Flow simulation system based on high order space-time extension of flux reconstruction method

Yi Lu[1]
CFS Ltd, Compass House, Vision Park, Histon, Cambridge CB24 9AD

Kai Liu[2]
BoXeR Solutions KK, Chuo-ku, Kobe, Japan, 651-0087

and

W.N.Dawes[3]
Whittle Lab, University of Cambridge, Cambridge, UK, CB3 0DY

High fidelity physics-based simulations demand access to real geometries, fast meshing, high accuracy in both space and time with less modelling. In this paper, a flow simulation system based on a high order space-time extension of the flux reconstruction method, is presented and tested. Arbitrary order of accuracy can be achieved on general hybrid unstructured meshes including tetrahedrons, pyramids, prisms and hexahedrons, to support complex geometries. Moreover, the efficient space-time extension of flux reconstruction method allows local time-stepping for accurate unsteady flow simulations. This reduces the number of functional evalutions compared to uniform time-stepping dramatically especially for flow with large-scale, multi-scale geometries. The flow simulation system, HOTNewt, combines high order mesh generation and smoothing, high order flux reconstruction implementation and its space-time extension to perform large eddy simulations(LES) on complex geometries and aims towards high accurate, high efficiency, low memory, and robust simulations for industrial level engineering flows.

Nomenclature

$\mathrm{d}t$ = time step
F = flux including inviscous part and viscous part of Navier-Stokes eqations
i = index of element
j = index of solution point on the single element
J = Jacobian matrix
K = order of polynomials
U = state conservative variable
x, y, z = coordinate of the physical domain
ξ, η, ζ = coordinate of the computational(local) domain
r = elemenet size ratio for 1D testing
t = physical time of simulation
τ = non-dimentional time between two steps
T_c = flow passing(through typical length scale) time of LES

---

[1] CFD Engineer, Cambridge Flow Solutions LTD, Cambridge, UK, and AIAA Associate Member.
[2] CFD Engineer, BoXeR Solutions KK, Kobe, Japan.
[3] Professor, Whittle lab, University of Cambridge, UK, and senior AIAA Member.

American Institute of Aeronautics and Astronautics

I. Introduction

Computational Fluid Dynamics(CFD) is widely used in both academia and different industry areas at present to understand flow mechanisms and support product design. However, almost all commercially available and in-house codes are based on algorithms developed more than 20 years ago. These codes are successfully used to solve complex geometry problems with the support of turbulence modeling but are generally applied limited to 2nd order accuracy. The improvement of simulation-based engineering design process relies on higher fidelity CFD simulations with less modelling. With the rapid progress in computational power, a rethink of current CFD algorithms and software is required.

There are a number of issues for advanced, practical and reliable CFD software to meet future requirements:

1) Real geometries supported by hybrid unstructured meshes are necessary to handle physical-based simulations with real geometries.
2) Reduced dependence on turbulence modelling: the lack for ability of turbulence modeling to predict many flow phenomenas, such as transitional flow and vortex dominated flows, are the main drawback of modern production CFD software which solve Reynolds Averaged Navier-Stokes(RANS) equations. High fidelity simulations are needed using enable large eddy simulations(LES) and direct numerical simulations(DNS), which have more general ability to handle turbulence without modeling.
3) High resolution for both space and time: LES/DNS requires high resolution by the numerical algorithms for both space and time discretisations, which suggest high order methods have strong advantages with better ability to correctly represent the whole energy spectrum resolved in LES/DNS.
4) High computational efficiency: one of the biggest challenge, may be the key problem, is the huge computational cost caused by not just by high resolution space discretisation requirements but also high accurate time resolution for vortex preservation, which indicate that practical numerical algorithms should achieve very high computational efficiency for both space and time discretisations. In this context, space(mesh and polynomial) and time adaptation is an effective method to reduce the computational cost.
5) Efficient usage of many-core computing system: during the last several years, the many-core systems like Nvidia GPU, AMD GPU and Intel Xeon PHI co-processors, have developed rapidly and are widely used for high performance computing(HPC) applications. They can provide much higher computing ability than traditional processors with lower price and less energy consumption. The rapid progress in many-core mobile phine chips is another future alternative for its much lower price and power consumption. Future high fidelity simulations should make use of these new hardwares/architectures to provide over-night predictions with affordable HPC systems.
6) Low-memory consumption: for LES using high order numerical schemes to resolve large-scale, real geometry problems, memory is another bottleneck, maybe as critical as computing power. In particular, the available memory on many-core systems is much less than normal processors, hence, more efficient useage of the memory is a key issue in heterogeneous computing system.
7) Supporting wide-range of flow problems: the flow simulation system needs to support low speed flow with low mach number preconditioning, subsonic/supersonic/hypersonic flows with shock/physical and moving geometries with ALE which are very important for engineering design, off-design running performance, and deformations.
8) Real-time monitoring and fast post-processing: for transient simulations like LES/DNS, the real-time monitoring for objective quantities such as mass flow, drag/lift coefficients, skin frinctions, etc, ought to be presented in real time for monitoring the simulations. Also parallel data-extracting and post-processing are necessary to deal with massive unsteady results generated by LES/DNS. Above all, extracting physical unstersting from these large simulations and cascading knowledge down to improve lower order modelling.
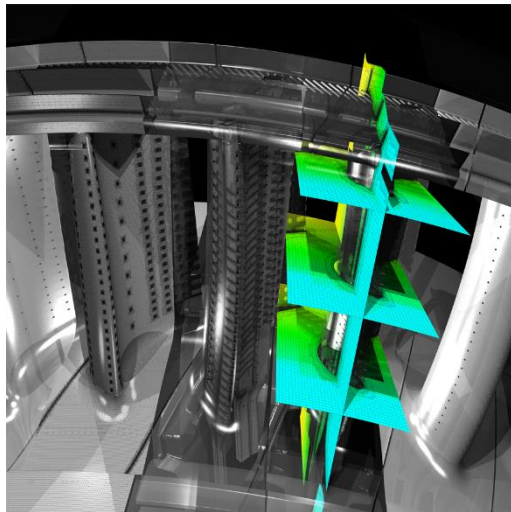
In practice, all the above issues are interrelated. Overall, we require a useful numerical scheme to achieve high order accuracy on hybrid unstructured mesh, with high computational efficiency, robust, low-memory and reliable for wide range of flow simulations. The flow solver should be integrated with advanced mesh generation, which can produce proper high order curved meshes for real geometries. This paper shows our progress to construct such a flow simulation system.

This paper is structured as follows. First, the framework of the end-to-end parallel high order simulating system, with the high order mesh generation for general hybrid unstructured meshes and smoothing are introduced. Next, the basic numerical formulation is reviewed, including high order space and time discretisations with analysis. Afterwards, some functionalities including shock-capturing, low mach number preconditioning and wall modelling

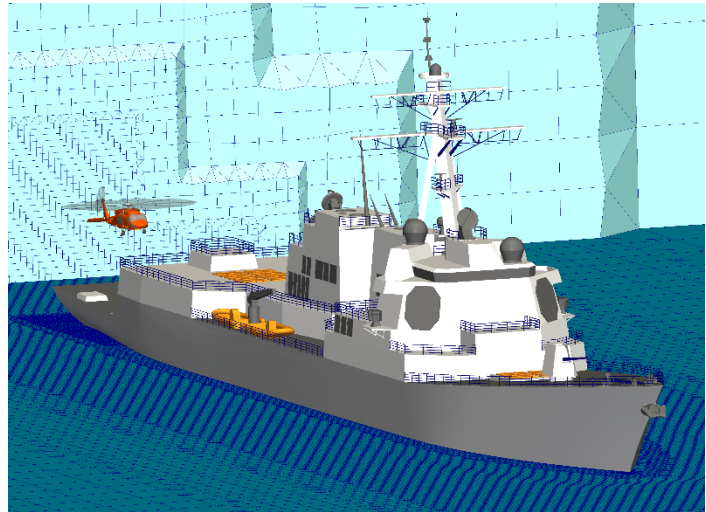American Institute of Aeronautics and Astronautics

are introduced. Then, the implementation of this code on many-core system is presented, following by a series of numerical simulations are performed and analyzed. Finally, some conclusions are drawn.
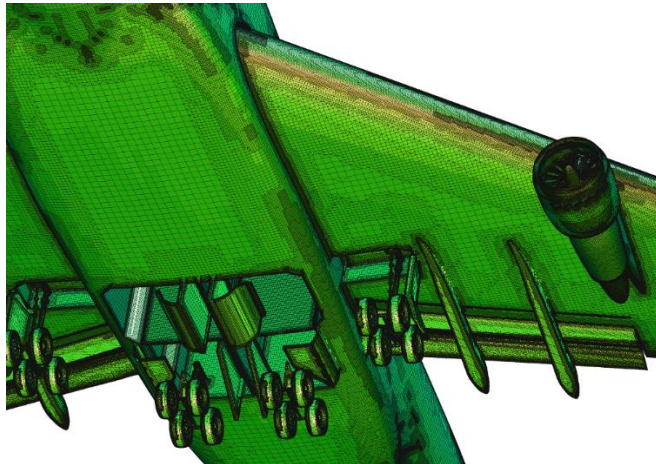
II.   End-to-End parallel high order simulating system

BOXERMesh is commericiallu available mesh generation software developed by Cambridge Flow Solutions LTD[1], which is fully parallized and scale on distributed memory, highly CAD-tolerant, scripted integration and automation for high quality mesh generation of large-scale, complex geometries, Fig.1 shows the application of BOXERMesh for different industrial problems. The main purpose of this work, is to combine this advanced parallel meshing system, with the newly developed high order, high efficient STEFR[2] solver, and the parallel post-processing software ParaView[3], to construct an end-to-end parallel high order simulating system for real geometry industrial problems, running on different type of modern computing resource including many-core systems.
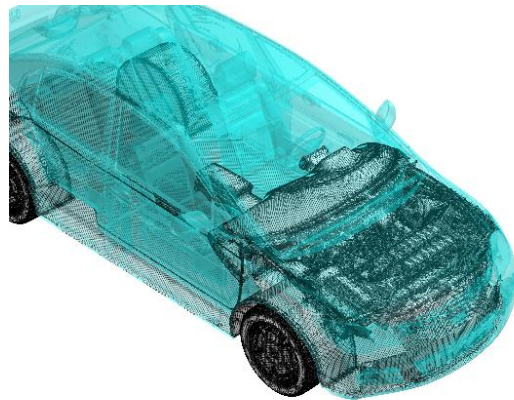

<table>
<tr><td>(a). Turbomachinery</td><td>(b). Marine</td></tr>
</table>

<table>
<tr><td>(c) Aeronautics</td><td>(d) Automotive</td></tr>
</table>

Figure 1. Application for different industrial fields for BOXERMesh.

The flowchart of this simulation system is shown in Fig.2. The automatic unsteady simulations and optimizations is aided by arbitrary volume refinement(AVR), which is used for routine industrial applications, but is also very powerful for flow-feature based remeshing during unsteady simulations as shown in Fig.3~Fig.5, in addition, p-adaptive refinement is an alternative because arbitrary order of acuuracy can be achieved by using the local reconstruction type high order special discretization.

3

American Institute of Aeronautics and Astronautics

The current high order curved meshing(modification) and smoothing are separate with respect to the basic mesh generation software, which is introduced in section III. In the near future, an integrated high order meshing system will be published for large scale, real geometry system, with improved high order geometry description using a relatively coarse mesh, to support both high order CFD simulation and general finite element analysis(FEA).
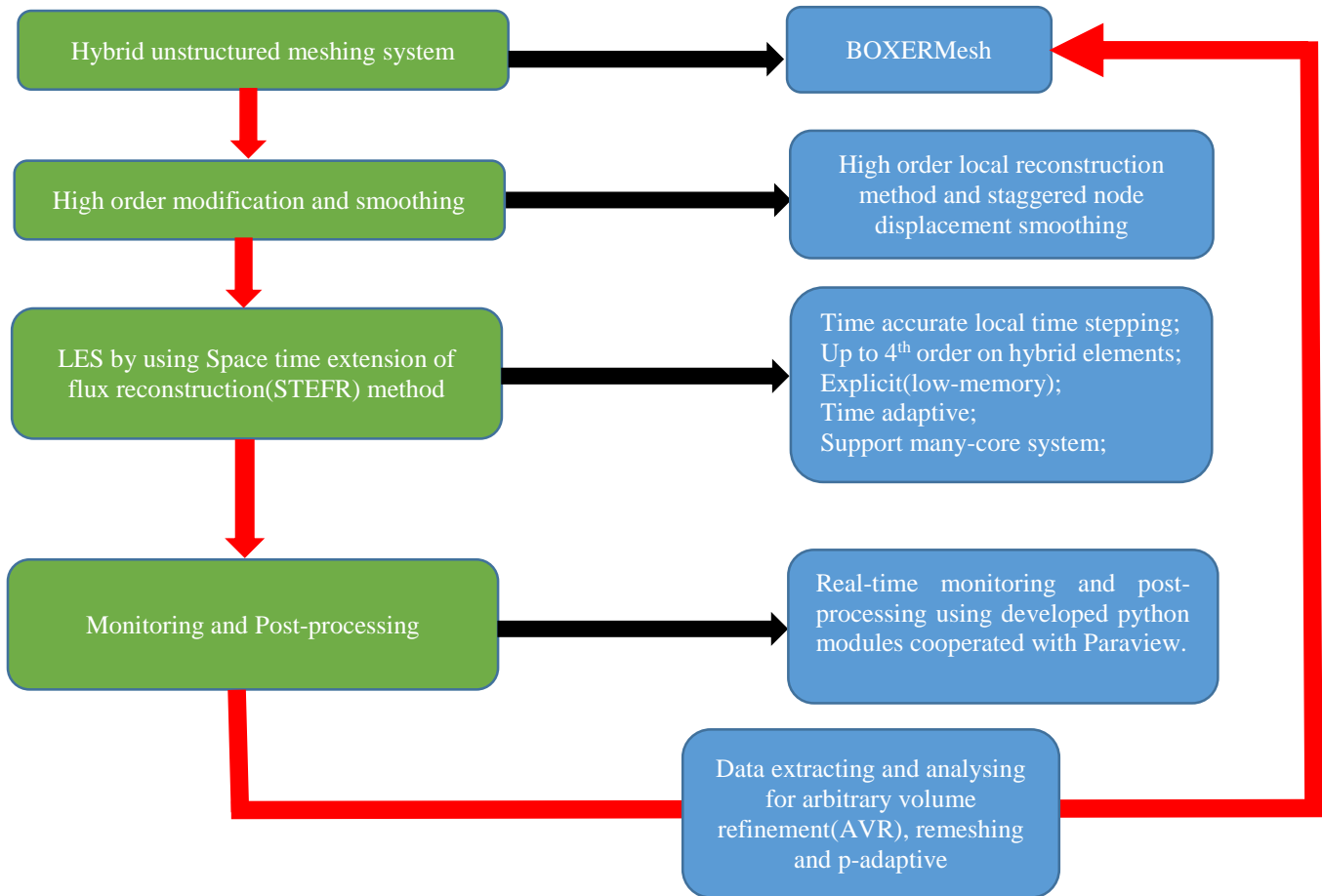

Figure 2. Flowchart of end-to-end parallel high order simulating system

This paper reviews our previous work, including high order mesh generation(modification)[5,6], high orde flux reconstruction for hybrid elements[2,5,6], space-time extention of FR method(STEFR) for time-accurate local time-stepping[2,7], and introduces some recent research for high order mesh smoothing, conservative STEFR form and in-cell piecewise solution integration method for shock capturing[7], STEFR method using many-core system and adaptive wall-modelling, real-time post-processing and monitoring, and trying to combine all these works with adavanced meshing software, as an End-to-End parallel high order simulation system to provide high fidelity, high accuracy and high efficiency LES for large scale industrial problems with complex geometry.
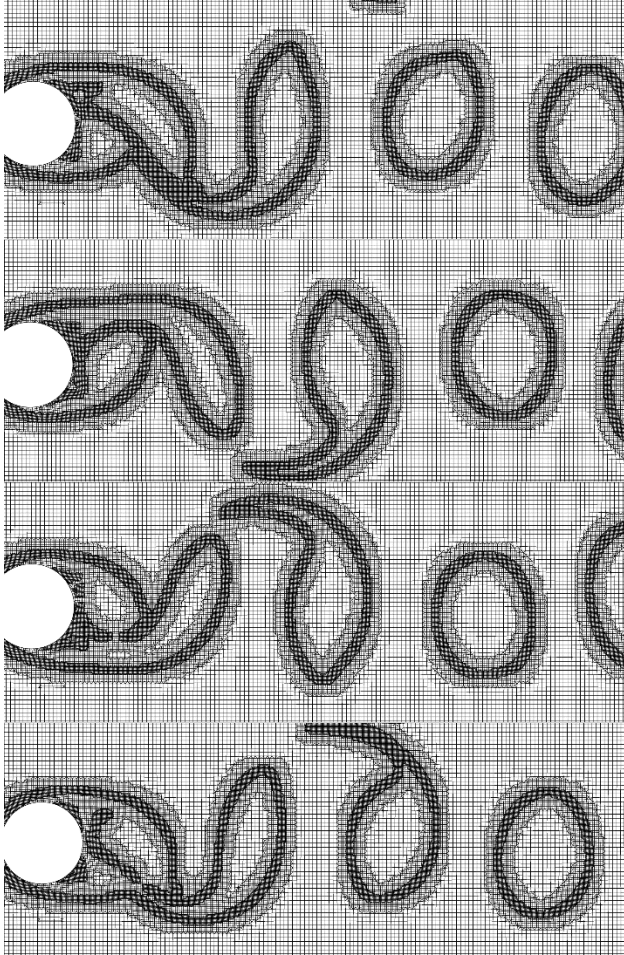
American Institute of Aeronautics and Astronautics

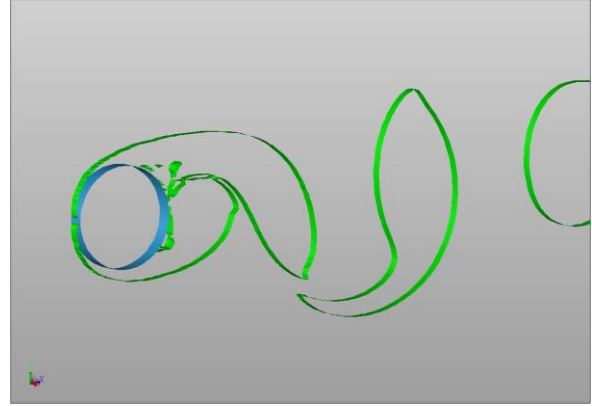**Figure 3. Unsteady solution-based AVR remeshing using BOXERMesh**

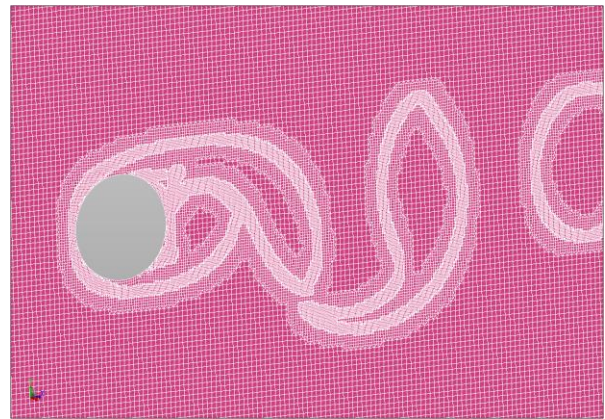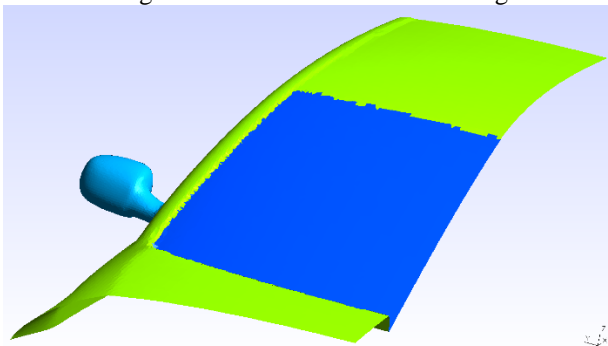**Figure 4. ISO-surface extracted from transient solution and used for AVR refinement in BOXERMesh**

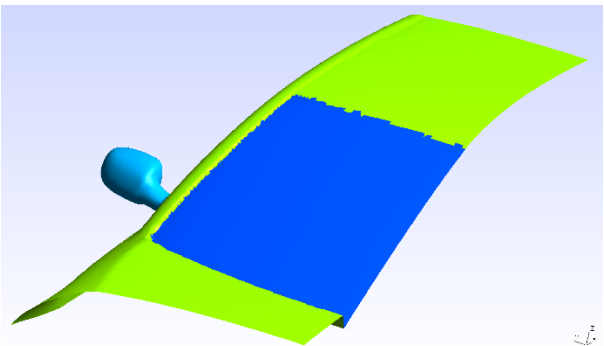**Figure 5. Transient AVR refinement mesh using BOXERMesh**

III. High order curved mesh generation and smoothing

III.A. High order mesh generation(modification)

The high order mesh generation is based on the first order meshes(piece-wise linear) generated by BOXERMesh, and the high order geometry desctiption is constructed by the local high order reconstruction of first order mesh nodes, details can be found in paper 5 and 6. High order nodes on surfaces can be interpolated and projected onto these local high order description, and related into volume meshes. Fig.6 presents the comparisons between first order original mesh and its high order modification for the wing mirror acoustic LES with details in section VII.B.

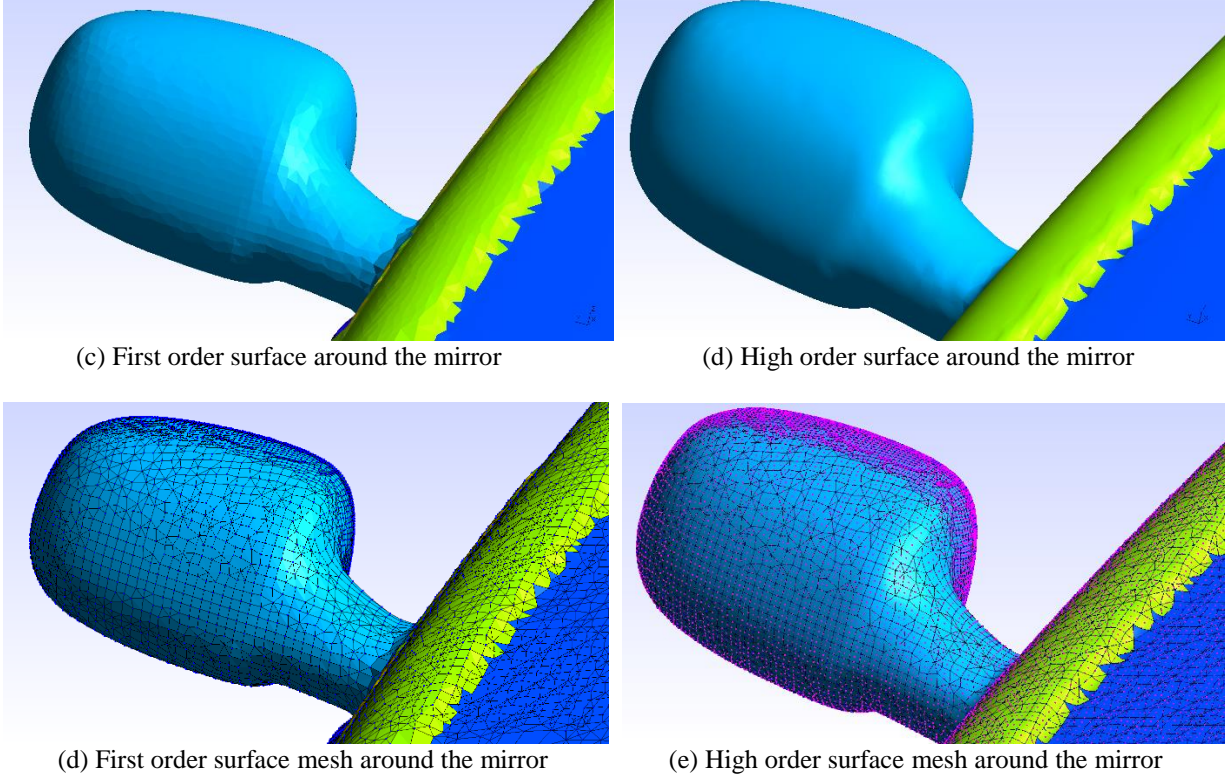(a) First order, piecewise linear surface

(b) High order surface

American Institute of Aeronautics and Astronautics

(c) First order surface around the mirror

(d) High order surface around the mirror

(d) First order surface mesh around the mirror

(e) High order surface mesh around the mirror

Figure 6. High mesh representations and compared with first order mesh for the mirror acoustic case[4], rendered by Gmsh[8]

III.B. High order mesh smoothing

Due to the local reconstructon inside of each cell, the cell quality criterions of high order schemes such as flux reconstruction method, Discontinuous Galerkin method, are quite different from the traditional finite volume method(FVM) and finite difference method(FDM). Three criterias are adopted in our higher orders meshing system as

1. Jacobian:
   The Jacobian matrix J corresponding to the transformation takes the following form
   $$J = \frac{\partial(x,y,z)}{\partial(\xi,\eta,\zeta)} = \begin{bmatrix} x_\xi & x_\eta & x_\zeta \\ y_\xi & y_\eta & y_\zeta \\ z_\xi & z_\eta & z_\zeta \end{bmatrix} \tag{1}$$
   Where (x,y,z) is physical coorninate and (ξ,η,ζ) is local coordinate. And the related quality is defined as
   $$Q_J(E_i) = \frac{|J|_{min}(\vec{r}_i)}{|J|_{max}(\vec{r}_i)} \tag{2}$$

   Where $|J|_{min}$ and $|J|_{max}$ are the minimum and maximum determinate of jacobian matrix over all geometry nodes $\vec{r}_i$ of element E_i. In particular, $\vec{r}_i$ includes all vomule integral points.

2. Squish
   For all tetrahedrons with straight edges, the quality Q_J keep the maximum value 1 becaue $|J|$ is uniform, however, there are still quality requirements for this kind of elements, for which the quality named Squish for each side of element is defined as the following form
   $$Q_S = 1 - 2\frac{acos\left(\frac{(\vec{P}_S - \vec{P}_C)}{||\vec{P}_S - \vec{P}_C||} \cdot \vec{n}\right)}{\pi} \tag{3}$$
   where $\vec{P}_S$ is the centroid point of face S and $\vec{P}_C$ is the centroid of the element. Q_S is related to the robustness of corrention part of the STEFR which in introduced in detail in section IV, versus the element quality Q_J related to the divergent part.

6

American Institute of Aeronautics and Astronautics

3. Orthogonality
   Another constraint for the correction part is the orthogonality which is introduced as

$$Q_O = 1 - 2\frac{acos\left(\frac{(\vec{P}_C - \vec{P}_{S,n}) \cdot \vec{n}}{||\vec{P}_C - \vec{P}_{S,n}||}\right)}{\pi} \tag{4}$$

   Where $\vec{P}_C$ is the centroid of the element and $\vec{P}_{S,n}$ is the centroid of the neighboring element shared side S.

The aim of high order mesh smoothing is to improve cell qualities through moving geometry nodes under control. There are four different node movements modes are adopted in this work as

1. Laplacian: assume all adjacent nodes construct edges of the mesh with arbitrary geometry node $\vec{P}_j$ are $\vec{P}_k, k \in [3, N_k]$, the new trial position of $\vec{P}_j$ is

$$\vec{P}_j' = \frac{\sum_{k=1}^{N_k} ||\vec{P}_j - \vec{P}_k|| \vec{P}_k}{\sum_{k=1}^{N_k} ||\vec{P}_j - \vec{P}_k||} \tag{5}$$

   This method is to average the distance between node $\vec{P}_j$ and all adjacent points $\vec{P}_k$.

2. Volume gradient

$$\vec{P}_j' = \vec{P}_j + L\left(\frac{\partial(Vol)_{min}}{\partial x}, \frac{\partial(Vol)_{min}}{\partial y}, \frac{\partial(Vol)_{min}}{\partial z}\right) \tag{6}$$

   Where L is the length scale with the element size, $(Vol)_{min}$ is the smallest volume of elements shared node $\vec{P}_j$.

3. Parallel quadrilaterial
   For each quadrilaterial face, assume four vertices are $\vec{P}_1, \vec{P}_2, \vec{P}_3$ and $\vec{P}_4$, each point $\vec{P}_j$ can be projected to the plane constructed by other three vertices $\boxplus (\vec{P}_k), k \neq j, k \in [1,4]$ and the new position is $\vec{P}_j'$. Only the point with smallest movement is chosen to adjust the node position as

$$\left|\left|\vec{P}_j' - \vec{P}_j\right|\right| = \left|\left|\vec{P}_k' - \vec{P}_k\right|\right|_{min}, \quad k \in [1,4] \tag{7}$$

   This mode is aim to improve the coplanarity of quadrilaterial faces, which influence Q_J and the robustness especially for pyramids.

4. Squish gradient

$$\vec{P}_j' = \vec{P}_j + L\left(\frac{\partial(Q_S)_{min}}{\partial x}, \frac{\partial(Q_S)_{min}}{\partial y}, \frac{\partial(Q_S)_{min}}{\partial z}\right) \tag{8}$$

   Where L is the length scale with the element size, $(Q_S)_{min}$ is the smallest Squish quality value of all elements shared point $\vec{P}_j$.

Once the trial point $\vec{P}_j'$ is confirmed, the line search process is performed to find the optimal position $\left(\vec{P}_j'\right)_{opt}$ between $\vec{P}_j$ and $\vec{P}_j'$, which is supposed to guarantee the monotone quality improvement in some global sense. These above four modes are performed in a staggerred way to achieve better efficiency, because each mode has its fatigue limitation for this nonlinear smoothing, and the "multigrid-like" procedure is efficient to absorb structure stress generated by different modes.

Tab.1 gives the comparison before and after high order smoothing for the wing mirror acoustic simulation, which is introduced in section VII.B, the geometry is a real-size, real-geometry type automotive model from drivaer[4]. It can be found the minimum Q_J is larger than 0.25, which is the greater than experienced threshold for third order simulations and the main optimimal direction, and both Q_S and Q_O are controlled to increase monotonically.

Table 1. High order mesh smoothing for the wing mirror acoustic case

Total $N_{cells=3258812}$	Q_J before smoothing	Q_J after smoothing	Q_S before smoothing	Q_S after smoothing	Q_O before smoothing	Q_O after smoothing
Minumum quality	0.11235	0.25023	0.04613	0.04613	0.12315	0.16850
Quality range	N_cells, ratio%	N_cells, ratio%	N_cells, ratio%	N_cells, ratio%	N_cells, ratio%	N_cells, ratio%

American Institute of Aeronautics and Astronautics

0~0.1	0, 0.0000000%	0, 0.0000000%	96, 0.0029459%	27, 0.0007870%	0, 0.0000000%	0, 0.0000000%
0.1~0.2	17, 0.0005217%	0, 0.0001457%	1234, 0.0378666%	392, 0.0114265%	45, 0.0013809%	6, 0.0001841%
0.2~0.3	167, 0.0051246%	2, 0.0000603%	6116, 0.1876758%	2399, 0.0699291%	801, 0.0245795%	411, 0.0126120%
0.3~0.4	1047, 0.0321283%	92, 0.0027716%	19109, 0.5863793%	25969, 0.7569773%	3553, 0.1090275%	2939, 0.0901862%
0.4~0.5	4205, 0.1290348%	28348, 0.8540244%	39774, 1.2205061%	59591, 1.7370340%	11543, 0.3542088%	11901, 0.3651944%
0.5~0.6	12285, 0.3769779%	35257, 1.0621680%	82509, 2.5318736%	97316, 2.8366901%	27959, 0.8579507%	28237, 0.8664814%
0.6~0.7	34440, 1.0568268%	47474, 1.4302228%	123939, 3.8031958%	181753, 5.2979667%	104779, 3.2152514%	103213, 3.1671971%
0.7~0.8	141835, 4.3523529%	151452, 4.5627102%	165424, 5.0762057%	128534, 3.7466719%	174650, 5.3593150%	175246, 5.3776039%
0.8~0.9	47647, 1.4620972%	52412, 1.5789872%	178923, 5.4904364%	244826, 7.1364984%	185195, 5.6828992%	185529, 5.6931483%
0.9~1.0	3017169, 92.5849359%	3004306, 90.5090556%	2641688, 81.0629150%	2689811, 78.4060190%	2750287, 84.3953870%	2751330, 84.4273926%

In the near future, the high order curved mesh generation and smoothing will be integrated into our current meshing system rather than applied as external modifications, with high order geometry discriptions taken from CAD directly, which will significantly improve the geometry fidelity and reduce the cell count, both of which are important key issues for industrial class high order large eddy simulations.

IV. Basic Numerical scheme

IV.A. Flux Reconstruction Discretisation

The flux reconstruction (FR) approach, proposed by Huynh[9] for 1D conservative laws, is able to unify the mathematics in several high-order methods, including nodal DG methods and the SD methods for at least linear problems. Wang and Gao[10] extended the idea to simplex meshes by using the "lifting operation" on original DG formulation for Euler equations and further for Navier-Stokes equations[11], and these works are summarized as the Correction Procedure via Reconstruction (CPR) method by Haga, Gao and Wang[12]. An infinite range of high-order energy stable flux reconstruction schemes were developed by Vincent, Castonguay and Jameson[13].The FR approach is simple, flexible and very efficient for its differential form without any numerical integration, and it has proved its higher efficiency than other high order schemes[11,14].

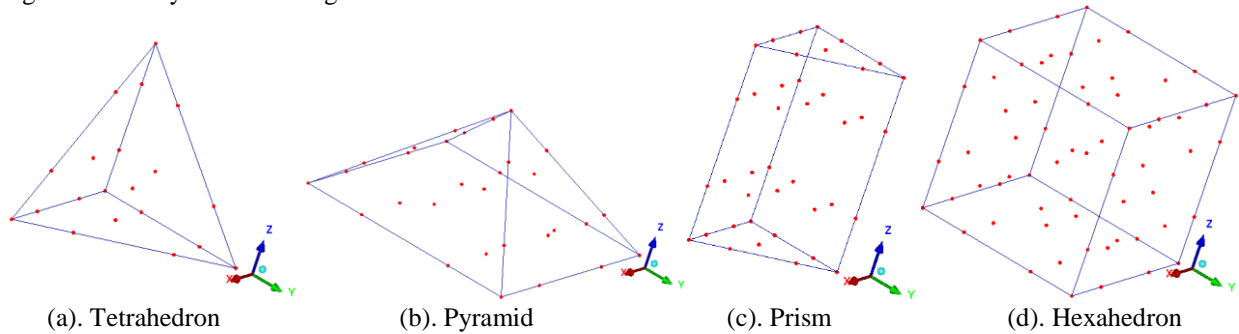(a). Tetrahedron   (b). Pyramid   (c). Prism   (d). Hexahedron
Figure 7. Solution point locations for 3D Elements. K=3

In order to deal with general unstructured meshes including high order curved cells, all elements are transformed from the physical domain (x, y, z) to computational(local) domain (ξ, η, ζ). Following the coordinates transformation, define

$$\hat{U} = |J|U$$
$$F^{\xi} = |J|(\xi_x F_x + \xi_y F_y + \xi_z F_z)$$

8

American Institute of Aeronautics and Astronautics

$$F^\eta = |J|(\eta_x F_x + \eta_y F_y + \eta_z F_z) \tag{1}$$

$$F^\zeta = |J|(\zeta_x F_x + \zeta_x F_x + \zeta_x F_x) \tag{2}$$

The governing equation of Navier-Stokes equation in computational(local) domain becomes

$$\frac{\partial \tilde{U}}{\partial t} + \nabla^\xi \cdot \vec{F}^\xi = \frac{\partial \tilde{U}}{\partial t} + \frac{\partial F^\xi}{\partial \xi} + \frac{\partial F^\eta}{\partial \eta} + \frac{\partial F^\zeta}{\partial \zeta} = 0 \tag{3}$$

Fig.7 gives the solution point locations($K = 3$ for different kinds of 3D elements including the tetrahedrons, pyramids, prisms and hexahedrons which make up a general, hybrid unstructured meshes. For the j −th solution points of i −th element in a non-overlapped mesh, the uniform FR discretisation for different types of elements are given as

$$\frac{\partial \tilde{U}_{i,j}}{\partial t} + \left(\nabla^\xi \cdot \vec{F}^\xi(U_i) \right)_{i,j} + \sum_{s=1}^{N_s} \sum_{m=1}^{K_s} \boldsymbol{\alpha}_{j,s,m} (\tilde{F}^\xi|_n - \bar{F}^\xi|_n)_{i,s,m} = 0 \tag{4}$$

where N_s is number of sides for the element and K_s equals number of flux points on the side, $\tilde{F}^\xi$, $\tilde{F}^\eta$ and $\tilde{F}^\zeta$ denote the common flux which take the form of Riemann fluxes for the inviscid flux and central averaged values for viscous part. Particularly, the difference between common flux and the outer normal projection of local flux $\tilde{F}^\xi|_n - \bar{F}^\xi|_n$, is called "correction flux" as the same as 1D FR formulation, which is used to update the DOFs by exchanging informations with adjoint elements. The FR coefficients $\boldsymbol{\alpha}$ can be obtained through "lift operation"[7] for the standard element, and the detail operations can be found in[11] and[12]. The Chain-Rule method is adopted to deal with the flux divergence because of its consistent order of accuracy and excellent performance to damp aliasing driven instabilities, which is given as

$$\left(\nabla^\xi \cdot \vec{F}^\xi(U_i) \right)_{i,j} = \frac{1}{|J|_{i,j}} \left(\frac{\partial \bar{F}^x_{i,j}}{\partial x} + \frac{\partial \bar{F}^y_{i,j}}{\partial y} + \frac{\partial \bar{F}^z_{i,j}}{\partial z} \right)$$

$$= \frac{1}{|J|_{i,j}} \left(\left(\frac{\partial \bar{F}^x}{\partial U} \right)_{i,j} \left(\frac{\partial U}{\partial x} \right)_{i,j} + \left(\frac{\partial \bar{F}^y}{\partial U} \right)_{i,j} \left(\frac{\partial U}{\partial y} \right)_{i,j} + \left(\frac{\partial \bar{F}^z}{\partial U} \right)_{i,j} \left(\frac{\partial U}{\partial z} \right)_{i,j} \right) \tag{5}$$

From Eq.4 it can be found that the FR discretisations consist of two parts: the flux divergence which takes the main part and it is completely local; and the linear combination of correction flux which is ought to be minimized. Detail discriptions can be found in[2].

IV.B. Space time extension

For high order method, the explicit multi-stage Runge-Kutta method is most widely used to solve unsteady problems[16], which is time-accurate, simple, low memory and efficient for massive computations. Another alternative method is the implicit Runge-Kutta method[10] which can overcome the strict CFL condition of the explicit method by using implicit time integration. Besides, the non-linear implicit time-marching method based on interative block lower-upper symmetry gauss-seidel(BLU-SGS) method are also adopted to perform LES[17,18]. However, all these methods use a global uniform time-step which is commonly defined by the smallest cell size on the whole mesh, which is very inefficient for large-scale simulations especially on multi-scale geometries. Whereas a time-accurate local time-stepping method is very attractive to reduce the total number of function evaluations by using the local CFL number for each element. For time-dependent ODE(ordinary differential equations), the so-called "space-time" type numerical method has great potential to achieve high order time accuracy and provide more flexibility by constructing polynomials of solution along time direction. So far, there are two types of space time extension for high order methods: one is the global space-time DG method developed by van der vegt and van der Ven[19] for inviscid flow, and extended for compressible Navier-Stokes equations by Klaij et al[20], which results in a set of global nonlinear equations; another one is based on "predictor-corrector" method, Gassner et al[21] reviewed different types of local predictors for DG and finite volume method(FVM), including Cauchy-Kowalevsky procedure, continuous Galerkin predictor and a local continuous extension Runge-Kutta(CERK) predictor.

In this paper, the "predictor-corrector" space-time extension for high order FR(STEFR) method is introduced, which use the "flux divergence" part to construct the local predictor by using continuous Runge-Kutta method[22], and the "correction flux" part of the FR discretisation is used as corrector. In the original FR formulation, the flux divergence takes the major part and is completely local, whereas the linear combination of correction flux part which takes adjacent elements into account, is used for updating degree of freedoms(DOFs). The STEFR combines these two parts smoothly by the staggered operations which is simple, efficient, accurate for both space discretisation and time-marching and allows local time-stepping.

Rewriting Eq.(4) as the following uniform formula

9

$$\frac{\partial U_{i,j}}{\partial t} = \mathbf{R}_{i,j}^D(U_i) + \sum_{s=1}^{N_s} \mathbf{R}_{i,j,s}^C(U_i, U_{i,s}^{nb}) \tag{6}$$

where $\mathbf{R}^D$ and $\mathbf{R}^C$ are "flux divergence" part and the linear combination of "correction flux" part, respectively. For 1D discretisation, $\mathbf{R}^D$ and $\mathbf{R}^C$ are given as

$$\mathbf{R}_{i,j}^D = -\frac{1}{|J|_{i,j}}\left(\frac{\partial \bar{F}}{\partial \xi}\right)_{i,j}, \qquad \mathbf{R}_{i,j,s}^C = -\frac{1}{|J|_{i,j}}\left[\left((\tilde{F}_{i,s} - \bar{F}_{i,s})g_s'(\xi_j)\right)\right], \ s = L \ or \ R \tag{7}$$

and for multi-dimensional discretisations, $\mathbf{R}^D$ and $\mathbf{R}^C$ are given as

$$\mathbf{R}_{i,j}^D = -\frac{1}{|J|_{i,j}}\left(\nabla^\xi \cdot \vec{F}^\xi(U_i)\right)_{i,j}, \qquad \mathbf{R}_{i,j,s}^C = -\frac{1}{|J|_{i,j}}\sum_{s=1}^{N_s}\sum_{m=1}^{K_s}\alpha_{j,s,m}(\tilde{F}^\xi|_n - \bar{F}^\xi|_n)_{i,s,m} \tag{8}$$

Integrating Eq.(8) over $t \in [t^n, t^{n+1}]$, one obtains

$$U_{i,j}^{n+1} - U_{i,j}^n = \int_{t^n}^{t^{n+1}} \mathbf{R}_{i,j}^D(U_i) + \sum_{s=1}^{N_s} \mathbf{R}_{i,j,s}^C(U_i, U_{i,s}^{nb}) \ dt \tag{9}$$

Inspired by the space-time extension of DG(STEDG)[7], constructing a local space-time approximation $v = v(\vec{x}_i, t)$ for $t \in [t^n, t^{n+1}]$ by solving the following time-dependent ODE

$$\frac{dv_{i,j}}{dt} = \mathbf{R}_{i,j}^D(t, v(\vec{x}_i, t)), \qquad v(\vec{x}_i, t = 0) = U_i^n(\vec{x}_i) \tag{10}$$

From Eq.(7) and Eq.(8), it can be found $\mathbf{R}^D$ is completely local, which indicate the time evolution of $v = v(\vec{x}_i, t)$ is local also. The continuous Runge-Kutta method[11] is adopted to solve Eq.(10) following the precedure as

$$v_{i,j}(\tau) = U_{i,j}^n + \Delta t \sum_{l=1}^{N_t} \dot{B}_l(\tau)\dot{H}_l$$

$$\dot{B}_l(\tau) = \sum_{m=1}^{O_t} b_{l,m}\tau^m$$

$$\dot{H}_l = \mathbf{R}_{i,j}^D(v_i^l)$$

$$v_{i,j}^l = U_{i,j}^n + \Delta t \sum_{n=1}^{l} a_{l,n}\dot{H}_{l-1} \tag{11}$$

Where $\tau \in [0,1]$ is the non-dimensional time, $\Delta t = t^{n+1} - t^n$, O_t is the order of time integration and N_t is the related number of stages, the coefficients $\boldsymbol{a}$ and $\boldsymbol{b}$ are given by Owren and Zennaro[9]. Integrating Eq.(10) for $t \in [t^n, t^{n+1}]$, one obtains
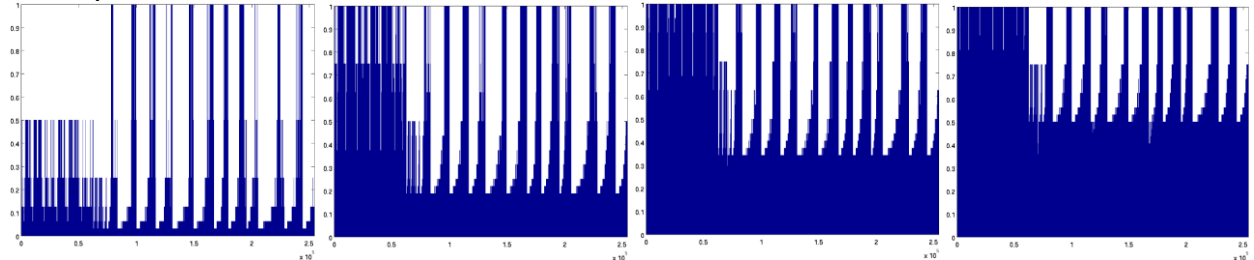
$$\int_{t^n}^{t^{n+1}} \mathbf{R}_{i,j}^D(v_i^l) \ dt = v_{i,j}(\tau = 1) - U_{i,j}^n \tag{12}$$

Taking the space-time polynomial $v = v(\vec{x}_i, t)$ as a local predictor, and the combination of correction flux $\mathbf{R}^C$ as corrector, substituting Eq.(12) into Eq.(9) which results in the space time extension of flux reconstruction(STEFR) scheme as

$$U_{i,j}^{n+1} = v_{i,j}(\tau_{max}) + \Delta t \int_0^1 \sum_{s=1}^{N_s} \mathbf{R}_{i,j,s}^C(v_i(\tau), v_{i,s}^{nb}(\tau^{nb}))d\tau \tag{13}$$

where τ^{nb} indicates the non-dimensional time for adjacent element with the same physical time.

For the STEFR method, all elements are allowed to use their maximum allowed local time-step which is adaptive during the simulations, this irregular time-stepping bring a lot of challenges for the time integration, physical time synchronous especially on parallel computing system. This section focus on application of STEFR and its implementation for parallelization. For the convenience of data exchange on element interfaces for time integration of correction fluxes, the actual time-steps for all elements are set as the times of power of 2 of the global smallest time-step.
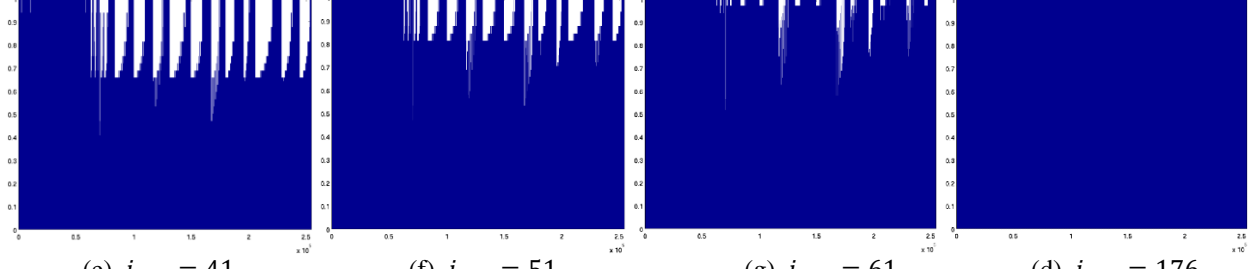

(a). $i_{iter} = 1$ (b). $i_{iter} = 1$ (c). $i_{iter} = 1$ (d). $i_{iter} = 1$

10

(e). $i_{iter} = 41$ (f). $i_{iter} = 51$ (g). $i_{iter} = 61$ (d). $i_{iter} = 176$

Figure 8. Snapshots time marching of one step: horizontal is element index and vertical is normalized prediction time.

Fig.8 presents the snapshots of transient prediction time during a global time synchronous step for a 3D simulation, for which this is totally 176 iterations. Compared to the uniform step time marching methods, this approach need to spend more effort for getting executable queues and asynchronous parallel communications. Therefore, the application of STEFR should be minimize the parallel communications. The heterogeneous computing architecture is adopted in this work, in which the OpenMP is used for parallel looping inside of each shared memory computing unit, and the communications between different computing units are undertaken by MPI. For most simulations on different computing systems, the ratio of wall-clock time for effective functional evolutions takes over 70% of computational resource for most cases, including the correction flux calculations, predictions and corrections.

The efficient implementation of STEFR using local time-stepping is much more complex than other explicit methods using uniform time-stepping, such as explicit Runge-Kutta method, especially for parallel simulations. However, the application presented in this section has proved very efficient for large scale simulations with speed up ratio up to ~ 100 compared with using global uniform time-stepping[2].

IV.C. Aliasing error analysis and conservative correction for Chain-Rule method

As expressed in Eq.(5), the Chain-Rule method is slighlt non-conservative which will not cause actual error for subsonic flow simulations, but it is absolutely unacceptable for transonic flow simulations with shock or other kind of discontinuities. The natural conservative treatment for the flux divergence part is to use "Lagrange Polynomials"(LP) as

$$\left(\nabla^\xi \cdot \vec{\bar{F}}^\xi(U_i)\right)_{i,j} = \sum_{k=1}^{N_{sps}} \left(\bar{F}^x_{i,k} \frac{d\phi_k}{dx}(\xi_j) + \bar{F}^y_{i,k} \frac{d\phi_k}{dx}(\xi_j) + \bar{F}^x_{i,k} \frac{d\phi_k}{dx}(\xi_j)\right) \tag{14}$$

However, the LP method is less accurate than Chain-Rule method[2] and not sufficiently to damp aliasing error in some region[19]. In order to construct a conservative form Chain-Rule method, a source term is introduced[6] as

$$S_i = \frac{\int_{E_i}\left[\left(\nabla^\xi \cdot \vec{\bar{F}}^\xi(U_i)\right)_{i,j}^{LP} - \left(\nabla^\xi \cdot \vec{\bar{F}}^\xi(U_i)\right)_{i,j}^{CR}\right]dE}{V_i} \tag{15}$$

Where V_i is the volume of $i-$th element, $\left(\nabla^\xi \cdot \vec{\bar{F}}^\xi(U_i)\right)_{i,j}^{LP}$ and $\left(\nabla^\xi \cdot \vec{\bar{F}}^\xi(U_i)\right)_{i,j}^{CR}$ are flux divergence part calculated by Eq.(5) and Eq.(14) respectively. The source term S_i is added to each solution point in the prediction part(flux divergence part) as a element constant value without any influence for aliasing error performance.

V.Some functionalities

V.A. In-cell piecewise integrated solution, additional, novel, method for Shock-Capturing

In-cell piecewise integrated solutions are used for shock-capturing in this work, which was originally introduced by Peraire and Huerta[23]. The basic idea is to construct piecewise constant solutions inside of single element, which are conservative and combined with the local high order polynomials at or near the physical discontinuities. For FR method, DOFs are stored at nodal-based solution points which inspire that the subcells could be around each solution point, Fig.7 shows the solution points and their subcell

11

American Institute of Aeronautics and Astronautics

interfaces(solution point itself if at the end of the element), for each solution point, the related subcell is from the left subcell interface to the right subcell interface. The piecewise constant solution is defined as:

$$U'_{i,j} = \frac{1}{V_{i,j}} \int_{E_{i,j}} U_i(\xi)dE, \quad j \in [1, K+1] \tag{16}$$

where $E_{i,j}$ is the subcell related to the $j-$th solution point at $i-$th element, and $V_{i,j}$ is the volume of subcell $E_{i,j}$. The actual approximation can be characterized by

$$\bar{U}_{i,j} = (1-\varepsilon)U_{i,j} + \varepsilon U'_{i,j} \tag{17}$$

where $\varepsilon \in [0,1]$ is a parameter that depends on the smoothness of the solution. The combination Eq.(17) is conservative because the piecewise constant solutions U' come from the piecewise integration of high order solutions U by Eq.(16). It is very suitable to use the above method to handle discontinuities for STEFR with local time-stepping because of its completely local. Other two popular methods for high order shock-capturing, limiter and artificial viscosity, are not sufficient for STEFR because of the necassary information exchange with adjacent elements, for the former is the neighboring conservative variables for constructing limiter, and the artificial viscous fluxes must be included in common fluxes at adjancent interfaces for the latter.
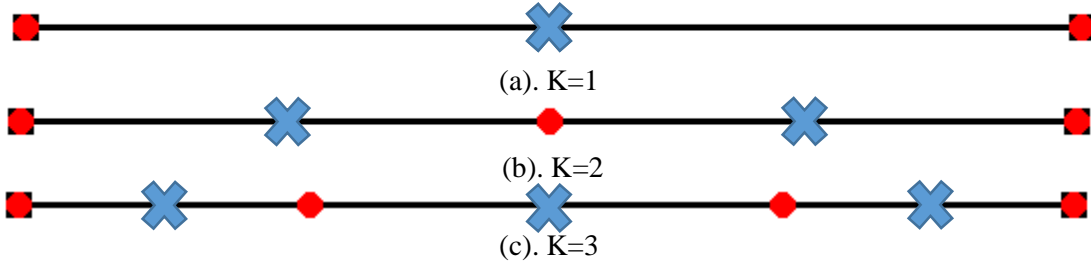

(a). K=1

(b). K=2

(c). K=3

Figure 9. Solution points(circle) and their subcell interfaces(cross)

Element-wise constant parameter ε can vary in space and time depending on the regularity of the high order solutions. In order to construct a smoothness indicator, firstly the nodal solutions can be transformed to be expressed by some hierarchy, orthogonal and modal polynomials, such as normalized Legendre polynomials in 1D, and the related mode is given by

$$U^m_{i,j} = \int_{-1}^{1} \phi^m_j U_i(\xi)d\xi, \quad j \in [1, K+1] \tag{18}$$

where ϕ^m_j is the modal shape function. For $i-$th element, the following smoothness indicator is defined

$$s_\kappa = Log_{10}\left(\frac{\sum_{j=N_{K-1}}^{N_K}(U^m_{i,j})^{\wedge}2}{\sum_{j=1}^{N_K}(U^m_{i,j})^{\wedge}2}\right) \tag{19}$$

For 1D problems, $N_K = K + 1$ is the number of shape functions of order K. Eq.(19) is the proportion of high frequency energy in the total energy, which is very small for smooth solutions. In this work, parameter ε is given as

$$\varepsilon = \begin{cases} 0, & s_\kappa \leq (s_0 - \kappa) \\ \frac{s_\kappa - s_0 + \kappa}{2\kappa}, & (s_0 - \kappa) < s_\kappa < (s_0 + \kappa) \\ 1, & s_\kappa \geq (s_0 + \kappa) \end{cases} \tag{20}$$

where $\kappa = 3$, $s_0 = -7.96 Log_{10}(K)$. For this case, strict conservative must be ensured because of discontinuity, hence, for 1D model cases, the LP method with over-integration is used to calculate the flux divergence part[24].

The subcell piecewise intergrated solution is constructed to combine with high order approximations to handle discontinuity, which can be proved robust, high accurate(subcell shock-capturing) and easy to implement for STEFR method. The extension for multi-dimentional elements is to find proper sub-cell for each solution point as shown in Fig.9, and project the nodal solutions to some hierarchy of orthogonal modal polynomials, later calculate smooth indicator and combine parameter completely the same as 1D

American Institute of Aeronautics and Astronautics

element, and finally integrate the solution on each selected subcell and combine it with high order solution on corresponding solution point to deal with discontinuity.

V.B. Low mach number preconditioning

The HLLC Riemann flux is adopted in this work to compute common flux for invicid part, for which a quite simple low-speed preconditioning scheme was introduce by Luo[24]. It is only need to replace the Roe's average variables $\hat{v}$(velocity along outer normal direction) and $\hat{c}$(speed of sound) by preconditioned velocity $\hat{v}'$ and speed of sound $\hat{c}'$, to calculate the siginal velocities S_i and S_j, details could be found in the paper[24].

V.C. Dynamic wall-model

Even just for a modest Reynolds number flow past a turbine blade(8.5×10^5), the computation cost for boundary layers to achieve wall-resolved LES takes more than 80% for the whole computing time. However the cell count near the wall is only about 20% of the whole mesh, which indicate that for high Reynolds number flow, most computational resource are cost to resolve neall wall regions because of their small size and large aspect ratios even when using this local time stepping high order time-accurate solver, section III.B later has an example. Although local timestepping can reduce the computational cost dramatically(up to ~100 times[2] faster than conventional global uniform time-stepping method), the wall-resolved LES for large scale industridual problems(whole air plane, F1 car, …) are still too expensive using current normal HPC resource. Advanced dynamic wall modelling is a realistic alternative, which could speed up the simulation more than 10 times faster without resolving near wall regions. In the present work, an improved dynamic non-equilibrium wall-model, which is presented by Park and Moin[25] recently, is adopted in this work, all details could be found in the paper.

VI. Many-core system implementation

From the detail description of STEFR method in section IV, the time marching process is very irregular compare to conventional global uniform time stepping, and local time step for each element is adaptive for every single global synchronous time step, in particular, asynchronous MPI is adopted for communications between different computing processes. All these factors makes the STEFR method very hard and inefficient to be implemented on GPU system like NVidia CUDA and AMD GPU cards, which require the loops in the calculations as uniform as possible. However, the asynchronous MPI + OpenMP inside each shared memory process for the STEFR method can be compiled directly and run normally on the intel PHI many-core cards. An workstation was built with 2 ten-core Xeon CPUs with 256GB memory and 8 Xeon PHI coprocessor 5110P cards, each card has 8GB memory, 61 physical cores with 1.053GHz frequency. The off-load mode code was written with asynchronous communication between PHI card to host CPU and PHI card to PHI card, which is viewed as a single "shared memory many-core unit", and it could be easily extended to PHI cluster by using asynchronous MPI for communication between different "shared momory units". The computational ability of this workstation is equivalent to around 140~180 CPU cores of Intel Sandy-bridge HPC cluster. Since the computational cost for each element is scale to its geometry size(local timestep), the flexible partitions of meshes for this PHI workstation can avoid memory limit on many core system, by putting more elements with larger gemometry size on CPU and elements with smaller geometry size(especially inside of boundary layers) on PHI card, to match the memory with computational ability. The data communications between different computing units including CPUs and MIC(PHI) cards are presented in Fig. 10, and this construction can be easily extended to different kinds of heterogeneous computing systems for the STEFR method.
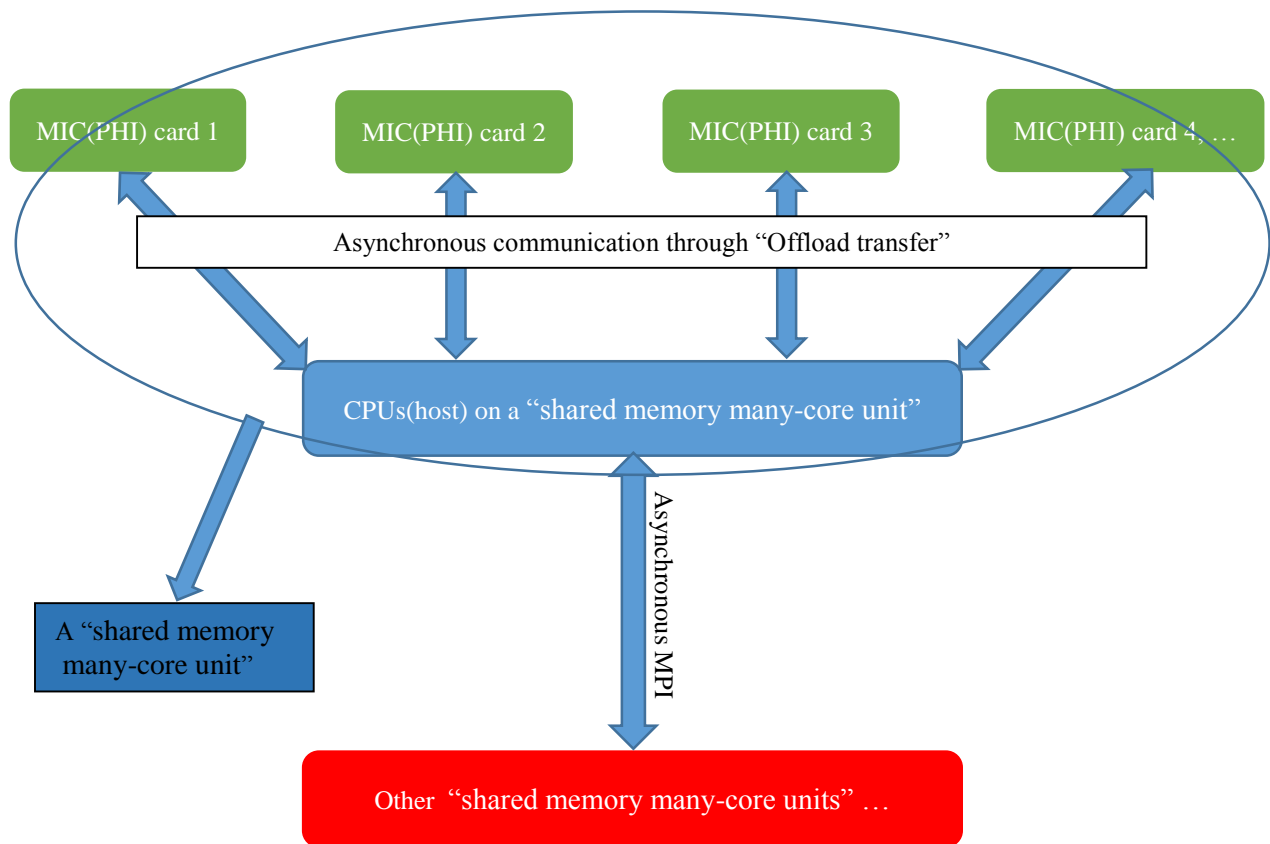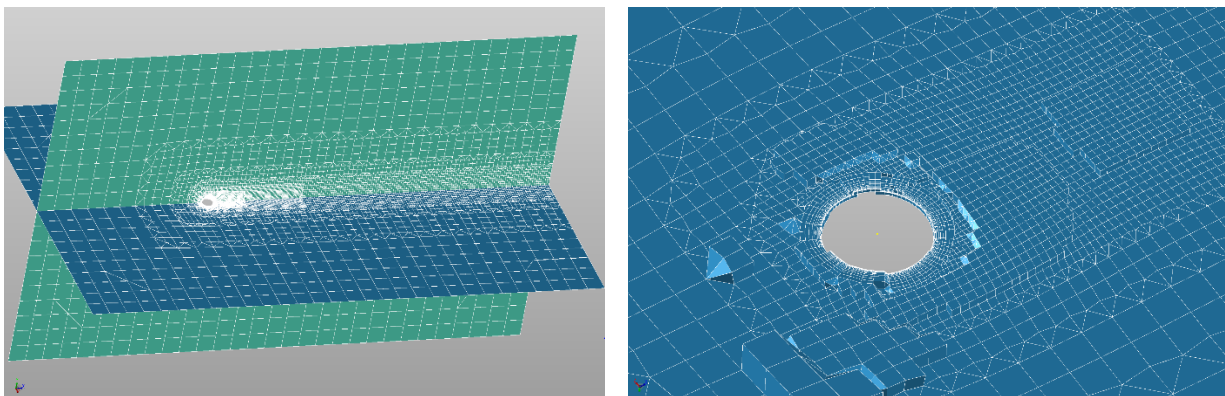
American Institute of Aeronautics and Astronautics

Figure 10. Communications between different computing units including CPUs and MIC cards.
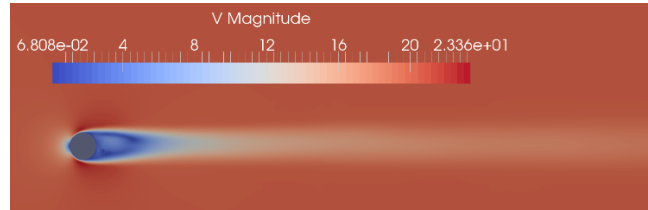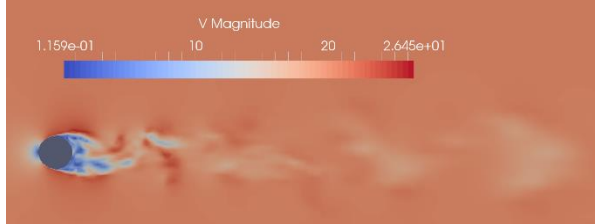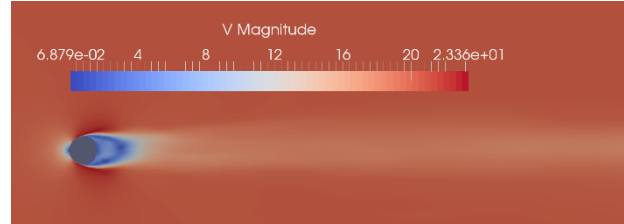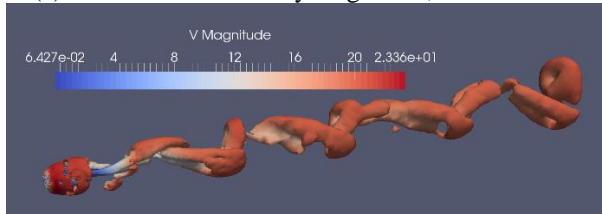
VII. Test cases

VII.A. Flow passing a sphere with $Re = 300$ and $Re = 10000$

The low Reynolds number(Re=300) flow and high Reynolds number(Re=10000) flow around sphere are simulated using hybrid unstructured meshes as Fig.11 shown. The slice of velocity magnititude and iso-surface of Q criterion for different simulations are given in Fig.12 which indicate that higher order simulations have better resolution for vortex region. The drag coefficient C_d and Strouhal number S_t are listed in Tab. 2 with comparisons with other calculations and experiments[28~32]. It can be found the 4th order simulations(K=3) for both two different Reynolds numbers are better than lower order simulations with less degree of freedoms.

American Institute of Aeronautics and Astronautics

(a). Computational domain(Re=300, K=1)   (b). Mesh slice(Re=10000, K=3)
Figure 11. Computational domain and mesh slice for flow passing sphere case

(a) Instantaneous velocity magnitude, $Re = 300$ (b)Average velocity magnitude, $Re = 300$

(c) Instantaneous velocity magnitude, $Re = 10000$ (d)Average velocity magnitude, $Re = 10000$

(a) Instantaneous Q-criterion, $Q = 5 \times 10^6, Re = 300$ (b)Average Q-criterion, $Q = 5 \times 10^6$, $Re = 300$
Figure 12. Results for flow passing sphere case, $K = 3$

Table 2. Results of Low-Re/High-Re flow passing a sphere

Reynolds number	Order of accuracy	N_{cells}	N_{dofs}	C_d	c_d (calc/expt)[28-32]	S_t	S_t (calc/expt) [28-32]
300	SECOND ($K = 1$)	419250	13915335	0.676	0.656 0.657 0.658 0.671	0.130	0.134 0.136 0.136 0.137
300	THIRD($K = 2$)	155286	15172365	0.664		0.134	
300	FOURTH ($K = 3$)	52447	10728620	0.6585		0.136	
10000	SECOND ($K = 1$)	3783863	135266050	0.448	0.393 0.438 0.40±0.01		$0.195s_{t1}$ $0.181S_{t1}$ $1.30{\sim}1.85S_{t2}$
10000	THIRD($K = 2$)	735883	79212055	0.4387		$0.1879S_{t1}$ $1.88S_{t2}$	
10000	FOURTH ($K = 3$)	164013	37728290	0.429		$0.188S_{t1}$ $1.75S_{t2}$	

Particularly, for this case, all simulations are performed with real time monitoring(drag coefficient) as shown in Fig.13 by using the developed python modual and executed in Paraview python shell[3].
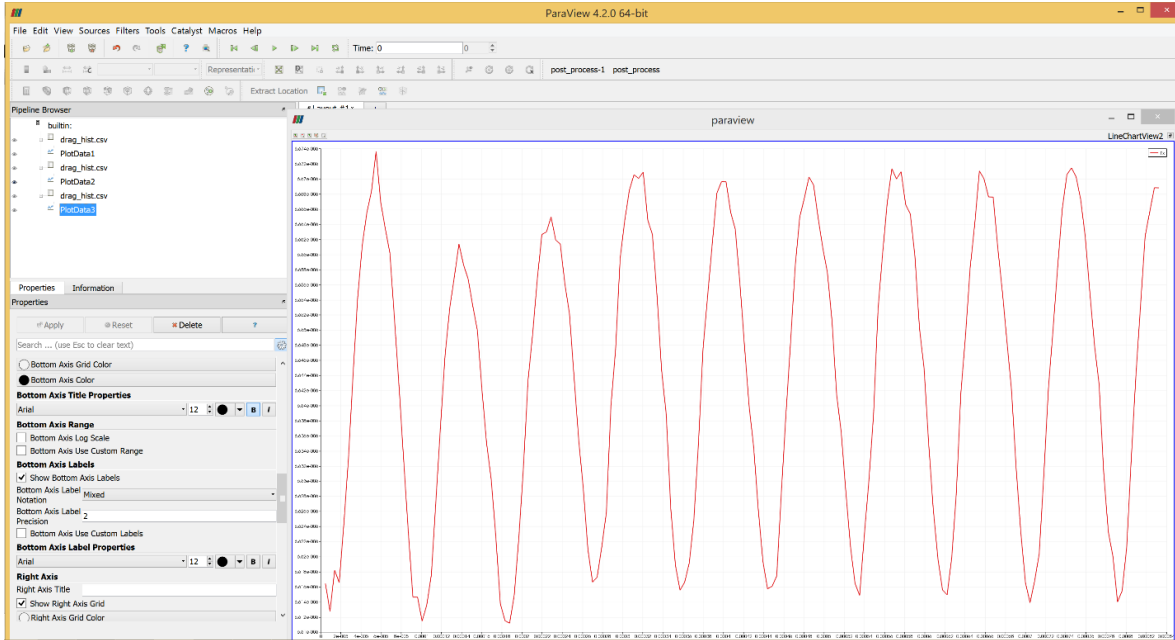
American Institute of Aeronautics and Astronautics

Figure 13. Real time monitoring of the variation for surface integration of drag force, for $Re = 300, K = 2$

VII.C. Acoustic simulation for a wing mirror of a real size automotive

A real size, real geometry type automotive model[4], is used for a demonstration wing mirror acoustic simulation, with computational domain and mesh slice shown in Fig.14. Due to the large size and very high Reynolds number, wall-resolved LES is very expensive, therefore, wall model is adopted to solve near wall region, and third order simulations are performed with very limited computational resource(32 CPUs × 8 cores each, or 2 CPUs with 8 intel-phi cards). In total, 3,258,812 cells are used with arbitrary volume refinement(AVR) at the downstream of the mirror and close to the window for accurate simulation of pressure fluctuations in this region. The totol degree of freedoms is 411,437,225, around 180 hours are required for 1 flow passing time(flow pass the whole computational domain) by using 32 CPUs × 8 cores each. The speed up ratio is aound 26 compared to global uniform time-stepping, by using the STEFR solver with time-accurate local timestepping. Compared to the wall-resolved transonic turbine blade case presented in section VII.B, the computational resource is mostly in volume refinement meshes(level 4 in Tab.3) in the mainstream rather than boundary layers due to the wall-modelling. The iso-surface of Q-criterion($Q = 80000$) is presented in Fig.15.
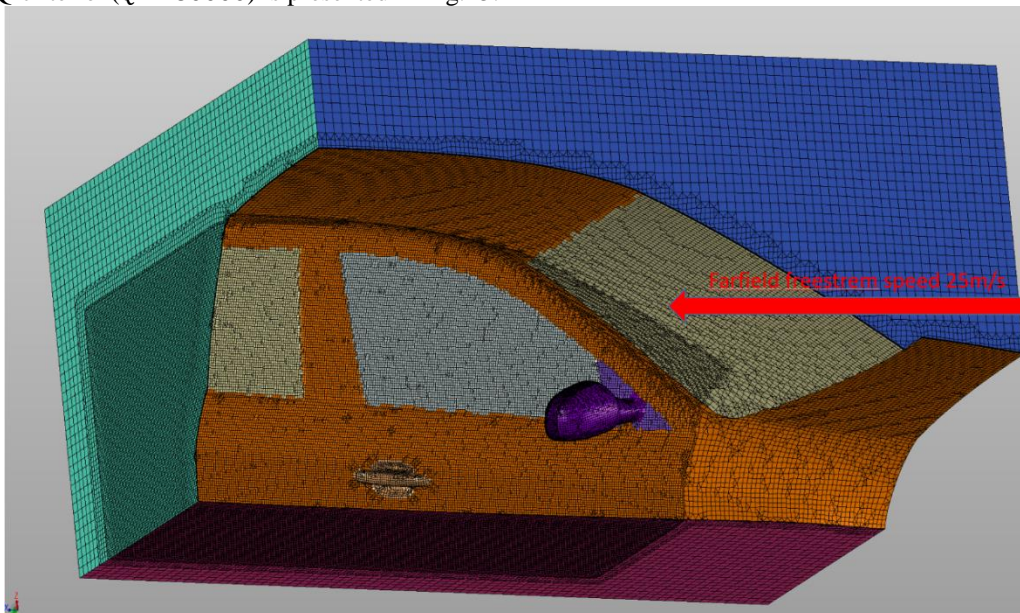

16

American Institute of Aeronautics and Astronautics

Figure 14. Mesh slice for wing mirror acoustic LES with wall modelling
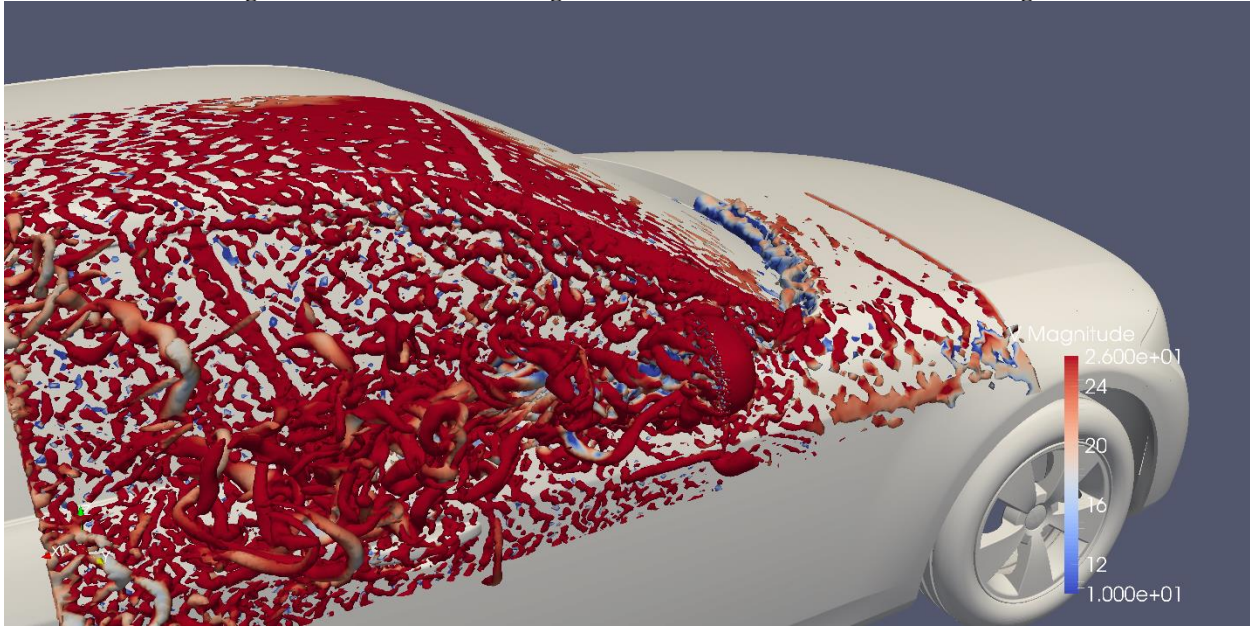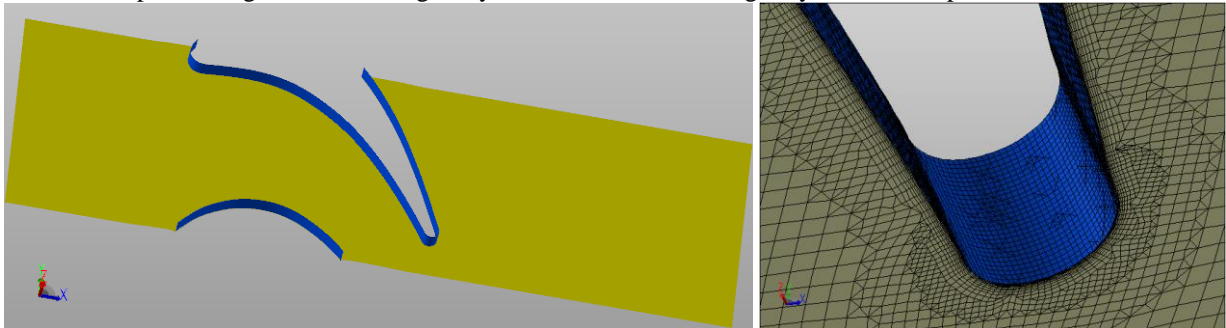

Figure 15. Instantaneous Q-criterion($Q = 80000$) iso-surface colored by velocity magtinitude

Table 3. Statistics of time level for wing-mirror acoustic case, $\Delta t_{min} = 3.6 \times 10^{-8}s, \Delta t_{max} = 6.9 \times 10^{-6}s$, Speed up ratio ≈ 26.9105

Level	0	1	2	3	4	5	6
$\dfrac{\Delta t}{\Delta t_{max}}$	$\dfrac{1}{128}$	$\dfrac{1}{64}$	$\dfrac{1}{32}$	$\dfrac{1}{16}$	$\dfrac{1}{8}$	$\dfrac{1}{4}$	$\dfrac{1}{2}$
N_{cells}	165	29993	133676	76394	2799619	171759	107737
Computing cost ratio	0.046475%	5.967045%	11.69318%	2.7784%	77.0633%	1.6902%	0.7614%

VII.C. Transonic turbine blade case

The computational domain for the well-known VKI-LS59 blade is presented in Fig16. The exit Reynolds number is 8.5×10^{5} with mach number 0.95. In the LES, the length of span is set to about 4.2% of the chord length. This is judged to be enough to capture the 3D structure scales, The hybrid unstructured mesh as shown in Fig.16 is generated by BoXeR[1] with total cells number 440061, where $y_+ \leq 1.5$ is enforced for the first boundary layer cells for wall-resolved large eddy simulations using the STEFR high order solver.. The flow passing time $T_c \approx 2 \times 10^{-4}s$; all results for statistics and anlysis were taken from the simulation with flow physical time $> 10T_c$. The simulations are performed on HPC cluster "Darwin" in University of Cambridge, using 16 nodes with 16cores on each node, and each flow passing time need around 16 hours, which demonstrated this STEFR solver is very efficient for performing LES on this high Reynolds number flow using very limited computational resource.

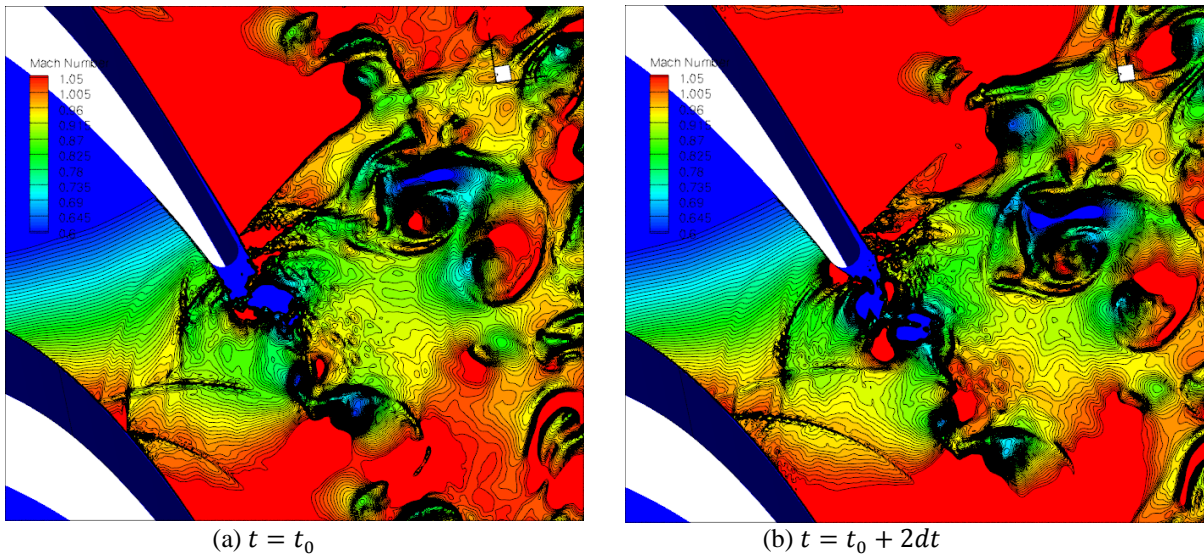(a) Computational domain   (b) slice of mesh around trailing edge

Figure16. Computational domain and mesh slice

17

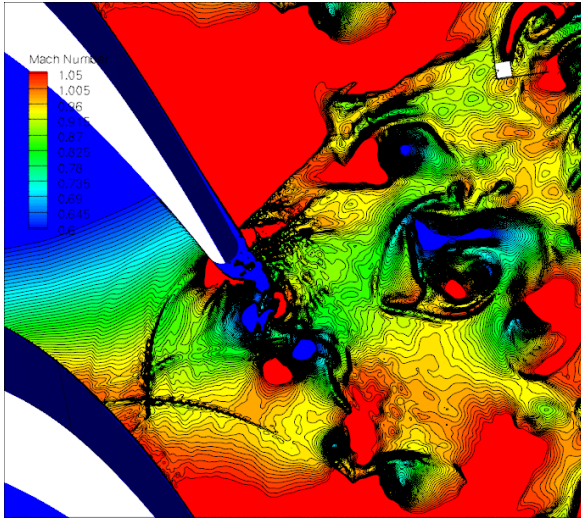American Institute of Aeronautics and Astronautics

The number of time levels are $8(\frac{\Delta t_{max}}{\Delta t_{min}} = 2^8)$. The rough statistics of time levels is listed in Tab.4 which is adaptive for every synchronous time step. It can be seen that the cell number for level $7(\Delta t_{max})$ is about 5 times more than level $0(\Delta t_{min})$, however, the computational cost for level $7(\Delta t_{max})$ is only 6.13% of the cost for level $0(\Delta t_{min})$, which means the local time-stepping "space-time" approach can eliminate many unnecessary functional evolutions for unsteady simulation and reduce the total computational cost significantly. The speed up ratio for this case is around 8.07 by using local time-stepping compared to traditional global uniform time-stepping. This is a quite modest value because of the simple geometry with limited scale range.
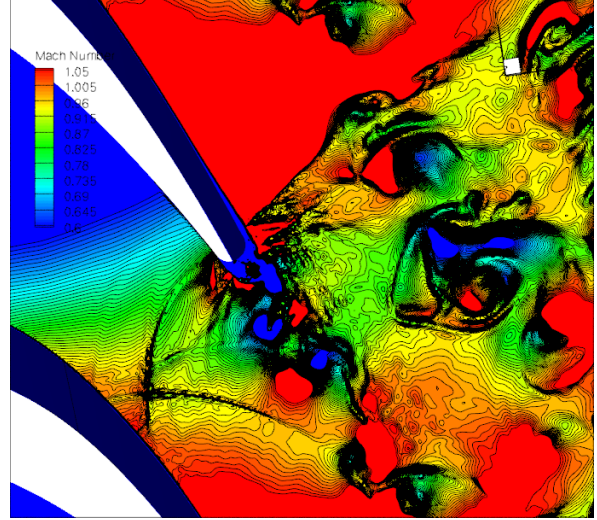
Table 4. Statistics of time level for VKILS-59 case

Level	0	1	2	3	4	5	6	7
$\dfrac{\Delta t}{\Delta t_{max}}$	$\dfrac{1}{128}$	$\dfrac{1}{64}$	$\dfrac{1}{32}$	$\dfrac{1}{16}$	$\dfrac{1}{8}$	$\dfrac{1}{4}$	$\dfrac{1}{2}$	1
N_{cells}	35418	35418	23640	37308	37551	36361	54320	180045
Computing cost ratio	47.6%	25.3%	9.83%	7.96%	3.65%	1.57%	1.22%	2.92%

A sequence of instantaneous contours of Mach number for the VKI-LS59 balde is shown in Fig.17. It can be clearly seen that a series of shocks within the throat part of the blade interact strongly with the trailing edge vortex shedding. The shock structures move both with and against the main flow direction. The reflected shock at the suction side boundary layer interaction is nearly perpendicular the impinging shock, and the shock-boundary layer interaction zone is in motion as well. These phenomenon can be clearly seen also in the experimental schlieren picture in Fig.18 and Fig.19 [22,23]. What is striking the crispness with which the present hig accuracy STEFR method resolved these physical phenomena and their interactions. An experimental Schlieren picture[23] of the vortex street behind a similar large scale turbine blade cascade model at transonic outlet Mach number is shown in Fig.19 displaying similar shock-trailing edge vortex wake interactions.

(a) $t = t_0$

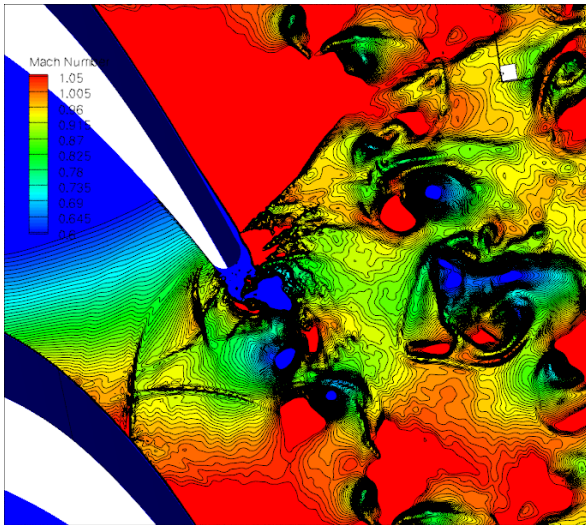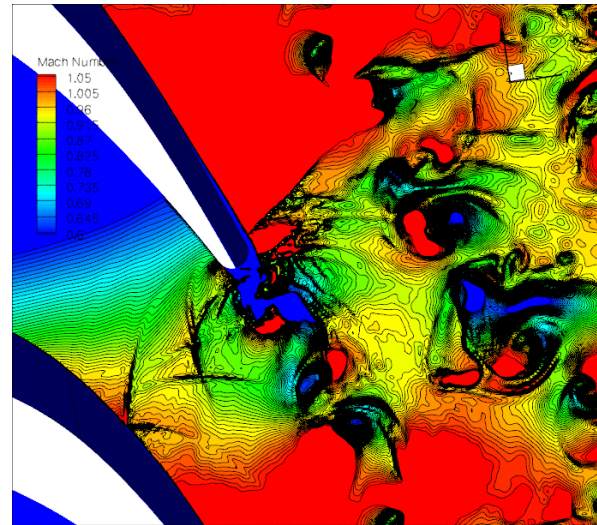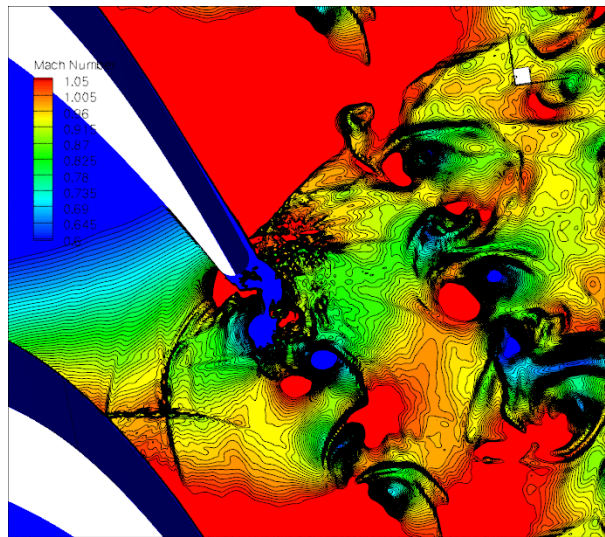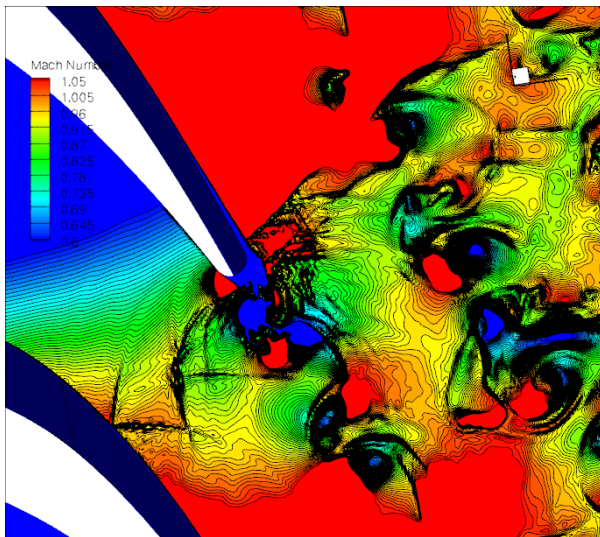(b) $t = t_0 + 2dt$

American Institute of Aeronautics and Astronautics

(c) $t = t_0 + 4dt$

(d) $t = t_0 + 6dt$

(f) $t = t_0 + 8dt$

(g) $t = t_0 + 10dt$

American Institute of Aeronautics and Astronautics

(h) $t = t_0 + 12dt$ (i) $t = t_0 + 14dt$
Figure17. Instantaneous Mach number for VKI-LS59 blade, $dt = 0.025T_c$
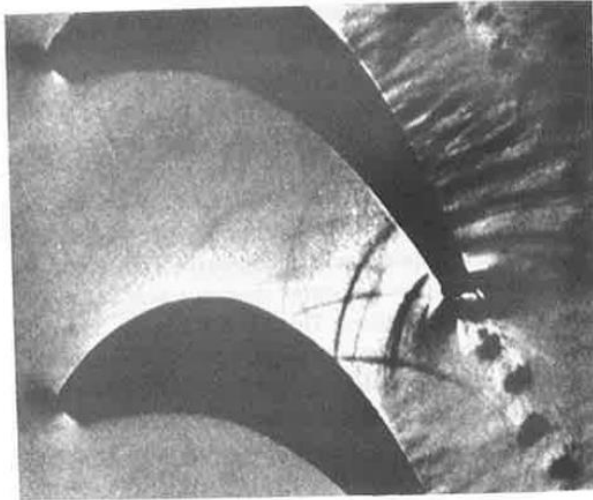

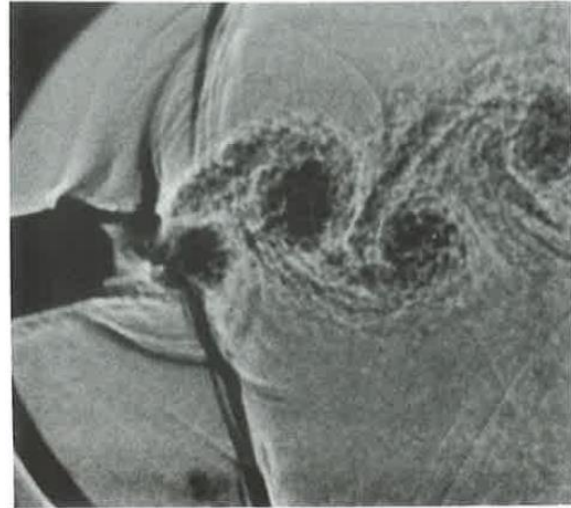Figure 18. Schlieren picture of a rotor blade[22].   Figure 19. Schlieren picture behind a large scale turbine blade[23]

The LES results are spanwise averaged and then time averaged in Fig20. and it can be seen the results match the experimental data well in the smooth region but not as well in the shock region especially at the throat part of the blade passage with the weak shocks. In order to inverstigate the mismatch, instantaneous spanwise averaged predicted isentropic Mach numbers on the blade surface are shown in Fig21. at different time intervals during the periodic cycle. It can be seen that the variation of the isentropic Mach number on blade surface is very distinctive with nearly all experiment points captured within the envelope. It is not clear how the experimental set-up responds and therefore averages the unsteady flowfield. In the case of the simulation it would appear that the time average of a periodically moving discontinuity is smooth an unexpected result. Detail analysis can be found in paper 7.
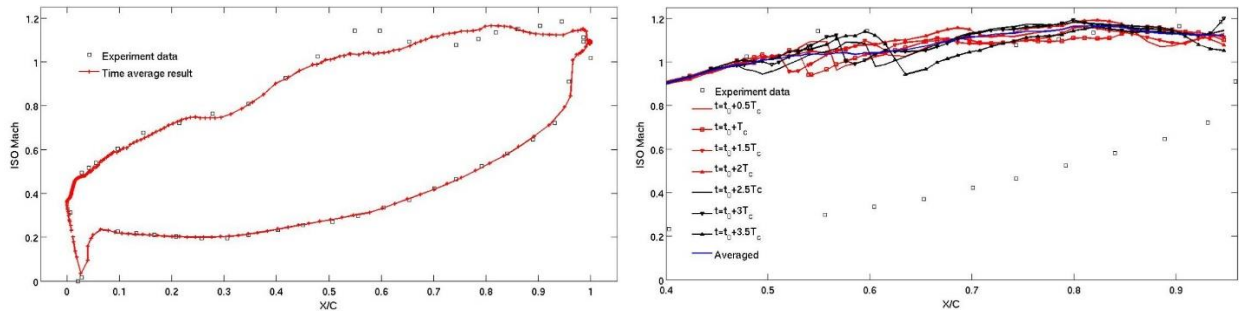

Figure 20. Compasion of iso-Mach number between calculation&expriment[7]. Figure 21. Envelope of transient iso-Mach[7]

VIII.   Conclusion

This paper describes the construction of an end-to-end parallel flow simulation system based on the space time extension of flux reconstruction method(STEFR), which support hybrid unstructured meshes to handle complex geometries. The method demonstrated to be robust, high computational efficiency, low memory, and flexible to solve high fidelity, high resolution and time-accurate unsteady flows. A series of numerical validations suggest this flow simulation system is able to be developed to provide fast, accurate and reliable simulations for industrial, real geometry flow problems, with the integrated advanced high order mesh generation.

American Institute of Aeronautics and Astronautics

References

[1]http://www.cambridgeflowsolutions.com

[2]Yi Lu, Kai Liu, and W. N. Dawes. Large eddy simulations using high order flux reconstruction method on hybrid unstructuredmeshes. InAIAAScienceandTechnologyForum and Exposition(SciTech2014), *AIAA2014-0424*, 2014.

[3]http://www.paraview.org/

[4]http://www.aer.mw.tum.de/en/research-groups/automotive/drivaer/

[5]Y.Lu. "Local Reconstruction High Order Method and Experimental Research for Internal Flow of Turbomachinery". *phD thesis*, Tsinghua University, China.

[6]Y.Lu, W. N. Dawes, and X. Yuan. "Investigation of 3D internal flow using new flux-reconstruction high order method". *In Proceedings of ASME Turbo Expo 2012: Power for Land, Sea and Air"*, GT2012-69270, 2012.

[7]Y.Lu, W.N.Dawes, "High order Large Eddy Simulations for a transonic turbine blade using hybrid unstructured meshes", ASME Paper, GT2015-42283, submitted.

[8]http://geuz.org/gmsh/

[9] Huynh, H. T. "A flux reconstruction approach to high-order schemes including discontinuous Galerkin methods", *18th AIAA Computational Fluid Dynamics Conference*, 2007, AIAA 2007-4079

[10]Z. J. Wang and Haiyang Gao, "A unifying lifting col- location penalty formulation including the discontinuous Galerkin, spectral volume/difference methods for conser- vation laws on mixed grids," *Journal of Computational Physics*, Vol. 228, No. 21, 2009, pp. 8161, 8186.

[11]Haiyang Gao and Z. J. Wang. "A high-order lifting collo- cation penalty formulation for the Navier-Stokes equations on 2d mixed grids". *In 19th AIAA Computational Fluid Dynamics*, AIAA 2009-3784, 2009.

[12]T. Haga, H. Gao and Z. J. Wang . "A high-order unifying discontinuous formulation for the navier-stokes equations on 3d mixed grids". *Mathematical Modelling of Natural Phenomena* , Vol. 6, No. 21, 2011, pp. 28, 56.

[13]P.Vincent, P. Castonguay, and A.Jameson. "A new class of high-order energy stable flux reconstruction schemes". Journal of Scientific Computing, Vol.47, No. 1, 2010, pp. 50,72.

[14]C. Liang, C. Cox, and M. Plesniak. "A comparison of computational efficiencies of spectraldiffer- ence method and correction procedure via reconstruction". Journal of Computational Physics, Vol.239, , 2013, pp. 244, 261.

[15]J. S. Hesthaven and T. Warburton. *Nodal Discontinuous Galerkin Methods: Algorithms,Analysis,andApplications*. Springer, Berlin, 2007.

[16]B. Cockburn and C. W. Shu. "TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws II: General framework". Mathematics of Computation, Vol.52, 1989, pp.411,435.

[17]M. Parsani, G. Ghorbaniasl, C. Lacor, and E. Turkel. "Implicit solution of the unsteady euler equations for high- order accurate discontinuous galerkin discretisations". Journal of Computational Physics, Vol.225, 2007, pp. 1994, 2015.

[18]M. Parsani, G. Ghorbaniasl, C. Lacor, and E. Turkel. "An implicit high-order spectral difference approach for large eddy simulation". Journal of Computational Physics, Vol. 229, 2010, pp.5373–5393.

[19]J.J. van der Vegt and H van der Ven. "Space time discontinuous galerkin finite element method with dynamic grid motion for inviscid compressible flows, i. general formulation". Journal of Computational Physics, Vol.182, 2002, pp.546, 585.

[20]C.M Klaij, J.J.W. van der Vegt, and H van der Ven. "Space time discontinuous galerkin method for the compressible navier-stokesequations". Journal of Computational Physics, Vol.217, 2006, pp. 589, 611.

[21]G. Gassner, M. Dumnser, F. Hindenlang, and C.D Munz. Explicit one-step time discretization for discontinuous galerkin and finite volume schemes based on local predic- tors. Journal of Computational Physics, Vol.230, 2011, pp. 4232, 4247.

[22]B. Owren and M Zennaro. Derivative of efficient continuous explicit runge-kutta methods. Journal of Science Computing, Vol 239, 2013, pp.138, 146.

[23]J. Peraire A. Huerta, E. Casoni. A simple shock-capturing technique for high-order discontinuous galerkin methods. International Journal for Numerical Methods in Fluids, Vol. 69, 2012, pp. 1614, 1632.

[24]Y. Lu, K. Liu, and W.N.Dawes. "Non-linearstability analysis for high order flux reconstruction method". *International Journal of Computational Fluid Dynamics*, Submitted.

[25]H.Luo, J.D.Baum and R.Lohner, *Extension of Harten-Lax-van Leer Scheme for Flows at All Speeds*. AIAA Journal, Vol. 43, No. 6, June 2005

[26]G.L.Park and P.Moin, *An improved dynamic non-equilibrium wall-model for large eddy simulation*. Physical of Fluids, Vol 26, 2014.

[27]G.S.Constantinescu, M.C.Chapelet, and K.D.Squires, *Prediction of Turbulent Flow over a Sphere*, AIAA Journal, 41, 1733-1742 (2003).

[28]J.Kim, D. Kim, and H. Choi, *An Immersed-Boundary Finite-Volume Method for Simulations of Flow in Complex Geometries,* Journal of Computational Physics, 171, 132 150 (2001).

[29]M.Giacobello, *Wake structure of a transversely rotating sphere at moderate Reynolds numbers*, Ph.D. thesis, The University of Melbourne, Department of Mechanical Engineering, September 2005.

[30]A.G.Tomboulides, , S.A. Orszag and G.E. Karniadakis, *Direct and Large eddy simulations of axisymmetric wakes*, AIAA paper number 93-0546, 1993.

[31]S.E.Kim, *Large Eddy Simulation Using Unstructured Meshes and Dynamic Subgrid-Scale Turbulence Models*, 34th AIAA Fluid Dynamics Conference and Exhibit, Portland, Oregon, AIAA paper number 2004-2548, 2004.

American Institute of Aeronautics and Astronautics

[32]E.Achenbach, *Vortex shedding from spheres*, J. Fluid Mech., 62, pp. 209-221, 1974.

[33]N.C. Baines, C.H.Sieverding, R. Kiock and F.Kehthaus. *The transonic flow through a plane turbine cascade as measured in four European wind tunnels*. Journal of Engineering for Gas Turbine and Power, Vol 108, 1986, pp. 277, 284.

[34]J.Heinemann, O. Lawczeck. *Von Karman vortex street in the wake of subsonic and transonic cascades. In AGARD conference proceedings N.177 on "Unsteady Phenomena in Turbumahinery"*, AGARD CP177

[35]C.H.Sieverding, G. Cicatelli. *A review of the research on unsteady turbine blade wake characteristics. In AGARD PEP 85th Symposium on "Loss Mechanisms and Unsteady Flows in Turbomachinery"*, AGARD CP571

American Institute of Aeronautics and Astronautics