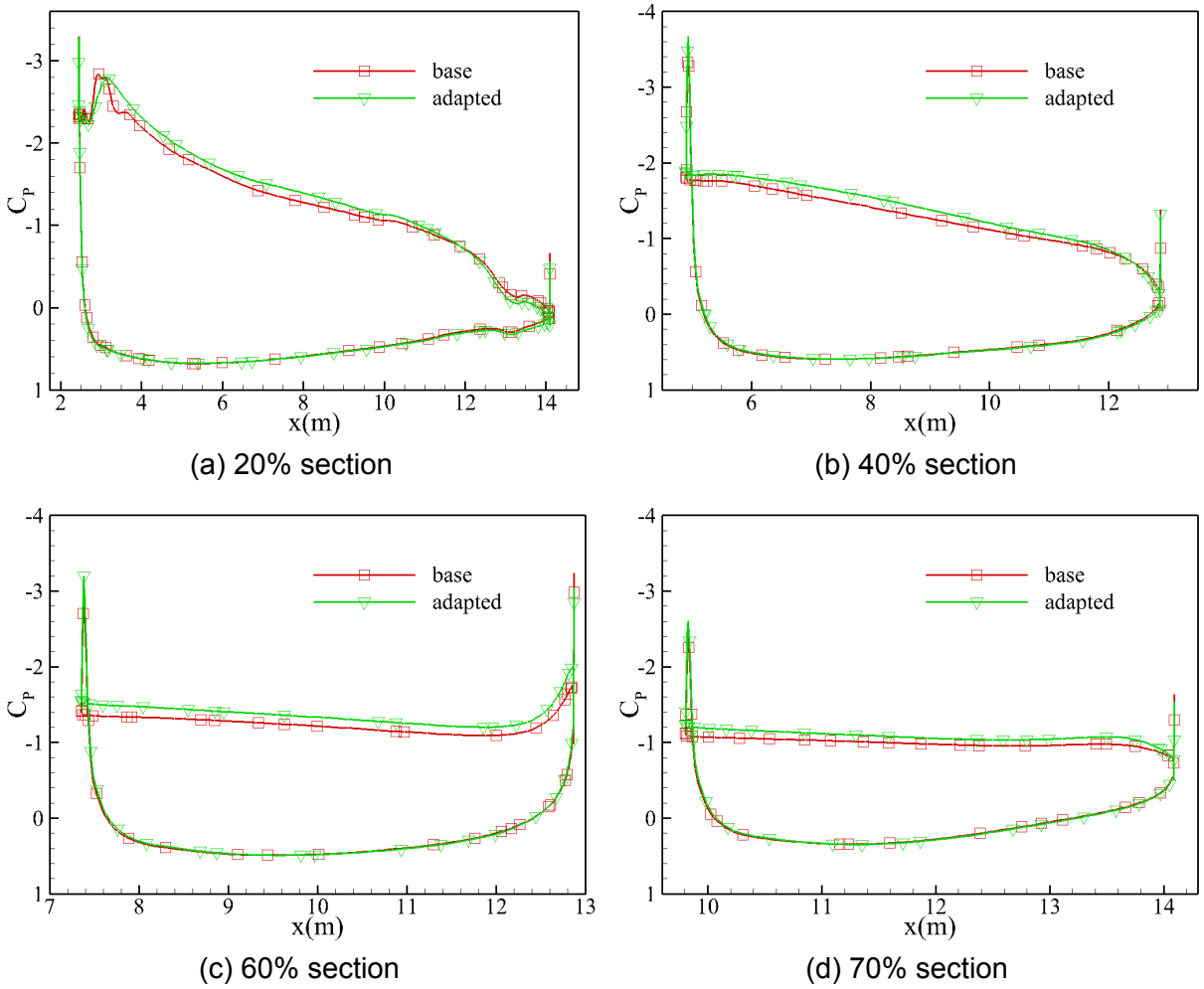
(a) lift coefficient (b) pitch moment coefficient

Figure 6 – The comparison of stall prediction by wind tunnel, CFD and CFD with mesh adaptation.



(a) the flow field (b) the distribution of mesh elements

Figure 7 – The vortices and mesh at three typical slices before and after adaptation.



(a) 20% section (b) 40% section

(c) 60% section (d) 70% section

Figure 8 – Comparison of the pressure distribution at 4 spanwise sections before and after mesh adaptation for the case of 40° angle of attack.

4. Concluding Remarks

In this paper, a robust mesh adaptation method for unstructured hybrid mesh is proposed for real engineering applications. The error estimate methods based on flow features are adopted to indicate the sensitive zones in which the elements should be refined. The isotropic subdivision is used to refine all the types of mesh elements, and every element containing suspending nodes is converted to a polyhedron. The parallel algorithms of repartition-migration for load rebalance is established. The stall aerodynamics of a flying wing of low aspect ratio was studied by CFD with mesh adaptation. The resolution of mesh in the zones containing vortices are improved after mesh refinement, which

is of benefit to maintain the strength of the vortices. As a result, the stall angle of attack, the lift coefficient and the pitch moment coefficient are all predicted more accurate after mesh optimization by mesh adaptation.

5. Contact Author Email Address

mailto: tangjingn@foxmail.com (Tang J), zhccxl@qq.com (Zhou N)

6. Copyright Statement

References

[1] Chen J, Zhang Y, Zhao H and Zhou G. Numerical investigations of the NASA common research model with aeroelastic twist. *Journal of Aircraft*, Vol. 55, No. 4, pp 1469-1481, 2018.

[2] Quadros W, Vyas V, Brewer M, Owen J and Shimada K. A computational framework for automating generation of sizing function in assembly meshing via disconnected skeletons. *Engineering with Computers*, Vol. 26, No. 3, pp 231-247, 2010.

[3] Chen J, Xiao Z, Zheng Y, Zheng J, Li C and Liang K. Automatic sizing functions for unstructured mesh generation. *International Journal for Numerical methods in Engineering*, Vol. 109, No. 4, pp 577-608, 2017.

[4] Park A, Krakos A, Michal T, Loseille A and Alonso J. Unstructured grid adaptation: status, potential impacts, and recommended investments toward CFD vision 2030. *46th AIAA Fluid Dynamics Conference*, Washington D.C., paper number AIAA 2016-3323, 2016.

[5] Tang J, Cui P C, Jia H Y, et al. Robust adaptation techniques for unstructured hybrid mesh. *Acta Aeronautica dt Astronautica Sinica*, Vol. 40, No. 10, pp 122894-1-14, 2019. (in Chinese)

[6] Fidkowski J and Darmofal L. Review of output-based error estimation and mesh adaptation in computational fluid dynamics. *AIAA Journal*, Vol. 49, No. 4, pp 673 -694, 2011.

[7] Alauzet F and Loseille A. A decade of progress on anisotropic mesh adaptation for computational fluid dynamics. *Computer-Aided Design*, Vol. 72, pp 13-39, 2016.

[8] Antepara O, Lehmkuhl O, Chiva J and Borrell R. Parallel adaptive mesh refinement simulation of the flow around a square cylinder at Re=22000. *Procedia Engineering*, Vol. 61, pp 246-250, 2013.

[9] Digonnet H, Coupez T, Laure P and Silva L. Massively parallel anisotropic mesh adaptation. *International Journal of High Performance Computing Applications*, Vol. 33, No 1, pp 1-22, 2017.

[10] Chen J, Zhang J, Tang J and Zhang Y. Numerical investigations of the Jaxa high-lift configuration standard model with MFlow solver, in *Numerical Simulation of the Aerodynamics of High-Lift Configurations*. 1st edition, Springer, pp 45-65, 2018.

[11] Roe L. Approximate Riemann solvers, parameter vectors, and difference schemes[J]. *Journal of Computational Physics*, Vol. 43, pp 357-372, 1981.

[12] Spalart R and Allmaras R. A one-equation turbulence model for aerodynamic flows. 30th Aerospace Sciences Meeting & Exhibit, Washington D.C., paper number AIAA paper 92- 0439, 1992.

[13] Tang J, Li B, Chen J and Gong X. Large scale parallel computing for fluid dynamics on unstructured grid. *15th International Symposium on Parallel and Distributed Computing*, Fuzhou, China. pp 64-69, 2016.

[14] Kamkar J, Wissink M, Sankaran V and Jameson A. Feature-driven Cartesian adaptive mesh refinement for vortex-dominated flows. *Journal of Computational Physics*, Vol. 230, pp 6271-6298, 2011.

[15] Karypis G and Kumar V. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM Journal on Scientific Computing*, Vol. 20, No. 1, pp 359-392, 1998.

[16] Gropp W, Lusk E and Thakur R. *Using MPI-2: advanced features of the message-passing interface*. 1st edition, MIT Press, 1999.

[17] Oliker L, Biswas R and Gabow N. Parallel tetrahedral mesh adaptation with dynamic load balancing. *Parallel Computing*, Vol. 26, No. 12, pp 1583-1608, 2000.

[18] Schloegel K, Karypis G and Kumar V. Wave front diffusion and LMSR: algorithms for dynamic repartitioning of adaptive meshes, *IEEE Transactions on Parallel and Distributed Systems*, Vol. 12, No. 5, pp 451-466, 2001.

[19] Wu J, Qin Y, Huang Z, Wei Z and Jia Y. Low speed experiment on longitudinal and lateral aerodynamic characteristics of the low aspect ratio flying wing calibration model. *ACTA Aerodynamica Sinica*. Vol. 34, No. 1, pp 125-130, 2016. (in Chinese)