Towards industrial large eddy simulation using the FR/CPR method

By

Feilin Jia

Submitted to the Department of Aerospace Engineering and the Graduate Faculty of the University of Kansas in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

Co-Chair: Dr. Ray Taghavi

Co-Chair: Dr. Z.J. Wang

Dr. Saeed Farokhi

Dr. Zhongquan Zheng

Dr. Suzanne M. Shontz

Date Defended: 25 April 2019

The dissertation committee for Feilin Jia certifies that this is the approved version of the following dissertation:

Towards industrial large eddy simulation using the FR/CPR method

Co-Chair: Dr. Ray Taghavi

Co-Chair: Dr. Z.J. Wang

Date Approved: 7 May 2019

Abstract

NASA's 2030 CFD Vision calls for the development of accurate and efficient scale-resolving simulations for turbulent flow, such as large eddy simulation (LES) and direct numerical simulation (DNS). This is primarily because the Reynolds-averaged Navier-Stokes (RANS) approach has failed to predict vortex-dominated flow involving large flow separations, e.g., flow through a jet engine or over aircraft near the edge of the flight envelope, i.e., during take-off and landing at high angles of attack. Although the DNS approach resolves all turbulence scales, it is too expensive in the foreseeable future for real world flow problems because of the disparate length and time scales in the flow. LES resolves the energetic large scales while modeling the smaller scales, so it provides a good compromise between accuracy and cost. As a result, LES is widely considered to be the method of choice for next generation CFD design tool.

The major obstacle for LES is its considerable computational cost since unsteady 3D simulations need to be performed to obtain the mean flow quantities such as the drag and lift coefficients. In order to resolve the dominant scales in a turbulent flow, numerical methods used for LES should have low dissipation and dispersion errors. This means standard second order finite-volume methods are usually not accurate or efficient enough for LES applications. High-order methods (order of accuracy > 2) have demonstrated their potential for LES and DNS in the past decade because of their low embedded numerical dissipation and dispersion errors. In the present study, we develop and demonstrate a recently developed high-order method, called flux reconstruction (FR) or correction procedure via reconstruction (CPR), for industrial LES. A major advantage of the FR/CPR method is its capability to handle unstructured mixed meshes, and its compactness and scalability, which is particularly desired on modern super-computers. We therefore address the following major pacing items in industrial LES in the present study:

- High-order methods

- Geometric flexibility
- Efficient time integration
- Efficient implementation on modern super computers
- Demonstration for real world applications

Acknowledgement

I would like to take this opportunity to thank the people who have made my work possible. I would like to express my most sincere gratitude to my adviser and co-chair of my dissertation committee, Professor Z.J. Wang, for his valuable advice throughout my graduate study. It is my honor being a member of his research lab. I would like to thank co-chair of my dissertation committee, Professor Ray Taghavi, for his guidance, and support. In addition, I would like to thank members of my dissertation committee, Professors Saeed Farokhi, Suzanne M. Shontz and Zhongquan Zheng for their time, very useful comments and critical suggestions during my PhD study.

Thanks are also given to my fellow group members, Lei Shi, Yanan Li, Cheng Zhou, Zhaowen Duan, Jeremy Ims, Mohammad Ahmad Alhawwary, Eduardo Jourdan and Salman Rahmani for their help and insight. Special thanks to my parents and elder brother for their warm support for all these years.

Table of contents

Chapter 1 Introduction

Over the past 30 years, a sustained scientific and economical investment has been made to develop robust, accurate and efficient algorithms for flow simulation because of the rapidly growing demand of civil aviation industry. Until now the main tool for industrial Computational Fluid Dynamics (CFD) simulations has been RANS solvers, which perform adequately well in a broad range of aeronautical engineering applications because of its good robustness and accuracy. However, more and more different kinds of emerging aircraft roles deal with more and more complex flow conditions, e.g. vortex-dominated flows, massively separated flows, unsteady flows around flapping wings, noise propagation problems, which the RANS approach failed to resolve. Therefore, the need for high-fidelity simulation techniques to predict complex flows is growing rapidly. Different from RANS, where all turbulent scales are modeled by a turbulence model, DNS methods resolve all scales, but DNS remains unaffordable for its extremely high computational cost in the foreseeable future. LES sits in the middle between of RANS and DNS. In LES, large energetic scale motions are resolved accurately while the small scales are modeled by the sub-grid scale (SGS) models. The resolution of the featured scales, the solution obtained from LES is expected to be more accurate than RANS, but still feasible on the current computing hardware resources. As a result, NASA's 2030 CFD Vision calls for scale-resolving simulations such as large eddy simulation in the near future.

LES has been used to simulate turbulent flows since its inception over fifty years ago. It has been developed for decades and many different SGS models have been proposed. In this thesis, only the monotone integrated LES or implicit LES (ILES), in which no explicit SGS model is used, is considered for its lower computational cost. However, to achieve the desired resolution, the computational cost of LES is still very high for most applications on present day supercomputers. There are several main reasons for this:

- LES involves time-accurate three-dimensional computations. The simulation has to be long enough to obtain a steady mean solution for the quantities of interest, such as the lift and drag coefficients.
- For intermediate to high Reynolds number problems, the range of scales which should be resolved is still quite large. These scales dictate the resolution of the computational mesh, and the order of accuracy of the numerical method, and the computational cost to a large degree.
- Low order methods, such as the 1^{st} or 2^{nd} order ones, usually require excessively fine computational meshes to obtain reasonable results. The development of high-order methods in the past two decades has injected new life into LES, and can speed up LES by orders of magnitude, making LES feasible for at least intermediate Reynolds number flow problems.

High order methods have received much attention for their superior accuracy and efficiency with a relatively small number of degrees of freedom. In the past two decades, many different kinds of high order numerical methods have been developed, e.g. spectral element method, discontinuous Galerkin (DG) method, spectral volume, spectral difference and the FR/CPR method. In the present thesis, the FR/CPR is selected as the high order method because of its accuracy, efficiency and simplicity. The main objective of the present study is to further develop the FR/CPR method for industrial LES applications. In particular, the following research issues are addressed:

- Develop the FR/CPR method for mixed unstructured meshes including tetrahedral, hexahedral, prismatic, pyramidal elements to achieve the maximum geometric flexibility to handle very complex real-world flow configurations.

- Evaluate the accuracy, efficiency and robustness of the FR/CPR method for turbo-machinery benchmark flow problems through comparison with commercial second order methods.
- Develop implicit solution methods, and compare the accuracy, efficiency and scalability of explicit and implicit temporal schemes associated with the high order FR/CPR method.
- Implement the FR/CPR method efficiently on modern computing architecture such as GPU clusters.

The included journal papers provide a reasonable summary of my research performed during my PhD study. The current work on the implementation on a GPU cluster is summarized in Chapter 5, and I plan to complete the work and submit it for journal publication.

AIAA JOURNAL
Vol. 57, No. 4, April 2019

Evaluation of Second- and High-Order Solvers in Wall-Resolved Large-Eddy Simulation

Feilin Jia,* Jeremy Ims,† and Z. J. Wang‡
University of Kansas, Lawrence, Kansas 66045
and
James Kopriva§ and Gregory M. Laskowski¶
GE Aviation, Lynn, Massachusetts 01905

DOI: 10.2514/1.J057232

In the context of wall-resolved industrial large-eddy simulation, a comparison is made between a high-order flux reconstruction (FR)/correction procedure via reconstruction (CPR) solver (hpMusic) with p refinement and a commercial second-order finite volume solver (Fluent) with mesh refinement (h refinement). A well-known benchmark problem in turbomachinery is employed: transonic flow over a von Karman Institute high-pressure turbine vane at a Reynolds number of 1.16×10^6. All of the meshes originated from the same coarse mesh, a mixed unstructured mesh, generated through global uniform refinement for the purpose of evaluating the solution dependence on mesh and polynomial order. Because the meshes used for hpMusic and Fluent belong to the same family, useful information about solution accuracy and efficiency can be obtained. Detailed comparisons are made in mean surface loading, heat transfer, power spectral density of pressure at selected monitor points, mean boundary-layer velocity and total temperature profiles, and wake loss. Numerical results are compared with experimental data, when available. The high-order FR/CPR method is shown to achieve a higher accuracy at a reduced cost than the second-order finite volume method.

Nomenclature

C	=	chord length
C_{ax}	=	chord length projected in x direction
C_d	=	drag coefficient
D_{TE}	=	thickness of trailing edge
$\mathrm{d}t$	=	time step
$\boldsymbol{F}$	=	flux vector
f	=	frequency
$\boldsymbol{n}$	=	unit normal vector
p	=	polynomial order of solution
t^*	=	nondimensional time
$\boldsymbol{U}$	=	conservative variable vector
V_{exit}	=	isentropic velocity at exit
W	=	test function
δ	=	correction term
ρ	=	density
Ω	=	domain

Subscript

com	=	common value

Superscript

n	=	normal component

Received 1 March 2018; revision received 17 October 2018; accepted for publication 26 October 2018; published online 28 December 2018.

*Ph.D. Student, Department of Aerospace Engineering, 2120 Learned Hall.
†Ph.D. Student, Department of Aerospace Engineering, 2120 Learned Hall.
‡Spahr Professor, Department of Aerospace Engineering, 2120 Learned Hall. Fellow AIAA.
§Senior Engineer, 1000 Western Avenue, MS74301.
¶Principal Engineer High-Fidelity CFD, Advanced Design Tools. Member AIAA.

I. Introduction

WITH continued growth in the civil aviation industry, air travel is expected to double in 20 years. This prediction of high-level growth highlights the need for improvement in aircraft and engine performance: a reduction of fuel burn, a lessening of greenhouse gas emission, a diminishing of noise pollution, and a reduction in weight and cost. These improvements call for simulation tools that are much more accurate and efficient in computing turbulence, the most prevalent flow condition in industrial problems. The most widely used industrial computational fluid dynamics (CFD) tools model turbulence via the Reynolds-averaged Navier–Stokes (RANS) equations, closed with a turbulence model. This approach has failed to resolve some of the more complex vortex-dominated flow problems that are arising in the context of aviation. As a result, NASA's 2030 CFD Vision [1] calls for scale-resolving simulations such as large-eddy simulation (LES), which have greater simulation capability.

The potential offered by LES [2] to compute turbulent flows has been well known since its inception over 50 years ago. It has been under development for decades and has now begun the shift from an analysis-only tool to a design-process tool to complement RANS simulation [3]. LES is believed to offer the best promise for vortex-dominated flows including flow transition, which characterize many aerospace applications such as flow over high-lift configurations, rotorcraft flows, and more recently flows in aircraft engines [3–5]. LES is also expected to benefit the simulation of massively separated flows, for which the RANS approach breaks down, due to the lack of a statistical steady mean flow or the lack of a universal turbulence closure.

In LES, large scales and small scales are separated by a low-pass filter, either explicitly or implicitly. The large scales are resolved while the effect of small scales is represented by a subgrid-scale (SGS) stress model [6]. Because small-scale motions are believed to be more universal and thus easier to model than large-scale ones, LES can offer reasonable accuracy, even for unsteady separated flows, while requiring much less computer resources than direct numerical simulation (DNS). One critical parameter in LES is the filter width Δ that determines which scales are computed and which are modeled. Ideally, the determination of Δ should be based on physical considerations and error tolerances, such as the computation of mean lift or drag error to a given accuracy. However, it is very difficult to determine the filter width without conducting

numerical simulations at various resolutions. In reality, an explicit filter is rarely employed. The mesh size often serves as an "implicit" filter size, although for methods with multiple internal degrees of freedom (DOFs), a fraction of the mesh size serves as the filter size [7,8]. When one refines the mesh, the filter width is also reduced. Therefore, mesh independence is difficult to demonstrate with mesh-refinement studies in LES, unless the mesh resolution approaches that required by DNS. In this case, the LES solution converges to the DNS solution.

It is obvious that the cost of LES is largely dependent on the filter width (mesh size) and the accuracy/efficiency of the numerical methods. For scale-resolving simulations such as LES and DNS, conclusions have been made from multiple high-order CFD international workshops that high-order methods are much more accurate/efficient than low-order ones [9–11], at least on benchmark problems such as the Taylor–Green vortex problem. For a Cartesian box domain, it is relatively straightforward to compare different methods and flow solvers on meshes of the same family (i.e., uniform Cartesian meshes).

In fact, many CFD solvers have been compared in the simulations of two- and three-dimensional flow in the past two decades [12–16], but those solvers are either finite volume or finite difference based and mostly second-order-accurate in space. Some high-order finite volume and finite element solvers are compared [17–21], but they are all in the RANS context. In particular, a recent study compared several second-order finite volume solvers (NSU2D, FUN3D, CFL3D) and a high-order finite element solver (DG3D) systematically [17], but only in the simulation of low-speed two-dimensional flow over an airfoil, and the flow is also smooth (Mach 0.15). Because of the reasons for large-eddy simulation stated previously, the performance of both the second- and the high-order solvers for industry-level three-dimensional flow in the LES context needs to be investigated systematically. To compare different methods and solvers, it is very important to employ meshes in the same family, generated without giving preference to any particular method or solver because different solvers have different dissipation and dispersion characteristics and so have different preferences on mesh properties. Through adjoint-based mesh adaptations, one can obtain nearly "optimal" meshes for any particular solver, which will provide the most accurate output prediction for any target number of degrees of freedom (nDOFs) [22]. Thus, evaluations purely based on the nDOFs are not objective unless consideration has been made for how well the grids resolve important features. A poor-quality grid with many DOFs can generate very poor numerical results, severely distorting the evaluation.

The primary objective of the present study is to compare a commercial finite volume solver, Fluent, to a high-order flux reconstruction (FR)/correction procedure via reconstruction (CPR) [23] solver, hpMusic, using mixed unstructured meshes. The comparative simulation is an LES benchmark turbomachinery problem: transonic flow over a von Karman Institute (VKI) vane at a medium Reynolds number of 1.16×10^6 [24]. The hpMusic solver has been shown to produce highly accurate solutions for industrial LES [25], with an accuracy comparable to the compact difference method [26] while being geometrically more flexible. To ensure an objective comparison, the computational meshes used by Fluent and hpMusic are in the same refinement family. To assess the mesh and order dependence of the solvers, mesh-refinement studies are carried out with Fluent, whereas p-refinement studies are carried out with hpMusic. We compare a wide range of parameters of interest to design engineers: surface loading, heat transfer, mean boundary-layer profiles, power spectral density (PSD) of pressure at selected monitor points, and wake loss.

The paper is organized as follows. In Sec. II, we briefly review the numerical methods used in Fluent and in hpMusic: the finite volume and the FR/CPR methods. In Sec. III, we describe the process to generate 1) the coarse linear mesh, 2) a quadratic coarse mesh, and 3) the finer meshes. In Sec. IV, we present detailed numerical results with an extensive comparison between the two solvers. Finally, we conclude the paper with a summary of major findings in Sec. V.

II. Solvers and Numerical Methods

A. hpMusic Solver

The hpMusic solver employs the high-order FR/CPR method, which was originally proposed by Huynh [23] in 2007 for hyperbolic partial differential equations and later extended to hybrid unstructured meshes [27,28]. Further developments on the FR/CPR method are given in [29–32]. This method belongs to the family of discontinuous finite element methods, similar to the discontinuous Galerkin (DG) [33,34] and spectral difference method [35], but it has some unique advantages. For example, FR contains a larger family of schemes [36], which may allow larger time steps than the DG method. Here, we present a brief introduction of the FR/CPR method, starting from a hyperbolic conservation law:

$$\frac{\partial \boldsymbol{U}}{\partial t} + \nabla \cdot \boldsymbol{F}(\boldsymbol{U}) = 0 \tag{1}$$

with proper initial and boundary conditions, where the vector $\boldsymbol{U}$ consists of conservative variables, and $\boldsymbol{F}$ is the flux. By discretizing the computational domain with nonoverlapping elements and introducing an arbitrary test function W in each element, the weighted residual formulation of Eq. (1) on element V_i can be expressed as

$$\int_{V_i} \left[\frac{\partial \boldsymbol{U}}{\partial t} + \nabla \cdot \boldsymbol{F}(\boldsymbol{U}) \right] W \, \mathrm{d}\Omega = 0 \tag{2}$$

The conservative variables inside each element are assumed to be polynomials and are expressed by nodal values at certain points called solution points. After applying integration by parts to the divergence of flux, replacing the normal flux term with a common Riemann flux F^n_{com}, and integrating back by parts, we obtain

$$\int_{V_i} \frac{\partial \boldsymbol{U}_i}{\partial t} W \, \mathrm{d}\Omega + \int_{V_i} W \nabla \cdot \boldsymbol{F}(\boldsymbol{U}_i) \, \mathrm{d}\Omega + \int_{\partial V_i} W [F^n_{\text{com}} - F^n(\boldsymbol{U}_i)] \, \mathrm{d}S = 0 \tag{3}$$

Here, the common Riemann flux is computed with a Riemann solver

$$F^n_{\text{com}} = F^n_{\text{com}}(\boldsymbol{U}_i, \boldsymbol{U}_{i+}, \boldsymbol{n}) \tag{4}$$

where $\boldsymbol{U}_{i+}$ stands for the solution outside the current element, and n denotes the outward normal direction of the interface. The normal flux at the interface is

$$F^n(\boldsymbol{U}_i) = F(\boldsymbol{U}_i) \cdot \boldsymbol{n} \tag{5}$$

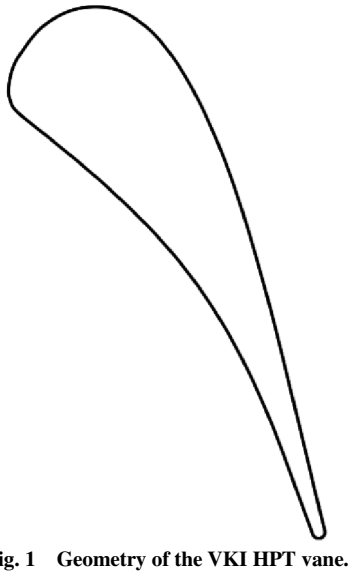


Fig. 1 Geometry of the VKI HPT vane.

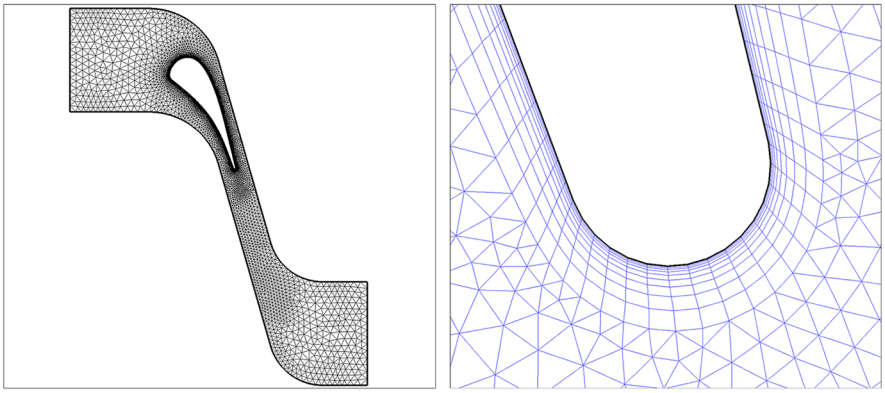


Fig. 2 Unstructured hybrid coarse mesh used for HPT vane geometry.

Note that if the face integral in Eq. (3) can be transformed into an element integral, then the test function will be eliminated. To do so, a "correction field" δ_i is defined within each element as

$$\int_{V_i} W\delta_i \,\mathrm{d}\Omega = \int_{\partial V_i} W[F^n]\,\mathrm{d}S = 0 \tag{6}$$

where $[F^n] = [F^n_{\text{com}} - F^n(\boldsymbol{U}_i)]$ is the normal flux jump. Equations (3) and (6) result in a formula that applies to every element independently:

$$\int_{V_i} \left[\frac{\partial \boldsymbol{U}_i}{\partial t} + \nabla \cdot \boldsymbol{F}(\boldsymbol{U}_i) + \delta_i\right] W \,\mathrm{d}\Omega = 0 \tag{7}$$

The final formulation for each solution point j is

$$\frac{\partial \boldsymbol{U}_{i,j}}{\partial t} + \Pi_j[\nabla \cdot \boldsymbol{F}(U_i)] + \delta_{i,j} = 0 \tag{8}$$

where Π_j denotes a projection to the polynomial space, and subscript j denotes the jth solution point within the element.

For viscous flux involving the gradient of conservative variables, we use the Bassi–Rebay 2 scheme [33]. For time integration, we use a second-order backward-difference formula (BDF2) with an Lower-Upper Symmetric-Gauss-Seidel solver for the nonlinear system [37]. To simulate turbulence, hpMusic uses implicit large-eddy simulation (ILES) [8,38,39] because of its lower computational cost compared with the conventional SGS models. In [8], we showed why ILES performed better than the static or dynamic Smagorinsky model with the FR/CPR method. To stabilize the flow near shocks, a robust limiter is also applied [40].

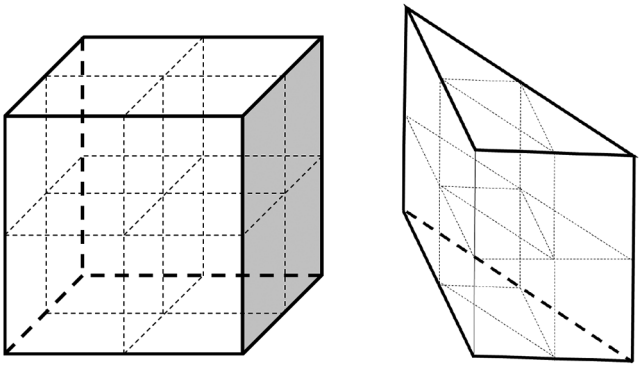


Fig. 3 Partitioning strategy for a hexahedral and a prismatic cell.

B. Fluent Solver

Large-eddy simulations with Fluent v17.2 are performed with the wall-adapting local eddy (WALE) subgrid-scale model [24]. The second-order bounded central differencing scheme is applied to the spatial discretization of the momentum equations. The bounded central differencing scheme is based on the normalized variable diagram (NVD) approach combined with the convection boundedness criterion. The bounded central differencing scheme is a composite NVD scheme that contains a pure central differencing, a blended scheme of the central differencing, a second-order upwind scheme, and a first-order upwind scheme. The same scheme is used for all other transport equations within the WALE calculations. A bounded second-order implicit time advancement scheme is also selected based on previous studies [24]. The pressure-based system is then solved in a coupled manner. The WALE model is designed to return the correct wall asymptotic ($\sim y^3$) behavior for wall bounded flows and to return a zero-turbulent viscosity for laminar shear flows. This allows the correct treatment of laminar zones in the domain.

III. Mesh Generation for the von Karman Institute High-Pressure Turbine Vane

The experimental study of Arts and Rouvroit [41] has received significant attention by the academic and industrial communities [4,5,42] for its usefulness in testing the prediction of heat transfer

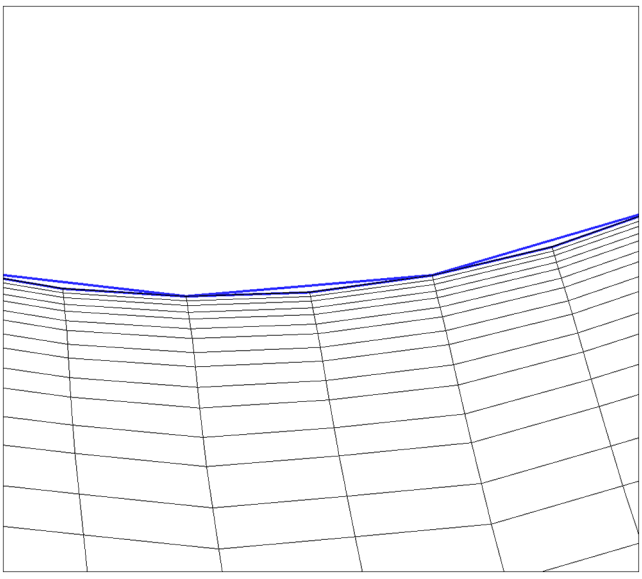


Fig. 4 Linear boundary and quadratic boundary reconstructed by meshCurve near the trailing edge.

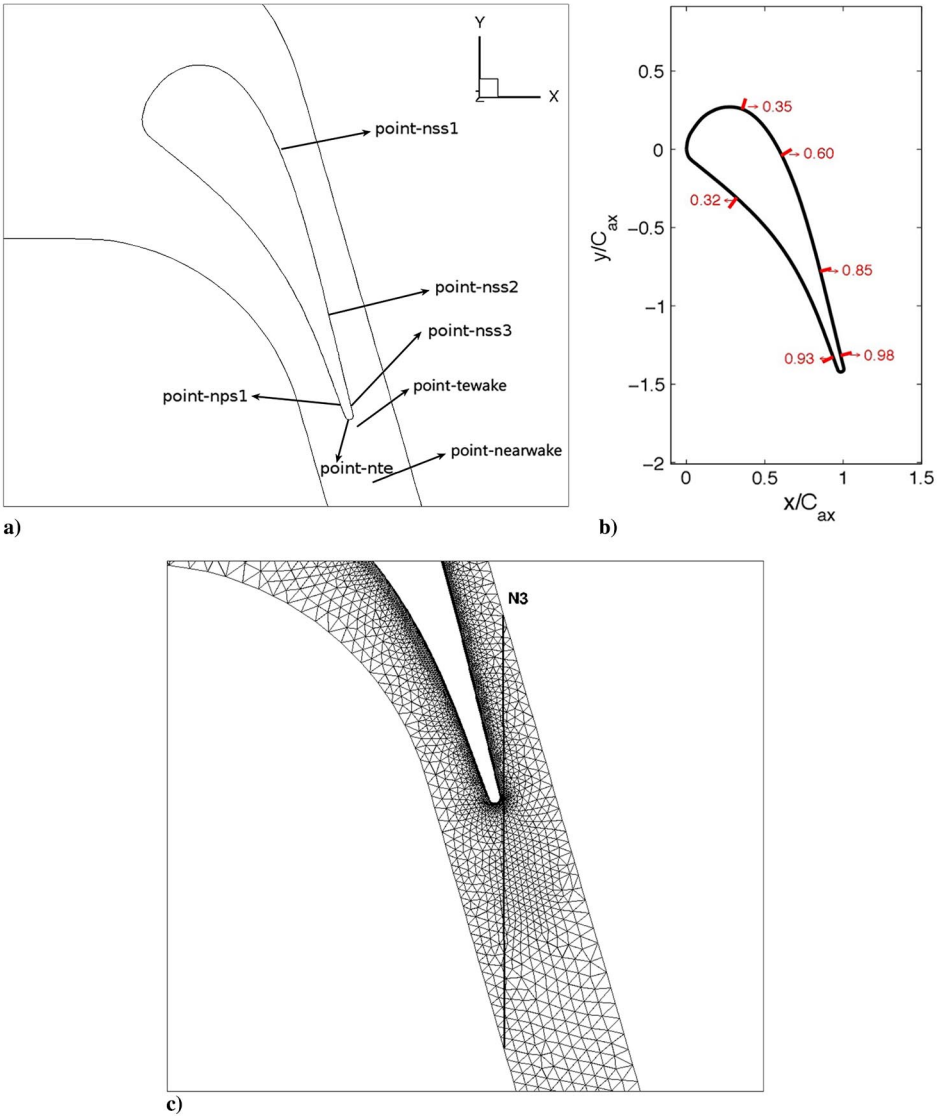


Fig. 5 Illustration of the measured locations: a) monitor points, b) boundary-layer profile, and c) wake loss.

coefficients for high-pressure turbine (HPT) vanes at engine-scale conditions. The experiments involved uncooled turbine vane studies that were carried out in a linear cascade in the von Karman Institute. The range of recorded Mach numbers, Reynolds numbers, and freestream turbulence conditions provide a challenging and well-suited test case for the prediction of boundary-layer transitions and the resulting surface heat transfer. In the present study, we choose case MUR129, corresponding to a very low level of incoming turbulence. The VKI vane geometry is shown in Fig. 1, and the complete vane description and coordinates can be found in [41]. The chord length of the vane (C) is 0.0676 m, and the span length is 0.0112 m (i.e., 16.6% of chord). The horizontal projection of the chord length (C_{ax}) is 0.0368 m. The distance between the inlet and the leading edge is 0.0546 m. The thickness of the trailing edge (D_{TE}) is 1.42e − 3 m.

Because we plan to conduct a global mesh refinement study, we generate finer meshes from an initial coarse mesh. The coarse mesh has 169,750 hexahedral and 278,425 prismatic elements, with

Table 1 Coordinates of the monitoring points

Point	x, m	y, m	z, m
tewake	0.03722	−0.05482	0.00000
nearwake	0.03950	−0.06300	0.00000
nps1	0.03448	−0.04890	0.00000
nss1	0.02658	−0.01169	0.00000
nss2	0.03164	−0.02951	0.00000
nss3	0.03634	−0.04886	0.00000
nte	0.03641	−0.05231	0.00000

Table 2 Horizontal coordinates of the location measuring wake loss

Location	x, m	x/C_{ax}
N3	0.038	1.033

Table 3 Simulation parameters with Fluent

Case	nDOFs	Time-step size, s	Inner iterations per time step	Steps before averaging	Averaging steps
Coarse	448,175	8e − 7	9	750	3,000
Medium	3,585,400	4e − 7	9	1500	6,000
Fine	28,683,200	2e − 7	9	3000	12,000

35 cells in the spanwise direction, as shown in Fig. 2. Hexahedral meshes are used near the blade to resolve the viscous boundary layer, whereas prismatic cells are used elsewhere. There are 10 structured hexahedral cells in the normal direction near the blade. The cell size in the wall-normal direction has an average $y+$ value of about 3.3. The cell's growth ratio along the normal direction is 1.5. The finer mesh is generated by uniformly refining the coarse mesh in all directions, as shown in Fig. 3. This procedure guarantees that all the meshes are in the same refinement family to assess mesh dependence. The original starting mesh was linear. It was converted to a quadratic mesh using meshCurve [43]. The curving process of meshCurve can be divided into three steps. The first step is to identify critical features such as sharp edges by a combination of several topological metrics [44]. Then, it employs a least-squares method to reconstruct interior edge and face nodes [45] to form higher-order elements. Finally, it employs a radial basis function approach to deform the interior cells [46]. Comparison between the linear and quadratic boundaries near the trailing edge is illustrated in Fig. 4. Note that the higher-order mesh indeed captures the curvature of the trailing edge.

The next finer mesh is called the medium mesh, which is produced through uniform refinement and has 3,585,400 elements. Likewise, the fine mesh, which has 28,683,200, is generated by refining the medium mesh.

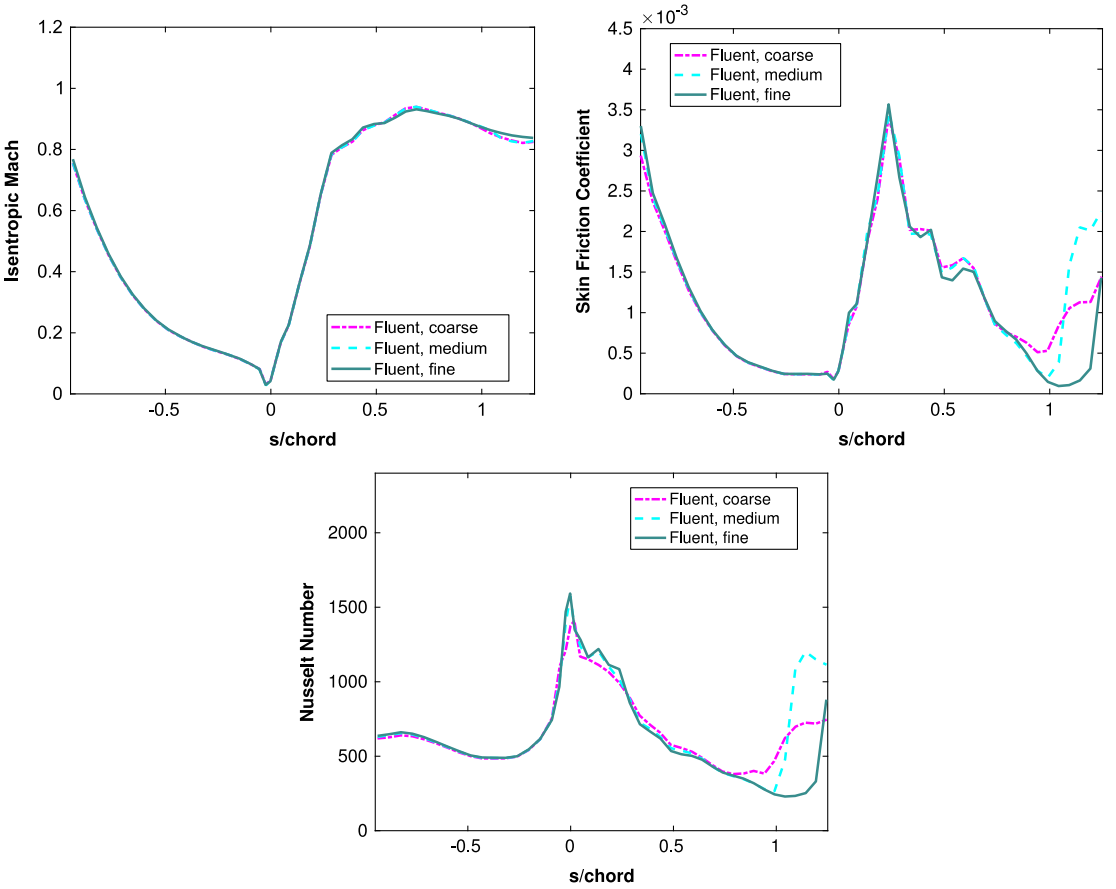


Fig. 6 Mean surface predictions with Fluent on the coarse, medium, and fine meshes.

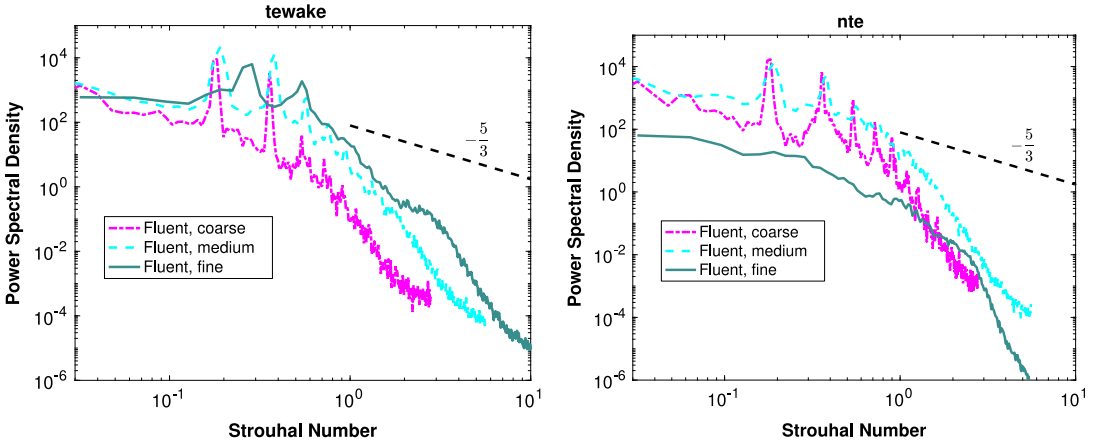


Fig. 7 PSD of pressure at selected monitoring points with Fluent.

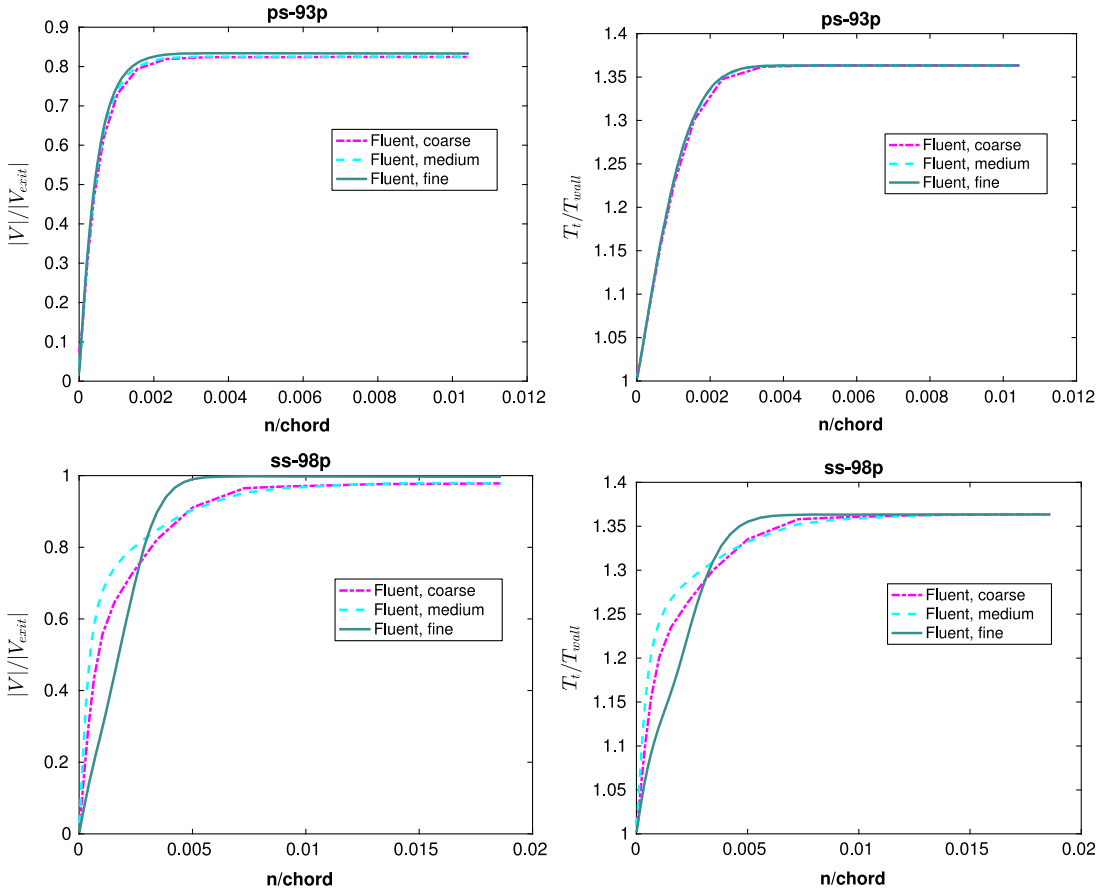


Fig. 8 Boundary-layer profiles computed with Fluent.

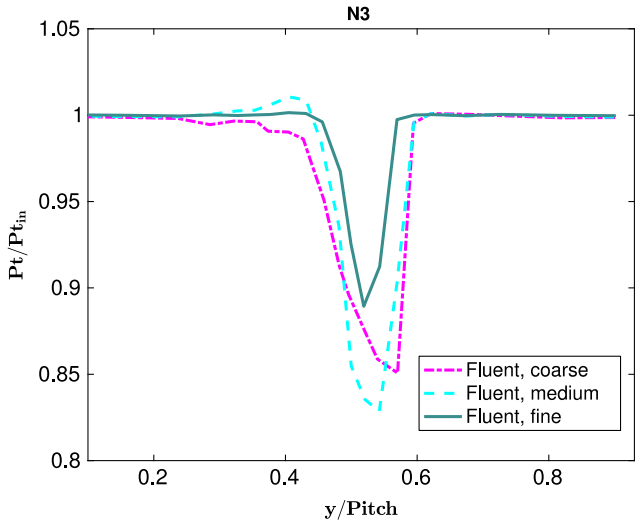


Fig. 9 Wake losses computed with Fluent and comparison.

To assess the solution quality, both the mean flow properties (vane loading and heat transfer) and the flow statistics at several monitoring locations are compared for both solvers. The locations and specific coordinates of those monitoring points are shown in Fig. 5a and Table 1, respectively. The mean boundary-layer velocity and total temperature profiles are also compared at selected stations, shown in Fig. 5b. Two stations are on the pressure side, and four stations are on the suction side. Note that we use the x/C_{ax} value to label the stations (e.g., "ps-32p" means the station located on the pressure side at $x/C_{ax} = 0.32$). In addition, one location is chosen for comparison of wake loss, shown in Fig. 5c and Table 2.

IV. Numerical Results and Discussions

A. Flow Conditions

As mentioned earlier, the present study corresponds to case MUR129 in [41], with zero freestream turbulence. At the inlet boundary, the total pressure and total temperature are fixed at 1.849e5 Pa and 409 K, respectively, with a zero angle of attack for the inflow velocity. At the exit, the static pressure is fixed at 1.165e5 Pa, resulting in an isentropic Mach number of 0.84. The vane is set to be a no-slip isothermal wall with a temperature of 300 K. Periodic boundary conditions are used in both the spanwise and pitchwise directions. Sutherland's law is used to determine the dynamic viscosity coefficient. The Reynolds number based on the chord length, the exit isentropic density, velocity V_{exit}, and viscosity is 1.16 million. The gas constant and the Prandtl number are 287.55 J/(kg·K) and 0.713, respectively. The isentropic exit velocity and the (true) chord are used to define a time scale

$$t^* = \frac{C}{V_{\text{exit}}} \tag{9}$$

which is the physical time to convect the flow by one chord length at the exit velocity. The time scale is used to nondimensionalize the physical time.

Table 4 Simulation parameters with hpMusic

p	nDOFs	Time-step size, s
1	3,028,550	2e − 7
2	9,594,900	5e − 8
3	22,001,000	2.5e − 8

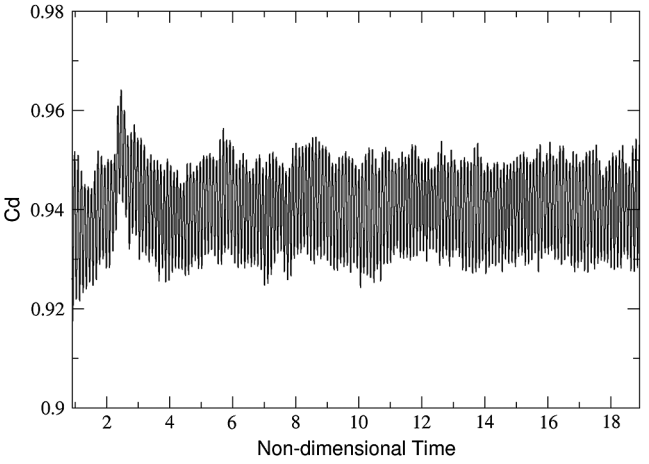


Fig. 10 Drag coefficient history for the $p1$ simulation with hpMusic.

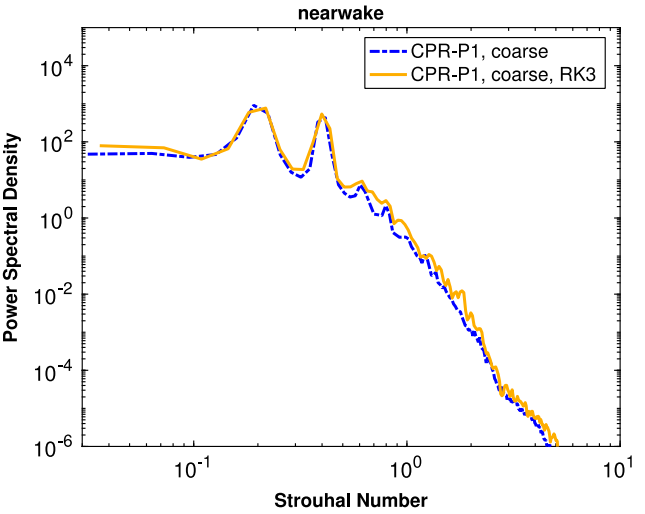


Fig. 12 Comparison of PSDs of pressure at a monitoring point near wake with explicit and implicit schemes.

B. Simulation Details and Results

Fluent simulations were conducted on the coarse, medium, and fine meshes to assess flow convergence. The total nDOFs (per equation) were 448,175, 3,585,400, and 28,683,200, respectively. A converged RANS solution was used to initialize the unsteady solution. Table 3 summarizes the time step, number of inner iterations, number of time steps to the statistically steady state, and averaging [24].

To assess solution mesh dependence, the computed mean surface isentropic Mach number, skin friction coefficient, and heat transfer on all three meshes are plotted in Fig. 6. Clearly, there is a lack of convergence on the transition location based on the mean skin friction and heat transfer. Away from the transition region, one can see that the heat transfer coefficient agrees well between the medium and fine meshes, whereas the skin friction does not show as clear convergence. For example, the difference between the skin friction on the fine and medium meshes is bigger than that between the medium and coarse ones in many places on the suction side (e.g., around $x/\text{chord} = 0.5$). A much earlier transition was predicted on the coarse and medium meshes than on the fine mesh, indicating that

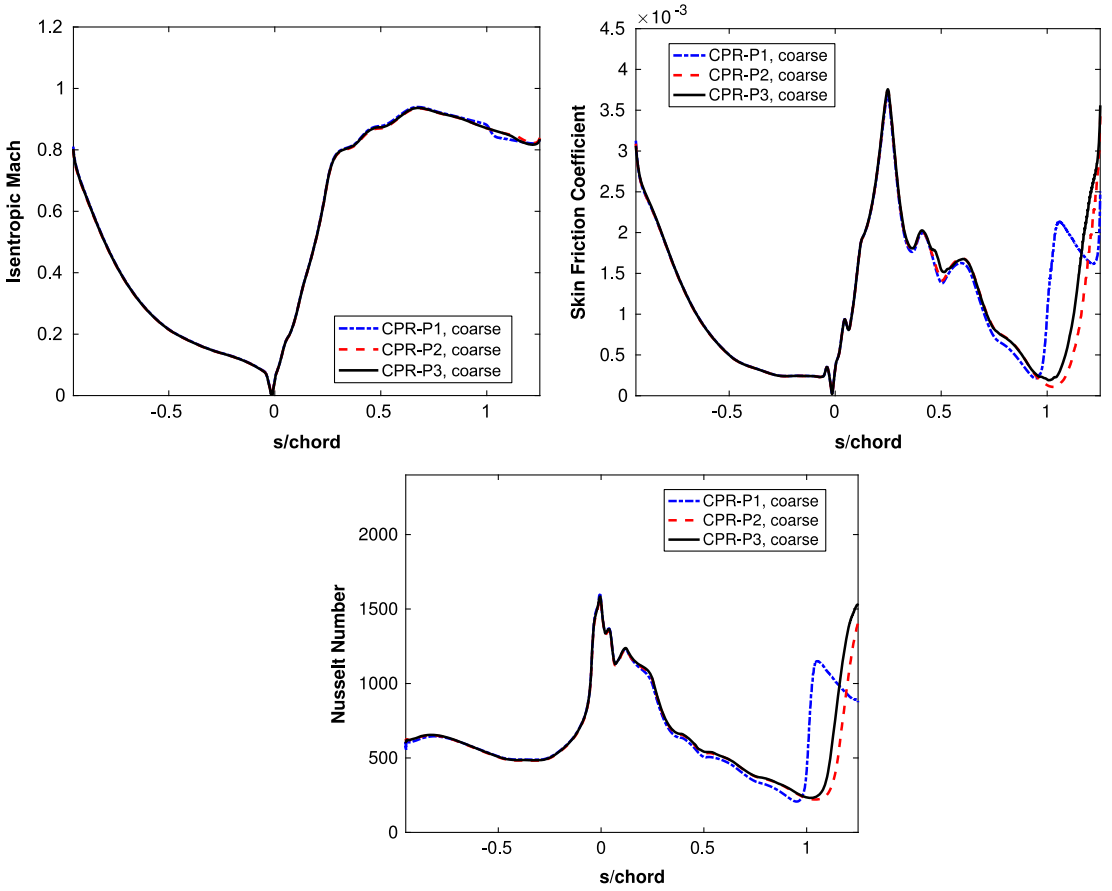


Fig. 11 Mean surface loading and heat transfer predictions with hpMusic on the coarse mesh at $p = 1, 2, 3$.

increased mesh resolution is needed to correctly predict the location of transition.

The PSDs of pressure at two monitoring points near the trailing edge are displayed in Fig. 7. Note that Strouhal number is calculated as

$$Sr = fD_{\mathrm{TE}}/V_{\mathrm{exit}} \tag{10}$$

where f is the frequency.

Note that the PSDs on different meshes do not show a clear trend of convergence. Also, all of them fail to capture the Kolmogorov $-5/3$ law, most likely due to the low order of accuracy and lack of resolution.

The total temperature and velocity profiles in the normal direction at one pressure-side and one suction-side stations are displayed in Fig. 8. At the pressure-side station, the temperature distribution along the normal direction shows a clear convergence toward the fine mesh result, but the velocity distribution does not because the velocity difference on the edge of the boundary layer between the medium and fine meshes is slightly larger than that between coarse and medium ones. At the suction side, the station is located inside the transition region. The profiles on different meshes do not show a visible trend of

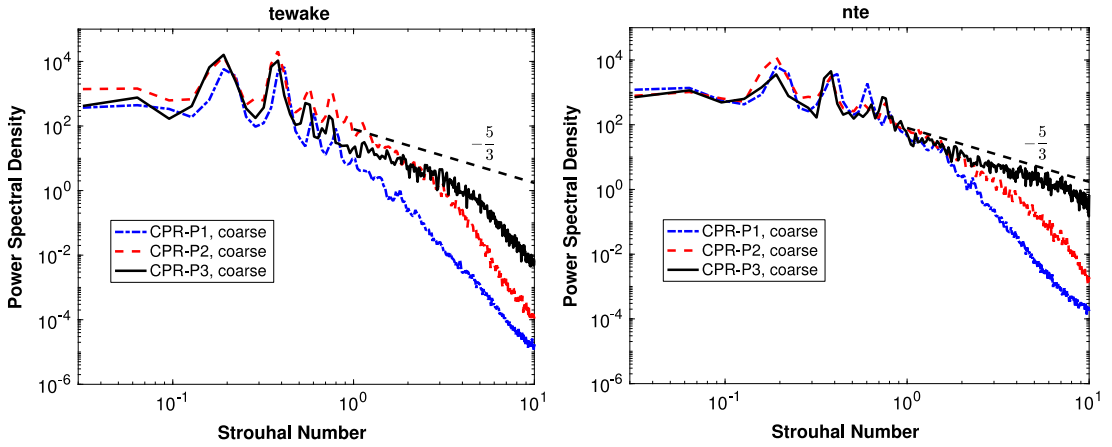


Fig. 13 PSD of pressure at selected monitoring points with hpMusic.

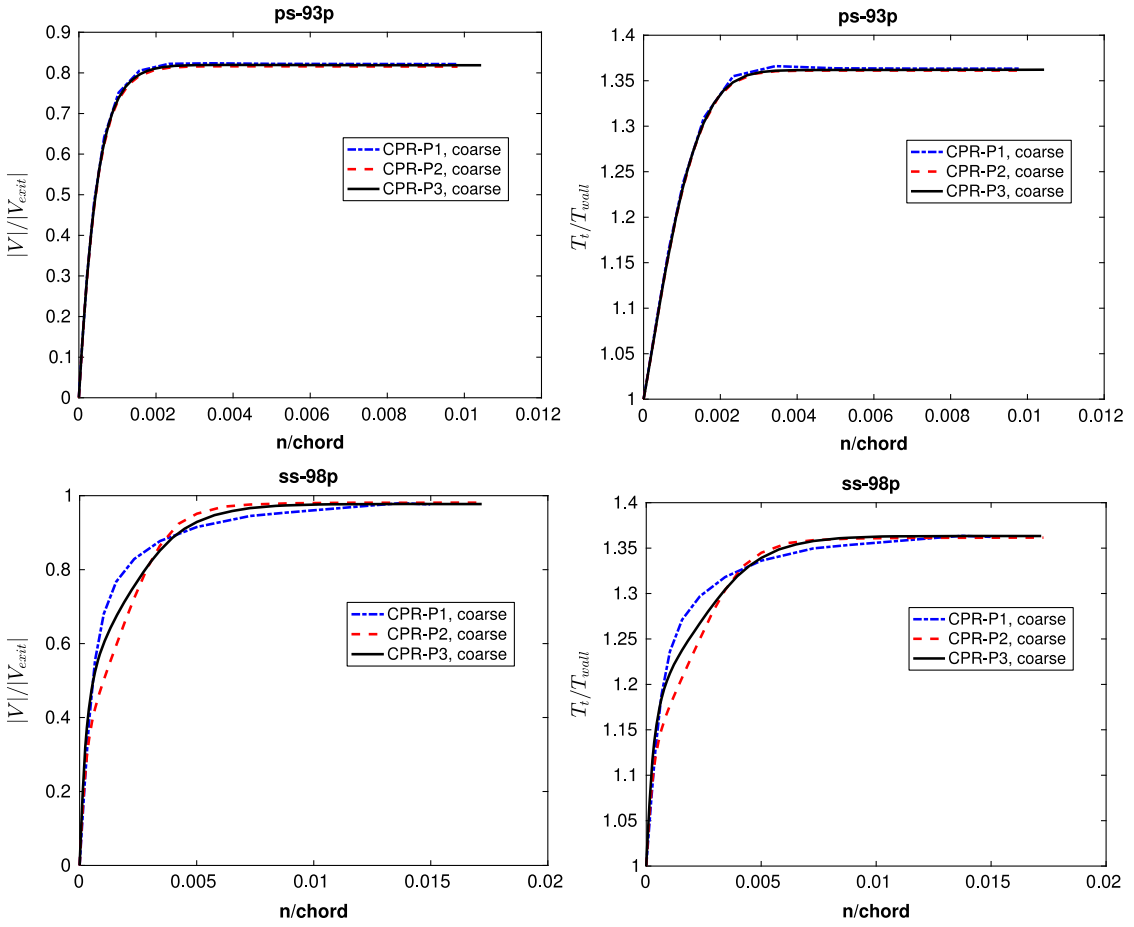


Fig. 14 Boundary-layer profiles computed with hpMusic.

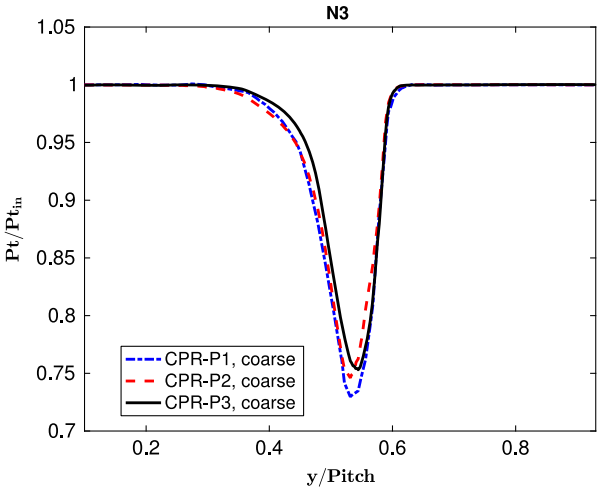


Fig. 15 Wake losses computed with hpMusic.

convergence. Also, the boundary-layer thicknesses are different between the medium and fine meshes. In addition, the velocity at the edge of the boundary layer of the fine mesh is more visibly different from those of the coarse and medium meshes. Therefore, more resolution is needed for the Fluent solver. The convergence of the other four stations is very similar to ps-93 point, and so we omit them here.

The wake losses computed with Fluent are plotted in Fig. 9. Note that the wake losses on these three meshes do not show any trend of convergence, and so more resolution is needed.

hpMusic simulations were conducted on the coarse mesh only with p refinement (i.e., for $p = 1, 2$, and 3, corresponding to second, third, and fourth orders of accuracy). Many internal DOFs were added in each element at different orders. A $p0$ (first-order) solution was obtained first on the coarse mesh to serve as the initial condition for the $p1$ simulation. We employed a BDF2 scheme for time integration. At each time step, the unsteady residual was reduced by at least two orders of magnitude. This level of convergence is verified in this paper. The nDOFs and time steps are listed in Table 4.

The drag coefficient history on the blade was used to monitor whether the simulation has reached a statistical steady state. The history of the $p1$ simulation is displayed in Fig. 10, which clearly indicates that the flow has reached a "statistical steady state" after six nondimensional time units. After that, time averaging was performed for several time units until the mean flow quantities were converged. For the present problem, spanwise averaging was employed to further reduce the length of the time-averaging duration. Similarly, the mean surface isentropic Mach number, skin friction coefficient, and heat transfer at different orders of accuracy are plotted in Fig. 11. There is a clear convergence trend toward the $p3$ result in all three surface quantities. The $p1$ simulation had an earlier transition, similar to the Fluent simulation. The $p2$ and $p3$ simulations agree very well almost everywhere except a slight difference in the transition location.

To assess the impact of time integration, we also performed a simulation using the CPR-P1 combined with the third-order Strong Stability-Preserving Runge–Kutta (SSP RK) scheme with a nondimensional time step of $1.41e-5$ ($3e-9s$), which is only 1.5% of the time step of the implicit one [47]. Therefore, the implicit solver is much faster than the explicit SSP RK solver. In Fig. 12, the PSDs of pressure at a monitor point labeled as "near wake" for both the explicit and implicit schemes are compared. Note that they show very good agreement, indicating that our second-order BDF implicit time solver with two orders of magnitude reduction of residual in each time step is accurate enough. Other monitor points show a similar agreement.

Next, the PSDs of pressure at two monitoring points are shown in Fig. 13. Note that the PSDs for $p2$ and $p3$ show good convergence, especially in the low-frequency range and the locations of the two dominant frequencies. Furthermore, the slopes of the PSD gradually converge to $-5/3$ from $p1$ to $p3$ on both monitor locations, indicating that the higher-order schemes have a better capability of resolving turbulence.

Next, we compare boundary-layer profiles at selected locations at both the pressure and suction sides. The profiles of velocity magnitude and total temperature at two stations are displayed in Fig. 14 for different orders of accuracy. The pressure-side station shows a very good convergence for both the velocity and temperature distributions. The suction-side station also does not show a clear convergence, but at least the thicknesses of boundary layer is nearly the same between $p2$ and $p3$. Also, the velocity and temperature at the edge of the boundary layer converges to the same value.

Finally, we compare the wake loss in Fig. 15. Note that the wake losses of $p2$ and $p3$ show a trend of convergence at the $N3$ station.

C. Comparison Between Fluent and hpMusic

When both solvers employ computational meshes in the same refinement family, it is very intriguing to compare their relative accuracy and efficiency. The Fluent simulation on the medium mesh has nearly the same number of total DOFs as hpMusic $p1$ simulation on the coarse mesh, and both are formally second-order-accurate. The Fluent solver uses a bounded central difference scheme with a SGS model, whereas the hpMusic $p1$ simulation employs a second-order FR/CPR scheme with a full upwind flux at cell interfaces without a SGS model (i.e., ILES). Both simulations predicted a transition location too early, indicating a lack of resolution. The PSDs at two wake points are compared in Fig. 16. The agreement is good over the entire frequency range, and the two dominant frequencies

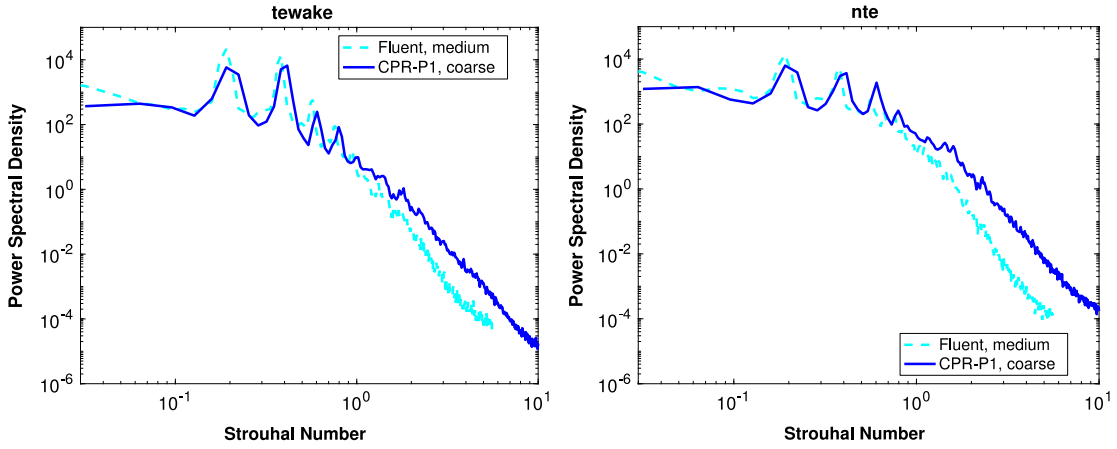


Fig. 16 Comparison of PSDs at two monitoring points near the trailing edge between Fluent and hpMusic.

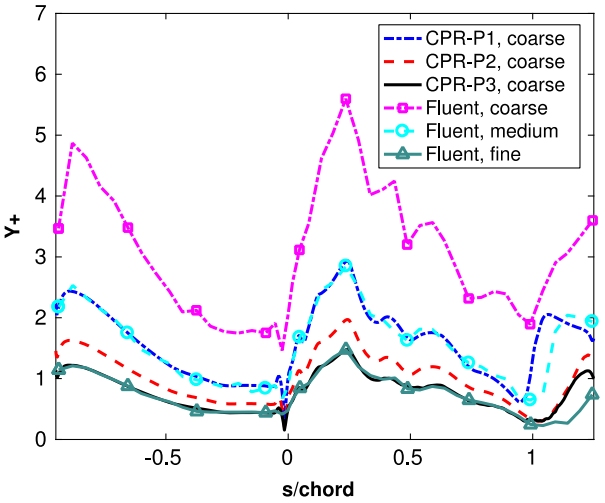
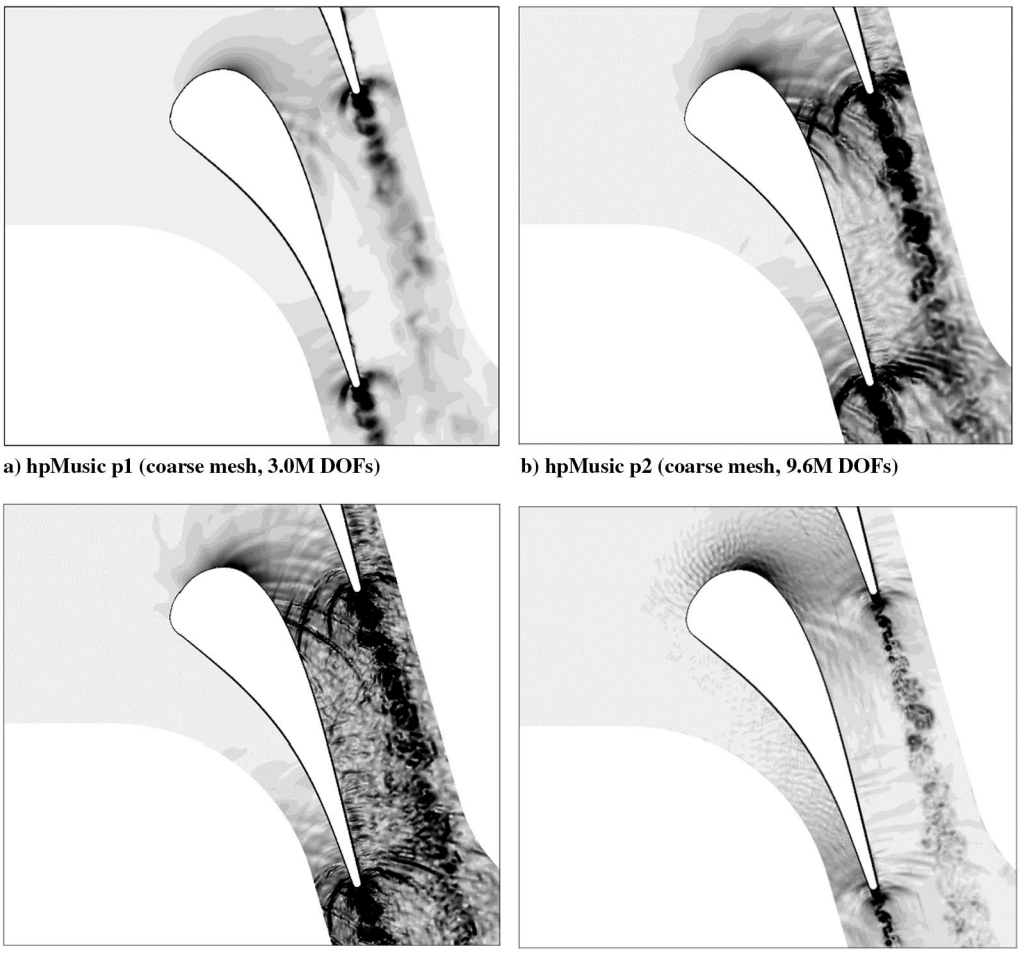


Fig. 17 y^+ distributions of both Fluent and hpMusic.

also agree well, indicating a similar accuracy between Fluent on the medium mesh and hpMusic at $p1$ on the coarse mesh.

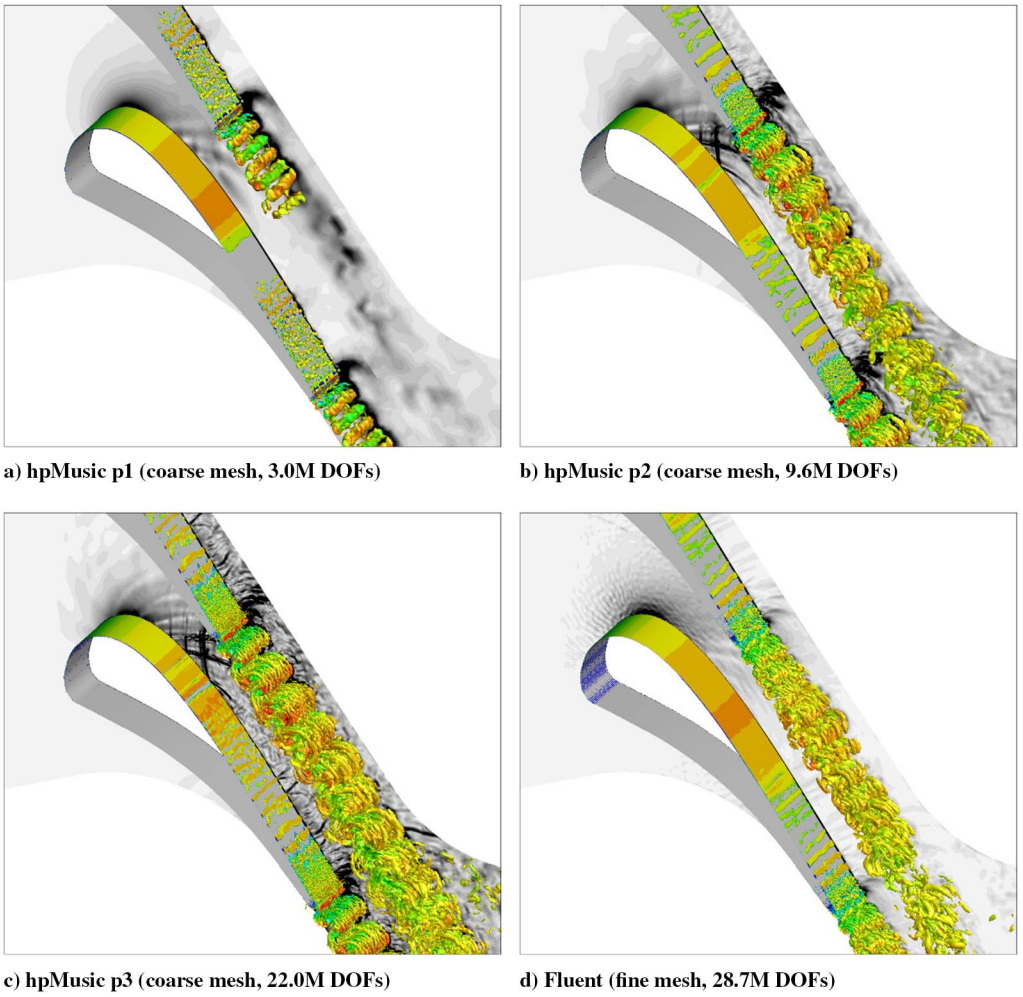
The cell–size-based y^+ distribution is shown in Fig. 17. It should be noted that hpMusic does not do any h refinement, and so for comparison purpose, y^+ for different orders of hpMusic is divided by the number of DOFs in the normal direction (e.g., $p1$ is divided by 2, $p2$ is divided by 3, and $p3$ is divided by 4). Note that the $p1$ result has a similar y^+ to Fluent on the medium mesh, and the $p3$ result has a similar y^+ to Fluent on the fine mesh. Next, the highest-resolution Fluent run on the fine mesh is compared to all of the hpMusic runs. The instantaneous schlieren distributions are displayed in Fig. 18. One observes that schlieren is a great tool to reveal the wakes, acoustic waves, and shock waves, as shown from this figure. We notice immediately the strikingly different resolutions between the hpMusic $p1$ and $p2$ results. The shock waves generated near the trailing edge are clearly seen in the $p2$ schlieren but smeared in the $p1$ schlieren, whereas the $p2$ and $p3$ results agree with each other very well. We also note that Fluent produced short-wave oscillations on both the suction side and pressure side near the trailing edge, perhaps due to a lack of short-wave damping in the numerical scheme. In addition, the shock waves are severely smeared in the Fluent schlieren. The instantaneous isosurfaces of Q-criteria colored by the velocity magnitude are compared between Fluent and hpMusic in Fig. 19. Again, the same message can be repeated. Finally, the mean isentropic Mach number and the heat transfer predictions with both Fluent and hpMusic are compared with experimental data in Fig. 20. Note that hpMusic is able to nail the transition location at $p3$, whereas Fluent predicted a transition too late. We also report the simulation time cost in Table 5. Fluent runs on 120 CPU cores (Intel Xeon CPU E5-2670 v3, 2.3 GHz), and hpMusic runs on 400 CPU cores (Intel Xeon CPU E5-2660, 2.2 GHz). All CPUs are equipped with infiniband interconnect network. Based on the CPU hours of explicit and implicit hpMusic $p1$ simulations, it is clear that the implicit time scheme is 10 times faster than the explicit one. From hpMusic $p1$ to $p2$, the time cost becomes more than seven times greater. From hpMusic $p2$ to $p3$, the time cost is about four times greater. It is shown that the hpMusic simulation at $p2$ costs about $1/3$ of the Fluent simulation on the fine mesh in CPU time.



a) hpMusic p1 (coarse mesh, 3.0M DOFs) **b) hpMusic p2 (coarse mesh, 9.6M DOFs)**

c) hpMusic p3 (coarse mesh, 22.0M DOFs) **d) Fluent (fine mesh, 28.7M DOFs)**

Fig. 18 Comparison of instantaneous schlieren ($|\nabla\rho|C/\rho$) distributions (20 levels between 0 and 3.38).



a) hpMusic p1 (coarse mesh, 3.0M DOFs)

b) hpMusic p2 (coarse mesh, 9.6M DOFs)

c) hpMusic p3 (coarse mesh, 22.0M DOFs)

d) Fluent (fine mesh, 28.7M DOFs)

Fig. 19 Comparison of instantaneous isosurfaces of Q-criterion (at 44.9) colored by velocity magnitude (20 levels from 0.0627 to 1.254) and schlieren.

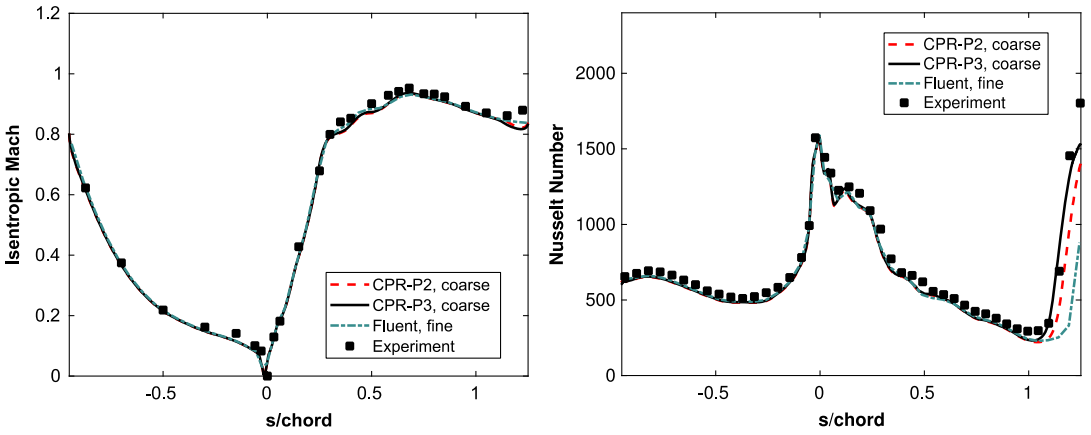


Fig. 20 Comparison of mean surface isentropic Mach number and heat transfer between Fluent and hpMusic predictions and experimental data.

Table 5 CPU hours needed for one nondimensional time unit

Solver	Mesh	Time scheme	nDOFs	CPU hours
hpMusic p1	Coarse	BDF2–LU-SGS	3,028,550	116
hpMusic p1	Coarse	RK3	3,028,550	1200
hpMusic p2	Coarse	BDF2–LU-SGS	9,594,900	810
hpMusic p3	Coarse	BDF2–LU-SGS	22,001,000	3300
Fluent	Fine	Bounded second-order implicit	28,683,200	2400

V. Conclusions

The performances of a second-order commercial solver (Fluent) and a high-order flux reconstruction /correction procedure via reconstruction solver (hpMusic) in conducting wall-resolved large-eddy simulation (LES) using mixed unstructured meshes for a benchmark turbomachinery problem at a Reynolds number over 1 million were evaluated. A wide variety of flow parameters are compared between the two solvers with up to ~29 million degrees of freedom per equation. The Fluent simulations failed to demonstrate

mesh convergence, indicating that these levels of resolution are still not enough for LES, whereas hpMusic clearly demonstrated p convergence in the simulations. Both the $p2$ and $p3$ simulations captured the transition location very well. In particular, the PSDs of pressure computed with $p3$ clearly match the $-5/3$ law. This study has conclusively demonstrated the advantage of high-order methods in industrial LES of achieving a higher accuracy at a reduced cost. The $p2$ simulation on the coarse mesh costs about $1/3$ of the Fluent simulation on the fine mesh ($64\times$ the number of cells) but produced much more accurate results.

Acknowledgments

The research outlined in the present paper has been supported by the U.S. Air Force Office of Scientific Research under grant FA9550-16-1-0128, the U.S. Army Research Office under grant W911NF-15-1-0505, and GE Global Research.

References

[1] Slotnick, J., Khodadoust, A., Alonso, J., Darmofal, D., Gropp, W., Lurie, E., and Mavriplis, D., "CFD Vision 2030 Study: A Path to Revolutionary Computational Aerosciences," NASA CR-2014-218178, 2014.

[2] Smagorinsky, J., "General Circulation Experiments with the Primitive Equations: 1. The Basic Experiment," *Monthly Weather Review*, Vol. 91, No. 3, 1963, pp. 99–164. doi:10.1175/1520-0493(1963)091<0099:GCEWTP>2.3.CO;2

[3] Laskowski, G. M., Kopriva, J., Michelassi, V., Shankaran, S., Paliath, U., Bhaskaran, R., Wang, Q., Talnikar, C., Wang, Z. J., and Jia, F., "Future Directions of High Fidelity CFD for Aerothermal Turbomachinery Analysis and Design," *46th AIAA Fluid Dynamics Conference*, AIAA Paper 2016-3322, 2016.

[4] Kopriva, J. E., Laskowski, G. M., and Sheikhi, M. R. H., "Hybrid LES of a High Pressure Turbine Nozzle/Blade Interaction," *Direct and Large-Eddy Simulation X*, edited by D. Grigoriadis, B. Geurts, H. Kuerten, J. Frohlich, and V. Armenio, Vol. 24, ERCOFTAC Series, Springer, Cham, 2018, pp. 273–279.

[5] Kopriva, J. E., Laskowski, G. M., and Shiekhi, M. R. H., "Computational Assessment of Inlet Turbulence on Boundary Layer Development and Momentum/Thermal Wakes for High Pressure Turbine Nozzle and Blade," *ASME International Mechanical Engineering Congress and Exposition*, American Soc. of Mechanical Engineers (ASME) Paper IMECE2014-38620,, Montreal, 2014.

[6] Germano, M., Piomelli, U., Moin, P., and Cabot, WH., "A Dynamic Subgrid Scale Eddy Viscosity Model," *Physics of Fluids A*, Vol. 3, No. 7, 1991, pp. 1760–1765. doi:10.1063/1.857955

[7] Boris, J. P., Grinstein, F. F., Oran, E. S., and Kolbe, R. L., "New Insights into Large Eddy Simulation," *Fluid Dynamics Research*, Vol. 10, Nos. 4–6, 1992, pp. 199–228. doi:10.1016/0169-5983(92)90023-P

[8] Li, Y., and Wang, Z. J., "A Priori and a Posteriori Evaluations of Sub-Grid Scale Models for the Burgers' Equation," *Computers & Fluids*, Vol. 139, No. 5, 2016, pp. 92–104. doi:10.1016/j.compfluid.2016.04.015

[9] Beck, A. D., Bolemann, T., Flad, D., Frank, H., Gassner, G. J., Hindenlang, F., and Munz, C. D., "High Order Discontinuous Galerkin Spectral Element Methods for Transitional and Turbulent Flow Simulations," *International Journal for Numerical Methods in Fluids*, Vol. 76, No. 8, 2014, pp. 522–548. doi:10.1002/fld.v76.8

[10] Wang, Z. J., "High-Order Methods on Unstructured Grids for Navier–Stokes Equations," *Journal of Progress in Aerospace Sciences*, Vol. 43, Nos. 1–3, 2007, pp. 1–41. doi:10.1016/j.paerosci.2007.05.001

[11] Wang, Z. J., Fidkowski, K., Abgrall, R., Bassi, F., Caraeni, D., Cary, A., and Kroll, N., "High-Order CFD Methods: Current Status and Perspective," *International Journal for Numerical Methods in Fluids*, Vol. 72, No. 8, 2013, pp. 811–845. doi:10.1002/fld.v72.8

[12] Georgiadis, N., Yoder, D., and DeBonis, J., "A Comparison of Three Navier–Stokes Solvers for Exhaust Nozzle Flowfields," *37th Aerospace Sciences Meeting and Exhibit*, AIAA Paper 1999-0748, 1999.

[13] Gaffuri, M., Brezillon, J., Kwak, D. Y., Ohira, K., and Carrier, G., "Comparison of CFD Solvers for Low Speed Vortex Dominated Flows," *31st AIAA Applied Aerodynamics Conference*, AIAA Paper 2013-2914, 2013.

[14] Sørensen, N. N., Méndez, B., Muñoz, A., Sieros, G., Jost, E., Lutz, T., and Baldacchino, D., "CFD Code Comparison for 2D Airfoil Flows," *Journal of Physics: Conference Series*, Vol. 753, No. 8, 2016, Paper 082019. doi:10.1088/1742-6596/753/8/082019

[15] Abras, J., and Hariharan, N. S., "CFD Solver Comparison of Low Mach Number Flow over the ROBIN Fuselage," *52nd Aerospace Sciences Meeting*, AIAA Paper 2014-0752, 2014.

[16] Abras, J. N., and Hariharan, N., "Comparison of Computational Fluid Dynamics Hover Predictions on the S-76 Rotor," *Journal of Aircraft*, Vol. 55, No. 1, 2018, pp. 12–22. doi:10.2514/1.C034121

[17] Brazell, M. J., Mavriplis, D. J., and Yang, Z., "Mesh-Resolved Airfoil Simulations Using Finite Volume and Discontinuous Galerkin Solvers," *AIAA Journal*, Vol. 54, No. 9, 2016, pp. 2659–2670. doi:10.2514/1.J054534

[18] Gan, J., Shen, Y., and Zha, G., "Comparison of Drag Prediction Using RANS Models and DDES for the DLR-F6 Configuration Using High Order Schemes," *54th AIAA Aerospace Sciences Meeting*, AIAA Paper 2016-0553, 2016.

[19] Antoniadis, A. F., Tsoutsanis, P., and Drikakis, D., "Assessment of High-Order Finite Volume Methods on Unstructured Meshes for RANS Solutions of Aeronautical Configurations," *Computers & Fluids*, Vol. 146, March 2017, pp. 86–104. doi:10.1016/j.compfluid.2017.01.002

[20] Brazell, M. J., Ahrabi, B. R., and Mavriplis, D. J., "Discontinuous Galerkin Turbulent Flow Simulations of NASA Turbulence Model Validation Cases and High Lift Prediction Workshop Test Case DLR-F11," *54th AIAA Aerospace Sciences Meeting*, AIAA Paper 2016-0861, 2016.

[21] Ahrabi, B. R., Brazell, M. J., and Mavriplis, D. J., "An Investigation of Continuous and Discontinuous Finite-Element Discretizations on Benchmark 3D Turbulent Flows (Invited)," *2018 AIAA Aerospace Sciences Meeting*, AIAA Paper 2018-1569, 2018.

[22] Shi, L., and Wang, Z. J., "Adjoint-Based Error Estimation and Mesh Adaptation for the Correction Procedure via Reconstruction Method," *Journal of Computational Physics*, Vol. 295, No. 15, 2015, pp. 261–284. doi:10.1016/j.jcp.2015.04.011

[23] Huynh, H. T., "A Flux Reconstruction Approach to High Order Schemes Including Discontinuous Galerkin Methods," *18th AIAA Computational Fluid Dynamics Conference*, AIAA Paper 2007-4079, June 2007.

[24] Kopriva, J. E., "The Role of Free-Stream Turbulence on High Pressure Turbine Aero-Thermal Stage Interaction," Ph.D. Dissertation, Dept. of Mechanical and Industrial Engineering, Northeastern Univ., Boston, MA, 2017.

[25] Wang, Z. J., Li, Y., Jia, F., Laskowski, G. M., Kopriva, J., Paliath, U., and Bhaskaran, R., "Towards Industrial Large Eddy Simulation Using the FR/CPR Method," *Computers & Fluids*, Vol. 156, No. 12, 2017, pp. 579–589. doi:10.1016/j.compfluid.2017.04.026

[26] Visbal, M. R., and Gaitonde, D. V., "On the Use of Higher-Order Finite-Difference Schemes on Curvilinear and Deforming Meshes," *Journal of Computational Physics*, Vol. 181, No. 1, 2002, pp. 155–185. doi:10.1006/jcph.2002.7117

[27] Haga, T., Gao, H., and Wang, Z. J., "A High Order Unifying Discontinuous Formulation for the Navier–Stokes Equations on 3D Mixed Grids," *Mathematical Modelling of Natural Phenomena*, Vol. 6, No. 3, 2011, pp. 28–56. doi:10.1051/mmnp/20116302

[28] Wang, Z. J., and Gao, H., "A Unifying Lifting Collocation Penalty Formulation Including the Discontinuous Galerkin, Spectral Volume/Difference Methods for Conservation Laws on Mixed Grids," *Journal of Computational Physics*, Vol. 228, No. 21, 2009, pp. 8161–8186. doi:10.1016/j.jcp.2009.07.036

[29] Huynh, H. T., Wang, Z. J., and Vincent, P. E., "High-Order Methods for Computational Fluid Dynamics: A Brief Review of Compact Differential Formulations on Unstructured Grids," *Computers & Fluids*, Vol. 98, No. 2, 2014, pp. 209–220. doi:10.1016/j.compfluid.2013.12.007

[30] Park, J. S., You, H., and Kim, C., "Higher-Order Multi-Dimensional Limiting Process for DG and FR/CPR Methods on Tetrahedral Meshes," *Computers & Fluids*, Vol. 154, No. 1, 2017, pp. 322–334. doi:10.1016/j.compfluid.2017.03.006

[31] Vermeire, B. C., Witherden, F. D., and Vincent, P. E., "On the Utility of GPU Accelerated High-Order Methods for Unsteady Flow Simulations: A Comparison with Industry-Standard Tools," *Journal of Computational Physics*, Vol. 334, No. 1, 2017, pp. 497–521. doi:10.1016/j.jcp.2016.12.049

[32] Wang, Z. J., "A Perspective on High-Order Methods in Computational Fluid Dynamics," *Science China Physics, Mechanics & Astronomy*, Vol. 59, No. 1, 2016, Paper 614701.

[33] Bassi, F., and Rebay, S., "A High Order Discontinuous Galerkin Method for Compressible Turbulent Flows," *Discontinuous Galerkin Methods*, edited by B. Cockburn, G. E. Karniadakis, and C. W. Shu, Vol. 11, Springer, Berlin, 2000, pp. 77–88.

[34] Cockburn, B., Karniadakis, G., and Shu, C. W., *Discontinuous Galerkin Methods: Theory, Computation and Applications*, Springer–Verlag, Berlin, 2000, pp. 3–50.

[35] Liu, Y., Vinokur, M., and Wang, Z., "Discontinuous Spectral Difference Method for Conservation Laws on Unstructured Grids," *Computational Fluid Dynamics 2004*, edited by C. Groth, and D. W. Zingg, Springer, Berlin, 2006, pp. 449–454.

[36] Vincent, P. E., Castonguay, P., and Jameson, A., "A New Class of High-Order Energy Stable Flux Reconstruction Schemes," *Journal of Scientific Computing*, Vol. 47, No. 1, 2011, pp. 50–72. doi:10.1007/s10915-010-9420-z

[37] Sun, Y., Wang, Z. J., and Liu, Y., "Efficient Implicit Non-Linear LU-SGS Approach for Viscous Flow Computation Using High-Order Spectral Difference Method," *18th AIAA Computational Fluid Dynamics Conference*, AIAA Paper 2007-4322, June 2007.

[38] Grinstein, F., Margolin, L., and Rider, W., *Implicit Large Eddy Simulation: Computing Turbulent Fluid Dynamics*, Cambridge Univ. Press, Cambridge, England, U.K., 2007, pp. 39–58.

[39] Vermeire, B. C., Nadarajah, S., and Tucker, P. G., "Implicit Large Eddy Simulation Using the High-Order Correction Procedure via Reconstruction Scheme," *International Journal for Numerical Methods in Fluids*, Vol. 82, No. 5, 2016, pp. 231–260. doi:10.1002/fld.v82.5

[40] Li, Y., and Wang, Z. J., "A Convergent and Accuracy Preserving Limiter for the FR/CPR Method," *55th AIAA Aerospace Sciences Meeting*, AIAA Paper 2017-0756, 2017.

[41] Arts, T., and Rouvroit, M., "Aero-Thermal Performance of a Two-Dimensional Highly Loaded Transonic Turbine Nozzle Guide Vane: A Test Case for Inviscid and Viscous Flow Computations," *Journal of Turbomachinery*, Vol. 114, No. 1, 1992, pp. 147–154. doi:10.1115/1.2927978

[42] Wheeler, A. P., Sandberg, R. D., Sandham, N. D., Pichler, R., Michelassi, V., and Laskowski, G., "Direct Numerical Simulations of a High Pressure Turbine Vane," *Journal of Turbomachinery*, Vol. 138, No. 7, 2016, Paper 071003. doi:10.1115/1.4032435

[43] Ims, J., Duan, Z., and Wang, Z. J., "MeshCurve: An Automated Low-Order to High Order Mesh Generator," *22nd AIAA Computational Fluid Dynamics Conference*, AIAA Paper 2015-2293, 2015.

[44] Jiao, X., and Bayyanan, N. R., "Identification of C1 and C2 Discontinuities for Surface Meshes in CAD," *Computer-Aided Design*, Vol. 40, No. 2, 2008, pp. 160–175. doi:10.1016/j.cad.2007.10.005

[45] Jiao, X., and Wang, D., "Reconstructing High Order Surfaces for Meshing," *Engineering with Computers*, Vol. 28, No. 4, 2012, pp. 361–373. doi:10.1007/s00366-011-0244-8

[46] Luke, E., Collins, E., and Blades, E., "A Fast Mesh Deformation Method Using Explicit Interpolation," *Journal of Computational Physics*, Vol. 231, No. 2, 2012, pp. 586–601. doi:10.1016/j.jcp.2011.09.021

[47] Gottlieb, S., and Shu, C. W., "Total Variation Diminishing Runge–Kutta Schemes," *Mathematics of Computation of the American Mathematical Society*, Vol. 67, No. 221, 1998, pp. 73–85. doi:10.1090/mcom/1998-67-221

C. Bailly
Associate Editor

Accuracy, Efficiency and Scalability of Explicit and Implicit FR/CPR Schemes in Large Eddy Simulation

Feilin Jia[1] and Z.J. Wang[2]

Department of Aerospace Engineering, University of Kansas, Lawrence, KS, 66045

Rathakrishnan Bhaskaran[3] and Umesh Paliath[4]

GE Global Research, Niskayuna, NY, 12309

Gregory M. Laskowski[5]

GE Aviation, Lynn, MA, 01905

The accuracy, efficiency and scalability of explicit and implicit temporal schemes associated with FR/CPR spatial schemes are studied in the context of large eddy simulation. One low pressure turbine and one high pressure turbine blade cases with different Mach numbers and Reynolds numbers are employed in this study. The 3 stage SSP Runge-Kutta (RK) scheme is compared with implicit backward difference formulas (BDF) of first and second order accuracy with a non-linear LU-SGS solver in the present evaluation. Various factors such as inner convergence tolerance, the frequency of the implicit operator update, and the order of time accuracy are investigated for large eddy simulation. The implicit BDF2-LUSGS algorithm can achieve good time accuracy, and is more efficient than the 3rd order explicit RK scheme, but not as scalable on a small cluster. At a very low Mach number, the explicit scheme is clearly not adequate, suggesting the need for pre-conditioning and/or an implicit scheme.

Keywords: Large eddy simulation; Implicit time integration; Accuracy; Efficiency; Scalability

[1] PhD Student, Department of Aerospace Engineering, 2120 Learned Hall.
[2] Spahr Professor, Department of Aerospace Engineering, 2120 Learned Hall, Fellow of AIAA.
[3] Lead Engineer, Aero Systems Organization, AIAA Senior Member.
[4] Senior Engineer, Aero Systems Organization, AIAA Member.
[5] Principal Engineer high-fidelity CFD, Advanced Design Tools, AIAA Member.

I. Introduction

Large Eddy Simulation (LES) [1] was proposed to compute turbulent flows over fifty years ago. After decades of development, LES is starting to move from being an analysis tool to the design process [2] in limited context where the accuracy is needed and the cost can be justified. In terms of cost and accuracy, LES [3,4] lies between the Reynolds Averaged Navier-Stokes (RANS) approach, where all scales are modeled and the Direct Numerical Simulation (DNS), where all scales are captured. Although RANS models have been effective for many practical problems, they may be inadequate for certain flows such as massively separated flows because a statistically steady mean flow may not exist for such problems, or the lack of accurate turbulence models. The use of DNS in computing high Reynolds number flows is, for the foreseeable future, limited by computing resources because of the disparate time and length scales. LES offers the best promise for vortex dominated flows found in many aerospace applications such as flow over high lift configurations, rotorcraft flows and more recently flows in aircraft engines [2,5-10].

Although 1st and 2nd order methods have been used in LES for relatively low Reynolds number flow problems, higher-order methods offer much more promise because of their better resolution for smaller scales given the same computational resources [11,12]. Recently, adaptive high-order methods such as discontinuous Galerkin (DG) [13,14], spectral difference (SD) [15] and flux reconstruction (FR) or correction procedure via reconstruction (CPR) methods [16] have shown much potential in LES with complex configurations. Impressive results have appeared in the literature on either CPU and GPU clusters, or a mixture of both, e.g., see the work from many research groups [17-22]. These methods are capable of handling unstructured meshes, and are therefore more suitable for industrial applications where complex geometries must be dealt with in an efficient manner.

It is well-known that implicit methods are essential in steady RANS simulations because of their much-reduced computational cost to reach convergence [11]. For LES, the choice between explicit and implicit schemes is not that clear-cut any more for several reasons. First, there is an upper limit on the time step size for LES because of the necessity to capture the dynamics of eddies of various sizes. Although implicit schemes allow a much larger time step than their explicit counterparts, they normally cost much more CPU time per time step. Second, on extreme-scale massively parallel computers with millions of cores, it is trivial to parallelize explicit schemes, while the efficiency of implicit schemes may be degraded greatly because of a lack of scalable and effective preconditioner or solvers. Third, implicit schemes require much more memory than explicit schemes, and the memory requirement

scales with p^6, with p being the order of the solution polynomial. Modern computer architectures such as GPU cards are severely limited by the amount of global memory size, making implicit schemes impossible to fit on such architectures. For a p5 FR/CPR or DG scheme, the implicit matrices for one hexahedral element take roughly 65 MB (in double precision). A 6GB GPU card can only store the implicit matrices of about ~90 elements. It is no-wonder that most LES with high-order discontinuous methods are performed with explicit RK type schemes.

In the present study, we attempt to provide an evaluation comparing the performance of explicit and implicit schemes in the context of implicit LES (ILES) [23,24] for both low and moderate Mach and Reynolds number flow problems. More specifically, we compare the performance of the backward Euler or BDF1 and BDF2 implicit schemes with a non-linear block LU-SGS (BLU-SGS) solver [25-27] with the explicit 3-stage Runge-Kutta scheme [28]. The reason we have chosen the BLU-SGS scheme is because of its relatively low memory requirement since only the main diagonal matrices are stored, reducing the memory by a factor of 7 for hexahedral meshes comparing with a fully implicit scheme.

The paper is organized as follows. In Section 2, we briefly review the spatial discretization method, the FR/CPR framework. In Section 3, we describe both the explicit and implicit time integration approaches. In Section 4, a low Reynolds number benchmark problem is considered firstly, and a detailed evaluation of accuracy, efficiency, and scalability is performed together with discussions. Then a high Reynolds number problem is also investigated. Finally we conclude the paper with several possible future research directions.

II. A Brief Overview of the FR/CPR Method

The FR/CPR method was originally developed by Huynh [16] in 2007 for hyperbolic partial differential equations, and later it was extended to hybrid unstructured meshes [29,30]. Further developments on the FR/CPR method are reviewed in [31,32]. This method belongs to discontinuous finite element methods, similar to the DG method, but also has some unique advantages. For example, FR contains a larger family of schemes [16], which may allow larger time steps than the DG method [33]. Many groups also reported that FR is more efficient than the DG method [34,35]. Here we present a brief introduction of the FR/CPR method starting from a hyperbolic conservation law

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{U}) = 0, \tag{1}$$

with initial and boundary conditions, where the vector **U** consists of conservative variables, and **F** is the flux. By discretizing the computational domain with non-overlapping elements, and introducing an arbitrary test function W in each element, the weighted residual formulation of Eq. (1) on element V_i can be expressed as

$$\int_{V_i} \left[\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{U})\right] W \mathrm{d}\Omega = 0. \tag{2}$$

The conservative variables inside one element are assumed to be polynomials, and expressed by nodal values at certain points called solution points (SPs). After applying integration by parts to the divergence of flux, replacing the normal flux term with a common Riemann flux F_{com}^n and integrating back by parts, we obtain

$$\int_{V_i} \frac{\partial \mathbf{U}_i}{\partial t} W \mathrm{d}\Omega + \int_{V_i} W \nabla \cdot \mathbf{F}(\mathbf{U}_i)\, \mathrm{d}\Omega + \int_{\partial V_i} W [F_{com}^n - F^n(\mathbf{U}_i)]\, \mathrm{d}S = 0. \tag{3}$$

Here, the common Riemann flux is computed with a Riemann solver

$$F_{com}^n = F_{com}^n(\mathbf{U}_i, \mathbf{U}_{i+}, \mathbf{n}), \tag{4}$$

where $\mathbf{U}_{i+}$ stands for the solution outside the current element, and n denotes the outward normal direction of the interface. The normal flux at the interface is:

$$F^n(\mathbf{U}_i) = \mathbf{F}(\mathbf{U}_i) \cdot \mathbf{n}. \tag{5}$$

Note that if the face integral in Eq. (3) can be transformed into an element integral then the test function will be eliminated. In order to do so, a "correction field" δ_i is defined in each element as

$$\int_{V_i} W \delta_i \, \mathrm{d}\Omega = \int_{\partial V_i} W [F^n]\, \mathrm{d}S = 0, \tag{6}$$

where $[F^n] = [F_{com}^n - F^n(\mathbf{U}_i)]$ is the normal flux jump. Eqs. (3) and (6) result in

$$\int_{V_i} \left[\frac{\partial \mathbf{U}_i}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{U}_i) + \delta_i\right] W \mathrm{d}\Omega = 0. \tag{7}$$

The final formulation for each solution point *j* is

$$\frac{\partial \mathbf{U}_{i,j}}{\partial t} + \Pi_j [\nabla \cdot \mathbf{F}(\mathbf{U}_i)] + \delta_{i,j} = 0. \tag{8}$$

where Π_j denotes a projection to the polynomial space, and subscript j denotes the j-th solution point in a certain element.

For viscous flux involving the gradient of conservative variables, we use the Bassi-Rebay 2 (BR2) scheme [14].

III. Time Integration Algorithms

III.1 3-Stage SSP Runge Kutta Scheme

The explicit scheme used in this study is the 3-stage SSP RK scheme [28], which has become a workhorse time integration approach for many LES codes. Let the semi-discretized equation after the spatial discretization be written as

$$\frac{\mathrm{d}\tilde{Q}}{\mathrm{dt}} = R(\tilde{Q}), \tag{9}$$

where $\tilde{Q}$ is the global degrees of freedom (DOFs). Given the solution at time level n $\tilde{Q}^{\boldsymbol{n}}$, the 3-stage SSP RK scheme can be written as

$$\tilde{Q}^{(1)} = \tilde{Q}^{n} + \Delta t R(\tilde{Q}^{\mathrm{n}}), \tag{10}$$

$$\tilde{Q}^{(2)} = \tfrac{3}{4}\tilde{Q}^{n} + \tfrac{1}{4}[\tilde{Q}^{(1)} + \Delta t R(\tilde{Q}^{(1)})], \tag{11}$$

$$\tilde{Q}^{\boldsymbol{n+1}} = \tfrac{1}{3}\tilde{Q}^{n} + \tfrac{2}{3}[\tilde{Q}^{(2)} + \Delta t R(\tilde{Q}^{(2)})] \tag{12}$$

III.2 Implicit LU-SGS Scheme

At each cell *c*, using the backward Euler difference, Eq. (1) can be written as

$$\frac{\widetilde{Q}_c^{n+1} - \widetilde{Q}_c^{n}}{\Delta t} - \left[R_c(\widetilde{Q}^{n+1}) - R_c(\widetilde{Q}^{n})\right] = R_c(\widetilde{Q}^{n}) \tag{13}$$

Let $\Delta\widetilde{Q}_c = \widetilde{Q}_c^{n+1} - \widetilde{Q}_c^{n}$ and linearizing the residual, we obtain

$$R_c(\widetilde{Q}^{n+1}) - R_c(\widetilde{Q}^{n}) \approx \frac{\partial R_c}{\partial \widetilde{Q}_c}\Delta\widetilde{Q}_c + \sum_{nb\neq c}\frac{\partial R_c}{\partial \widetilde{Q}_{nb}}\Delta\widetilde{Q}_{nb}, \tag{14}$$

where subscript *nb* indicates all the neighboring cells contributing to the residual of cell *c*. Therefore, the fully linearized equations for (13) can be written as

$$\left(\frac{I}{\Delta t} - \frac{\partial R_c}{\partial \widetilde{Q}_c}\right)\Delta\widetilde{Q}_c - \sum_{nb\neq c}\frac{\partial R_c}{\partial \widetilde{Q}_{nb}}\Delta\widetilde{Q}_{nb} = R_c(\widetilde{Q}^{n}). \tag{15}$$

However, it is expensive in memory to store the full LHS implicit Jacobian matrices. Therefore, we employ a preconditioned LU-SGS scheme to solve Eq. (15). The contributions from the neighboring cells are included in the right hand side, i.e,

$$\left(\frac{I}{\Delta t}-\frac{\partial R_c}{\partial \widetilde{Q}_c}\right)\Delta\widetilde{Q}_c^{(k+1)} = R_c(\widetilde{Q}^n)+\sum_{nb\neq c}\frac{\partial R_c}{\partial \widetilde{Q}_{nb}}\Delta\widetilde{Q}_{nb}^{*}\,, \tag{16}$$

where superscript (k+1) is an iteration index, and superscript * indicates the most recently updated solutions. The matrix

$$D=\left(\frac{I}{\Delta t}-\frac{\partial R_c}{\partial \widetilde{Q}_c}\right) \tag{17}$$

is the element (or cell) matrix, which serves as the preconditioning matrix. Eq. (16) is then solved with a direct LU decomposition solver. Since we do not want to store the matrices $\frac{\partial R_c}{\partial \widetilde{Q}_{nb}}$, (16) is further manipulated as follows. Note that

$$\begin{aligned} &R_c(\widetilde{Q}^n)+\sum_{nb\neq c}\frac{\partial R_c}{\partial \widetilde{Q}_{nb}}\Delta\widetilde{Q}_{nb}^{*} = R_c(\widetilde{Q}_c^n,\{\widetilde{Q}_{nb}^n\})+\sum_{nb\neq c}\frac{\partial R_c}{\partial \widetilde{Q}_{nb}}\Delta\widetilde{Q}_{nb}^{*} \\ &\approx R_c(\widetilde{Q}_c^n,\{\widetilde{Q}_{nb}^{*}\}) \approx R_c(\widetilde{Q}_c^{*},\{\widetilde{Q}_{nb}^{*}\})-\frac{\partial R_c}{\partial \widetilde{Q}_c}\Delta\widetilde{Q}_c^{*} = R_c(\widetilde{Q}^{*})-\frac{\partial R_c}{\partial \widetilde{Q}_c}\Delta\widetilde{Q}_c^{*}. \end{aligned} \tag{18}$$

In Eq. (18), note that both *approximations* can be obtained using a first-order Taylor series expansion. Combining (16) and (18) together, we obtain

$$\left(\frac{I}{\Delta t}-\frac{\partial R_c}{\partial \widetilde{Q}_c}\right)(\widetilde{Q}_c^{(k+1)}-\widetilde{Q}_c^{(k)}) = R_c(\widetilde{Q}^{*})-\frac{\Delta\widetilde{Q}_c^{*}}{\Delta t}\,. \tag{19}$$

Eq. (19) is then solved with symmetric forward and backward sweeps. Note that once Eq. (19) is solved to machine zero, the unsteady residual is zero at each time step. For steady state problems, the last term in Eq. (19) can often be dropped resulting in faster convergence rate.

Because of the way in which the viscous fluxes are computed, the present FR/CPR method also uses cells, which are neighbors' neighbors. If the analytical approach is used to compute the element Jacobian matrix $\frac{\partial R_c}{\partial \widetilde{Q}_c}$, the formulation would be complex. Instead, the following numerical approach is used based on the definition

$$\frac{\partial R_c}{\partial \widetilde{Q}_c} \approx \frac{R_c(\{\widetilde{Q}_{nb}\}, \widetilde{Q}_c + \varepsilon) - R_c(\{\widetilde{Q}_{nb}\}, \widetilde{Q}_c)}{\varepsilon}, \tag{20}$$

where ε is a small parameter, e.g., $\varepsilon \approx \left\| \widetilde{Q}_c \right\| \times 10^{-8}$. Although this approach is very easy to implement for arbitrarily complex residual operators, it is quite expensive because each variable has to be changed, and the flux be computed. In practice, we have found it unnecessary to compute the matrix at each iteration. Therefore, we often re-compute the matrix every 40-100 iterations. Numerical tests showed that this matrix-freezing approach did not significantly degrade the convergence rate.

IV. Performance Study of Flow over Turbine Cascades

IV.1 Low Reynolds number flow over a low pressure turbine

The first case we selected is a benchmark problem of flow over a T106A low-pressure turbine cascade from the 4th International Workshop on High-Order Methods (https://how4.cenaero.be/) with the mesh provided by the workshop. The p2 mesh is shown in Figure 1, which has 14,035 p2 hexahedral elements with 5 elements in the spanwise direction. The isentropic Mach number at the exit is 0.4 and the Reynolds number is 60,000 based on the chord and exit condition. The span is 10% of the chord. The incoming flow angle of attack is 46.1° with respect to the positive x direction with no incoming turbulence. To show the capability of the high-order method, we present results from a p-refinement study first. For this purpose, we use the 3-stage SSP RK approach with FR/CPR schemes of p = 2 to p = 4 (3rd to 5th order), resulting in 379K, 898K, 1,754K DOFs per equation. The average y+ value at the first element is around 4.9 based on the cell size in the wall normal direction, and its "equivalent y+" values are 1.6, 1.2, and 0.98 for p = 2, 3, 4 respectively.

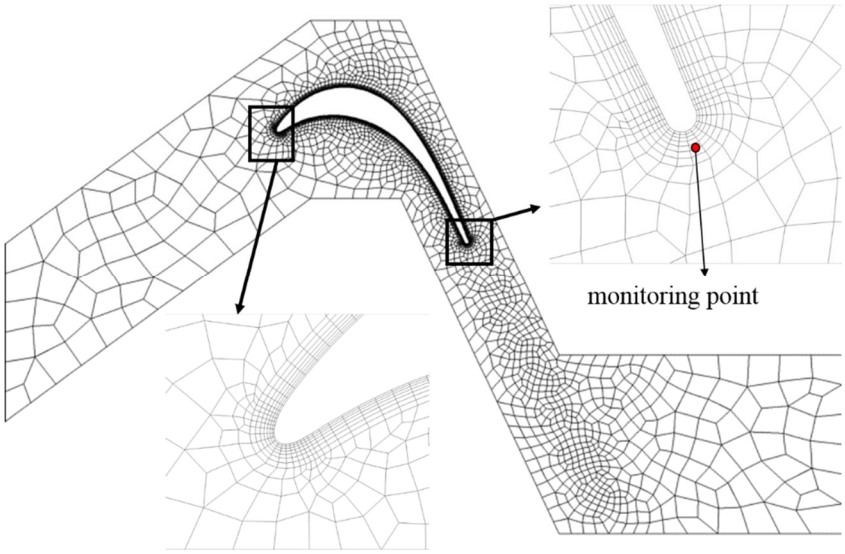


Figure 1. Unstructured hexahedron mesh used for T106A case.

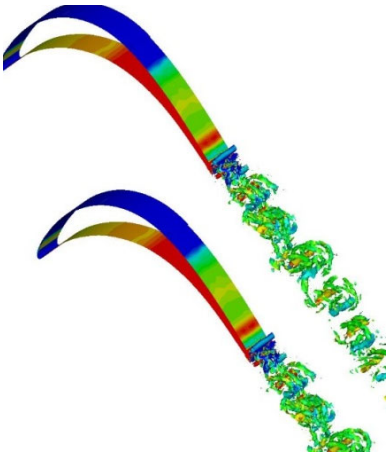
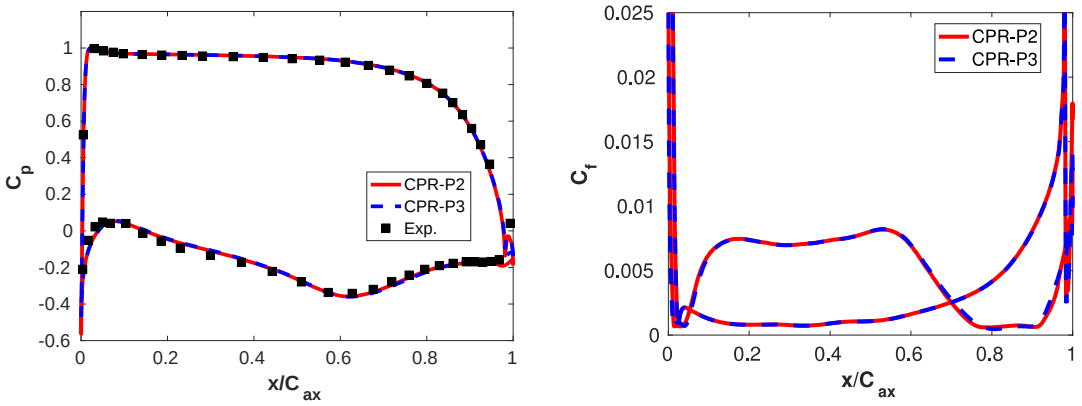


Figure 2. Iso-surfaces of Q-criteria colored by the spanwise vorticity.

The simulation started at p = 0 (or p0), and then restarted at p1 to establish a good initial condition for p2. The p2 simulation restarted from the p1 solution, and the averaging process began when the initial transient passed through the computational domain, and it was made sure that the mean flow was sufficiently converged. The instantaneous iso-surfaces of the Q-criterion colored by the spanwise vorticity is shown in Figure 2 for p = 3, showing the laminar flow on the airfoil, and its breakdown and transition to turbulence close to the trailing edge. The mean pressure coefficient c_p and skin friction coefficient c_f on the airfoil surface for p2 and p3 are displayed in Figure 3a and 3b respectively, together with experimental data. It clearly shows the p-convergence for c_p and c_f, because the p2 and

p3 results are already on top of each other. The agreement with the experimental data [10] is very good.

In addition, we also perform the same study with a very low Mach number of M = 0.01 to assess how the method handled nearly incompressible flow problems. For the remaining study, we always use the third order FR/CPR scheme, i.e., p = 2.



(a) (b)

Figure 3. Mean surface pressure coefficient skin friction coefficient by CPR-P2 and CPR-P3.

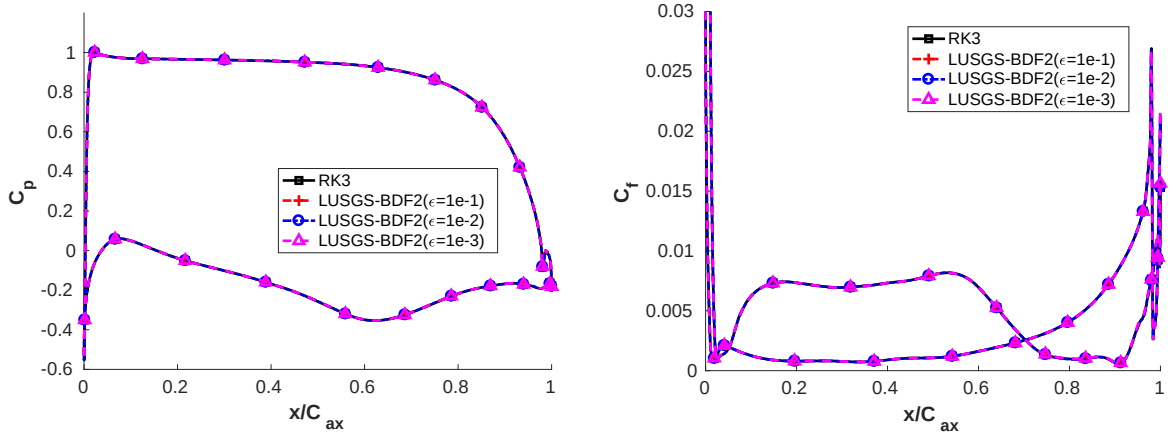
IV.1 (a) Accuracy assessment

For the explicit 3-stage RK (RK3) scheme, we always use the maximum time step for stability, and this time step roughly corresponds to a CFL number on the order of 1/5 for the smallest cell. As expected, this time step is so-small that the error is always dominated by the spatial operator. We therefore employ the RK3 results as the benchmark to evaluate the BDF1 and BDF2 results. Because of the turbulent nature of the flow, we cannot use the solution history to evaluate how accurate the unsteady simulation is. Instead, we use the power spectral density (PSD) of a flow variable at one of the most energetic locations (as shown in Figure 1) in the flow field to assess if the time marching scheme is capable of capturing the dynamics of the flow. In addition, we also compare the mean pressure and skin-friction coefficients on the surface of the turbine blade.

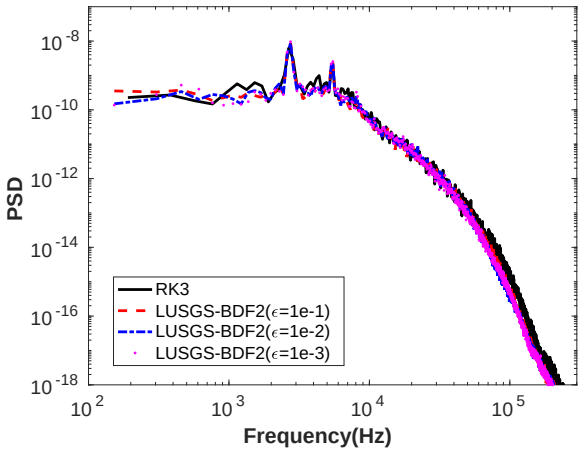
Several factors affect the accuracy and efficiency of the implicit LU-SGS approach. The first parameter is the convergence tolerance (ε) at each implicit step. Previous studies of unsteady (laminar) flow problems have indicated that a three-order drop in the unsteady residual [26] is necessary to achieve temporal convergence. Here we assess

the influence of this tolerance based on the PSD and the mean flow profiles. The second parameter which greatly affects the solution efficiency is how often the implicit operator is updated since it is expensive to compute. Previously we updated the implicit operator every 40-100 steps. This criterion is examined in the context of turbulent flow simulations. In order to carry out a fair comparison, all p2 simulations with explicit and implicit schemes started from the same initial condition at p2, which was obtained from reasonably good p0 and p1 solutions. Then the simulations were performed for 10 non-dimensional time units before averaging was carried out over another 10 non-dimensional time units (defined based on the incoming flow velocity and the chord length). The flow time history used to generate the PSD was also recorded during this time interval. Although BDF1 and BDF2 are linearly unconditionally stable, when the time step is too large, the nonlinear LU-SGS solver can still diverge. Therefore there is a stable time step limit for the LU-SGS approach, and we always try to use the maximum time step.

First, we investigate the influence of the convergence tolerance (ε) on the solution accuracy. We selected 3 different tolerances, e.g. ε=1e-1, 1e-2 and 1e-3, together with the BDF2 approach. The resulting c_p and c_f distributions and the PSD of density associated with the above 3 tolerances for the Mach = 0.4 case are shown in Figure 4. From Figure 4a and 4b, the collapsed c_p and c_f distributions indicate that ε = 1e-1 is reasonable for predicting the mean friction on the blade. From Figure 4c, note that the trailing edge vortex shedding frequency is captured by all schemes. The close agreement on PSDs also demonstrates that ε = 1e-1 is accurate enough to capture important turbulent scales. Therefore, in the following simulations, we will always set the convergence tolerance to be 1e-1.



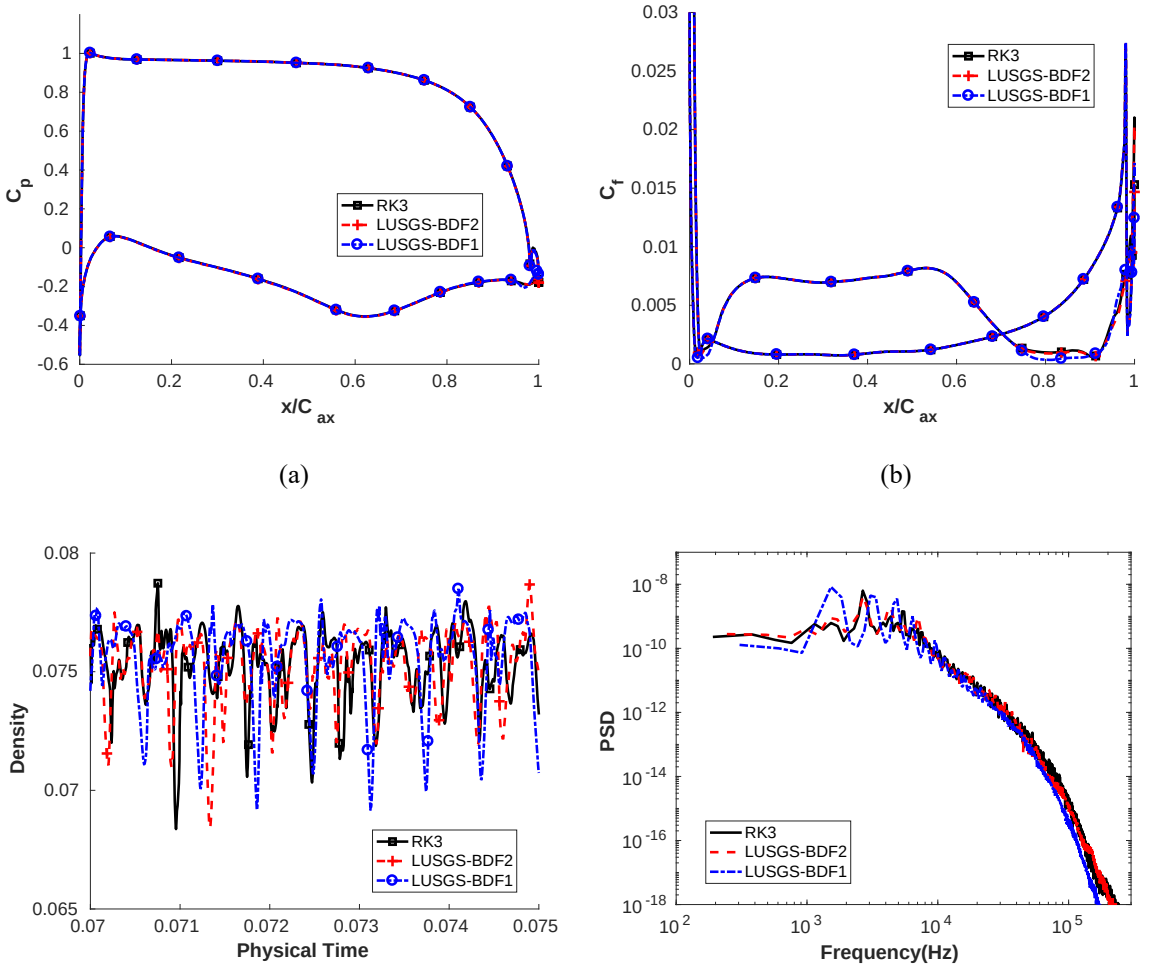
(a) (b)



(c)

Figure 4. Flow information of BDF2 with different convergence tolerance of Mach=0.4 case

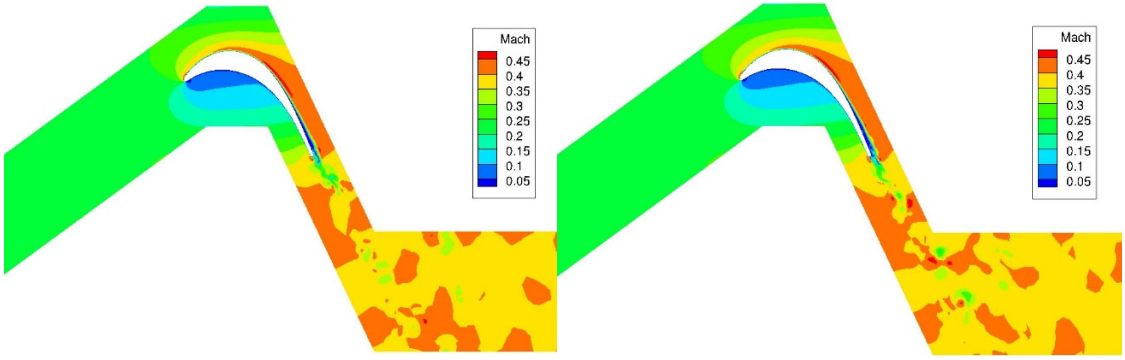
Second, we assess the accuracy of the BDF1 and BDF2 schemes. The distribution of the mean pressure and skin friction coefficients as well as the power spectral density of the monitoring point are shown in Figure 5. From Figure 5a, all schemes can predict the pressure coefficient very well. From Figure 5b, the skin friction coefficient computed with RK3 and BDF2 are also on top of each other, but the BDF1 result has a visible discrepancy with the other two schemes on the suction side near the transition region. Figure 5c and 5d show the density history as well as its PSD. The density histories of all schemes are evidently different because of the turbulent nature of the flow. For the PSD, there is a very good agreement between the BDF2 and RK3 results, and the trailing edge vortex shedding frequency is captured well by both schemes. The BDF1 results differ visibly from those with other two schemes. In the low frequency range, BDF1 failed to capture the main vortex shedding frequency. In the high frequency range, there is visibly more damping in the results of BDF1. Therefore, BDF2 associated with ε=0.1 appears to be adequate for the accuracy purpose. Figure 6 and 7 show the contours of instantaneous and averaged Mach number by RK3 and BDF2 respectively. Note that the implicit BDF2 and explicit schemes generated similar results. The transition of turbulence on the suction side is also captured by both schemes, indicating that both schemes resolved the boundary layer very well.



(a) (b)

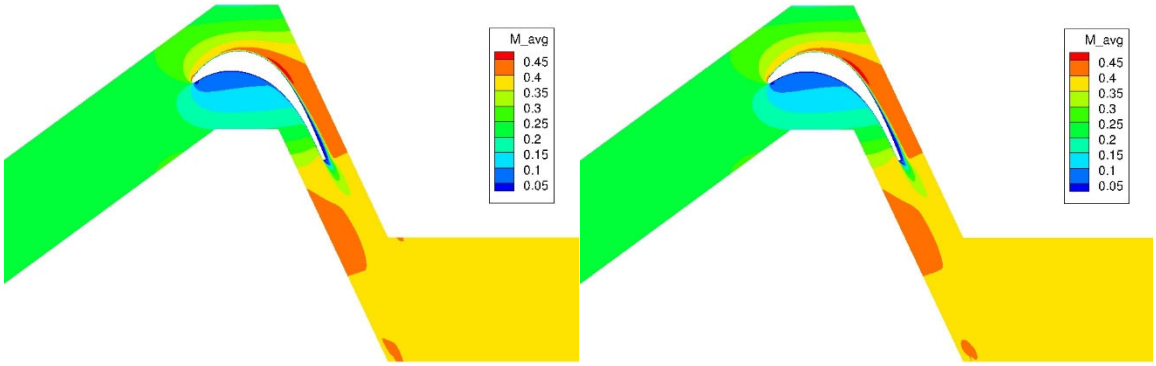
(c) (d)

Figure 5. Flow information of Mach=0.4 case



(a) RK3 (b) LUSGS-BDF2

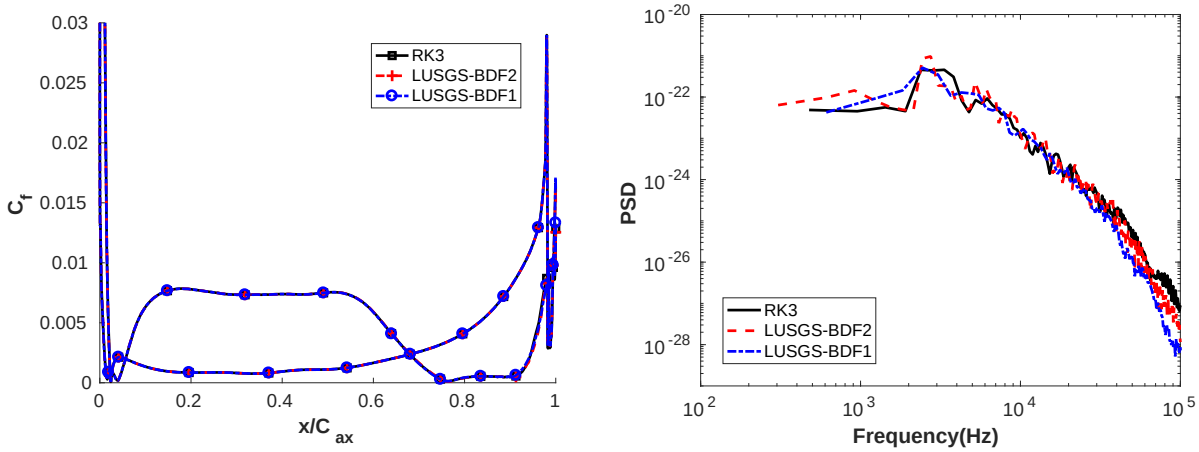
Figure 6. Instantaneous Mach number field in Mach=0.4 case



(b) RK3 (b) LUSGS-BDF2

Figure 7. Averaged Mach number field in Mach=0.4 case

Finally, we also evaluate the solution accuracy for a very low Mach flow at M = 0.01, with everything else being the same. The skin friction coefficient distribution and PSD of the monitoring point are shown in Figure 8.

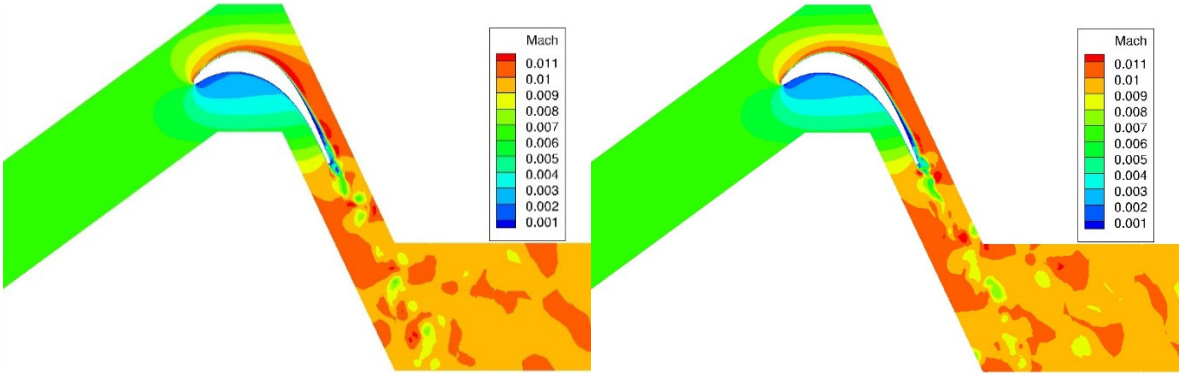


(a) (b)

Figure 8. Flow information of Mach=0.01 case

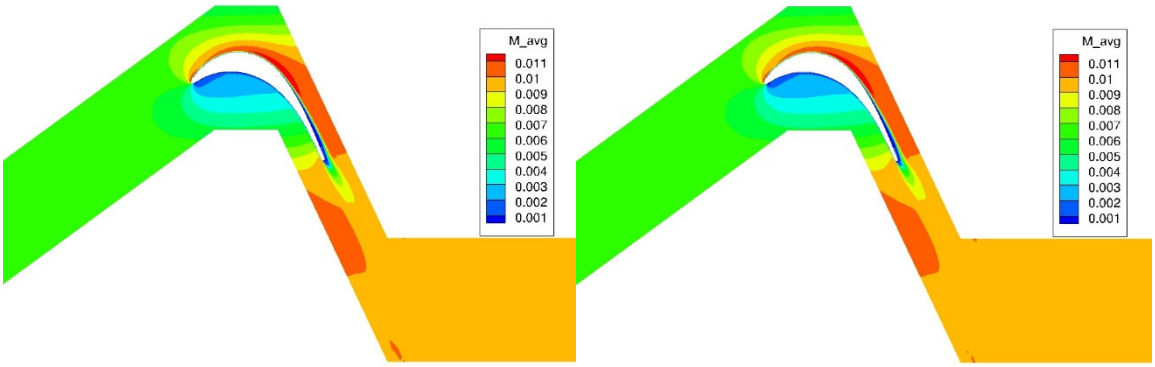
From Figure 8a, the skin friction coefficients of all three schemes are on top of each other, which means the mean flow by BDF1 is already as accurate as that by RK3. While from Figure 8b, BDF1 has evident damping in the high

frequency range, although it also captures the main vortex shedding frequency quite well. Figure 9 & 10 also show the contours of averaged Mach number by RK3 and BDF2 respectively, which are also similar.



(a) RK3 (b) LUSGS-BDF2

Figure 9. Instantaneous Mach number field in Mach=0.01 case



(b) RK3 (b) LUSGS-BDF2

Figure 10. Averaged Mach number distributions for the Mach = 0.01 case

IV. 1 (b) Efficiency Evaluation

This section investigates the computational efficiency. The efficiency of RK3 hinges on the maximum time step. For BDF2, the time step and the frequency to update the LHS are two key factors for efficiency. Moreover, those two factors are coupled with each other because of the nonlinearity of the flow physics. Generally speaking, a large time step requires the LHS to be updated more frequently, and vise-versa. The strategy in the present study is to first choose several different LHS updating frequencies, and under each frequency, the corresponding maximum time step is obtained to ensure stability. After that, we compare the efficiency among the above combinations and find

some trends. All simulations were running on 200 cores with 10 computational nodes. Each node has 2 CPUs (Intel Xeon E5-2670 v2), each with 10 cores, and 128GB of total memory.

For the Mach = 0.4 case, Figure 11 shows the BDF2's maximum time step and speedup with respect to RK3 in several different LHS update frequencies. Figure 11 indicates that the dominating factor for this case is the LHS updating frequency. When the LHS is updated every 25 steps or more, its benefit is negligible. The maximum speedup of BDF2 with respect to RK3 is about 3. The optimal time step in that situation is about 9 times larger than that of RK3.

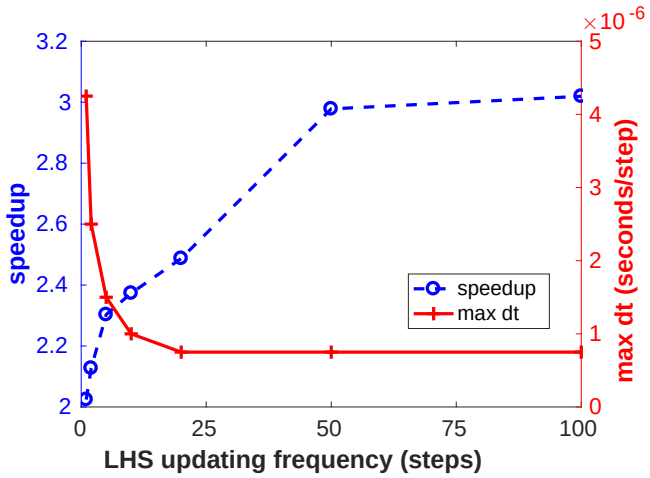


Figure 11. Speedup and maximum time step for LUSGS-BDF2 for Mach=0.4 case

For the Mach = 0.01 case, Figure 12 shows the maximum time step and the speedup. From this figure, we conclude that the maximum time step is the same regardless how often the LHS is updated. If the LHS is updated every 400 steps, the speedup of the implicit scheme over the explicit scheme is about 14. For this nearly incompressible flow regime, the time step is the deciding factor for efficiency because the LHS can essentially remain the same.

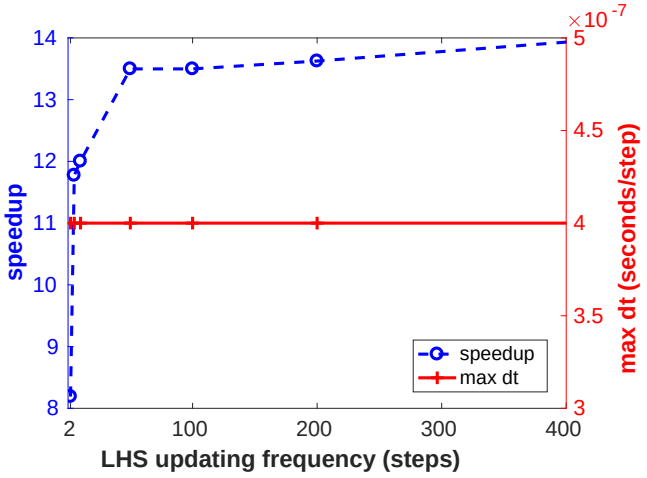


Figure 12. Speedup and maximum time step of LUSGS-BDF2 for Mach=0.01 case

The optimal time step sizes for both the low and Mach number problems are compared in Table 1.

Table 1. Optimal time step of RK3 and LUSGS-BDF2

optimal dt (sec/step)	Mach = 0.4	Mach = 0.01
RK3	8.e-8	2.e-9
LUSGS-BDF2	7.5e-7	4.e-7

It is obvious that the explicit RK3 scheme is very inefficient for the low Mach case because the time step is determined by the acoustic speed. At Mach = 0.01, the time step is 40 times smaller than that at M = 0.4 because of the stiffness of disparate acoustic and convective scales. For BDF2, the time steps are comparable with different Mach numbers.

The total CPU times are also compared in Table 2. For the Mach = 0.4 case, BDF2 is 3 times faster than RK3. For the Mach = 0.01 case, BDF2 is 14 times faster than RK3. Note that the low Mach case took much more CPU time (nearly 9 times) than the high Mach case with BDF2. This is because the low Mach case took more inner iterations to converge to the specified tolerance. The test results show that the implicit scheme is indeed more efficient, especially for low Mach problems.

Table 2. Total CPU times required to simulate 1 non-dimensional time with RK3 and LUSGS-BDF2

	Mach = 0.4		Mach = 0.01	
	CPU time (core hours)	inner iterations/step	CPU time (core hours)	inner iterations/step
RK3	30.19		1203.58	
LUSGS-BDF2	10 (speedup=3)	3	86.39 (speedup=14)	12~15

IV.1 (c) Scalability

Through previous accuracy and efficiency studies, the optimal parameters for the implicit LUSGS-BDF2 scheme have been obtained. This section focuses on the computational scalability. For this purpose, we conduct a strong scalability study using the same flow problem. For the Mach = 0.4 case, the comparison is shown in Figure 13. With up to 200 cores, RK3 can still achieve almost ideal speedup. BDF2's parallel performance is nearly ideal until with 200 cores, with which it achieves an efficiency of 85%. With 200 cores, each core contains roughly 70 elements, and 1,890 DOFs/equation at p = 2. The parallel performance of both the implicit and explicit schemes are quite good.

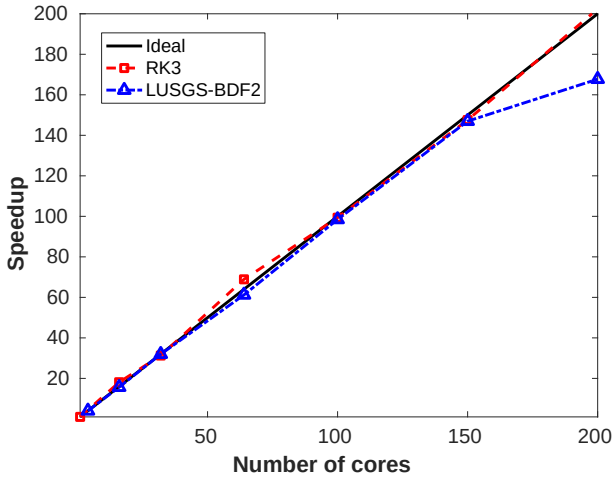


Figure 13. Strong scalability study of RK3 and LUSGS-BDF2 for the Mach = 0.4 case

IV. 2 Higher Reynolds number flow over a high pressure turbine vane

To verify the previous study for a broader range of Reynolds numbers, the second case of flow over a high pressure turbine (HPT) vane at a higher Reynolds number of 1.16 million is investigated. An extensive study on heat transfer at engine scale conditions was investigated experimentally by Arts and Rouvroit in a linear cascade in the von Karman Institute (VKI) [36]. The VKI vane geometry is shown in Figure 14, and its detailed description and coordinates can be found in [36]. The chord length of the vane (C) is 0.0676m, and the span length is 0.012m, i.e., 17.75% of chord. The horizontal projection of the chord length (C_{ax}) is 0.0368m. The thickness of the trailing edge (D_{TE}) is 1.42e-3m. The distance between the inlet of the domain and the vane's leading edge is 0.0547m. The p2 mesh shown in Figure 14 has 68,640 p2 curved hexahedral elements with 16 elements in the spanwise direction. There are 7 hexahedral mesh layers attached on the vane with the progression rate of 1.5 in the normal direction. At the inlet boundary, the total pressure and total temperature are fixed at 1.849e5 Pa and 409 K respectively with a zero angle of attack for the inflow velocity. At the exit, the static pressure is fixed at 1.559e5 Pa, resulting in an isentropic Mach number of 0.50. The vane is set to be a no-slip iso-thermal wall with a wall temperature of 300K. Periodic boundary conditions are used in both the span-wise and pitch-wise directions. Sutherland's law is used to determine the dynamic viscosity coefficient. The Reynolds number based on the chord length, the exit isentropic density, velocity (V_{exit}) and viscosity is 1.16 million. The gas constant and the Prandtl number are 287.55 J/(kg*K) and 0.713 respectively. The averaged y+ at the first element is around 9.2 based on the element size in the wall normal direction with an equivalent y+ of 3.1 at p2.

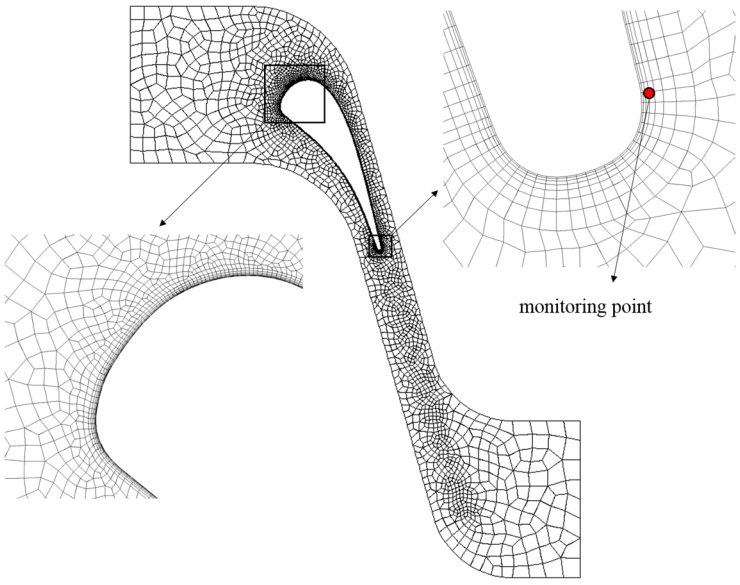
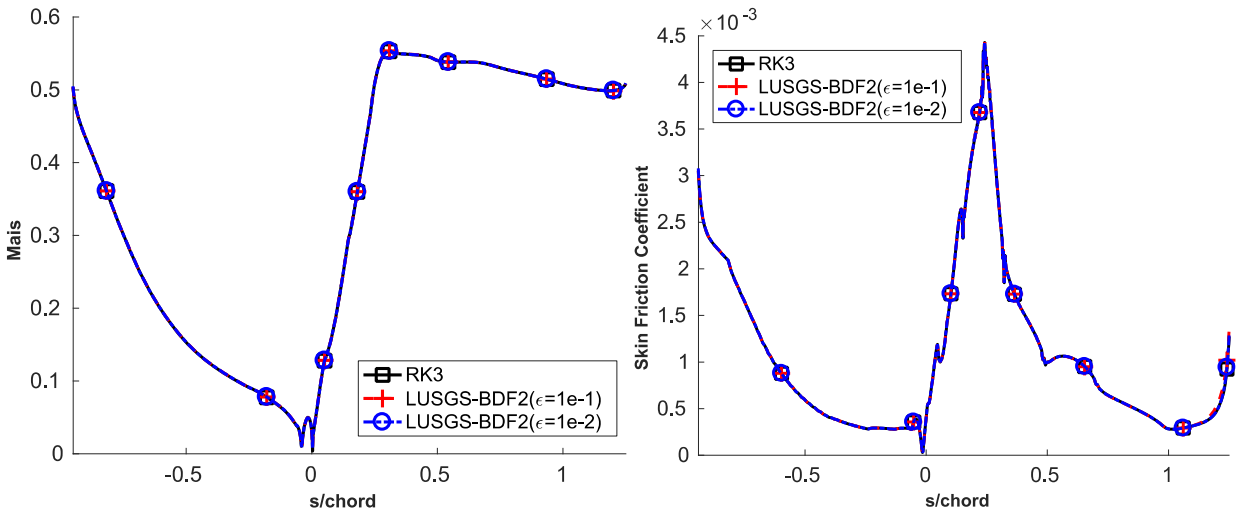


Figure 14. Unstructured hexahedron mesh used for MUR129 case.

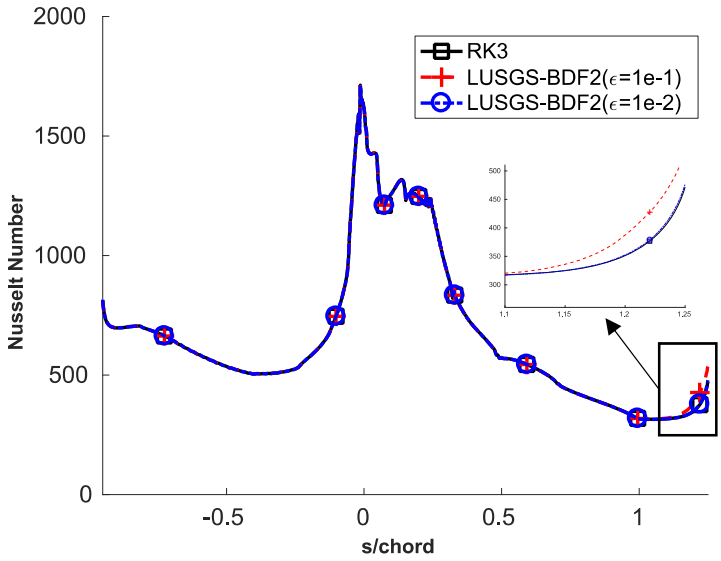
The simulation process is quite similar to that of the T106A case. The simulation started at p = 0, and then restarted at p1 to establish a reasonable initial condition for p2. The averaging began when the initial transient went through the domain completely. During the averaging process, the pressure history of an energetic location is recorded, which are shown in Figure 14.

IV. 2 (a) Accuracy assessment

To investigate the convergence tolerance in the LUSGS solver for this higher Reynolds number case, we selected 2 different tolerances, e.g. ε=1e-1 and 1e-2, together with the BDF2 approach. The resulting mean surface loading (isentropic Mach number and skin friction coefficient) and heat transfer (Nusselt number) are shown in Figure 15. From Figure 15a and 15b, the collapsed isentropic Mach number and skin friction coefficient distributions indicate that ε = 1e-1 is small enough for predicting the averaged surface loading on the blade. From Figure 15c, note that the results of RK3 and LUSGS-BDF2 with ε = 1e-2 are on top of each other, but that of LUSGS-BDF2 with ε = 1e-1 has an evident over prediction near the trailing edge for the heat transfer.



(a) (b)



(c)

Figure 15. Mean surface loading and heat transfer predictions on CPR-P2.

Furthermore, the gage pressure history and its PSD at the monitoring location is shown in Figure 16. The histories of the pressure are quite different among all schemes, but their power spectral densities agree well in the low frequency range at least. However, in the high frequency range, which represents smaller temporal scales of flow, the LUSGS-BDF2 with ε = 1e-2 agrees well with the RK3, but the LUSGS-BDF2 with ε = 1e-1 gives much higher prediction. Therefore, in the higher Reynolds number case, we should set the convergence tolerance ε to be 1e-2.

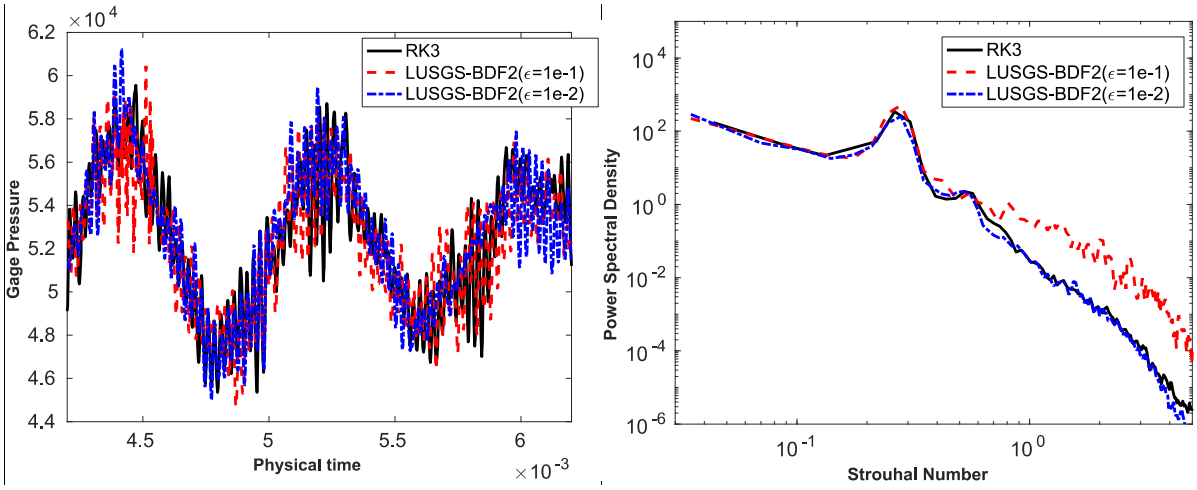
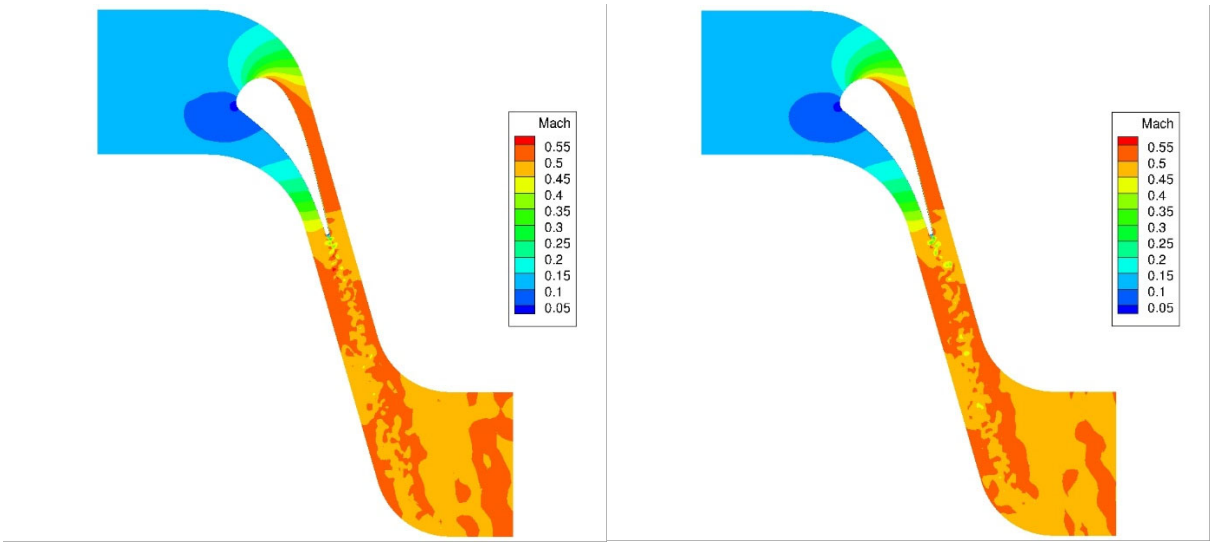


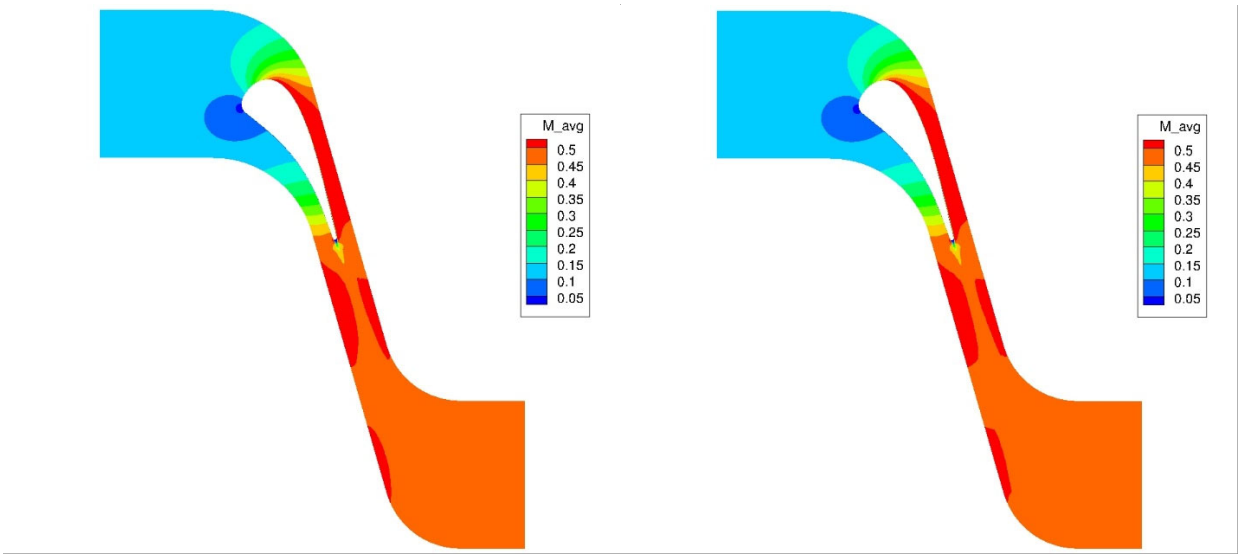
Figure 16. Gage pressure history and its PSD on the monitoring point.

Figure 17 and 18 show the contours of instantaneous and averaged Mach number by RK3 and BDF2(ε = 1e-2) respectively. The implicit and explicit schemes again generated similar results.



(a) RK3 (b) LUSGS-BDF2

Figure 17. Instantaneous Mach number field.



(a) RK3 (b) LUSGS-BDF2

Figure 18. Averaged Mach number field.

IV.2 (b) Efficiency evaluation

Different from the T106A case, this VKI case has a higher Reynolds number. We found that if the LHS updating frequency is fixed, it needs to be updated frequently, which obviously reduced the efficiency. Instead, we employ a

adaptive strategy. Only when the number of inner iterations exceeded a certain number, selected to be 7, in this case, the LHS is updated. In addition, we also fix the maximum number of inner iterations of the LUSGS solver to be 10. The CPU times required to simulate 1 non-dimensional time using RK3 and LUSGS-BDF2 is shown in Table 3. The time scale is defined by the chord length and the exit isentropic velocity. It is shown that the implicit scheme is about 3.4 times faster than the explicit one in this high Reynolds number flow.

Table 3. Total CPU times required to simulate 1 non-dimensional time with RK3 and LUSGS-BDF2.

	CPU time (core hours)	Time step (sec)	Inner iterations/step
RK3	184.8	1.e-8	
LUSGS-BDF2	54.4	2.e-7	5

V. Conclusions

In the present study, we evaluate the performance of explicit and implicit algorithms for a high-order FR/CPR scheme to conduct LES using both low and high Reynolds number cases. The low Reynolds number case is a benchmark turbo-machinery flow problem from the 4th International Workshop on High-Order CFD Methods. The high Reynolds number case is a well-known HPT vane case from VKI. The 3-stage SSP Runge-Kutta algorithm is compared to implicit BDF schemes with a non-linear BLU-SGS solver in accuracy, efficiency and scalability. Conclusions from the present study are summarized next.

- The LUSGS-BDF1 scheme is not accurate, and can degrade the mean pressure and skin friction profiles, as well the PSD.
- The LUSGS-BDF2 scheme can use a time-step of 1 to 2 orders larger than that allowed by the RK3 scheme. For the p2 spatial FR/CPR scheme, the LUSGS-BDF2 scheme is a factor of 3 to 15 more efficient than the RK3 scheme depending on the Mach number.
- The implicit schemes are not very sensitive to the convergence tolerance at each time step for LES based on the computed mean surface loading and heat transfer, as well as the PSD at energetic locations. To be accurate enough for both low and high Reynolds number cases, the convergence tolerance should be set to 0.01.

- At a very low Mach number, the explicit scheme becomes very inefficient. Low Mach preconditioning becomes necessary for such problems.
- Although both the explicit and implicit schemes achieved good parallel performance on a small cluster for a small problem, the implicit LUSGS scheme does not scale as well as the explicit RK3 scheme when the problem size becomes small on each process core.

VI. Acknowledgements

We gratefully acknowledge support by AFOSR under grant FA9550-16-1-0128, and US Army Research Office under grants W911NF-15-1-0505 and W911-NF-1510377, as well as GE Global Research.

References

1. Smagorinsky J. General circulation experiments with the primitive equations: I. The basic experiment. Mon Weather Rev 1963;91(3):99-164.
2. Laskowski GM, Kopriva J, Michelassi V, Shankaran S, Paliath U, Bhaskaran R, Wang Q, Talnikar C, Wang ZJ, Jia FL. Future Directions of High Fidelity CFD for Aerothermal Turbomachinery Analysis and Design. In: 46th AIAA Fluid Dynamics Conference; 2016. p. 3322.
3. Piomelli U, Balaras E. Wall-layer models for large-eddy simulations. Annu Rev Fluid Mech 2002;34(1):349-374.
4. Pope SB. Ten questions concerning the large-eddy simulation of turbulent flows. New J Phys 2004;6(1):35.
5. Bhaskaran R, Wood TH, Paliath U, Breeze-Stringfellow A. Large Eddy Simulation of a 3D Transonic Fan. In: 46th AIAA Fluid Dynamics Conference; 2016. p. 3816.
6. Kopriva JE, Laskowski GM, Sheikhi MRH. Hybrid LES of a high pressure turbine nozzle/blade interaction. In: Grigoriadis D, Geurts B, Kuerten H, Fröhlich J, Armenio V. (eds) Direct and Large-Eddy Simulation X. ERCOFTAC Series, vol 24. Springer, Cham. 2018.
7. Kopriva JE, Laskowski GM, Shiekhi MRH. Computational assessment of inlet turbulence on boundary layer development and momentum/thermal wakes for high pressure turbine nozzle and blade. IMECE 2014-38620, 2014.

8. Paliath U, Premasuthan S. Large Eddy Simulation for Jet Installation Effects. In: 19th AIAA/CEAS Aeroacoustics Conference; 2013. p. 2137.
9. Paliath U, Shen H, Avancha R, Shieh C. Large eddy simulation for jets from chevron and dual flow nozzles. In: 17th AIAA/CEAS Aeroacoustics Conference; 2011. p. 2881.
10. Stadtmüller P. Investigation of Wake-Induced Transition on the LP Turbine Cascade T106 A-EIZ. DFG-Verbundproject Fo 2001;136(11).
11. Wang ZJ. High-order methods on unstructured grids for Navier–Stokes equations. Prog Aerosp Sci 2007;43(1-3):1-41.
12. Wang ZJ, Fidkowski K, Abgrall R, Bassi F, Caraeni D, Cary A, … Kroll N. High-order CFD methods: current status and perspective. Int J Numer Methods Fluids 2013;72(8):811-845.
13. Cockburn B, Karniadakis G, Shu CW. Discontinuous Galerkin Methods: Theory, Computation and Applications: Theory, Computation and Applications 2000.
14. Bassi F, Rebay S. A high order discontinuous Galerkin method for compressible turbulent flows. In Discontinuous Galerkin Methods, Springer Berlin Heidelberg 2000. p. 77-88.
15. Liu Y, Vinokur M, Wang ZJ. Discontinuous spectral difference method for conservation laws on unstructured grids. Computational Fluid Dynamics 2004. p. 449-454.
16. Huynh HT. A flux reconstruction approach to high-order schemes including discontinuous Galerkin methods. In: 18th AIAA Computational Fluid Dynamics Conference 2007. p. 4079.
17. Beck AD, Bolemann T, Flad D, Frank H, Gassner GJ, Hindenlang F, Munz CD. High-order discontinuous Galerkin spectral element methods for transitional and turbulent flow simulations. Int J Numer Methods Fluids 2014;76(8):522-548.
18. Lu Y, Liu K, Dawes WN. Flow simulation system based on high order space-time extension of flux reconstruction method. In: 53rd AIAA Aerospace Sciences Meeting 2015. p. 0833.
19. Lodato G, Castonguay P, Jameson A. Structural wall-modeled LES using a high-order spectral difference scheme for unstructured meshes. Flow Turbul Combust 2014;92(1-2):579-606.
20. Uranga A, Persson PO, Drela M, Peraire J. Implicit large eddy simulation of transition to turbulence at low Reynolds numbers using a discontinuous Galerkin method. Int J Numer Methods Eng 2011;87(1-5):232-261.
21. Vermeire BC, Nadarajah S, Tucker PG. Implicit large eddy simulation using the high-order correction

procedure via reconstruction scheme. Int J Numer Methods Fluids 2016;82(5):231-260.

22. Vermeire BC, Witherden FD, Vincent PE. On the utility of GPU accelerated high-order methods for unsteady flow simulations: A comparison with industry-standard tools. J Comput Phys 2017;334(1):497-521.
23. Boris JP, Grinstein FF, Oran ES, Kolbe RL. New insights into large eddy simulation. Fluid Dyn Res, 1992;10(4-6):199-228.
24. Li Y, Wang ZJ. A priori and a posteriori evaluations of sub-grid scale models for the Burgers' equation. Comp Fluids 2016;139(5):92-104.
25. Chen RF, Wang ZJ. Fast, block lower-upper symmetric Gauss-Seidel scheme for arbitrary grids. AIAA J 2000;38(12):2238-2245.
26. Jameson A, Caughey D. How many steps are required to solve the Euler equations of steady, compressible flow-In search of a fast solution algorithm. In: 15th AIAA Computational Fluid Dynamics Conference 2001. p. 2673.
27. Yoon S, Jameson A. Lower-upper symmetric-Gauss-Seidel method for the Euler and Navier-Stokes equations. AIAA J 1988;26(9):1025-1026.
28. Gottlieb S, Shu CW. Total variation diminishing Runge-Kutta schemes. Math Comput 1998;67(221):73-85.
29. Haga T, Gao H, Wang ZJ. A high-order unifying discontinuous formulation for the Navier-Stokes equations on 3D mixed grids. Mathematical Modelling of Natural Phenomena 2011;6(3):28-56.
30. Wang ZJ, Gao H. A unifying lifting collocation penalty formulation including the discontinuous Galerkin, spectral volume/difference methods for conservation laws on mixed grids. J Comput Phys 2009;228(21):8161-8186.
31. Huynh HT, Wang ZJ, Vincent PE. High-order methods for computational fluid dynamics: a brief review of compact differential formulations on unstructured grids. Comp Fluids 2014;98(2):209-220.
32. Wang ZJ. A perspective on high-order methods in computational fluid dynamics. Sci China Phys, Mech & Astro 2016;59(1):614701.
33. Vincent PE, Castonguay P, Jameson A. A new class of high-order energy stable flux reconstruction schemes. J Sci Comput 2011;47(1):50-72.
34. Park JS, You H, Kim C. Higher-order multi-dimensional limiting process for DG and FR/CPR methods on tetrahedral meshes. Comp Fluids 2017;154(1):322-334.

35. Yu M, Wang ZJ, Liu Y. On the accuracy and efficiency of discontinuous Galerkin, spectral difference and correction procedure via reconstruction methods. J Comput Phys 2014;259(15):70-95.
36. Arts T, Rouvroit M. Aero-thermal performance of a two-dimensional highly loaded transonic turbine nozzle guide vane: A test case for inviscid and viscous flow computations. J Turbomach 1992;114(1):147-154.

Computers and Fluids 156 (2017) 579–589

Contents lists available at ScienceDirect

Computers and Fluids

journal homepage: www.elsevier.com/locate/compfluid

Towards industrial large eddy simulation using the FR/CPR method

Z.J. Wang[a,*], Y. Li[a], F. Jia[a], G.M. Laskowski[b], J. Kopriva[b], U. Paliath[c], R. Bhaskaran[c]

[a] University of Kansas, Lawrence, KS 66045, U.S.A
[b] GE Aviation, Lynn MA 01905, U.S.A
[c] GE Global Research, Niskayuna NY 12309, U.S.A

ARTICLE INFO

Article history:
Received 14 January 2017
Revised 29 March 2017
Accepted 27 April 2017
Available online 28 April 2017

Keywords:
Large eddy simulation
Adaptive high-order methods
Computational fluid dynamics
Turbulent flow

ABSTRACT

Large eddy simulation (LES) has been shown to be very promising in computing vortex-dominated turbulent flows. The proliferation of high-order methods capable of handling complex geometries has significantly reduced the cost of such simulations to achieve a specified level of accuracy comparing with 1st or 2nd order methods. In this article, we examine key factors affecting the quality of LES solutions, and outline our progress in applying one particular high-order method, the flux reconstruction (FR) or correction procedure via reconstruction (CPR) method in LES. The progress has enabled LES of a benchmark flow over a turbine blade at a Reynolds number of nearly 600,000 with p-independent mean surface skin friction and heat transfer, which agree well with experimental data. Pacing items in the use of LES in the design process are given.

1. Introduction

The potential offered by Large Eddy Simulation (LES) [33] to compute turbulent flows has been well-known since its inception over fifty years ago. After decades of development, LES is starting to move from being an analysis tool to design process in limited context where the accuracy is needed and the cost can be justified. In terms of cost and accuracy, LES lies between the Reynolds Averaged Navier-Stokes (RANS) and the Direct Numerical Simulation (DNS) approaches. In a RANS simulation, all turbulence scales are modeled, and therefore, it is the least expensive. For non-separating turbulent flow problems, the RANS approach has been shown to be accurate and efficient. In the other extreme, the DNS approach resolves all turbulence scales, and thus it is very accurate and prohibitively expensive for high Reynolds number flow problems. Although RANS models have been effective for many practical problems, they may be inadequate for certain flows such as massively separated flows because a statistically steady mean flow may not exist for such problems. Even if a steady mean exists, a universal turbulence closure does not exist. The use of DNS in computing high Reynolds number flows is, for the foreseeable future, limited by computing power because of the disparate time and length scales. LES is a compromise of the two approaches, and offers the best promise for vortex dominated flows found in many aerospace applications such as flow over high lift configurations, rotorcraft flows and more recently flows in aircraft engines [18–20,26,27].

In LES, large scales and small scales are separated by a low-pass filter, either explicitly or implicitly. The large scales are resolved while the effect of small scales is represented by a sub-grid scale (SGS) stress model. Since small-scale motions are believed to be more universal, and thus easier to model than large scale ones, LES can offer reasonable accuracy even for unsteady separated flows while requiring much less computer resources than DNS. One critical parameter in LES is the filter width, Δ, which determines what scales are computed, and what are modeled [31]. Ideally, the determination of Δ should be based on physical considerations, for example, to obtain a mean lift or drag error of, say, 5%. However, it is very difficult to determine the filter width without conducting numerical simulations with various resolutions. Once Δ is specified, the "truth" LES solution is well defined theoretically, which is the filtered DNS solution with the low pass filter. Because the DNS solution is usually unknown, the truth LES solution is also unknown. Nevertheless, the usual mesh or order-refinement studies can be conducted with a fixed filter width to investigate the effects of the mesh resolution and mesh-dependence. In reality, however, an explicit filter is rarely employed, and the mesh size often serves as an "implicit" filter size although for methods with multiple internal degrees of freedom (DOFs), the effective implicit filter size is a fraction of the element size. As a result, when one refines the mesh, the filter width is also reduced. Therefore, mesh independence is difficult to demonstrate with mesh-refinement studies in LES, unless of course, the mesh resolution approaches that required

* Corresponding author.
E-mail address: zjw@ku.edu (Z.J. Wang).

http://dx.doi.org/10.1016/j.compfluid.2017.04.026
0045-7930/

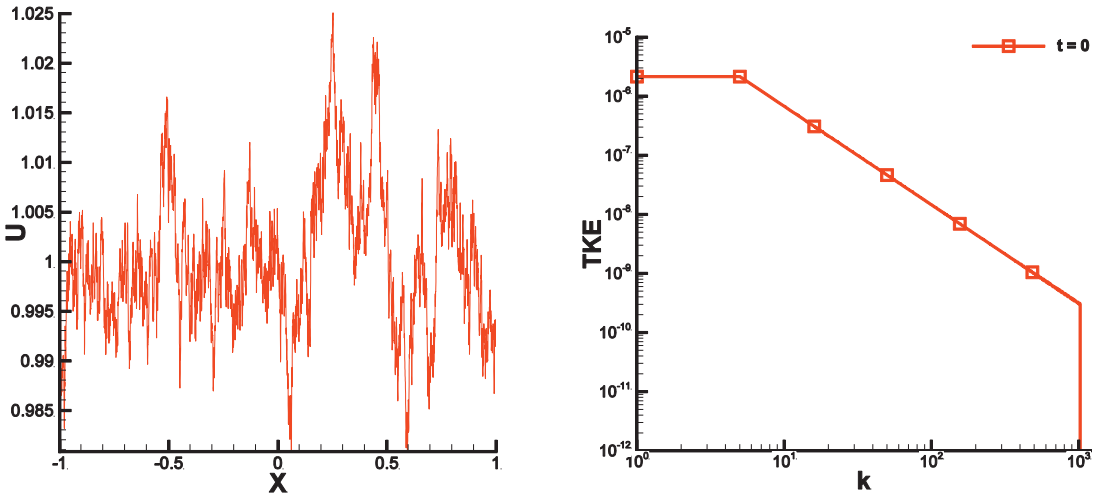


Fig. 1. Initial condition (left) and the initial energy spectrum (right).

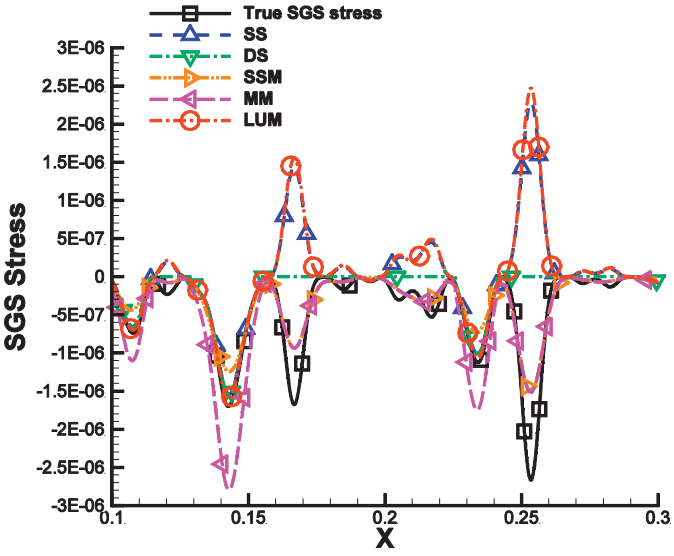


Fig. 2. The SGS stress comparison in the a priori tests.

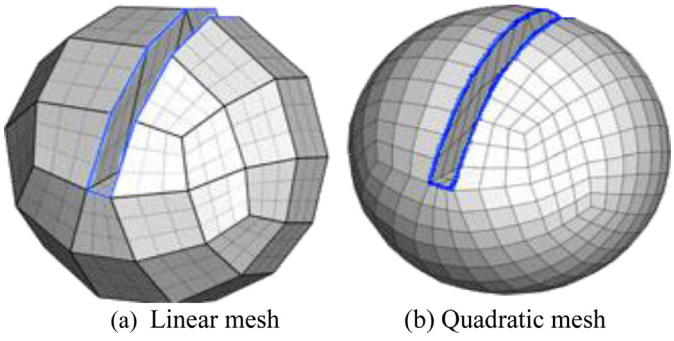


Fig. 3. Illustration of low order and high order quadrilateral meshes.

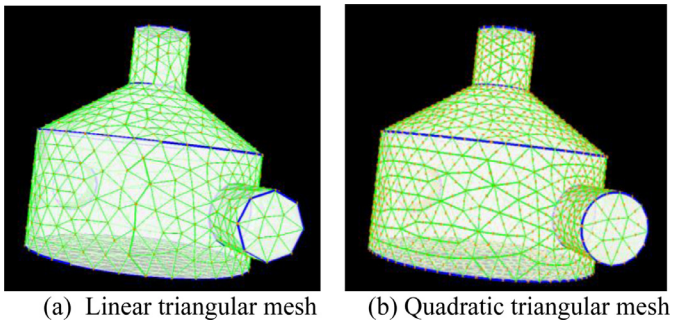


Fig. 4. Illustration of low order and high order triangular meshes.

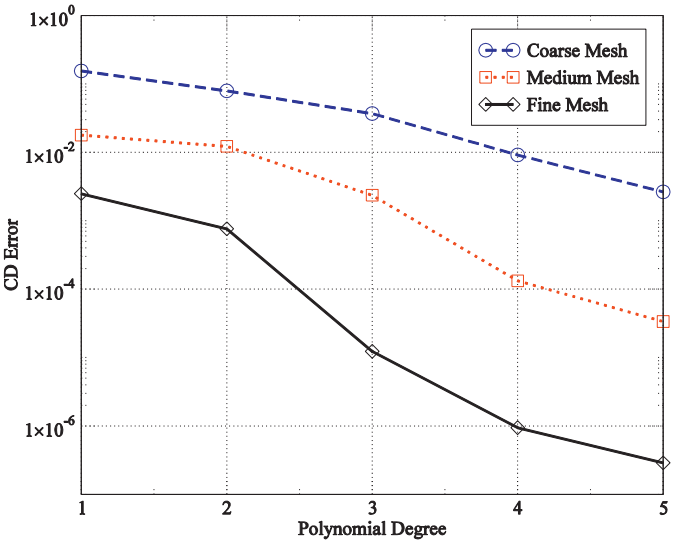


Fig. 5. C_D errors at t/T = 1 with HP-refinements.

by DNS. In this case, the LES solution converges to the DNS solution.

The shortest waves remaining in the LES solution have wave lengths on the order of Δ. There must be enough DOFs per wave (DPW) to properly resolve the shortest waves, with at least 2 DPW for spectral methods, and significantly more for other lower order methods [37]. It is thus obvious that the cost of LES is largely dependent on the filter width and the efficiency of the numerical methods. Theoretically speaking, the filter width Δ and the mesh size h are two fundamentally different parameters. As mentioned earlier, Δ and h are often taken to be the same in practice. In this case, there may not be enough DPW to adequately obtain the "truth" LES solution. For example, in a finite volume or finite difference method, there is only 1 DOF in each element. Therefore, there is only 1 DPW for a wave with length Δ. Since we normally need

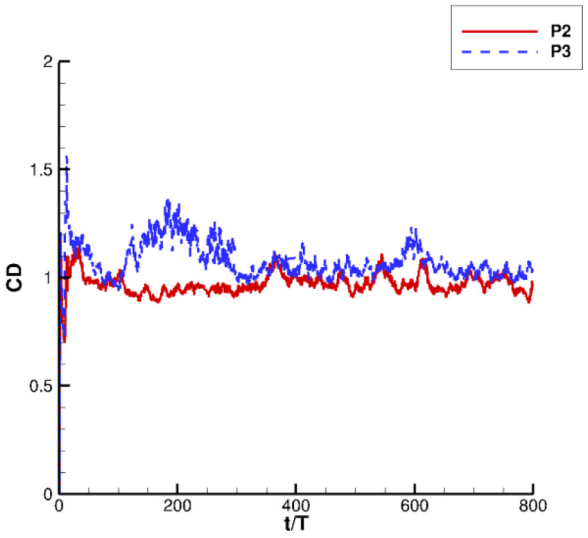


Fig. 6. The C_D histories on the medium mesh.

~10 DPW to properly resolve a wave with a 2nd order FV method [37], waves with shorter wave lengths than 10Δ can never be adequately resolved because of the large truncation error at such a mesh resolution. It was shown in [21] that the truncation error is so dominant that errors due to SGS models are negligible. Therefore, for methods which are guaranteed to be linearly stable, implicit LES (ILES) [5,11] in which no SGS models are used is preferred because of no extra cost.

Recently, adaptive high-order methods such as discontinuous Galerkin (DG) [6], spectral difference (SD) [23] and flux reconstruction (FR) or correction procedure via reconstruction (CPR) methods [13,12] have shown much potential in LES with complex configurations. Impressive results have appeared in the literature on either CPU and GPU clusters, or a mixture of both, e.g., see the work from many research groups [3,24,25,38,40,41]. These methods are capable of handling unstructured meshes. Unlike FV methods, multiple DOFs are defined in each element depending on the order of accuracy. In 1D, for example, a solution polynomial of p requires p+1 DOFs to be defined on each element. Even if the filter width is the same as the mesh size, it is possible to have multiple DPW for the shortest waves in the LES solution. If the order is high enough, the truncation error can be made small. Then the error due to the SGS models may be the main source, and the effects of SGS models may be significant. In such cases, the scale similarity model [1] may have an advantage.

For high Reynolds number turbulent flow problems, the mesh requirement to resolve the boundary layer may be too severe even for LES. However, the skin friction is strongly dependent on the mesh resolution near the wall, not only in the wall-normal direction, but also in the flow and cross-flow directions. A pure-LES approach at such high Reynolds numbers is still too expensive [30]. Several approaches have been developed to reduce the mesh requirement for LES: hybrid RANS/LES approaches such as DES (detached eddy simulation) [35], and wall modeled LES approaches (WMLES) [24,43,49]. In a DES-type approach, RANS [34] is used in the flow field near solid walls while LES is employed in massively separated flow regions with sufficient mesh resolution. In WMLES, the mesh resolution near the wall is not sufficient to resolve the "large eddies" since the size of the so-called "large eddy" in the boundary layer is much smaller than the integral scale of the flow field. If they are to be resolved, the mesh has to be very fine in all three directions, rendering the computational cost close to DNS. Thus, in a practical LES, wall modeling is necessary for high Reynolds problems in the foreseeable future. This type of approach treats most of the flow field with LES away from the wall, but introduces a wall model to compute the wall shear stress more accurately than otherwise.

Recent Workshops on High-Order CFD Methods [46] have concluded that a significant pacing item in employing high-order methods in real world flow simulations is high-order meshes, in which edges are represented by degree 2 or higher polynomials. This is because such simulations require "coarse meshes" relative to those used in 2nd order methods. Linear meshes would introduce too much solution errors near wall boundaries, which destroy the quality of the high-order solution. Recently, a low to high order mesh converter called meshCurve has been developed [15] for such a purpose, which will be discussed in the present paper.

The paper is organized as follows. In Section 2, we briefly review the FR/CPR method. In Section 3, we describe results from a recent study on SGS models. In Section 4, high-order mesh gener-

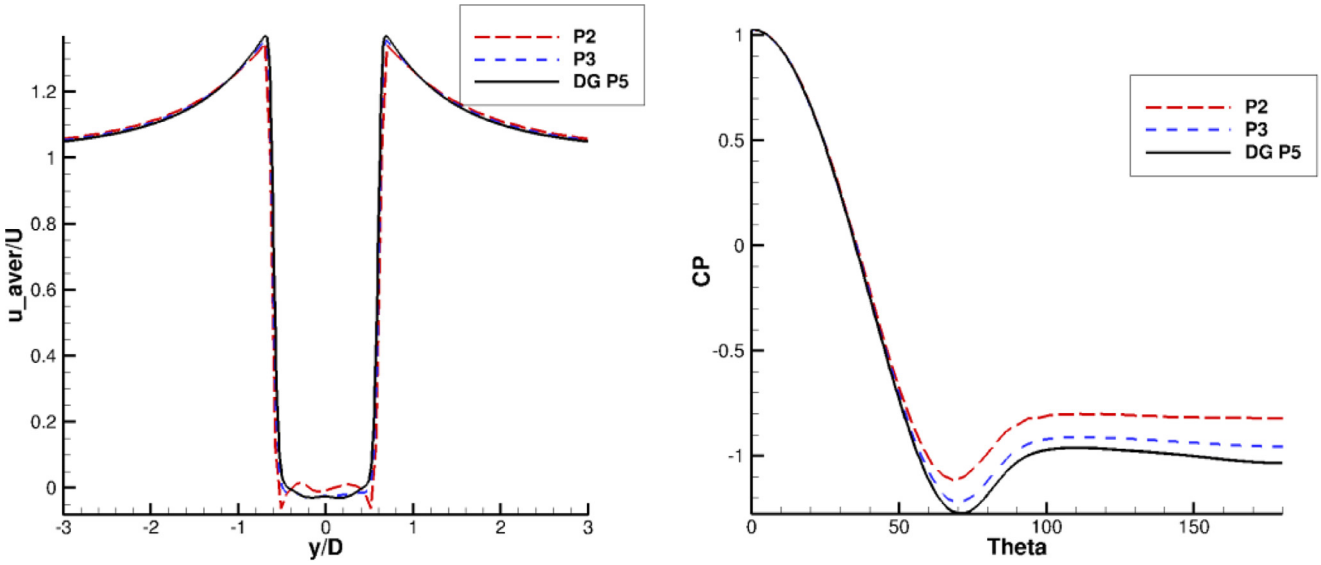


Fig. 7. Mean velocity profile at stream-wise location x/D = 0.58, and the mean c_p profile on the cylinder wall (u_aver is the averaged stream-wise velocity, U is the free-stream velocity magnitude, Theta is the polar angle of the cylinder with 0 corresponding to the stagnation point).

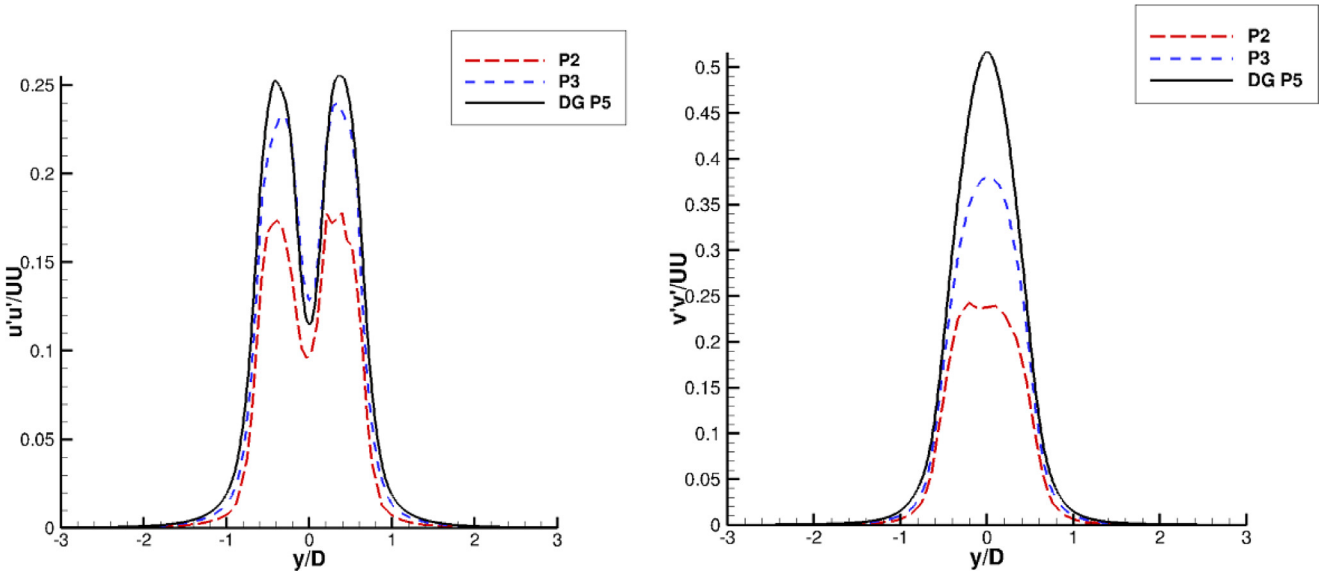


Fig. 8. Comparison of computed Reynolds stresses between p2, p3 and a high resolution DG p5 simulation at x/D = 1.54 (u', v' are the instantaneous velocity fluctuations in the stream-wise and cross-flow directions).

ation is discussed. Then we will present results for several benchmark LES problems in Section 5. Finally we conclude the paper with several future pacing items.

2. A brief overview of the FR/CPR method

The FR/CPR method was originally proposed by Huynh [13] in 2007 for hyperbolic partial differential equations, and later it is extended it to hybrid unstructured meshes [47,12]. Further developments on the FR/CPR method are given in [14,45]. This method belongs to discontinuous finite element methods, similar to the DG method, but also has some unique advantages. For example, FR contains a larger family of schemes [39], which may allow larger time steps than the DG method. Many groups also reported that FR is more efficient than the DG method [28,48]. Here we present a brief introduction of the FR/CPR method starting from a hyperbolic conservation law

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{U}) = 0, \tag{1}$$

with initial and boundary conditions, where the vector **U** consists of conservative variables, and **F** is the flux. By discretizing the computational domain with non-overlapping elements, and introducing an arbitrary test function W in each element, the weighted residual formulation of Eq. (1) on element V_i can be expressed as

$$\int_{V_i} \left[\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{U}) \right] W d\Omega = 0. \tag{2}$$

The conservative variables inside one element are assumed to be polynomials, and expressed by nodal values at certain points called solution points (SPs). After applying integration by parts to the divergence of flux, replacing the normal flux term with a common Riemann flux F^n_{com} and integrating back by parts, we obtain

$$\int_{V_i} \frac{\partial \mathbf{U}_i}{\partial t} W d\Omega + \int_{V_i} W \nabla \cdot \mathbf{F}(\mathbf{U}_i) d\Omega + \int_{V_i} W [F^n_{com} - F^n(\mathbf{U}_i)] dS = 0. \tag{3}$$

Here, the common Riemann flux is computed with a Riemann solver

$$F^n_{com} = F^n_{com}(\mathbf{U}_i, \mathbf{U}_{i+}, \mathbf{n}), \tag{4}$$

where $\mathbf{U_{i+}}$ stands for the solution outside the current element, and **n** denotes the outward normal direction of the interface. The normal flux at the interface is:

$$F^n(\mathbf{U}_i) = \mathbf{F}(\mathbf{U}_i) \cdot \mathbf{n}. \tag{5}$$

Note that if the face integral in Eq. (3) can be transformed into an element integral then the test function will be eliminated. In order to do so, a "correction field" δ_i is defined in each element as

$$\int_{V_i} W \delta_i d\Omega = \int_{\partial V_i} W [F^n] dS = 0, \tag{6}$$

where $[F^n] = [F^n_{com} - F^n(\mathbf{U}_i)]$ is the normal flux jump. Eqs. (3) and (6) result in

$$\int_{V_i} \left[\frac{\partial \mathbf{U}_i}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{U}_i) + \delta_i \right] W d\Omega = 0. \tag{7}$$

The final formulation for each solution point j is

$$\frac{\partial \mathbf{U}_{i,j}}{\partial t} + \Pi_j [\nabla \cdot \mathbf{F}(\mathbf{U}_i)] + \delta_{i,j} = 0. \tag{8}$$

where Π_j denotes a projection to the polynomial space, and subscript j denotes the j-th solution point in a certain element.

For viscous flux involving the gradient of conservative variables, we use the Bassi-Rebay 2 (BR2) scheme [2]. In this paper, the time marching method is the explicit third-order SSP Runge-Kutta scheme [10].

3. Sub-grid scale models

As discussed in the introduction section, many factors affect the quality of LES, including

- The filter width Δ
- The space discretization scheme
- The time marching approach
- The mesh size h and time step
- The sub-grid-scale (SGS) stress model
- The initial and boundary conditions.

There are two fundamentally different length scales in LES: the mesh size h and the filter width Δ. When $\Delta \gg h$, the truncation error is small relative to the SGS modeling error. On the other

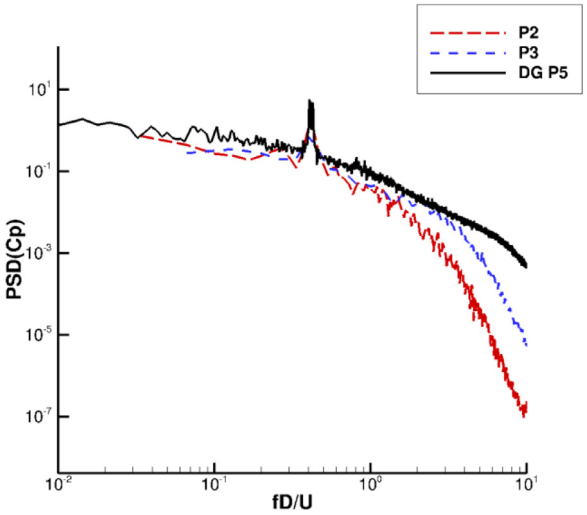


Fig. 9. Comparison of the power spectral densities of the pressure coefficient at x/D = 1.54, y = 0.

hand, when $\Delta \approx h$, the truncation error is dominant and the SGS error is relatively small. This point was amply demonstrated with a study of the filtered Burger's equation in [21]. Several well-known SGS models including the static Smagorinsky (SS) model [33], the dynamic Smagorinsky (DS) model [7], the scale similarity model (SSM) and the mixed model (MM) of Bardina [1], the linear unified RANS-LES model (LUM) [9] are applied to the filtered 1D Burger's equation

$$\frac{\partial \hat{u}}{\partial t} + \hat{u}\frac{\partial \hat{u}}{\partial x} = \nu \frac{\partial^2 \hat{u}}{\partial x^2} - \frac{\partial \tau^{SGS}}{\partial x}, \tag{9}$$

where $\hat{u}$ is the spatially filtered state variable, ν is viscosity, and τ^{SGS} the SGS stress defined by

$$\tau^{SGS} = \frac{1}{2}\widehat{\hat{u}\hat{u}} - \frac{1}{2}\hat{u}\hat{u}. \tag{10}$$

Since the term $\widehat{\hat{u}\hat{u}}$ is not available, a SGS model is needed to close the equation. In ILES, the SGS stress is assumed zero. The filtered Eq. (9) is then integrated in time with a multi-scale initial condition mimicking a turbulent spectrum. Both a priori and a posteriori studies of the SGS models were conducted. Fig. 1 shows the initial condition and its energy spectrum.

In an a priori study, the initial data as shown in Fig. 1 is considered a DNS solution, and filtered with a low-pass filter of width Δ to produce the LES solution. Then the LES solution is employed to compute the SGS stress using various SGS models. These stresses are called modeled stresses. Since the DNS solution is available, it can be used to compute the actual SGS stress, which is called true stress. By comparing the modeled stress with the true stress, one can get a sense on how physical the SGS model is.

In an a posteriori study, a LES is performed with a given SGS model until a certain time. Then this LES solution is used to compute the SGS stress. In order to obtain the true SGS stress at the time, a DNS is also carried out until the same time. This DNS solution is used to compute the true SGS stress, which can be used to evaluate the performance of SGS models. In addition, one can also obtain the "truth" LES solution by filtering the DNS solution. Hence one can also compare the actual LES solution against the "truth" LES solution.

To evaluate the performance of various SGS models, Fig. 2 displays the true stress and the modelled stresses in an a priori study. We can easily draw some conclusions from this figure:

- No models are able to predict the true stress in both the amplitude and phase (peaks and valleys).

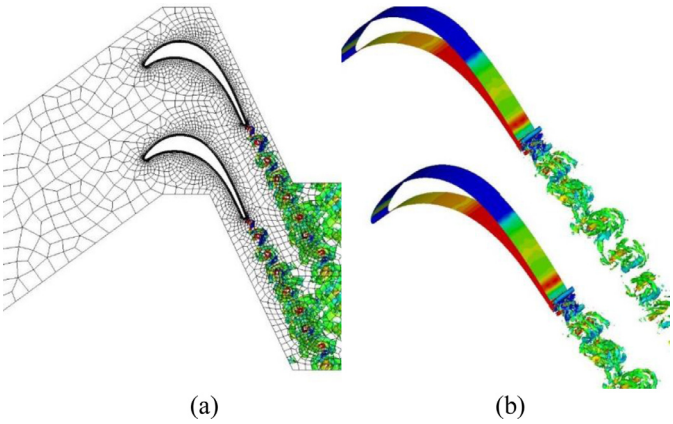


Fig. 10. Computational mesh for the T106A LPT Blade Test Case and the Iso-surfaces of Q-criteria colored by the spanwise vorticity.

- Both the SSM and MM correctly predict the phase of the true stress.
- SS correctly predicts the phase of the true stress about half the time. Since the stress is either positive or negative in 1D, this means the correct phase prediction is by chance, and not physical at all.
- DS agrees with the SS when the phase is correct. When the stress computed with SS has a wrong sign, DS sets the stress to 0.
- LUM agrees very well with SS in the SGS prediction.

Obviously, the good phase prediction capability of the MM is due to the dominant SSM term. The recommendation from this study is to use ILES when $\Delta \approx h$, and SSM when $\Delta \gg h$. In a real-world LES, the former is almost always true. Therefore, ILES is recommended.

4. High-order mesh generation

Many production simulations with a 2nd order method require tens or hundreds of millions of cells to produce results of engineering accuracy. Some mistakenly believe that high-order simulations would need meshes of similar size. Because high-order methods took much more CPU time than low order methods on the same mesh, high-order methods were sometimes dismissed as prohibitively expensive. In reality, high-order methods are capable of achieving similar accuracy on a much coarser mesh than low order methods [44]. Therefore, meshes with only tens or hundreds of thousands of elements may be adequate for a high-order simulation. For such a coarse mesh, it is critical to represent curved boundaries with high-order polynomials to achieve high overall accuracy in the simulation.

High-order mesh generation poses two new challenges. First, it is more difficult to generate coarse meshes for a complex geometry as automated mesh generation algorithms can break down when generating surface meshes at regions with high curvature. Second, generating highly clustered viscous meshes near a curved wall is daunting as interior mesh lines can cross the curved boundary, or intersect each other. Curved interior elements are necessary to remove possible crossings [32,29,17].

Several approaches have been used to overcome some of the difficulties. Many research groups generated fine multi-block structured meshes first. Then these fine meshes are merged once or twice to produce high-order quadratic (degree 2) or quartic (degree 4) quadrilateral and hexahedral meshes. This approach was adopted by HOPR (https://www.hopr-project.org). Although it enabled high-order simulations to be carried out, it is time consum-

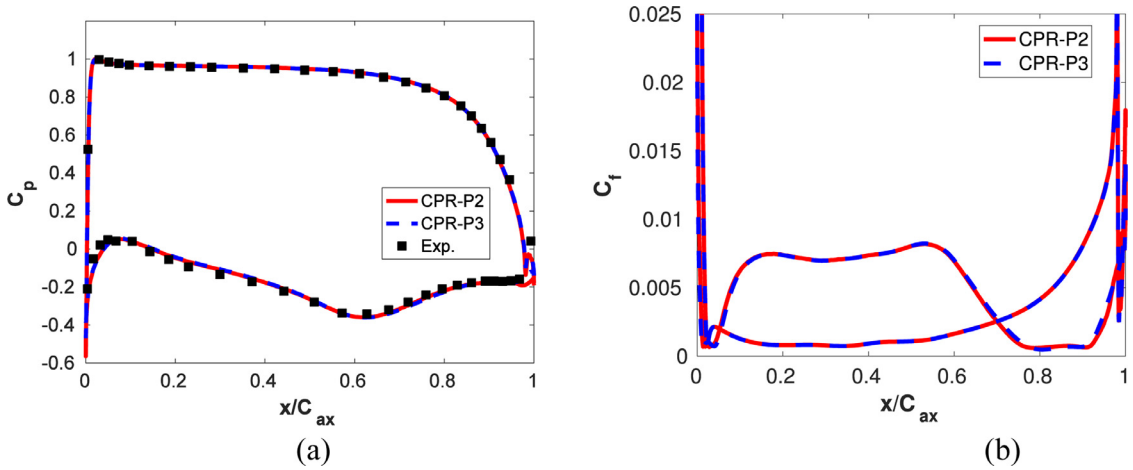


Fig. 11. Mean surface pressure coefficient skin friction coefficient by CPR-P2 and CPR-P3.

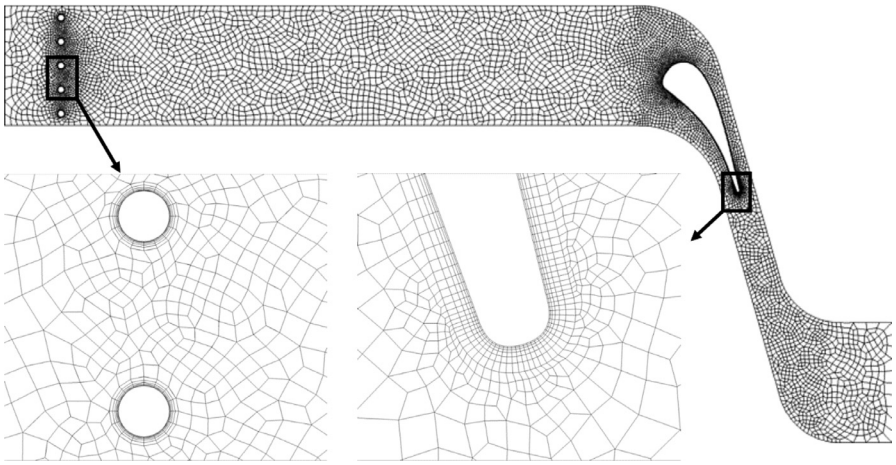


Fig. 12. FR/CPR unstructured computational mesh for the uncooled VKI S1N.

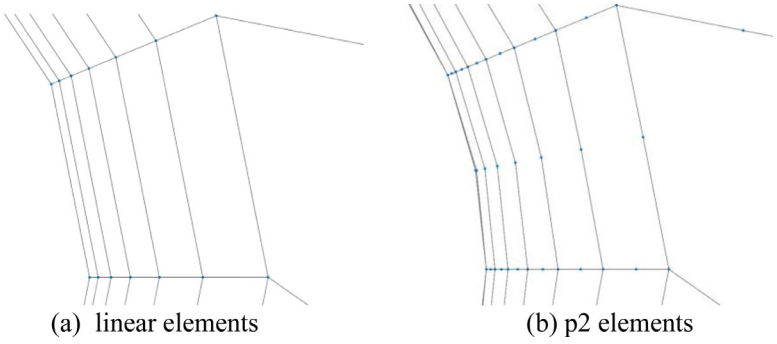


Fig. 13. Elements before and after curving process via meshCurve around a cylinder.

ing to generate structured meshes for complex geometries, and is not, therefore, a long term industrial solution. Another approach is to generate a linear mesh as coarse as possible using commercial mesh generators. Then the elements with a curved wall boundary are made high order by generating curved edges and surfaces, as shown in Fig. 3. Finally, the interior elements are also curved to avoid grid lines crossing into each other, thus ensuring the positivity of the Jacobian of the geometric transformation [29,17]. In case the geometry is not available, a surface reconstruction technique [16] is used to rebuild a high-order surface before the surface and volume meshes are curved. A tool named meshCurve has been recently developed [15] at the University of Kansas to convert a linear mesh into a high order mesh. MeshCurve can automatically detect and preserve geometrically important features such as sharp edges and corner points. An example of triangular surface reconstruction is displayed in Fig. 4. Note that the sharp edges are preserved, and upgraded into higher order polynomials.

Here is a list of basic features of meshCurve:

- Input a linear mesh through an international CFD data standard, CGNS.
- Select boundaries of the linear mesh which are to be upgraded to high-order.
- Automatically detect critical points and edges of the selected boundaries.

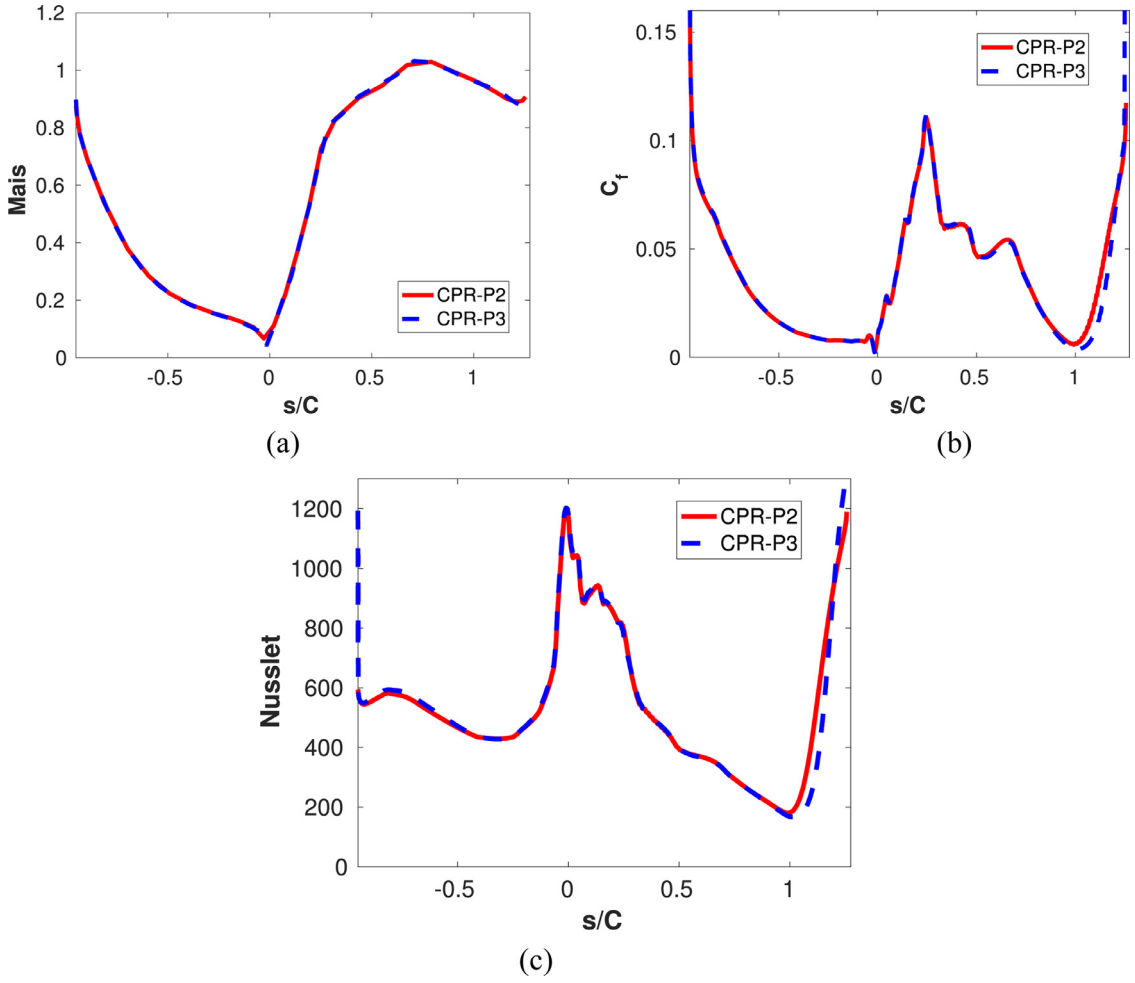


Fig. 14. Isentropic Mach number, skin friction coefficient and Nusselt number by CPR-P2 and CPR-P3.

- Perform a least squares reconstruction to upgrade the linear patches to quadratic ones.
- Curve the interior meshes to avoid negative Jacobians.
- Save the high-order mesh in CGNS format.

Now, only quadratic elements are supported. We plan to extend the reconstruction to produce cubic and quartic elements in the future. To encourage research on high-order methods, the tool has been released to the world here (https://sites.google.com/site/meshcurvesoftware/).

5. Demonstration cases

We present results for several demonstrations cases with two from the 4th International Workshop on High-Order CFD Methods (http://how4.cenaero.be).

5.1. Flow over a cylinder at Re= 3900

Flow over a cylinder at a Reynolds number of 3900 based on the diameter is a classical test case for LES codes. Researchers also discovered that there are non-unique modes in long term simulations [41]. The case was selected as a benchmark problem by the Workshop with an infinitely smooth initial condition. Thus, the solution before turbulence starts can be used to measure space and time accuracy of numerical methods. We selected this case here to show hp-convergence in a hp-refinement study. Coarse, medium and fine meshes were also provided by the workshop. Mesh and

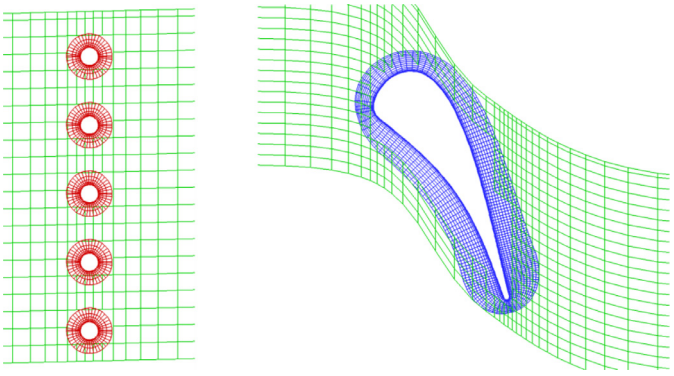


Fig. 15. Zoomed in view around the blade and turbulence grid of the overset mesh used for the fine mesh FDL3Di calculation. Only every 10th mesh point is shown for clarity.

order refinement studies were performed to compute the C_L and C_D at the non-dimensional time of t/T = 1. The converged C_L and C_D are 0.070603 and 0.151505 respectively, which were verified by the CFD groups in Imperial College (Brian Vermeire) using the FR/CPR method and University of Stuttgart (Andrea Beck) using the DG method. The C_D errors with different mesh resolutions and polynomial orders (p) are displayed in Fig. 5. Good mesh convergence is shown in this figure for all polynomial orders.

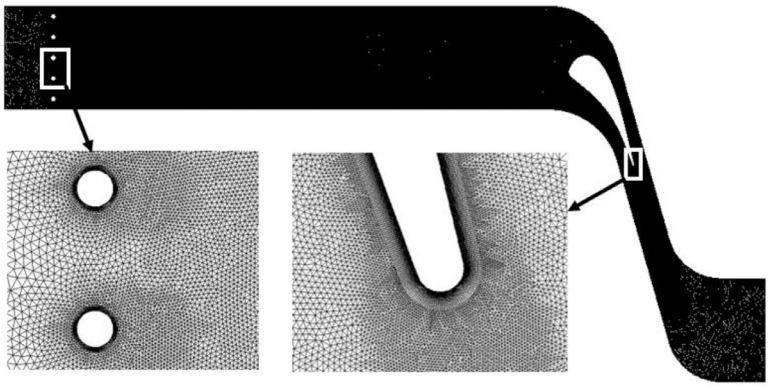
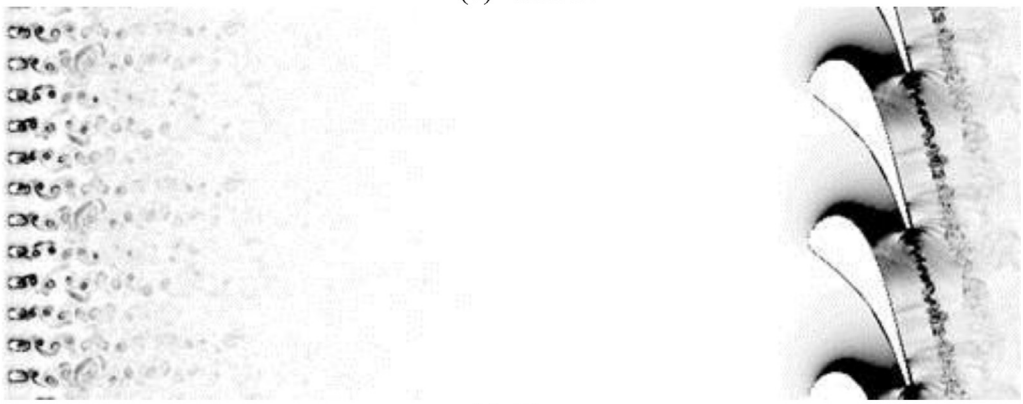


Fig. 16. Unstructured computational mesh for the uncooled VKI S1N for the Fluent simulation.

(a) FDL3DI

(b) CPR-P5



(c) Fluent

Fig. 17. Comparison of computational Schlieren distributions ($c|\nabla\rho|/\rho$, c being the chord length, from 0 to 1).

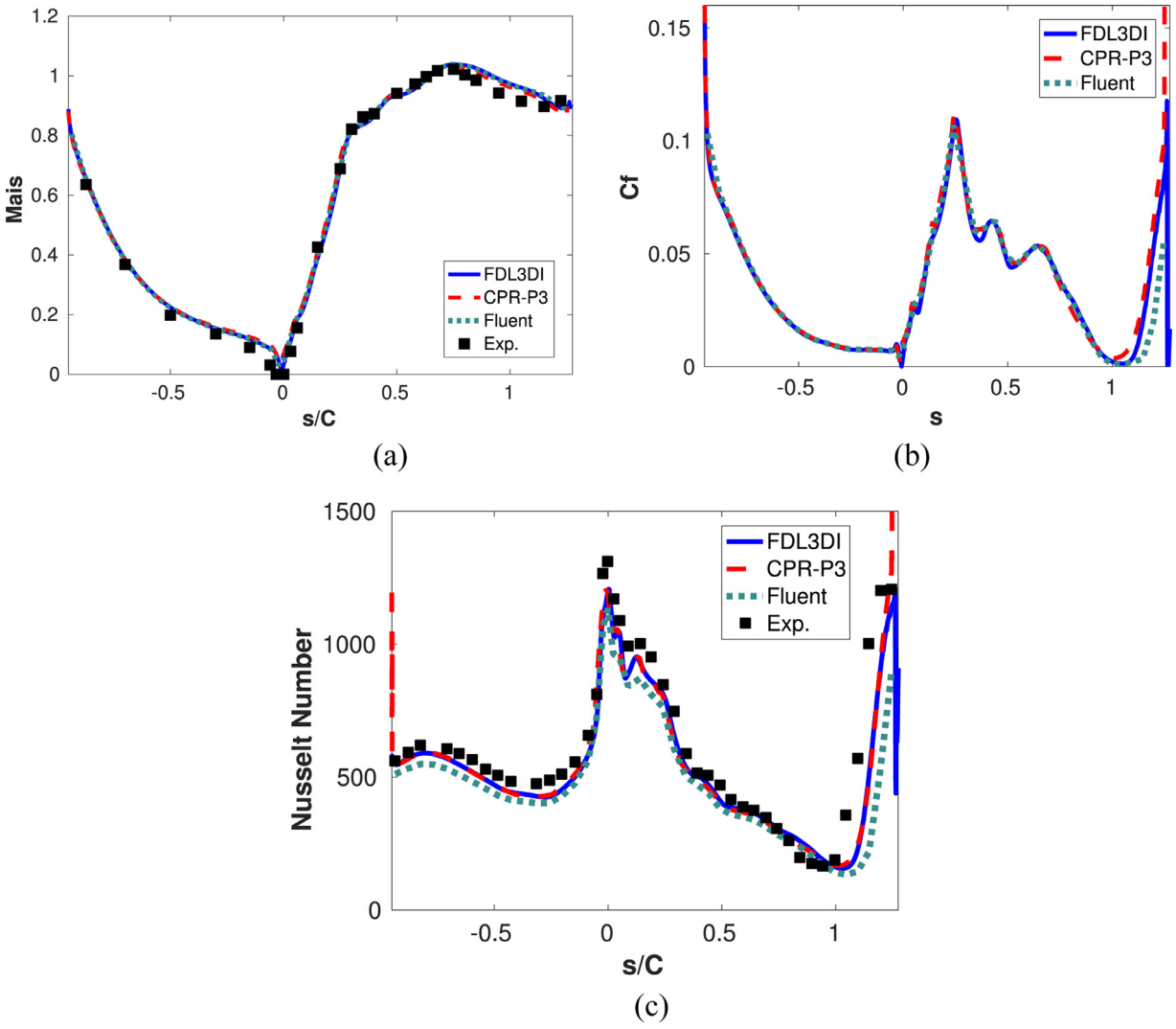


Fig. 18. Comparison of surface isentropic Mach number, skin friction coefficient, heat transfer.

Long term simulations were then carried out on the medium mesh for p = 2 and 3 because of resource limitations. The C_D histories are compared in Fig. 6. It appears that the flow reached a statistically mean state after t/T = 100. Time averages were performed starting from t/T = 100 until 800. The mean streamwise velocity profiles at a streamwise location x/D = 0.58, and the mean pressure coefficients on the cylinder surface are compared in Fig. 7 with a DG p5 solution provided by Andrea Beck of University of Stuttgart. The Reynolds stresses at x/D = 1.54 are compared in Fig. 8. The power spectral densities of c_p at x/D = 1.54, y= 0 are shown in Fig. 9. There is still p-dependence at this resolution, and there is a clear convergence towards the higher resolution results.

5.2. Flow over a T106A low pressure turbine cascade

This case was again a benchmark problem from the Workshop, and a high-order mesh was also provided, as shown in Fig. 10a, which has 14,035 p2 hexahedral elements with 5 elements in the spanwise direction. We use FR/CPR schemes of p = 2 to p = 4 (3rd to 5th order) to assess the p-dependence of the simulations, resulting in 379 K, 898 K, 1754 K DOFs per equation. The average y+ value at the first element is around 4.9 based on the cell size in the wall normal direction.

The simulation started at p = 0 (or p0), and then restarted at p1 to establish a good initial condition for p2. The p2 simulation restarted from the p1 solution, and the averaging process began when the initial transient passed the computational domain, and it was made sure the mean flow sufficiently converged. The instantaneous isosurfaces of the Q-criterion colored by the spanwise vorticity is shown in Fig. 9b for p = 3, showing the laminar flow on the airfoil, and its breakdown and transition to turbulence close to the trailing edge. The mean pressure coefficient c_p and skin friction coefficient c_f on the airfoil surface at p2 and p3 are displayed in Fig. 11a and b respectively, together with experimental data. It clearly shows the p-convergence for c_p and c_f, because the p2 and p3 results are already on top of each other. The agreement with the experimental data [36] is very good.

5.3. Uncooled VKI vane high pressure turbine case

This problem was selected from [18,19] to demonstrate the capability of several different solvers including one commercial 2nd order code, and two high-order codes for turbomachinery flow problems. The isentropic exit Reynolds number is approximately 584,000 based on the blade chord length, and the isentropic exit Mach number is 0.92. Several different views of the unstructured hexahedral mesh for the FR/CPR solver are displayed in Fig. 12. In

the upstream, stationary circular bars were employed to generate flow turbulence. The linear mesh was generated with Gmsh [8], and then was curved using meshCurve [15] to generate a quadratic mesh. The final curved mesh contains 264,640 p2 hexahedral elements with 32 elements in the spanwise direction. The elements around the cylinders are also shown before and after curving in Fig. 13a and b respectively. Note that because of the limitation of the visualization package, the p2 edge seems to have 2 straight segments. Actually, each side is a degree 2 polynomial. The average y+ value is around 4–5 (based on the cell size in the wall normal direction) to obtain an accurate skin friction prediction. FR/CPR schemes of $p=2$ to $p=5$ (3rd to 6th order) were used in the simulations to assess p-dependence. Therefore, the total numbers of DOFs for $p=2$ to $p=5$ are 7.1 M, 16.9 M, 33.1 M, and 57.2 M per equation respectively. The simulations started at p0, and proceeded to p1, p2, et al. An accuracy-preserving limiter [22] was employed in the simulations to capture the shock-waves. The C_L and C_D histories were monitored to determine a time to start averaging. Roughly 15 non-dimensional time units were run before time-averaging was performed for another 3 time units. It was made sure that the time averaged flow field reached eye-ball convergence (1–2%). The mean isentropic Mach number, skin friction coefficient (c_f) and Nusselt number profiles computed with p2 and p3 along the airfoil surface are compared in Fig. 14a-c respectively. Results for p4 and p5 are on top of each other, and are therefore not plotted. The figure clearly shows excellent p-convergence.

Another high-order code FDL3DI [42] based on the compact finite difference method with overset meshes was also used in the present simulation. The overset mesh [4,26,27] is shown in Fig. 15, and has 110 million mesh points (or DOFs per equation). This solver employs structured overset grids. It uses a sixth order compact finite difference scheme in space along with an implicit Beam-Warming scheme for time advancement.

In order to evaluate the performance of a production 2nd order code, Fluent was also used. The computational mesh has 135 million elements, and is shown in Fig. 16 [18,19]. The computational Schlieren distributions from FDL3DI, FR/CPR and Fluent are compared in Fig. 17. There is an excellent visual agreement between the two high-order simulations with fine flow features well resolved such as the standing shock waves on the top of the airfoil, and the acoustic waves radiating from the trailing edge. These fine features are not resolved as well by the Fluent simulation primarily due to its 2nd order accuracy.

The surface isentropic Mach number, skin friction and heat transfer predicted with all three solvers are compared with experimental data in Fig. 18. Again there is a very good overall agreement between the experimental data and various simulations. The two high-order codes, although employing very different numerical methods, showed excellent agreement in the heat transfer prediction.

6. Conclusions and future work

We presented several recent progresses which have enabled large eddy simulations using high-order FR/CPR schemes to be conducted for multi-scale benchmark flow problems. These recent developments include high-order mesh generation, subgrid-scale models, and wall models for high Reynolds number flow problems. With these progresses, accurate large eddy simulations are becoming feasible. Comparisons with experimental data, a commercial 2nd-order solver Fluent, and another high-fidelity simulation code, FDL3DI indicate that the unstructured grid based FR/CPR solver is capable of providing highly accurate predictions in aerodynamic loads as well as in heat transfer. A detailed comparison on cost and efficiency will be carried out in a future publication.

Future developments will include research on more efficient time marching algorithms for extreme-scale parallel computers, as well as improved wall models for high Reynolds number flow problems.

Acknowledgements

The research outlined in the present paper was supported by NASA under NNX12AK04A, AFOSR under grant FA9550-16-1-0128, US Army Research Office under grants W911NF-15-1-0505 and W911-NF-1510377, as well as GE Global Research. The VKI vane LES simulations were run on National Center for Supercomputing Applications (NCSA) Blue Waters and the authors would like to thank the Private Sector Program and the Blue Waters sustained petascale computing project at NCSA. Blue Waters is supported by the National Science Foundation (award numbers OCI 07–25070 and ACI-1238993) and the state of Illinois.

References

[1] Bardina J, Ferziger JH, Reynolds WC. Improved subgrid scale models for large eddy simulation. Am Inst Aeronaut Astronaut 1980:80–1357.
[2] Bassi F, Rebay S. A high order discontinuous Galerkin method for compressible turbulent flows. In: Cockburn B, Karniadakis G, Shu C-W, editors. Discontinuous Galerkin Methods: Theory, Computation, and Application. Lecture Notes in Computational Science and Engineering. Springer; 2000. p. 77–88.
[3] Beck AD, Bolemann T, Flad D, Frank H, Gassner GJ, Hindenlang F, et al. High-order discontinuous Galerkin spectral element methods for transitional and turbulent flow simulations. Int J Numer Methods Fluids 2014;76(8):522–48.
[4] Bhaskaran R., Wood T.H., Paliath U., Breeze-Stringfellow A. Towards large eddy simulation of a 3D transonic fan. 46th AIAA Fluid Dynamics Conference, AIAA Aviation Forum and Exposition, Washington D.C., 2016–3816.
[5] Boris JP, Grinstein FF, Oran ES, Kolbe RL. New insight into large eddy simulation. Fluid Dyn Res 1992;10:199–228.
[6] Cockburn B, Karniadakis GE, Shu C-W. Discontinuous Galerkin Methods. Theory, Computation and Applications, Lecture Notes in Computational Science and Engineering. eds. Berlin: Springer-Verlag; 2000.
[7] Germano M, Piomelli U, Moin P, Cabot WH. A dynamic subgrid-scale eddy viscosity model. Phys Fluids A 1991;3:1760–5.
[8] Geuzaine C, Remacle JF. Gmsh: a 3-D finite element mesh generator with built-in pre-and-post-processing facilities. Int J Numer Methods Eng 2009;79(11):1309–31.
[9] Gopalan H, Heinz S. A unified RANS-LES model: computational development, accuracy and cost. J Comput Phys 2013;249:249–74.
[10] Gottlieb S, Shu C-W. Total variation diminishing Runge-Kutta schemes. Math Comput 1998;67:73–85.
[11] Grinstein F, Margolin L, Rider W. Implicit Large Eddy Simulation. Cambridge University Press; 2007.
[12] Haga T, Gao H, Wang ZJ. A high-order unifying discontinuous formulation for the Navier–Stokes equations on 3D mixed grids. Math Model Nat Phenom 2011;6(3):28–56.
[13] Huynh HT. A flux reconstruction approach to high-order schemes including discontinuous Galerkin methods. AIAA Pap 2007 4079.
[14] Huynh HT, Wang ZJ, Vincent PE. High-order methods for computational fluid dynamics: a brief review of compact differential formulations on unstructured grids. Comput Fluids 2014;98:209–20.
[15] Ims J, Duan Z, Wang ZJ. MeshCurve: an automated low-order to high-order mesh generator. AIAA 2015 2293.
[16] Jiao X, Wang D. Reconstructing high-order surfaces for meshing. Eng Comput 2012;28(4):361–73.
[17] Karman SL, Erwin JT, Glasby RS, Stefanski DL. High-order mesh curving using WCN optimization. AIAA 2016 3178.
[18] Kopriva J.E., Laskowski G.M., Sheikhi M.R.H. Hybrid LES of a high pressure turbine nozzle/blade interaction. DLES-10, Limassol, Cyprus, 2015.
[19] Kopriva J.E., Laskowski G.M., Shiekhi M.R.H. Computational assessment of inlet turbulence on boundary layer development and momentum/thermal wakes for high pressure turbine nozzle and blade. IMECE2014-38620, 2014.
[20] Laskowski GM, Kopriva J, Michelassi V, Shankaran S, Paliath U, Bhaskaran R, et al. Future directions of high fidelity CFD for aerothermal turbomachinery research, analysis and design. AIAA 2016 3322.
[21] Li Y, Wang ZJ. A priori and a posteriori evaluations of sub-grid scale models for the Burgers' equation. Comput Fluids 2016;139:92–104.
[22] Li Y, Wang ZJ. Recent progress in developing a convergent and accuracy preserving limiter for the FR/CPR method. 55th AIAA Aerospace Sciences Meeting, AIAA SciTech Forum. AIAA; 2017. 0756).
[23] Liu Y, Vinokur M, Wang ZJ. Discontinuous spectral difference method for conservation laws on unstructured grids. J Comput Phys 2006;216:780–801.
[24] Lodato G, Castonguay P, Jameson A. Structural wall-modeled LES using a high-order spectral difference scheme for unstructured meshes. Flow Turbulence Combust 2014;92(1-2):579–606.

[25] Lu Y., Liu K., Dawes W.N. Flow simulation system based on high order space-time extension of flux reconstruction method, 53rd AIAA aerospace sciences meeting, AIAA SciTech forum, (AIAA 2015-0833).
[26] Paliath U, Premasuthan S. Large eddy simulation for jet installation effects. 19th AIAA Aeroacoustics Conference. AIAA; 2013. 2137.
[27] Paliath U, Shen H, Avancha R, Shieh C. Large eddy simulation for jets from chevron and dual flow nozzle. 17th AIAA Aeroacoustics Conference. AIAA; 2011. 2881.
[28] Park JS, You H, Kim C. Higher-order multi-dimensional limiting process for DG and FR/CPR methods on tetrahedral meshes. in press. Comput Fluids 2017.
[29] Persson PL, Peraire J. Curved mesh generation and mesh refinement using Lagrangian solid mechanics. AIAA 2009 949.
[30] Piomelli U, Balaras E. Wall-layer models for large eddy simulations. Ann Rev Fluid Mech 2002;34(1):349–74.
[31] Pope SB. Ten questions concerning the large eddy simulation of turbulent flows. New J Phys 2004;6(1):35.
[32] Sherwin SJ, Peiro J. Mesh generation in curvilinear domains using high-order elements. Int J Numer Methods Eng 2000;53(1):207–23.
[33] Smagorinsky J. General circulation experiments with the primitive equations. I Basic Exp Mon Weather Rev 1963;91(3):99–164.
[34] Spalart PR, Allmaras SR. A one-equation turbulence model for aerodynamic flows. AIAA J 1992 94.
[35] Spalart PR, Deck S, Shur ML, Squires KD, MKh S, Travin A. A new version of detached-eddy simulation, resistant to ambiguous grid densities. Theor Comput Fluid Dyn 2006;20(3):181–95.
[36] Stadtmüller P. Investigation of wake-induced transition on the LP turbine cascade T106A-EIZ. DFG-Verbundprojekt Fo. 2001;136(11).
[37] Tam CKW. Computational aeroacoustics: issues and methods. AIAA J 1995;33(10):1788–96.
[38] Uranga A, Persson PO, Drela M, Peraire J. Implicit large eddy simulation of transition to turbulence at low Reynolds numbers using a discontinuous Galerkin method. J Numer Meth Eng 2011;87:232–61.
[39] Vincent PE, Castonguay P, Jameson A. A new class of high-order energy stable flux reconstruction schemes. J Sci Comput 2011;47(1):50–72.
[40] Vermeire BC, Nadarajah S, Tucker PG. Implicit large eddy simulation using the high-order correction procedure via reconstruction scheme. J Numer Meth Fluids 2016;82:231–60.
[41] Vermeire BC, Witherden FD, Vincent PE. On the utility of GPU accelerated high-order methods for unsteady flow simulations: a comparison with industry-standard tools. J Comput Phys 2017;334:497–521.
[42] Visbal MR, Gaitonde DV. On the use of higher-order finite-difference schemes on curvilinear and deforming meshes. J Comput Phys 2002;181(1):155–85.
[43] Wang M, Moin P. Dynamic wall modeling for large-eddy simulation of complex turbulent flows. Phys Fluids 2002;14(7):2043–51.
[44] Wang ZJ. High-order methods on unstructured grids for Navier–Stokes equations. J Prog Aerosp Sci 2007;43(1):1–41.
[45] Wang ZJ. A perspective on high-order methods in computational fluid dynamics. Sci Chi Phys Mech Astron 2016;59(1):1–6.
[46] Wang ZJ, Fidkowski KJ, Abgrall R, Bassi F, Caraeni D, Cary A, et al. High-order CFD methods: current status and perspective. Int J Numer Methods Fluids 2013;72(8):811–45.
[47] Wang ZJ, Gao H. A unifying lifting collocation penalty formulation including the discontinuous Galerkin, spectral volume/difference methods for conservation laws on mixed grids. J Comput Phys 2009;228(21):8161–86.
[48] Yu ML, Wang ZJ, Liu Y. on the accuracy and efficiency of discontinuous galerkin, spectral difference and correction procedure via reconstruction methods. J Comput Phys 2014;259:70–95.
[49] Zhu H, Fu S, Shi L, Wang ZJ. Implicit large-eddy simulation for the high-order flux reconstruction method. AIAA J 2016;54(9):2721–33.

Chapter 5 Implementation on Multi-GPU Parallel Architecture

1 Introduction

Accurate and efficient numerical methods for unstructured mixed meshes are required for a wide range of industrial computational fluid dynamics applications. The current generation of industry-standard CFD software is predominantly based on second-order accurate Finite Volume (FV) methods with the Reynolds Averaged Navier-Stokes (RANS) approach, due to their robustness and ability to work on unstructured mixed meshes of complex configurations. The low order FV methods have been successful for steady problems, but they are not as accurate or efficient as high-order methods to perform scale-resolving simulations of turbulent flows, such as large eddy simulation (LES) of flow through a jet engine or over aircraft near the edge of the flight envelope at high angles of attack [1]. Therefore, over the past decade, there has been a surge in the development of high-order unstructured methods that are at least third-order accurate in space to solve such problems [2,3]. Compared to the conventional second-order methods, high-order methods have been shown to provide improved accuracy at reduced computational cost for a wide range of applications. Popular high-order methods include the discontinuous Galerkin (DG) method, spectral volume, and spectral difference methods, amongst others [4,5]. In 2007, Huynh proposed the flux reconstruction or correction procedure via reconstruction (FR/CPR) approach, which is a common framework that unifies nodal DG, SV, SD schemes at least for the case of linear equations [6]. The FR/CPR was subsequently extended to three-dimensional mixed element types for the simulations of transitional and turbulent flows via scale resolving simulations [7].

Excellent parallel performance for the high-order FR/CPR solver has been obtained on CPU clusters because of the scalability of the FR/CPR method. To take advantage of the most powerful computers in the world, such as DOE's Summit, implementation on GPU clusters is necessary.

Recently, the General-Purpose Graphics Processing Unit (GPGPU) technology has earned increasing attention in science and engineering simulations. Because GPUs usually have thousands of computing cores per device, they are typically capable of achieving much higher floating-point performance and memory bandwidth than CPUs. Among all the development tools related to GPGPU technology, the compute unified device architecture (CUDA), introduced by NVIDIA in 2007, marked the beginning of widespread use of GPUs in general-purpose computing. By providing a minimal set of C/C++ language extension, the CUDA platform not only reduces the complexity of programming but also allows much flexibility to programmers. This makes CUDA appealing for developing high performance scientific programs for large-scale computing.

Currently, GPUs have been incorporated into many fields, such as molecular dynamics (MD), direct simulation Monte Carlo (DSMC), weather forecast, CFD and artificial intelligence (AI). In the field of CFD, several numerical methods, including finite difference (FD), finite volume (FV), spectral difference (SD), discontinuous Galerkin methods, Lattice Boltzmann methods and more, have been ported to GPU based on CUDA platform [8-12]. Compared with the standard FV or FD, the FR/CPR method has better locality and compactness especially for higher orders of accuracy, which makes it an extremely good candidate for acceleration using the massive parallel computing units in GPUs.

The main purpose of this paper is to design an efficient multi-GPU parallel algorithm to accelerate our 3D CPU-based Navier-Stokes high order FR/CPR explicit solver on mixed high order meshes via the CUDA platform. In Section 2, several key points to implement a GPU-friendly FR/CPR approach are explained. In Section 3, we evaluate the performance speedup

obtained withby GPUs using a benchmark flow over a low pressure turbine. Finally, in Section 4 conclusions are drawn and the future work is outlined.

2 Numerical method and implementation

Here, the formula of FR/CPR scheme is given for simplicity. In the i-th element, the final semi-discretization formulation for each solution point j is

$$\frac{\partial \mathbf{U}_{i,j}}{\partial \mathrm{t}} + \Pi_j[\nabla \cdot \mathbf{F}(\mathbf{U}_i)] + \delta_{i,j} = 0. \tag{1}$$

where Π_j denotes a projection to the polynomial space, and subscript j denotes the j-th solution point in a certain element. A "correction field" δ_i is defined in each element as

$$\int_{V_i} W\delta_i \, \mathrm{d}\Omega = \int_{\partial V_i} W[F^n] \, \mathrm{dS}, \tag{2}$$

where $[F^n] = [F^n_{com} - F^n(\mathbf{U}_i)]$ is the normal flux jump. It is easy to see that the second term in left hand side (LHS), called residual due to local gradient, is only related to the local solution in i-th element, while the third term, called residual due to correction, is related with the fluxes on the interfaces between itself and its neighbors, which is not local.

For the multi-GPU implementation, we follow the flow chart shown in Figure 1. The overall design of the GPU capability is to offload the computation-intensive portions of the application to the GPU, while the remainder of the program still runs on the CPU. Each CPU is responsible for one GPU card and the multi-GPU computing ability is obtained through the message passing interface (MPI). The CPU is still responsible for input and output, andinitialization, including computing mesh metrics as well as assigning the initial solution. Then GPU memory is allocated to store solutions, fluxes, etc. The metrics information is also copied from CPU to GPU in this

initialization stage. After that, it goes into the time marching loop. We employ the 3 stage Runge-Kutta (RK3) scheme in this paper. During each sub-step of RK3, it contains 3 steps. The first step is to apply the boundary condition as well as interpolate solutions to interfaces. Then in the second step, two independent procedures are overlapped by CUDA multi-stream technique to improve the parallel efficiency. One procedure is to compute the residual due to the local gradient on GPU, and the other procedure is to copy necessary boundary data from GPU to CPU for MPI communications and then copy back from CPU to GPU. After the time marching is finished, all solutions are copied from GPU to CPU for post-processing.

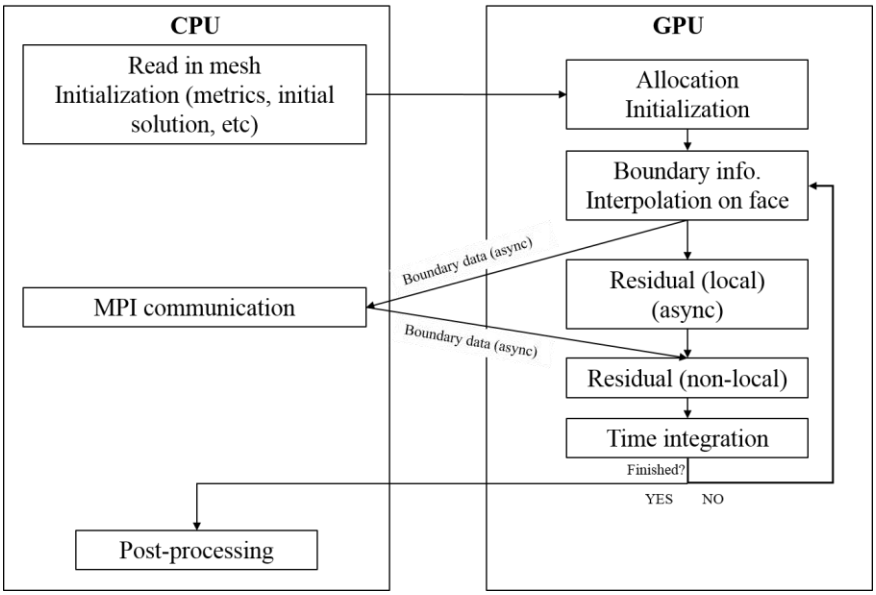


Figure 1. Multi-GPU design using the CUDA multi-stream technique..

3 Results

A benchmark problem of flow over a T106A low-pressure turbine cascade from the 4th International Workshop on High-Order Methods (https://how4.cenaero.be/) is chosen to test the multi-GPU solver. The mesh is a p2 curved mesh provided by the workshop, which has 14,035 p2 hexahedral elements with 5 elements in the spanwise direction. The isentropic Mach number at

the exit is 0.4 and the Reynolds number is 60,000 based on the chord and exit condition. The span is 10% of the chord. The incoming flow angle of attack is 46.1° with respect to the positive x direction with no incoming turbulence. The verification of the GPU code and the CPU code is completed by monitoring residuals at several time steps, which is shown in Table 1. Double precision accuracy was used and the polynomial order was P^2. The residual at all time can be recovered for at least 13 digits, ensuring that the GPU code produces the same results as the CPU code.

Table 1 GPU code verification (P^2)

Iteration	CPU residual	GPU residual
1	6.676897852446208e+02	6.676897852446209e+02
50	6.640058726471827e+02	6.640058726471859e+02
100	6.604041215405563e+02	6.604041215405637e+02

To test the benefits of the overlapping local residual calculation and MPI communication, the case was ran on up to 16 GPUs. For each case, the same number of GPUs and CPUs are compared to calculate the speedup. The results are shown in Figure 2. Without overlapping, the speedup goes down rapidly with the increasing number of GPUs, but with overlapping, it can still be kept around 55 on 16 GPUs (875 element per GPU). It is demonstrated that the CUDA multi-stream really overlaps the communication and computation, and improves the scalability.

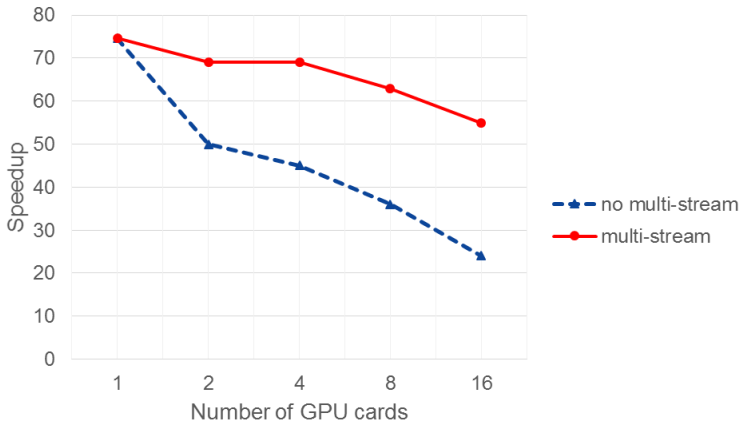


Figure 2. Speedup of a multi-GPU simulation

4 Conclusions and future work

This work has demonstrated an efficient implementation of he high-order FR/CPR method on a cluster of GPUs. Applying GPU computing to a high-order CFD method leads to a huge reduction in computational cost when compared to the parallel CPU implementation. With the help of the CUDA multi-stream technique, the speedup of multi-GPU computing can still be about 55 with less than 1000 elements per card. The future work is to extend the current inviscid multi-GPU solver on hexahedral meshes to a viscous multi-GPU Navier-Stokes solver on mixed meshes.

References

1. Laskowski GM, Kopriva J, Michelassi V, Shankaran S, Paliath U, Bhaskaran R, Wang Q, Talnikar C, Wang ZJ, Jia FL. Future Directions of High Fidelity CFD for Aerothermal Turbomachinery Analysis and Design. In: 46th AIAA Fluid Dynamics Conference; 2016. p. 3322.

2. Wang ZJ. High-order methods on unstructured grids for Navier–Stokes equations. Prog Aerosp Sci 2007;43(1-3):1-41.
3. Wang ZJ, Fidkowski K, Abgrall R, Bassi F, Caraeni D, Cary A, … Kroll N. High-order CFD methods: current status and perspective. Int J Numer Methods Fluids 2013;72(8):811-845.
4. Cockburn B, Karniadakis G, Shu CW. Discontinuous Galerkin Methods: Theory, Computation and Applications: Theory, Computation and Applications 2000.
5. Liu Y, Vinokur M, Wang ZJ. Discontinuous spectral difference method for conservation laws on unstructured grids. Computational Fluid Dynamics 2004. p. 449-454.
6. Huynh HT. A flux reconstruction approach to high-order schemes including discontinuous Galerkin methods. In: 18th AIAA Computational Fluid Dynamics Conference 2007. p. 4079.
7. Wang ZJ, Gao H. A unifying lifting collocation penalty formulation including the discontinuous Galerkin, spectral volume/difference methods for conservation laws on mixed grids. J Comput Phys 2009;228(21):8161-8186.
8. Vermeire BC, Witherden FD, Vincent PE. On the utility of GPU accelerated high-order methods for unsteady flow simulations: A comparison with industry-standard tools. J Comput Phys 2017;334(1):497-521.
9. Witherden FD, Vermeire BC, Vincent PE. Heterogeneous computing on mixed unstructured grids with PyFR. Computers & Fluids. 2015; 120:173-86.
10. Lai J, Li H, Tian Z, Zhang Y. A Multi-GPU Parallel Algorithm in Hypersonic Flow Computations. Mathematical Problems in Engineering. 2019;2019.
11. Loppi NA, Witherden FD, Jameson A, Vincent PE. A high-order cross-platform incompressible Navier–Stokes solver via artificial compressibility with application to a turbulent jet. Computer Physics Communications. 2018; 233:193-205.
12. Zimmerman BJ, Wang ZJ. The efficient implementation of correction procedure via reconstruction with graphics processing unit computing. Computers & Fluids. 20 (101:263-72.

Chapter 6 Conclusions and Future Work

In the present study, by developing the FR/CPR method on mixed unstructured meshes including hexahedral, tetrahedral, prismatic and pyramidal elements, the geometric flexibility is greatly enhanced to handle complex configurations in industrial problems. The performances of a commercial second order method with mesh refinement and the high order FR/CPR method with p refinement are evaluated for a well-known benchmark turbomachinery case: transonic flow over a VKI high-pressure turbine vane at a Reynolds number of 1.16×10^6. Through a systematic comparison on many various flow parameters, e.g. mean surface loading, heat transfer, power spectral density (PSD) of pressure at energetic locations, mean boundary-layer velocity and total temperature profiles and wake loss, the 4^{th} order FR/CPR method clearly demonstrates p convergence on the coarse mesh, but the commercial second order method has not obtained mesh convergence yet on the fine mesh (64 times the number of cells of the coarse one). Both the 3^{rd} and 4^{th} order FR/CPR methods captured the transition location very well, while the commercial second order method failed to do so. Additionally, by only using 1/3 of the cost of the commercial second order method on the fine mesh, the 3^{rd} order FR/CPR method on the coarse mesh can achieve finer flow resolution and produce more accurate results.

An implicit time marching scheme, backward difference formulas (BDF) with a non-linear blocked lower-upper symmetric Gauss-Seidel (BLU-SGS) solver, is developed and evaluated for turbo-machinery cases with various Mach and Reynolds numbers to compare the accuracy, efficiency and scalability against the explicit 3-stage Strong Stability Preserving Runge-Kutta (RK3) scheme. Various factors such as inner convergence tolerance, the frequency of the implicit operator update, and the order of time accuracy are studied. It is found that the LUSGS-BDF1 scheme is not accurate enough and can degrade the mean blade loading distributions as well as the PSD. The LUSGS-BDF2 scheme is more adequate and it can use a time-step of 1 to 2 orders larger than the maximum

time step allowed by the RK3 scheme. Although implicit schemes are not very sensitive to the convergence tolerance at each time step, it's better to use 0.01 as the convergence tolerance for both low and high Reynolds number cases to make simulations accurate. For the 3rd order FR/CPR scheme, depending on the Mach number, the LUSGS-BDF2 scheme can run 3 to 15 times faster than the RK3 scheme with satisfied time accuracy.

This study demonstrates the advantage of the high order FR/CPR method in industrial LES over second order finite volume methods by achieving higher accuracy and finer resolution at much less cost. The high order FR/CPR method can perform LES simulations on mixed meshes of real-world engineering problems with a moderate Reynolds number (~1 million) on a relatively small cluster, which represents a first step to the next generation of CFD design tool.

Several potential related areas could be done in the future to enhance the capabilities of the high order methods:

- Offload the intensive computation to the emerging computer architectures like graphical processing unit (GPU).
- Implement the capability to handle sliding meshes to simulate the internal flow in jet-engines.
- Associate mesh adaptation techniques with the high order method to ease the mesh generation process as well as to reduce the mesh size.