AFRL-AFOSR-VA-TR-2015-0403

The Development of High-Order Methods for Real World Applications

Zhi Wang
UNIVERSITY OF KANSAS CENTER FOR RESEARCH INC.

12/03/2015
Final Report

Air Force Research Laboratory
AF Office Of Scientific Research (AFOSR)/ RTA2
Arlington, Virginia 22203
Air Force Materiel Command

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ORGANIZATION.

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
24-11-2015	Final Report	9/1/2012 - 8/31/2015

4. TITLE AND SUBTITLE
The Development of High-Order Methods for Real World Applications

5a. CONTRACT NUMBER
FA9550-12-1-0286

5b. GRANT NUMBER
FA9550-12-1-0286

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Z.J. Wang, L. Shi, Y. Li and J. Ims

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
University of Kansas
2120 Learned Hall
Lawrence, KS 66045

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Office of Scientific Research
875 North Randolph Street, Suite 325
Arlington, VA 22203

10. SPONSOR/MONITOR'S ACRONYM(S)
AFOSR

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Unlimited DISTIRIBUTION A

13. SUPPLEMENTARY NOTES

14. ABSTRACT
With increased computational power and progress in numerical methods over the past several decades, Computational Fluid Dynamics (CFD) is now used routinely as a powerful tool in the design of aircraft. Current production CFD codes used in the aerospace industry are usually second order accurate. High-order methods have the potential to achieve higher accuracy at less cost than low-order methods. This potential has been demonstrated conclusively for smooth problems in the latest International Workshops on High-Order Methods. For non-smooth problems, solution based hp-adaptation offers the best promise. The primary objective of the present study is to develop robust and efficient high-order CFD methods and tools for the compressible Navier-Stokes equations that can provide engineering accuracy for real world industry problems. Several pacing items are addressed, which include hp-adaptations, sub-grid stress models for large eddy simulations, and high-order mesh generation.

15. SUBJECT TERMS
High-Order Methods, CFD, Large Eddy Simulations, Hp-Adaptation, High-Order Mesh Generation

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Z.J. Wang
UU	UU	UU	UU	81	19b. TELEPHONE NUMBER *(Include area code)* 785-864-2440

Reset

Standard Form 298 (Rev. 8/98)
Prescribed by ANSI Std. Z39.18
Adobe Professional 7.0

INSTRUCTIONS FOR COMPLETING SF 298

1. REPORT DATE. Full publication date, including day, month, if available. Must cite at least the year and be Year 2000 compliant, e.g. 30-06-1998; xx-06-1998; xx-xx-1998.

2. REPORT TYPE. State the type of report, such as final, technical, interim, memorandum, master's thesis, progress, quarterly, research, special, group study, etc.

3. DATES COVERED. Indicate the time during which the work was performed and the report was written, e.g., Jun 1997 - Jun 1998; 1-10 Jun 1996; May - Nov 1998; Nov 1998.

4. TITLE. Enter title and subtitle with volume number and part number, if applicable. On classified documents, enter the title classification in parentheses.

5a. CONTRACT NUMBER. Enter all contract numbers as they appear in the report, e.g. F33615-86-C-5169.

5b. GRANT NUMBER. Enter all grant numbers as they appear in the report, e.g. AFOSR-82-1234.

5c. PROGRAM ELEMENT NUMBER. Enter all program element numbers as they appear in the report, e.g. 61101A.

5d. PROJECT NUMBER. Enter all project numbers as they appear in the report, e.g. 1F665702D1257; ILIR.

5e. TASK NUMBER. Enter all task numbers as they appear in the report, e.g. 05; RF0330201; T4112.

5f. WORK UNIT NUMBER. Enter all work unit numbers as they appear in the report, e.g. 001; AFAPL30480105.

6. AUTHOR(S). Enter name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. The form of entry is the last name, first name, middle initial, and additional qualifiers separated by commas, e.g. Smith, Richard, J, Jr.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES). Self-explanatory.

8. PERFORMING ORGANIZATION REPORT NUMBER. Enter all unique alphanumeric report numbers assigned by the performing organization, e.g. BRL-1234; AFWL-TR-85-4017-Vol-21-PT-2.

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES). Enter the name and address of the organization(s) financially responsible for and monitoring the work.

10. SPONSOR/MONITOR'S ACRONYM(S). Enter, if available, e.g. BRL, ARDEC, NADC.

11. SPONSOR/MONITOR'S REPORT NUMBER(S). Enter report number as assigned by the sponsoring/ monitoring agency, if available, e.g. BRL-TR-829; -215.

12. DISTRIBUTION/AVAILABILITY STATEMENT. Use agency-mandated availability statements to indicate the public availability or distribution limitations of the report. If additional limitations/ restrictions or special markings are indicated, follow agency authorization procedures, e.g. RD/FRD, PROPIN, ITAR, etc. Include copyright information.

13. SUPPLEMENTARY NOTES. Enter information not included elsewhere such as: prepared in cooperation with; translation of; report supersedes; old edition number, etc.

14. ABSTRACT. A brief (approximately 200 words) factual summary of the most significant information.

15. SUBJECT TERMS. Key words or phrases identifying major concepts in the report.

16. SECURITY CLASSIFICATION. Enter security classification in accordance with security classification regulations, e.g. U, C, S, etc. If this form contains classified information, stamp classification level on the top and bottom of this page.

17. LIMITATION OF ABSTRACT. This block must be completed to assign a distribution limitation to the abstract. Enter UU (Unclassified Unlimited) or SAR (Same as Report). An entry in this block is necessary if the abstract is to be limited.

The Development of High-Order Methods for Real World Applications

November 24, 2015

Final Report for AFOSR GRANT FA9550-12-1-0286

Z.J. Wang, L. Shi, Y. Li and J. Ims

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

Abstract

With increased computational power and progress in numerical methods over the past several decades, Computational Fluid Dynamics (CFD) is now used routinely as a powerful tool in the design of aircraft. Current production CFD codes used in the aerospace industry are usually second order accurate. High-order methods have the potential to achieve higher accuracy at less cost than low-order methods. This potential has been demonstrated conclusively for smooth problems in the latest International Workshops on High-Order Methods. For non-smooth problems, solution based hp-adaptation offers the best promise. Adjoint-based adaptive methods have the capability to dynamically distribute computing resources to areas most important for predicting engineering parameters, such as lift and drag coefficients. The primary objective of the present study is to develop robust and efficient high-order CFD methods and tools for the compressible Navier-Stokes equations that can provide engineering accuracy for real world industry problems. This report outlines progresses in the following areas.

The flux reconstruction (FR) or the correction procedure via reconstruction (CPR) method used in this work is a high-order differential formulation. We develop a parallel adjoint-based adaptive CPR solver which can work with any element-based error estimate and handle arbitrary discretization order for mixed elements. First, a dual-consistent discrete form of the CPR method is derived. Then, an efficient and accurate adjoint-based error estimation for the CPR method is developed and its accuracy and effectiveness are verified for the linear and non-linear partial differential equations (PDE). The current method has been applied to aerodynamic problems. Numerical tests show that significant savings in the number of DOFs can be achieved through the adjoint-based adaptation.

The usage of large-eddy simulation (LES) methods for the computation of turbulent flows has increased substantially in recent years. By resolving large energetic scales, LES has the potential to exhibit good performance especially for vortex dominated or massively separated flows. Due to the disparate length scales in LES, high-order methods are preferred for their high accuracy. Recently, models of the sub-grid scale (SGS) stress with the high-order FR/CPR method have been evaluated on 3D turbulent flows and the 1D Burgers' equation. Preliminary studies show that implicit LES (ILES), which does not involve any SGS model, has the most efficient and accurate results. In addition, a mathematical analysis of the scale similarity is performed, revealing that the ratio of the resolved stress to the SGS stress is γ^2, where γ is the ratio of the second filter width to the first filter width, under the assumption of small filter width.

For high-order methods to be effective, they must be paired with high-order meshes that include high-order representations of curved boundaries. A user-friendly, GUI-based software named meshCurve is developed to convert linear unstructured meshes to curved high-order meshes . Using the state-of-the-art algorithms, the software reconstructs the curved geometry from the linear surface mesh, while retaining sharp feature curves . The upgrade process is automatic and does not require a CAD file. meshCurve may be used by CFD practitioners and researchers to easily produce quality high-order meshes from existing low-order meshes. The CGNS standard is used as the file format for input and output.

Contents

1 Introduction

1.1 Background

With continuous progresses in numerical methods and computer hardware over the past several decades, Computational Fluid Dynamics (CFD) is now used routinely as a powerful tool in the design of aircraft. Current production CFD codes used in the aerospace industry are usually second order accurate. High-order methods have the potential to achieve greater accuracy at less cost. This potential has been demonstrated conclusively for smooth problems in the recent several International Workshops on High-Order Methods [107]. A variety of high order methods have been developed. Refer to several books [49, 63, 106] and reviews [29, 105] for the state-of-the-art and recent progresses in the development of such methods. The primary objective of the present study is to develop robust and efficient high-order CFD software for compressible Navier-Stokes equations that can provide engineering accuracy for real world industry problems.

The numerical method we consider is a nodal differential high-order formulation named the correction procedure via reconstruction (CPR) method [54][108]. This formulation has some remarkable properties. The framework is easy to understand, efficient to implement and can recover several well known methods such as the discontinuous Galerkin (DG) [85, 8, 9, 23, 24, 83, 111], the spectral volume method (SV) [104, 110, 72] and the spectral difference methods (SD)[71, 64, 78, 97, 70]. For recent development with CPR, interested readers can refer to [55, 109, 58, 17, 35, 36, 18, 56, 118].

For non-smooth problems, solution based hp-adaptations offer the best promise. Adaptive methods have the capability to dynamically distribute computing resources to desired areas of the computational domain, achieving required accuracy at minimal cost [74, 20, 27, 53]. Because they can ensure reliability and increase the robustness of high-order methods, adaptive methods have received considerable attention in the high-order CFD community [48, 98, 99, 31, 102, 116, 30].

The effectiveness of adaptive methods highly depends on the accuracy of the error estimation. There are at least three major types of adaptation criteria: gradient or feature based Berger and Colella [15], Warren et al. [112], Barth [7], Harris and Wang [46], residual-based Ainsworth and Oden [1], Baker [5], Johnson [61], Shih and Qin [93], Gao and Wang [34], Cagnone and Nadarajah [17], and adjoint-based Giles and Pierce [41], Venditti and Darmofal [99], Hartmann and Houston [48], Venditti and Darmofal [98], Becker and Rannacher [13], Giles and Pierce [42], Park [82], Venditti and Darmofal [100], Fidkowski and Darmofal [31], Wang and Mavriplis [101], Leicht and Hartmann [65], Li et al. [67], Yano and Darmofal [117], Ceze and Fidkowski [21]. Heuristic feature-based criteria perform refinements around some unique flow features, such as large gradients or strong vorticity, but because they do not directly relate to the output variables of interest, they cannot provide universal and robust error estimation Zhang et al. [119], Venditti and Darmofal [98]. The residual-based error criteria targets the elements which have large discretization error, flagging them for refinement. The locally defined element-wise error may lead to false refinements in convection-dominated problems. The dual-weighted residual method proposed by Becker and Rannacher [12] relates a specific functional output directly to the local residual by solving an additional adjoint equation. It can capture the error propagation effects inherent in the hyperbolic equations. This kind of adjoint-based error indicator has been shown very effective in driving an hp-adaptation procedure to obtain a very accurate prediction of the functional outputs [41, 42, 13, 16, 117, 21, 102]. Recently, Fidkowski and P.L Roe developed a new error indicator based on the entropy variables, which can be interpreted as the dual solution of the output of entropy balance on the whole domain. It can be obtained directly from the state variables without solving extra adjoint equations and has been successfully applied to

inviscid, viscous and turbulence flows Fidkowski and Roe [32, 33].

The adjoint solution is required for the error estimate and output-based adaptation. There are two approaches to obtain the adjoint: the continuous adjoint and the discrete adjoint. It has been shown that the discrete adjoint solution leads to a more accurate error estimation for the fine grid functional, while the continuous adjoint gives a better output estimation when the primal and adjoint solutions are well resolved [68]. However, the discrete adjoint solution should be consistent with the exact adjoint from the continuous adjoint equation. It is well known that dual consistency can significantly impact the convergence rate of both the primal and adjoint approximations. There are several possible sources of dual inconsistency in a high-order discretization. A dual-consistent discretization with variational forms, such as the finite element and DG methods, were examined for the Euler and Navier-Stokes equations in the literature [41, 42, 47, 62, 90]. More recently, adjoint-based error estimation for summation-by-parts finite-difference methods have been studied [50, 51, 14]. However, the analysis of the dual consistency for compact high-order differential-type methods appears lacking. This is one of the focuses of the present study.

The high-order CPR method can handle arbitrary solution order on mixed grids. The marked candidate elements for adaptation can be modified by enriching its solution order or subdividing its element or resizing its grid. Thus, the ways to increase the discretization resolution can be generally classified into 3 categories: h-refinement, r-refinement and p-refinement. For h-refinement, subdivision is performed locally for each candidate element to increase the total DOFs. R-refinement or the moving mesh method keeps the total number of nodes the same but moves the location of the grid locally or globally [53]. With p-refinement, the local degree of approximation polynomial is modified. The moving mesh method with curved elements in 3D is still an on-going research area. In this work, we only modify the solution polynomial order locally or subdivide elements hierarchically for adaptations. Intuitively, h-refinement should be applied to the discontinuities and p-enrichment is appropriate in smooth flow regions. However, the optimal choice between h- or p-refinement is not a trivial problem, which is studied in the previous research [16, 52, 96, 37, 11, 21, 25, 63, 102].

Reynolds-averaged Navier-Stokes (RANS) methods have been used almost exclusively for the computational fluid dynamics (CFD) analysis of practical engineering turbulent flows for the past several decades. In RANS, all turbulent fluid dynamic effects are replaced by a turbulence model. RANS-based techniques are successfully used in the industry for many cases. However, the behavior in the vortex dominated or massively separated flows are far from being satisfactory. On the other end, direct numerical simulation (DNS) methods resolve all turbulent motions. Without the influence of the turbulence modeling, it gives the whole spectrum of the turbulent flow. But DNS will still remain impractical for its high cost of computational power. Large-eddy simulation (LES) is a compromise of these two ends. In LES, large energetic scale motions are resolved while the small scale motions are taken care of by the SGS models. With the resolving of the important scales, the solution given by LES is expected to be more accurate than RANS, but, still affordable by the computation power today.

In LES, the large scales and small scales are separated by a low-pass filter. While the large scales are fully resolved, the small scales are believed to be more universal, and thus easier to model than the large-scale ones. Many SGS models have been developed in the last four decades. We focuses on five of them: the static Smagorinsky model (SS)J. [57] DK [26], the dynamic Smagorinsky model (DS)Germano M and WH [38], the scale-similarity model (SSM)Bardina J [6], the mixed model (MM) Bardina J [6] and the linear unified RANS-LES model (LUM) Harish Gopalan [45]. Among explicit models, the SS is a popular one because of its simplicity. The effect of the SGS stress upon the resolved scales is modeled as an eddy viscosity. The eddy viscosity is expressed in the mixing

length form with a dimensionless empirical coefficient. However, it has been found that the empirical coefficient depends on the flow. It also adds too much dissipation to the large scale motions if we keep the coefficient the same as we approach wall boundaries. To resolve these deficiencies, the DS model was developed in Germano M and WH [38]. In the DS model, the coefficient is calculated based on the Germano identity, which involves two levels of filtering and relates the SGS stress to the resolved stress. The coefficient is locally decided and no longer a prescribed constant, and it goes to zero as a wall boundary is approached. The DS model has been applied to a large variety of flow simulations Piomelli U [84]Ghosal S [40]Akselvoll K [3]Wu X [113]. An alternative way to model the SGS stress is offered by the SSM Bardina J [6]. As the name indicates, it assumes similarity between two scales of stresses, the resolved stress and the SGS stress. Numerical tests showed that energy accumulated at small scales with this model Meneveau C [80]. To remedy the problem, the DS was added to dissipate the energy, which led to the MM. Recently, hybrid RANS-LES models have drawn much research interest. They combine RANS with LES so that in the near wall region, the RANS model is used, while LES is employed in the outer region. These models have demonstrated good accuracy with reasonable cost when compared to a pure LES approach. The linear unified RANS-LES model (LUM) was developed in Harish Gopalan [45]. In the present work, we also evaluate the LUM model. Finally, we also consider the monotone integrated LES Durbin P. A [28] or implicit LES (ILES) Grinstein F. F. [44], in which no explicit SGS model is used. In ILES, the numerical algorithm has its numerical dissipation which serves as the SGS. The obvious advantage of ILES is its lower computational cost compared with the conventional SGS models.

As high-order methods become increasingly mature for real world applications, high-order boundary treatment becomes a critical issue. In mainstream finite volume solvers, the wall boundary of the geometry is usually represented by piecewise straight segments or planar facets. However, this linear boundary representation is far from sufficient for high-order methods. When the error from the geometry dominate the discretization error, the benefit of high-order methods is largely lost. For high-order methods to be effective, they must be paired with high-order meshes that include high-order representations of curved boundaries [77, 115].

Unfortunately, algorithms for generating high-order meshes are in short supply, and software support is limited. The lack of adequate high-order mesh generators has prompted researchers at the 1st International Workshop of High-Order CFD to prioritize high-order meshing as a pacing item for future research [103]. A second, related issue, is the need for high-order versions of existing low-order meshes. In many cases, the existing meshes no longer have their original CAD geometry file, so it is impossible to generate high-order meshes even if software to do so is available. To address these issues, the general-purpose low-to-high-order mesh converter meshCurve was created. The software is a cross-platform, GUI-based tool to easily and efficiently perform low-to-high-order mesh conversion. Access to the CAD geometry is unnecessary because the software uses the mesh itself to infer the surface curvature, while automatically identifing and preserving feature curves. Since high-order meshing is an issue for multiple simulation disciplines, we anticipate that meshCurve may also be useful in disciplines beyond CFD.

1.2 Program Objectives

The objective of this work is to develop a robust, accurate and efficient high-order CFD tool for real world engineering problems. In particular, the specific aims of this research are identified as follows:

- Develop a dual-consistent discrete adjoint equation for the CPR method.

- Develop an efficient and accurate automated adjoint-based error estimation method for the CPR formulation.
- Implement an adjoint-based adaptive solver with the CPR method, which can handle arbitrary discretization orders, and compare the different adaptation strategies.
- Demonstrate the importance of hp-adaptation for the high-order CPR method to aerodynamic flows and engineering problems.
- Evaluate SGS models with 3D turbulent flows using FR/CPR method.
- A priori and a posteriori evaluation of SGS models with Burgers' Equation and Euler Equations using FR/CPR method.
- Demonstrate the accuracy and efficiency of LES with the high-order CPR method to aerodynamic flows and apply it to a wide range of engineering applications.
- Develop a GUI-based, user-friendly, linear to high-order mesh converter.

2 Results and Discussion on Adjoint-based Adaptive High-order Differential Formulation

2.1 Review of the Correction Procedure via Reconstruction Method

The CPR method [54, 108] can be derived by transforming a weighted residual form into a differential one. Consider a hyperbolic conservation law

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

with proper initial and boundary conditions, where Q is the state vector, and $\vec{F} = (F, G)$ is the flux vector. Assume that the computational domain Ω is discretized into N non-overlapping triangular elements $\{V_i\}_{i=1}^{N}$. Let W be an arbitrary weighting or test function. The weighted residual formulation of Eq.1 on element V_i can be expressed as

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

Let Q_i be an approximate solution to the analytical solution Q on V_i. On each element, the solution belongs to the space of polynomials of degree k or less, i.e., $Q_i \in P^k(V_i)$. After applying integration by parts twice to the flux divergence and replacing the normal flux term with a common Riemann flux F^n_{com} in the above equation, we get

$$\int_{V_i} \frac{\partial Q_i}{\partial t} W \mathrm{d}\Omega + \int_{V_i} W \nabla \cdot \vec{F}(Q_i) \mathrm{d}\Omega + \int_{\partial V_i} W \left[F^n_{com} - F^n(Q_i) \right] \mathrm{d}S = 0. \tag{3}$$

Here the common Riemann flux F^n_{com} is defined as

$$F^n_{com} = F^n_{com}(Q_i, Q_{i+}, \vec{n}), \tag{4}$$

where Q_{i+} denotes the solution outside the current element V_i, and the normal flux $F^n(Q_i)$ at the interface is

$$F^n(Q_i) = \vec{F}(Q_i) \cdot \vec{n}.$$

In order to eliminate the test function, the boundary integral above is cast as a volume integral via the introduction of a "correction field" on V_i, $\delta_i \in P^k(V_i)$,

$$\int_{V_i} W \delta_i \mathrm{d}\Omega = \int_{\partial V_i} W [F^n] \mathrm{d}S, \tag{5}$$

where $[F^n] = F^n_{com} - F^n(Q_i)$ is the normal flux difference. Substituting Eq.5 into Eq.3, we obtain

$$\int_{V_i} \left(\frac{\partial Q_i}{\partial t} + \nabla \cdot \vec{F}(Q_i) + \delta_i \right) W \mathrm{d}\Omega = 0. \tag{6}$$

If the flux vector is a linear function of the state variable, then $\nabla \cdot \vec{F}(Q_i) \in P^k$. In this case, the terms inside the square bracket are all elements of P^k. Because the test space is selected to ensure a unique solution, Eq.6 is equivalent to

$$\frac{\partial Q_i}{\partial t} + \nabla \cdot \vec{F}(Q_i) + \delta_i = 0. \tag{7}$$

For nonlinear conservation laws, $\nabla \cdot \vec{F}(Q_i)$ is usually not an element of P^k. As a result, Eq.6 cannot be reduced to Eq.7. In this case, the most obviously choice is to

project $\nabla \cdot \vec{F}(Q_i)$ into P^k. Denote $\Pi\big(\nabla \cdot \vec{F}(Q_i)\big)$ as a projection of $\nabla \cdot \vec{F}(Q_i)$ to P^k. One choice is

$$\int_{V_i} \Pi\big(\nabla \cdot \vec{F}(Q_i)\big) W \mathrm{d}\Omega = \int_{V_i} \nabla \cdot \vec{F}(Q_i) W \mathrm{d}\Omega. \tag{8}$$

Then Eq.6 reduces to

$$\frac{\partial Q_i}{\partial t} + \Pi\big(\nabla \cdot \vec{F}(Q_i)\big) + \delta_i = 0. \tag{9}$$

Next, let the DOFs be the solutions at a set of solution points (SPs) $\{\vec{r}_{i,j}\}$ (j varies from 1 to $K = (k+1)(k+2)/2$). Then Eq.9 is true at the SPs, i.e.,

$$\frac{\partial Q_{i,j}}{\partial t} + \Pi_j\big(\nabla \cdot \vec{F}(Q_i)\big) + \delta_{i,j} = 0, \tag{10}$$

where $\Pi_j\big(\nabla \cdot \vec{F}(Q_i)\big)$ denotes the values of $\Pi\big(\nabla \cdot \vec{F}(Q_i)\big)$ at SP j. The efficiency of the CPR approach hinges on how the correction field δ_i and the projection $\Pi\big(\nabla \cdot \vec{F}(Q_i)\big)$ are computed. Two approaches, the Lagrange polynomial (LP) and the chain-rule (CR) formulations, were suggested to compute the projection of the flux divergence [108]. For the LP approach, the flux is assumed to belong to the polynomial space of degree k, e.g. F, $G \in P^k$. Therefore, the projection of the flux divergence is expressed as

$$\Pi\big(\nabla \cdot \vec{F}\big) = \nabla \cdot (\sum_j L_j \vec{F}_j),$$

where L_j is the Lagrange interpolation polynomial defined on SP j. For the CR approach, the flux divergence $\nabla \cdot \vec{F}$ is assumed to belong to the polynomial space P^k, which can be expressed as

$$\Pi\big(\nabla \cdot \vec{F}\big) = \sum_j L_j (\frac{\partial \vec{F}}{\partial Q} \cdot \nabla Q)_j.$$

Note that, for a linear conservation law, the LP and CR approaches lead to the same formulation. However, for a nonlinear flux equation, the CR approach can reduce the aliasing errors [108].

To compute δ_i, we define $k+1$ points named flux points (FPs) along each interface, where the normal flux differences are computed. We approximate (for nonlinear conservation laws) the normal flux difference $[F^n]$ with a degree k interpolation polynomial along each interface,

$$[F^n]_f \approx \mathbf{I}_k [F^n]_f \equiv \sum_l [F^n]_{f,l} L_l^{FP}, \tag{11}$$

where f is a face (or edge in 2D) index, and l is the FP index, and L_l^{FP} is the Lagrange interpolation polynomial based on the FPs in a local interface coordinate. For linear triangles with straight edges, once the solution points and flux points are chosen, the correction at the SPs can be written as

$$\delta_{i,j} = \frac{1}{|V_i|} \sum_{f \in \partial V_i} \sum_l \alpha_{j,f,l} [F^n]_{f,l} S_f, \tag{12}$$

where $\alpha_{j,f,l}$ are lifting constants independent of the solution variables, S_f is the face area, $|V_i|$ is the volume of V_i. The details of how to compute the lifting constants$\alpha_{j,f,l}$ can be found in Ref. [108].

In 1D, a continuous flux polynomial $\hat{F}$ which is equal to the common flux at the interfaces can be reconstructed using a piece-wise analytic flux polynomial $F_i(x) = \Pi(F(Q_i))$ and a correction term $\sigma_i(x)$ as

$$\hat{F}_i(x) = F_i(x) + \sigma_i(x).$$

According to Ref.[54], $\sigma_i(x)$ should approximate the zero function and satisfy the following equation

$$\sigma_i(x) = [F^{com}_{i-1/2} - F_i(x_{i-1/2})]g_L(x) + [F^{com}_{i+1/2} - F_i(x_{i+1/2})]g_R(x).$$

Here, $g_L(x)$ and $g_R(x)$ are both degree $k+1$ polynomials called correction functions with the properties

$$g_L(x_{i-1/2}) = 1, \quad g_L(x_{i+1/2}) = 0, \quad g_R(x_{i-1/2}) = 0, \quad g_R(x_{i+1/2}) = 1.$$

Then the CPR method for the 1D conservation law can be expressed as

$$\frac{\partial Q_{i,j}}{\partial t} + \frac{\partial F_i(x)}{\partial x} + [F^{com}_{i-1/2} - F_i(x_{i-1/2})]g'_L(x_{i,j}) + [F^{com}_{i+1/2} - F_i(x_{i+1/2})]g'_R(x_{i,j}) = 0.$$

A series of correction functions with different accuracy and stability properties were developed in Ref. [54]. If the correction function g is chosen as right Radau polynomials, the DG method is recovered from the CPR scheme. In this case, the correction function denoted by g_{DG} or g_1 is perpendicular to the degree k-1 polynomial space. Similarly, a g_2 correction function is defined, which is perpendicular to the degree k-2 polynomial space. In summary, for any integer $m \geqslant 1$, a g_m correction function can be defined which is perpendicular to P^{k-m}. For the sake of simplicity, the projection operator Π is omitted in the rest of the paper.

2.2 The Dual-Consistent CPR Formulation

Aircraft design engineers are usually interested in scalar engineering outputs such as lift or drag coefficients generated from CFD simulations. An adjoint solution can directly relate the local residual to the engineering output. Adjoint has been used in a wide range of applications including optimal controls, design optimization, data assimilation and error estimation. There are two approaches to obtain an adjoint solution. One can solve the continuous adjoint equation which is a partial differential equation using any numerical method, or directly solve the discrete adjoint equation derived from the discretized primal equation. As for the primal problem, a numerical scheme is defined as a consistent method if its discrete operator converges to the continuous operator, or the exact solution satisfies the discrete numerical formulation as the mesh size approaches zero. Similarly, for a dual-consistent adjoint formulation, the exact adjoint solution from the continuous adjoint equation should satisfy the discrete adjoint equation in the limit of vanishing mesh size. The dual-consistency of a discrete adjoint operator from a numerical discretization is a key component to ensure that the optimal convergent rate is achieved for an engineering output. To establish a robust and accurate functional error estimation procedure for the CPR method, a dual-consistent CPR formulation is developed in this section.

2.2.1 The Dual Problem and Adjoint-based Error Estimation

Linear PDEs First, we review the dual problem in the theory of output-based error estimation. A detailed discussion can be found in a series of articles [41, 42] and references therein. Consider a linear differential equation

$$LQ = f \quad on\,\Omega,$$

with a homogeneous boundary condition, where $Q \in V$ is the solution, V is the infinite dimensional solution space and $f \in L_2(\Omega)$. Suppose the output functional of interest $\mathcal{J}$ is given as an inner product of a smooth function g and the solution Q

$$\mathcal{J}(Q) = (g, Q) \equiv \int_\Omega gQdV,$$

over the entire domain Ω. The dual problem is introduced by adding a weighted residual to the functional

$$\mathcal{J}(Q) = (g, Q) + (\psi, f - LQ).$$

where ψ is an arbitrary function for now. If the solution Q satisfies the linear differential equation, this weighted residual does not affect the value of the original functional. Denote L^* the adjoint differential operator with respect to L defined according to $(\psi, LQ) = (L^*\psi, Q)$ with homogeneous boundary conditions. Then we have

$$\mathcal{J}(Q) = (g, Q)+(\psi, f)-(\psi, LQ) = (g, Q)+(\psi, f)-(L^*\psi, Q) = (\psi, f)-(L^*\psi-g, Q). \quad (13)$$

Let ψ be the adjoint solution computed using

$$L^*\psi = g \quad on\,\Omega.$$

Then the last term in Eq. 13 vanishes, and we obtain

$$\mathcal{J}(Q) = (\psi, f).$$

Obviously J is independent of Q, but depends on ψ now. So we call J a function of ψ, i.e.,

$$\mathcal{J}(\psi) = (\psi, f) - (L^*\psi - g, Q). \quad (14)$$

Therefore, the duality of the functional is

$$\mathcal{J}(Q) = (g, Q) \quad if \quad LQ = f \quad on\,\Omega$$

or

$$\mathcal{J}(\psi) = (\psi, f) \quad if \quad L^*\psi = g \quad on\,\Omega.$$

Suppose Q_h and ψ_h are the discrete primal solution and the discrete adjoint solution obtained with a numerical method, and both of them belong to the discrete solution space V_h. Then, the discrete source term for the primal equation is $f_h = LQ_h$. The functional error can be estimated by

$$\begin{aligned}\delta\mathcal{J} = (g, Q_h) - (g, Q) &= (g, Q_h - Q) \\ &= (L^*\psi, Q_h - Q) \\ &= (\psi, L(Q_h - Q)) \\ &= \underbrace{(\psi_h, f_h - f)}_{computable\,error} + \underbrace{(\psi - \psi_h, f_h - f)}_{remaining\,error}, \end{aligned} \quad (15)$$

where the linearity of the inner product and the adjoint definition are used. Note that the first term on the RHS of Eq. 15 is defined as the computable error, since it does not involve any analytical primal solution. If a numerical method possesses the Galerkin orthogonality property,

$$(v_h, f_h - f) = 0, \quad for\ all\ v_h \in V_h,$$

the computable error vanishes since we take ψ_h in space V_h. Therefore, there is no need to evaluate the computable error, and the order of the output error only depends on the remaining error term. If we use a degree k polynomial to approximate the primal and adjoint solution and assume the optimal order of accuracy can be achieved, we obtain

$$\|Q_h - Q\| = O(h^{k+1}), \quad \|\psi_h - \psi\| = O(h^{k+1}).$$

Here$\|.\|$ is a L_2 norm. In addition, for a n^{th} order differential PDE, we have $\|f_h - f\| = O(h^{k+1-n})$. So the order of the computable error is $O(h^{2k+2-n})$, which leads to a superconvergent functional of order $O(h^{2k+2-n})$. To estimate the output error, we need to evaluate the computable error. A common approach is to use an adjoint solution from a finer approximation space, e.g. $p = k + 1$. Then the computable error is not equal to 0, and we can use this error estimate to correct the original output

$$\mathcal{J}_{corr} \equiv \mathcal{J}(Q_h) + (\psi_h, f_h - f).$$

The convergence rate of this corrected output is two orders higher than the original output, which is $O(h^{2k+4-n})$. A rigorous proof can be found in Ref. [52].

Nonlinear PDEs Consider a non-linear differential equation

$$\mathcal{N}(Q) = 0, \quad on\,\Omega. \tag{16}$$

Suppose an output functional of interest is given as $\mathcal{J}(Q)$. A dual problem is introduced by defining a Lagrangian of the output with the constraint of the solution Q satisfying the primal equation $\mathcal{N}(Q) = 0$

$$\mathcal{L} = \mathcal{J}(Q) + \int_\Omega \psi \mathcal{N}(Q) \mathrm{d}\Omega. \tag{17}$$

Here $\psi \in V$ has two roles. First, ψ is the adjoint solution. Second, it also serves as a Lagrangian multiplier. After performing the linearization and enforcing stationary of $\mathcal{L}$ to a solution perturbation $\delta Q \in V$, we obtain

$$\delta\mathcal{L} = \mathcal{L}'[Q](\delta Q) = \mathcal{J}'[Q](\delta Q) + \int_\Omega \psi \mathcal{N}'[Q](\delta Q) \mathrm{d}\Omega = 0 \quad \forall \delta Q \in V, \tag{18}$$

where the primed notation denotes Frechét linearization with respect to an argument in the square bracket. Eq. 18 defines the dual problem in a variational form by finding ψ such that

$$\mathcal{J}'[Q](v) + \int_\Omega \psi \mathcal{N}'[Q](v) \mathrm{d}\Omega = 0 \quad \forall v \in V. \tag{19}$$

Let Q_h denotes an approximate solution to the analytical solution Q. The difference between them can be interpreted as a solution perturbation $Q_h = Q + \delta Q$. The output error defined as $\delta\mathcal{J} = \mathcal{J}(Q_h) - \mathcal{J}(Q)$ can be estimated by the adjoint weighted residual method

$$\delta\mathcal{J} \approx \mathcal{J}'[Q](\delta Q) = -\int_\Omega \psi\mathcal{N}'[Q](\delta Q)\mathrm{d}\Omega \approx -\int_\Omega \psi(\mathcal{N}(Q_h)-\mathcal{N}(\mathrm{Q}))\mathrm{d}\Omega = -\int_\Omega \psi\mathcal{N}(Q_h)\mathrm{d}\Omega. \tag{20}$$

2.2.2 The Continuous Adjoint Equation

We consider the following conservation law

$$\mathcal{N}(Q) \equiv \nabla\cdot\vec{\mathcal{F}}(Q), \tag{21}$$

as an example to develop the continuous adjoint equation. Eq. 19 leads to

$$\mathcal{J}'[Q](v) + \int_\Omega \psi\left(\frac{\partial}{\partial Q}\left(\nabla\cdot\vec{\mathcal{F}}\right)\right)(v)\mathrm{d}\Omega = 0 \quad \forall v \in V. \tag{22}$$

Suppose the output functional $\mathcal{J}(Q)$ consists of surface ($\partial\Omega$) and volume (Ω) integrals

$$\mathcal{J}(Q) = \int_\Omega gQ\mathrm{d}\Omega + \int_{\partial\Omega} j_\tau(Q)\mathrm{d}s,$$

where j_τ is a boundary operator. Substituting the definition of $\mathcal{J}$ into Eq. 22, and performing integration by parts, we get

$$\int_\Omega (g - \nabla\psi\cdot\frac{\partial\vec{\mathcal{F}}}{\partial Q})\mathrm{d}\Omega + \int_{\partial\Omega}(\frac{\partial j_\tau}{\partial Q} + \psi\frac{\partial\vec{\mathcal{F}}}{\partial Q}\cdot\vec{n})\mathrm{d}s = 0.$$

$$\nabla\psi\cdot\frac{\partial\vec{\mathcal{F}}}{\partial Q} = g, \tag{23}$$

which is a linear partial differential equation for ψ, and the corresponding boundary conditions are from the surface integral

$$\int_{\partial\Omega}(\frac{\partial j_\tau}{\partial Q} + \psi\frac{\partial\vec{\mathcal{F}}}{\partial Q}\cdot\vec{n})\mathrm{d}s = 0. \tag{24}$$

2.2.3 The Dual-Consistent CPR Formulation

The discrete adjoint equation is obtained directly by linearizing the discretized primal equation. Consider a discretized formulation of the primal Eq. 16

$$\mathcal{N}_h(Q_h) = 0, \quad Q_h \in V_h$$

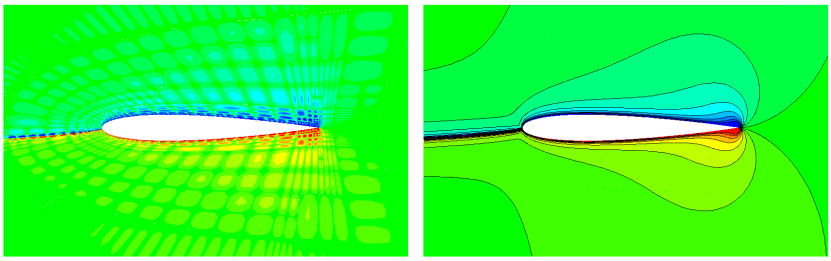
with mesh h, and a discrete solution perturbation $\delta Q_h \in V_h$. Linearizing the discrete residual $\mathcal{R}_h$ and the discrete output $\mathcal{J}_h$, we get

$$\delta\mathcal{R}_h = \mathcal{R}_h(Q_h + \delta Q_h) - \mathcal{R}_h(Q_h) \approx \frac{\partial\mathcal{R}_h}{\partial Q_h}\delta Q_h$$

$$\delta\mathcal{J}_h = \mathcal{J}_h(Q_h + \delta Q_h) - \mathcal{J}_h(Q_h) \approx \frac{\partial\mathcal{J}_h}{\partial Q_h}\delta Q_h.$$

The discrete adjoint $\tilde{\psi}_h$ is defined as the sensitivity of output perturbation $\delta\mathcal{J}_h$ to the primal residual perturbation $\delta\mathcal{R}_h$

$$\delta\mathcal{J}_h \equiv -\tilde{\psi}_h^T\delta\mathcal{R}_h,$$



(a) The discrete adjoint

(b) The discrete adjoint in variational forms

Figure 1: The C_L adjoint for a subsonic NACA0012 airfoil

which leads to

$$\frac{\partial \mathcal{J}_h}{\partial Q_h}\delta Q_h \approx \delta \mathcal{J}_h = -\tilde{\psi}_h^T \delta \mathcal{R}_h \approx -\tilde{\psi}_h^T \frac{\partial \mathcal{R}_h}{\partial Q_h}\delta Q_h.$$

We obtain the discrete adjoint equation by canceling the solution perturbation δQ_h

$$-\frac{\partial \mathcal{R}_h}{\partial Q_h}^T \tilde{\psi}_h = \frac{\partial \mathcal{J}_h}{\partial Q_h}^T. \tag{25}$$

For numerical methods with a weak form such as FEM or DG methods, after choosing a proper basis, Eq. 25 is equivalent to its variational formulation. A detailed derivations can be found in Ref. [47, 30]. So the discrete adjoint equation for numerical methods in semi-linear form is consistent with the continuous adjoint equation. However, this is not true for numerical methods in a differential form such as the CPR method, which does not possess a variational form.

Substituting a pointwise residual $r_{i,j}$ defined on each solution point j of cell i arising from a differential scheme, we obtain

$$-\sum_i \sum_j \frac{\partial r_{i,j}}{\partial Q_l}\tilde{\psi}_{i,j} = \frac{\partial \mathcal{J}}{\partial Q_l}, \tag{26}$$

where l is the global index of the DOFs in the entire domain. Figure 1a shows the x-momentum component of the lift adjoint for a subsonic airfoil using the discrete adjoint equation with the CPR method. It has a very oscillatory distribution in each cell, which indicates dual-consistency violations.

Since the CPR method is not in a variational form, its discrete adjoint equation should be directly derived from the linearized Lagrangian. Assume the adjoint solution belongs to the same space of the primal solution, the adjoint variable ψ_i of cell i can be approximated using the Lagrange basis L_j

$$\psi_i = \sum_j L_j \hat{\psi}_{i,j}.$$

Directly discretizing the linearized Lagrangian, Eq. 19, with a quadrature rule, we obtain

$$-\sum_i \sum_j \frac{\partial r_{i,j}}{\partial Q_l}\omega_j |J_{i,j}| \hat{\psi}_{i,j} = \frac{\partial \mathcal{J}}{\partial Q_l}, \tag{27}$$

where ω_j and $|J_{i,j}|$ are the quadrature weight and the element Jacobian at the solution point. Comparing with Eq. 26, the following relation can be derived from the discrete adjoint $\tilde{\psi}_{i,j}$ and the discrete adjoint $\hat{\psi}_{i,j}$ in the integral form

$$\tilde{\psi}_{i,j} = \omega_j |J_{i,j}| \hat{\psi}_{i,j}.$$

It is obvious that the discrete adjoint formula for a numerical scheme in a differential form is not consistent with the continuous adjoint equation. The inconsistent adjoint is related to the consistent counterpart in terms of the quadrature weight ω and the cell Jacobian $|J|$. This integral equation can be interpreted as an explicitly defined variational form for the CPR method. In this paper, we call it the dual consistent discrete adjoint formula. Figure 1b shows the consistent discrete adjoint solution with the CPR method. Clearly the consistent adjoint solution is smooth.

2.2.4 Analysis of Dual-Consistency for the CPR Method

For a dual consistent discretization, the discrete adjoint equation should approach the continuous adjoint partial differential equation when the mesh size diminishes. In other words, the analytic primal solution Q and the analytic dual solution ψ should satisfy the discrete adjoint equation when the mesh size goes to zero. Substituting the discrete residual of the CPR method into the discrete adjoint equation for the conservation law leads to

$$\sum_i \sum_j \frac{\partial}{\partial Q_l} \left(\nabla \cdot \vec{F}(Q)_{i,j} + \delta_{i,j} \right) \omega_j |J_{i,j}| \hat{\psi}_{i,j} = -\frac{\partial \mathcal{J}}{\partial Q_l}, \tag{28}$$

where $\delta_{i,j}$ is the correction term. For a 1D CPR formulation, the above equation can be further expressed as

$$\sum_i \sum_j \left(\sum_k \frac{\partial F_{i,k}}{\partial Q_l} \frac{dL_k(\xi_j)}{d\xi} + \frac{\partial}{\partial Q_l} [F]_{i-\frac{1}{2}} g'_L(\xi_j) + \frac{\partial}{\partial Q_l} [F]_{i+\frac{1}{2}} g'_R(\xi_j) \right) \omega_j \xi_{x,i,j} \hat{\psi}_{i,j} = -\frac{\partial \mathcal{J}}{\partial Q_l}. \tag{29}$$

Here, g'_L, g'_R and ψ_i belong to $P^k(\Omega_i)$. $\frac{dL}{d\xi}$ is a degree $k-1$ polynomial. Therefore, the degree of the integrand is at least $2k$. Assume that the quadrature rule defined on the solution points is exact at least for a degree $2k$ polynomial. Then we have

$$\sum_i \int_{\Omega_i} \psi_i \sum_k \frac{\partial F_{i,k}}{\partial Q_l} \frac{dL_k}{d\xi} \xi_x \mathrm{d}x + \sum_i \int_{\Omega_i} \psi_i \left(\frac{\partial}{\partial Q_l} [F]_{i-\frac{1}{2}} g'_L + \frac{\partial}{\partial Q_l} [F]_{i+\frac{1}{2}} g'_R \right) \xi_{\mathrm{x}} \mathrm{d}x = -\frac{\partial \mathcal{J}}{\partial Q_l}.$$

Performing integration by parts on the LHS, we obtain

$$\begin{aligned} LHS = &- \sum_i \int_{\Omega_i} \frac{\partial F}{\partial Q} \frac{d\psi}{dx} \mathrm{d}x + \sum_i \psi \frac{\partial \mathcal{F}}{\partial Q} |_{i-\frac{1}{2}}^{i+\frac{1}{2}} \\ &+ \sum_i \frac{\partial [F]_{i-\frac{1}{2}}}{\partial Q} \left(\psi_i g_L |_{-1}^{1} - \int_{-1}^{1} \psi'_i \mathrm{g_L} \mathrm{d}\xi \right) + \sum_i \frac{\partial [F]_{i+\frac{1}{2}}}{\partial Q} \left(\psi_i g_R |_{-1}^{1} - \int_{-1}^{1} \psi'_i \mathrm{g_R} \mathrm{d}\xi \right). \end{aligned} \tag{30}$$

Recall that, for the CPR method, the correction functions satisfy

$$g_L(-1) = g_R(1) = 1$$
$$g_L(1) = g_R(-1) = 0.$$

Furthermore, a DG correction function g_{dg} of degree $k+1$ is perpendicular to P^{k-1}, i.e.,

$$\int_{-1}^{1} g_{dg}(\xi)\phi(\xi)\mathrm{d}\xi = 0, \quad ,\forall\phi \in P^{k-1}.$$

Therefore, if a DG correction function is used, Eq. 30 can be simplified to

$$LHS = -\sum_i \int_{\Omega_i} \frac{\partial F}{\partial Q}\frac{d\psi}{dx}\mathrm{d}x + \sum_i \psi \frac{\partial \mathcal{F}}{\partial Q}\Big|_{i-\frac{1}{2}}^{i+\frac{1}{2}} - \sum_i \frac{\partial [F]_{i-\frac{1}{2}}}{\partial Q}\psi_{i-\frac{1}{2}} + \sum_i \frac{\partial [F]_{i+\frac{1}{2}}}{\partial Q}\psi_{i+\frac{1}{2}}, \quad (31)$$

or equivalently,

$$LHS = -\sum_i \int_{\Omega_i} \frac{\partial F}{\partial Q}\frac{d\psi}{dx}\mathrm{d}x + \sum_i \frac{\partial F^{com}_{i+\frac{1}{2}}}{\partial Q}\psi_{i+\frac{1}{2}} - \sum_i \frac{\partial F^{com}_{i-\frac{1}{2}}}{\partial Q}\psi_{i-\frac{1}{2}}. \quad (32)$$

The first term is the governing equation for the continuous adjoint equation, which vanishes for the analytic adjoint solution. Notice that

$$\begin{aligned}\sum_i \frac{\partial F^{com}_{i+\frac{1}{2}}}{\partial Q}\psi_{i+\frac{1}{2}} - \sum_i \frac{\partial F^{com}_{i-\frac{1}{2}}}{\partial Q}\psi_{i-\frac{1}{2}} &= \frac{\partial F^{com}_{N+\frac{1}{2}}}{\partial Q}\psi_{N+\frac{1}{2}} - \frac{\partial F^{com}_{\frac{1}{2}}}{\partial Q}\psi_{\frac{1}{2}} + \sum_{i=1}^{N-1}(\psi_i(1) - \psi_{i+1}(-1))\frac{\partial F^{com}_{i+\frac{1}{2}}}{\partial Q} \\ &= \frac{\partial F^{com}_{N+\frac{1}{2}}}{\partial Q}\psi_{N+\frac{1}{2}} - \frac{\partial F^{com}_{\frac{1}{2}}}{\partial Q}\psi_{\frac{1}{2}}\end{aligned}$$

when $\psi_i(1) = \psi_{i+1}(-1)$ for the analytical adjoint solution. So the final equation is

$$\frac{\partial F^{com}_{N+\frac{1}{2}}}{\partial Q}\psi_{N+\frac{1}{2}} - \frac{\partial F^{com}_{\frac{1}{2}}}{\partial Q}\psi_{\frac{1}{2}} = -\frac{\partial \mathcal{J}}{\partial Q}. \quad (33)$$

As discussed in Ref. [47, 75], a well-defined dual-consistent boundary flux should only be a function of the boundary state $F^{com} = F^{com}(Q_{bc}(Q_L))$, and a properly-defined dual-consistent output functional leads to Eq. 33 and is consistent with the dual boundary condition for the continuous adjoint equation (Eq. 24). Therefore, Eq. 28 can be satisfied exactly with an analytical adjoint solution. A similar procedure can be applied to the system of equations in 2D. The corresponding equation to Eq. 33 is

$$\int_{\Gamma} \psi^T \frac{\partial F^{com}}{\partial Q}\mathrm{d}s = \frac{\partial \mathcal{J}}{\partial Q}, \quad (34)$$

which will be used to analyze the dual consistency for the 2D linear wave equation and the Euler equations in the next section.

Based on the analysis, the key factors to ensure a dual-consistent CPR formulation are summarized next.

1. In order to ensure the integral accuracy of the discrete adjoint equation in a variational form, an accurate quadrature rule defined on the solution points should be exact for a degree $2k$ polynomial. Recall that a k+1 point Gaussian quadrature rule can yield an exact integration of a degree $2k+1$ polynomial. Therefore, the Gauss quadrature points are preferred as the solution points. The Lobatto quadrature rule

can only integrate a degree $2k-1$ polynomial exactly. If the Lobatto points are used as the solution points, the CPR formulation will have an accuracy loss for the discrete adjoint solution and the corresponding error estimation.

2. The correction function g in the CPR method must be perpendicular to the derivatives of the adjoint solution

$$\int_{\Omega} g\psi' \mathrm{d}\Omega = 0. \tag{35}$$

If we assume that the discrete adjoint ψ belongs to the same space of the primal solution, the degree of the derivative ψ' is $k-1$. Then the only qualified correction function is the DG correction function g_{dg}, which is perpendicular to P^{k-1}. On the other hand, the degree of the adjoint solution ψ is determined by the specific correction function. Suppose we use the g_2 correction function, which is only perpendicular to P^{k-2}. To satisfy this condition, the degree of the discrete adjoint solution ψ is automatically degenerated to $k-1$. In summary, for a CPR scheme with a correction function g_m, the discrete adjoint $\psi \in P^{k+1-m}$.

3. The Lagrange polynomial (LP) approach is required to evaluate the flux divergence term in the CPR formulation, instead of the chain rule (CR) approach. This requirement is very similar to the conservation requirements for the CPR scheme. Therefore, a similar fix can be obtained by following the conservation fix of the CR approach [35]. However, the numerical results in the next section show that the dual-inconsistent violation by the CR approach is relatively weak.

4. A properly defined common numerical flux on the boundaries, and a well-defined output functional, are critical in making the boundary terms in the adjoint equation vanish.

2.2.5 Numerical Tests

Linear Advection Equation First, the dual-consistent CPR discretization and the error estimation are verified with a first-order hyperbolic partial differential equation (PDE). Consider a 2D linear wave equation

$$\nabla \cdot \vec{c}Q = f, \quad x \in \Omega$$

with a Dirichlet boundary condition

$$Q(\vec{x}) = B(\vec{x}), \quad x \in \partial\Omega^{-}$$

given on the inflow boundaries $\partial\Omega^{-} = \{\vec{x} \in \partial\Omega |\ \vec{n}\cdot\vec{c} < 0\}$, where $\vec{n}$ is the outward surface normal direction and $\vec{c}$ is the advection velocity (Figure 2a) prescribed as

$$\vec{c} = (e^{-x}, e^{y}).$$

The solution is set to be

$$Q(\vec{x}) = e^{y} sin(\frac{\pi(e^{x}-1)}{e-1}),$$

which determines the source term f by the method of manufactured solutions [50].

The simulation is performed on a unit square $\Omega = [0,1]^2$ filled with quadrilateral elements. The output of interest is defined on the outflow boundaries $\partial\Omega^{+} = \{\vec{x} \in \partial\Omega |\ \vec{n}\cdot\vec{c} > 0\}$ as

$$\mathcal{J}(Q) = \int_{\partial\Omega^+} g(\vec{x})(\vec{n}\cdot\vec{c}Q)ds,$$

where the weighting function

$$g(\vec{x}) = \frac{\pi e^{x-y}}{e-1}.$$

The exact value of the output is $\mathcal{J} = 2e$.

In this case, Eq. 34 leads to

$$\int_{\partial\Omega^+} \psi \frac{\partial F^{com}}{\partial Q} ds = -\frac{\partial}{\partial Q}\int_{\partial\Omega^+} g(\vec{x})(\vec{n}\cdot\vec{c}Q)ds,$$

which is equivalent to the boundary condition for the continuous adjoint equation

$$\int_{\partial\Omega^+} (\vec{n}\cdot\vec{c})\,\psi + (\vec{n}\cdot\vec{c})g(\vec{x})ds = 0.$$

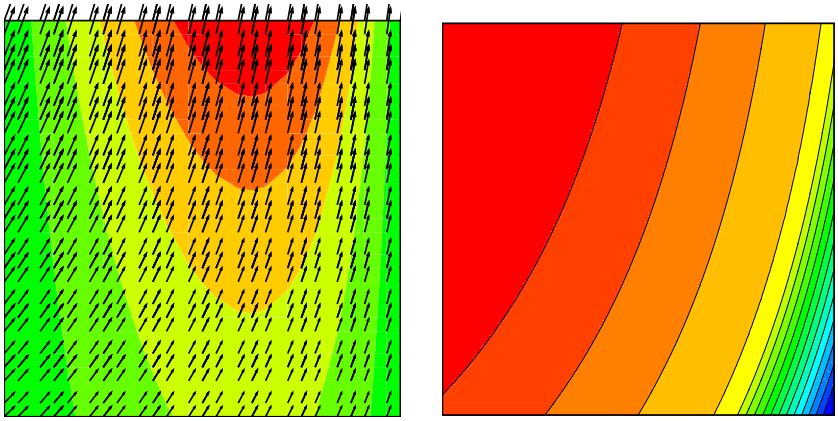
The common flux $F^{com}(Q_{BC}(Q_L), \vec{n}) = \vec{n}\cdot\vec{c}Q_L$ is chosen for the boundaries to ensure that the CPR method for the linear wave equation with this output is a dual-consistent formulation.

The contours of the primal solution and the adjoint solution are shown in Figure 2. Based on the dual-consistency analysis, we choose the Gauss points as the solution points and the DG correction function g_{dg}. Figure 3 shows the convergent rate of the primal solution, the output functional and the error estimate using the CPR method. The optimal order $O(h^{k+1})$ is obtained for the primal solution. Super convergence of the output functional and the error estimate is good indicator of a dual-consistent discretization. For this case, a super convergence of $O(h^{2k+1})$ is observed for both of the output error and the adjoint-based output error estimate. The corrected output converges 2 orders faster than the original output, which is $O(h^{2k+3})$. The results of the CPR method with Lobatto points and g_{dg} is shown in Figure 4a. Comparing with the results of the Gauss points, the accuracy loss of the quadrature rules defined on the Lobatto points leads to one order loss in the output functional, the error estimate and the corrected functional.

Now we test the influence of the correction functions on the output functional and the error estimate. As discussed in [54], for a integral $m \geq 1$, a correction function g_m of degree $k+1$ is perpendicular to P^{k-m}, and the Fourier analysis indicates that the order of the corresponding scheme is $O(h^{2k+2-m})$. The previous analysis shows that the approximation order of the adjoint solution ψ is determined by the correction function g_m as $O(h^{k+1-m})$.

Figure 4 shows the convergence rates using the g_2 and g_3 correction functions with the Gauss points as the solution points. Similar relationships are obtained between the order of accuracy of the functional outputs, the error estimate and the correction functions. For the advection equation, the output functional and the adjoint-based error estimate with the Gauss points and g_m are accurate to order $O(h^{2k+2-m})$. Furthermore, the corrected outputs of the corresponding schemes are accurate to order $O(h^{2k+4-m})$. The convergence rates with the different CPR schemes are summarized in Table 1. Those results are consistent with the analysis in section 2.2.3.

For a linear partial differential equation, the discrete adjoint based output error estimation should recover the exact output error obtained by a finite difference method. Table 2 compare the true output error between k = 1 and = = 2 results and the adjoint-based output error. The difference between them are within machine zero.



(a) Primal solution (b) Adjoint solution

Figure 2: The primal and adjoint solution of the linear advection equation

SPs	g	primal L_2 err.	output error	output error est.	corrected output error
Gauss	DG	k+1	2k+1	2k+1	2k+3
Lobatto	DG	k+1	2k	2k	2k+2
Gauss	G2	k+1	2k	2k	2k+2
Gauss	G3	k+1	2k-1	2k-1	2k+1

Table 1: Convergence rates for the linear wave equation with different schemes

The Supersonic Vortex Transportation Problem Second, we test the present adjoint-based error estimation for the Euler equations with curved elements. The problem we consider is a 2D supersonic vortex transported in a circular sector. The computational domain is defined on a section of an annulus with the inner radius of $r_{in} = 1$ and the outer radius of 1.384. The initial mesh consists of $k = 4$ quadrilateral elements and is shown in Figure 5a. The isentropic vortex rotates around the center of the circular sector. The density ρ is only a function of the radius r (Figure 5b)

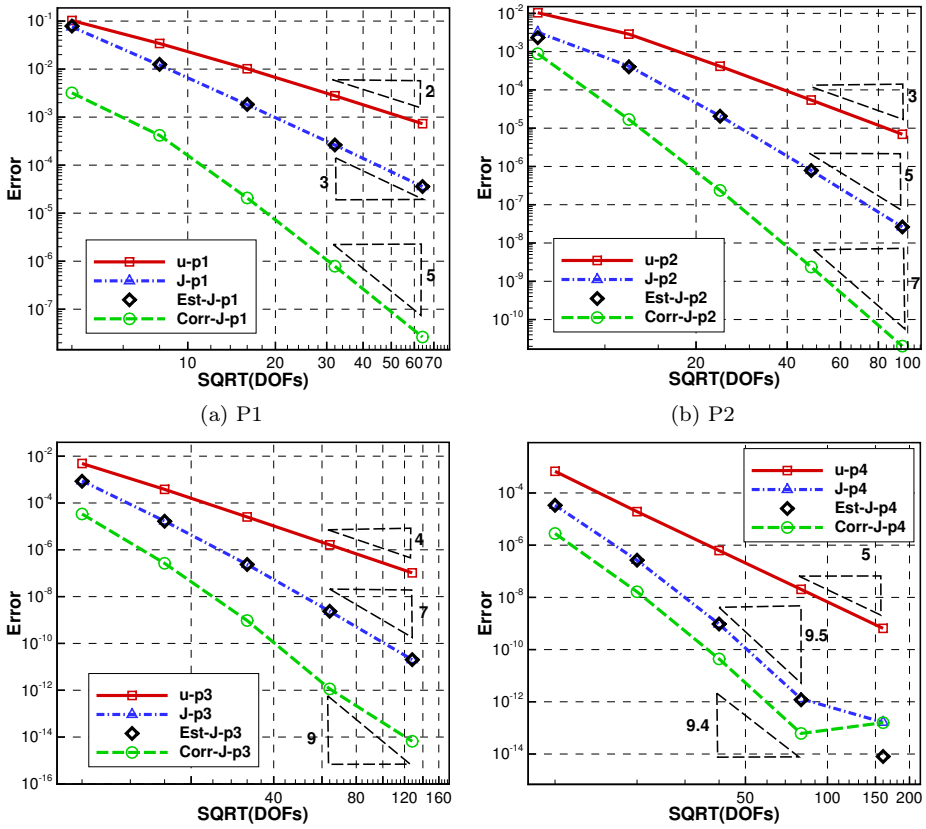
$$\rho(r) = \rho_{in}(1 + \frac{\gamma - 1}{2} M_{in}^2 (1 - \frac{r_{in}^2}{r^2}))^{\frac{1}{\gamma - 1}},$$

where γ is the ratio of heat capacities and the remaining parameters are the flow conditions on the inner surface chosen as $\rho_{in} = 2$, $M_{in} = 2$ and $p_{in} = \frac{1}{\gamma}$. The other variables can be computed with the isentropic relations. Characteristic boundary conditions with the analytical solution are used at both the inlet and the outlet, and slip wall boundary conditions are applied on the inner and the outer boundaries. The output of interest is the force in the x direction on the inner surface, where the pressure is equal to $\frac{1}{\gamma}$, so the exact value of the output is $\mathcal{J} = -\frac{1}{\gamma}$.

The output boundary operator j_τ is defined as $j_\tau = p\vec{n} \cdot \vec{i}_{dir}$, where $\vec{i}_{dir} = [1, 0]$. Therefore, the corresponding equation to Eq. 33 in 2D is

$$\int_{\partial\Omega} \psi^T \frac{\partial F^{com}}{\partial Q} ds = -\frac{\partial}{\partial Q} \int_{\partial\Omega} \frac{\partial p}{\partial Q} (\vec{n} \cdot \vec{i}_{dir}) ds,$$

which is equivalent to the boundary condition for the continuous adjoint equation



(a) P1 (b) P2

(c) P3 (d) P4

Figure 3: Results for the linear advection equation using a CPR scheme with the Gauss points and g_{dg}

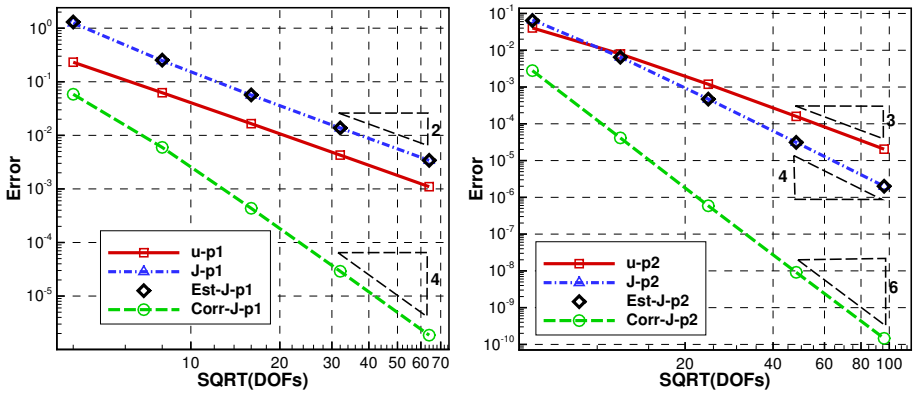
$$\int_{\partial\Omega} \psi^T \frac{\partial p}{\partial Q} \begin{bmatrix} 0 \\ n_x \\ n_y \\ 0 \end{bmatrix} + (\vec{n} \cdot \vec{i}_{dir}) \frac{\partial p}{\partial Q} ds = 0,$$

with the flux

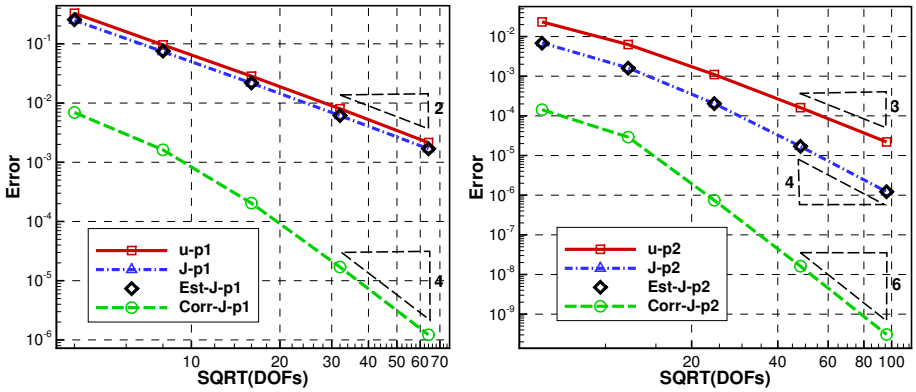
$$F^{com}(Q_{BC}(Q_L), \vec{n}) = \vec{n} \cdot \vec{F}(Q_{BC}(Q_L)) = p \begin{bmatrix} 0 \\ n_x \\ n_y \\ 0 \end{bmatrix},$$

chosen as the common flux on the boundaries. This ensures that the CPR method for the Euler equations with this output definition is a dual-consistent formulation.

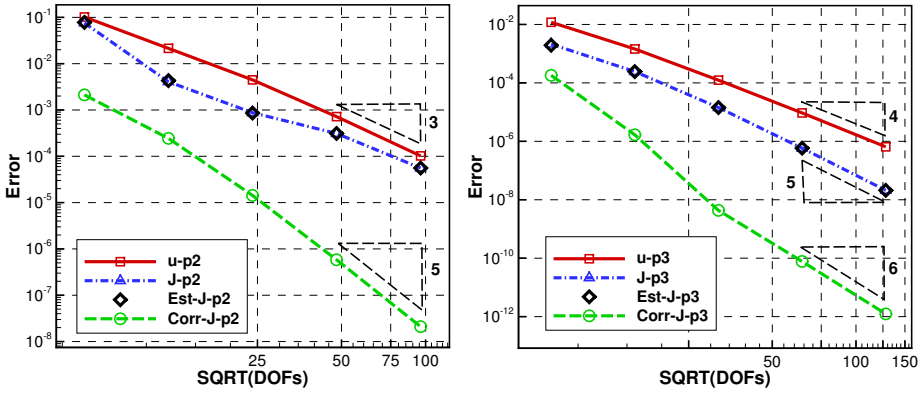
The adjoint solution with the dual-consistent boundary conditions and dual-inconsistent boundary conditions are shown in Figure 5. The dual-inconsistent boundary conditions generated some spurious oscillations near the wall, while the adjoint solution from the dual-consistent boundary conditions was very smooth. Figure 6 displays the solution error, the output error and the adjoint-based error estimate. The optimal order of accuracy $k+1$ in L_2 norm was obtained by both the dual-consistent and inconsistent boundary conditions in the primal solution. However, a super convergence of order $2k+1$ was



(a) Lobatto points with g_{dg} as the correction function

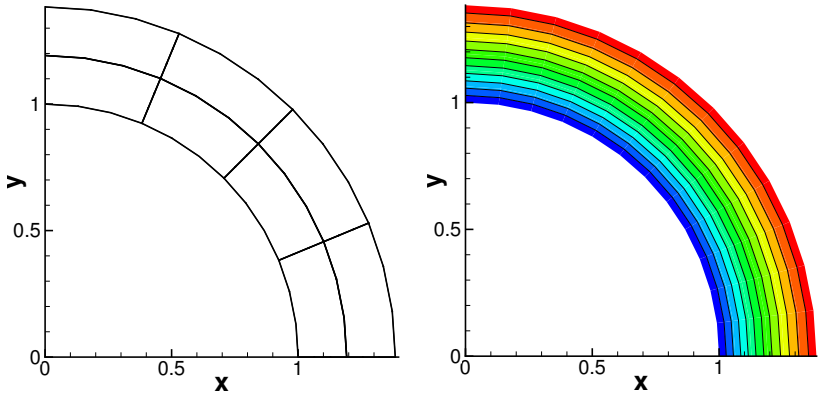


(b) Gauss points with g_2 as the correction function



(c) Gauss points with g_3 as the correction function

Figure 4: Results for the linear advection equation with different correction functions



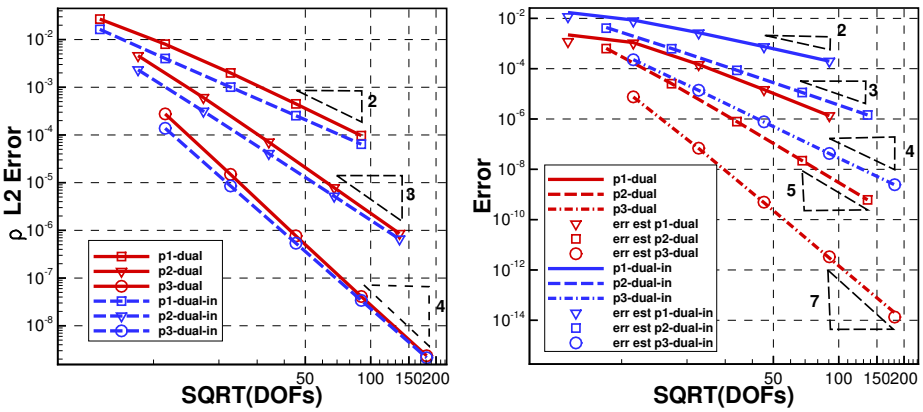
(a) The initial mesh

(b) Density

(c) Adjoint solution ψ_1 with dual-consistent BCs

(d) Adjoint solution ψ_1 with dual-inconsistent BCs

Figure 5: The primal and adjoint solution for the supersonic vortex transportation problem (Gauss points, g_{dg},$k = 3$)



(a) L_2 density error

(b) Output error

Figure 6: The results of the supersonic vortex transportation problem using the dual-consistent BC and the dual-inconsistent BC (Gauss points, g_{dg})

Cells	$\mathcal{J}_H(Q_H) - \mathcal{J}_h(Q_h)$	$-(\psi_h)^T R_h(Q_h^H)$
4	-7.85516345138E-002	-7.85516345117E-002
16	-1.25394717994E-002	-1.25394717999E-002
64	-1.83796281052E-003	-1.83796281080E-003
256	-2.63400045419E-004	-2.63400045485E-004
1024	-3.58710248100E-005	-3.58710248149E-005

Table 2: The output error by the finite difference method and adjoint-based error estimation (coarse space $k = 1$, fine space $k = 2$)

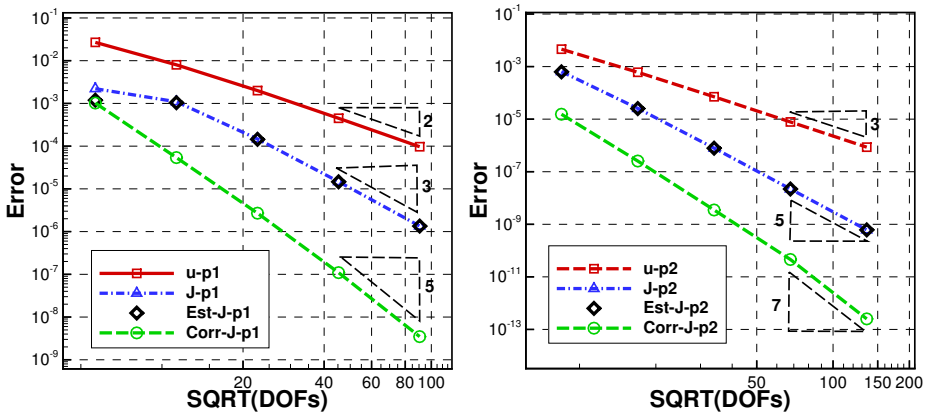
SPs	g	$\nabla \cdot \vec{F}$	primal L_2 err.	output error	output error est.	corrected output error
Gauss	DG	LP	k+1	2k+1	2k+1	2k+3
Gauss	DG	CR	k+1	2k+1	2k+1	2k+3
Lobatto	DG	LP	k+1	2k	2k	$\approx 2k+1$
Lobatto	DG	CR	k+1	2k	2k	2k+1
Gauss	G2	LP	k+1	2k	2k	2k+2
Gauss	G3	LP	k+1	2k-1	2k-1	2k+1

Table 3: Order of accuracy for the supersonic vortex transportation problem

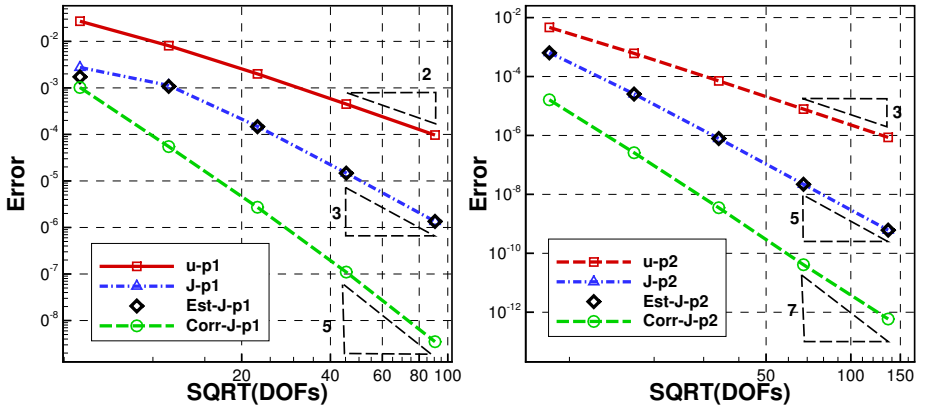
observed for the output with the dual-consistent boundary condition only. The spurious adjoint oscillation caused by the dual-inconsistent boundary condition degraded the adjoint solution, and destroyed the super convergence property of the output functional.

Figure 7 and 8 show the convergence rates of the CPR method with different solution points using both the LP and the CR approaches. Similar results were obtained with the LP and CR approaches for the Gauss points: a super-convergence of order $2k+1$ for the output functional and the error estimate and a super convergence of $2k+3$ for the corrected output. This indicates that the dual-consistency violation of the CR approach is relatively weak, and does not affect the adjoint-based error estimate. However, with the Lobatto points and the LP approach, accuracy loss did occur. The super convergence of the corrected output is lost for the $k = 1$ scheme. The CPR schemes with Lobatto points and the CR approach can reduce the alias error generated by the non-linear fluxes. Even though the CR approach is not fully dual-consistent, the super convergence rates are recovered for the Lobatto points, whose output functional and the error estimate are accurate to a super convergence order of $2k$ and the order of the corrected output is $2k+1$. Figure 9 shows the results of the CPR method with different correction functions. Similar results as the linear wave equation are obtained. For the Euler equations, the output functional and the adjoint-based error estimate with the Gauss point and g_m are accurate to order $O(h^{2k+2-m})$. Furthermore, the corrected outputs of the corresponding schemes are accurate to order $O(h^{2k+4-m})$. The convergence rates with the different CPR schemes are summarized in Table 3. The results of this test case indicate that the dual consistent formulation performs as expected for a non-linear equations and curved elements.

Inviscid Flow over the NACA0012 Airfoil The last test case in this section is a subsonic flow over a NACA0012 airfoil with a free-stream Mach number of $M_\infty = 0.5$ and an angle of attack, $\alpha = 2^\circ$. It is used to further assess the accuracy of the adjoint-based error estimation for a problem with a geometric singularity. The output of interest is chosen as the lift or drag of the airfoil. The contours of the C_L adjoint and the C_D adjoint are shown in Figure 1b and Figure 10. In this test case, the error in the functional

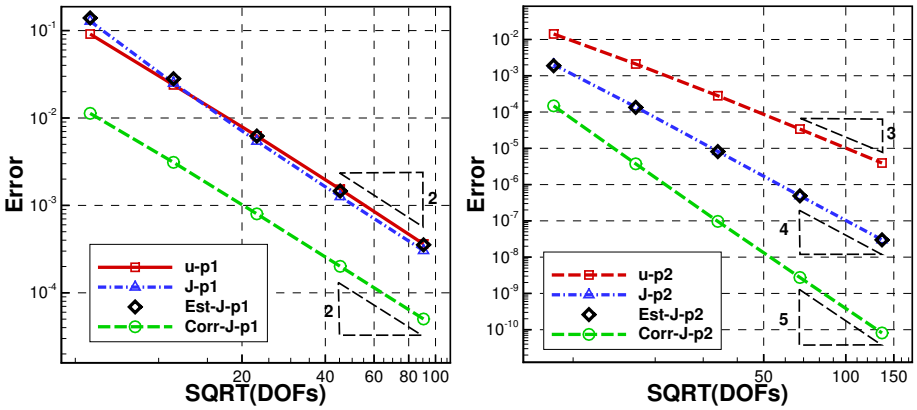


(a) Gauss points, LP and g_{dg}

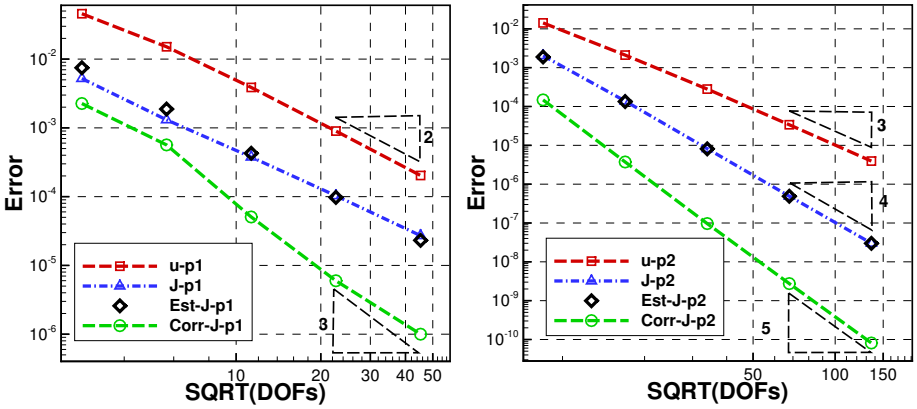


(b) Gauss points, CR and g_{dg}

Figure 7: The results of the supersonic vortex transportation problem with the Gauss points

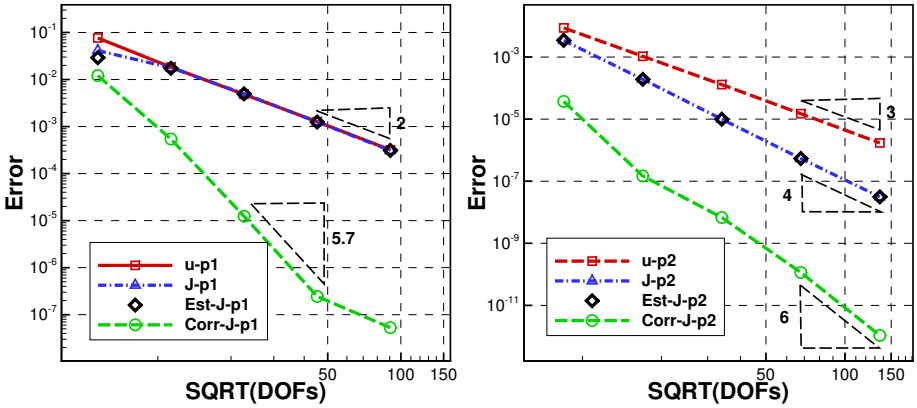


(a) Lobatto points, LP and g_{dg}

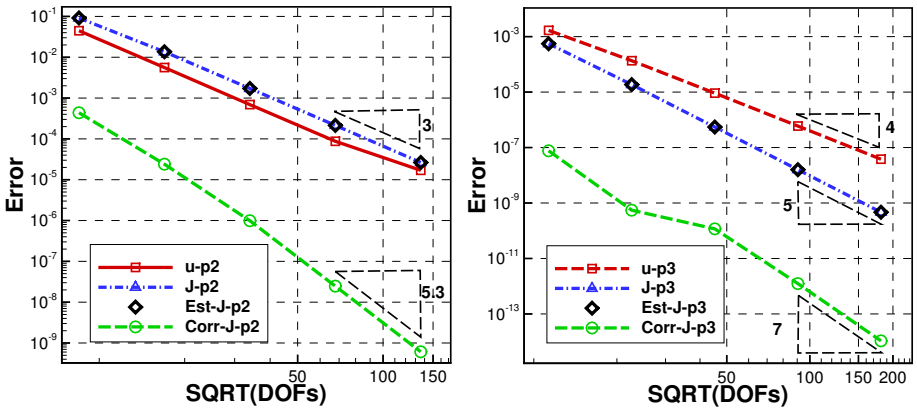


(b) Lobatto points, CR and g_{dg}

Figure 8: The results of the supersonic vortex transportation problem with the Lobatto points



(a) Gauss points, LP and g_2



(b) Gauss points, LP and g_3

Figure 9: The results of the supersonic vortex transportation problem

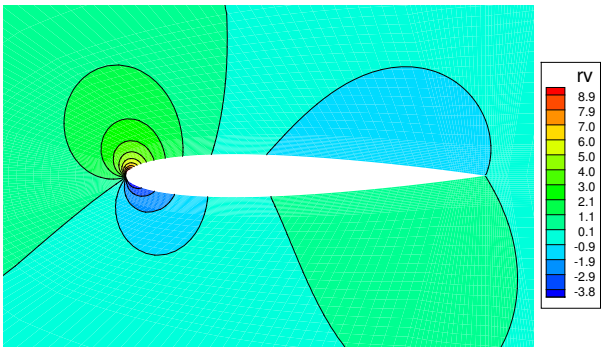


Figure 10: The 3^{rd} component of the C_D adjoint of an Inviscid NACA 0012 Airfoil at $M_\infty = 0.5, \alpha = 2^\circ$

$\mathcal{J}_H(Q_H) - \mathcal{J}_h(Q_h)$ is computed using p-enrichment from $k = 2$ to $k = 3$ and the effectivity of the error estimation is defined as

$$\eta_H^e = \frac{-\sum_j \omega_j |J_{i,j}| \, (\psi_h)_{i,j} \, r_{i,j}(Q_h^H)}{\mathcal{J}_H(Q_H) - \mathcal{J}_h(Q_h)}.$$

Due to the geometry singularity, the super convergence of the output functional and its error estimate is lost. Table 4 shows the results with 4 levels of uniformly refined meshes. Note that the error of the initial estimates on the very coarse meshes is large; however, the effectivity index η_H^e approaches unity as the mesh is refined for both C_L and C_D. In addition, the error estimate with the C_D adjoint is more accurate than with the C_L adjoint. This is due to the fact that the regularity of the C_L adjoint is low because of the singularity along the stagnation streaming line from the leading edge, while the C_D adjoint is relatively smooth.

Cells	C_L			C_D		
	$\mathcal{J}_H(Q_H) - \mathcal{J}_h(Q_h)$	$-(\psi_h)^T R_h(Q_h^H)$	η_H^e	$\mathcal{J}_H(Q_H) - \mathcal{J}_h(Q_h)$	$-(\psi_h)^T R_h(Q_h^H)$	η_H^e
280	-8.90E-03	-1.84E-02	2.064	8.39E-03	8.51E-03	1.015
1120	-5.52E-03	-7.15E-03	1.295	1.93E-03	1.95E-03	1.011
4480	-1.59E-03	-1.68E-03	1.058	2.29E-04	2.29E-04	1.002
17920	-3.08E-04	-3.24E-04	1.049	2.13E-05	2.13E-05	0.999

Table 4: Adjoint-based error estimate for an inviscid NACA 0012 airfoil at $M_\infty = 0.5, \alpha = 2^\circ$ (coarse space $k = 2$, fine space $k = 3$)

2.3 Adjoint-based Error Estimation and H-adaptation

2.3.1 Adjoint-based H-adaptation

An adjoint-based error estimation relates a specific functional output error directly to the local residual error. Therefore, it can be used to construct a very effective error indicator to drive an adaptive procedure for any engineering output. From Eq. 20, the output error can be estimated by performing a quadrature rule as

$$\delta \mathcal{J}_h(Q_h) \approx -\sum_i \sum_j \omega_j |J_{i,j}| \psi_{i,j} r_{i,j}(Q_h^H).$$

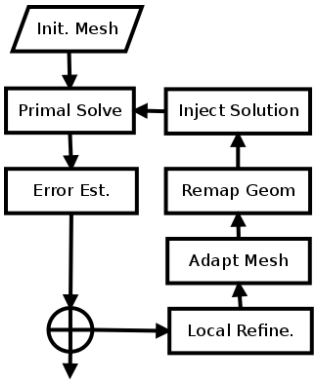


Figure 11: The procedure of the adjoint-based h-adaptation

The continuous adjoint solution ψ is approximated by solving ψ_h on the finner space through enriching the degree of the solution polynomial. The finer solution Q_h is obtained by performing several iterations of GMRES relaxation after prolongating from the coarse solution Q_H

$$Q_h^H = I_h^H Q_H,$$

with an injection operator I_h^H. The adjoint-based local error indicator η_i used in the present study is defined by taking an absolution value of the elemental output error contribution

$$\eta_i = |\sum_j \omega_j |J_{i,j}| \left(\psi_h - I_h^H \psi_H\right)_{i,j} r_{i,j}(Q_h^H)|.$$

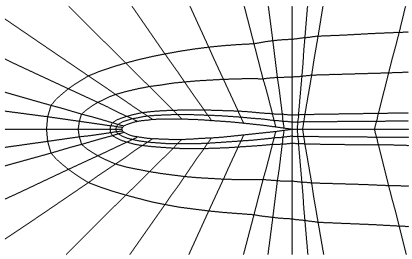
Here, to achieve a better estimate, the adjoint defect between the coarse level and fine level $\psi_h - I_h^H \psi_H$ is used. For a system of equations, the local error indicator is formed by summing together every component's contribution to the functional error estimate.

Figure 11 shows the procedure of the adjoint-based h-adaptation. A fixed-fraction hanging-node h-adaptation strategy is used in the present study. At each adaptation step, a fixed fraction $f = 10\%$ of the candidate elements with the largest error indicators are adapted. The marked elements are refined isotropically through its local mapping functions. As a result, the newly inserted boundary points may not lie on the geometry. In order to ensure the accuracy of the geometry approximation, all of the newly inserted boundary points are remapped to the real geometry by querying the stored geometry information for each boundary element. Then the coordinates of the interior points of the modified elements are updated by a transfinite interpolation from the boundary points.

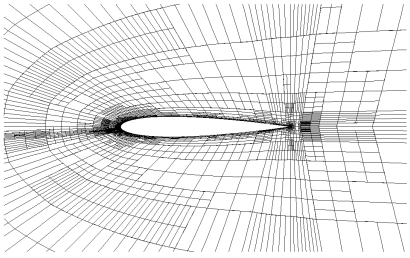
2.4 Numerical Results

2.4.1 Inviscid Flow over the NACA0012 Airfoil

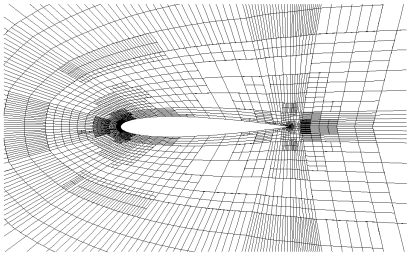
This is the same case used in section 2.2.5. To assess the effectiveness of the adaptation driven by the adjoint-based error indicator, h-adaptations with C_L and C_D as the output of interest are performed using quadrilateral elements with a 4th order scheme ($k = 3$). The newly inserted points on every adaptation stage are remapped to the real NACA0012 airfoil to reduce the geometry approximation error. The initial mesh consists of 560 $p4$ curved elements. The initial mesh and the final adapted meshes using lift and drag adjoints are shown in Figure 12. Regions near the trailing edge and around the airfoil



(a) The initial mesh



(b) C_L Adjoint



(c) C_D Adjoint

Figure 12: Adapted mesh for the adjoint-based h-adaptation for a inviscid NACA 0012 airfoil at $M_0 = 0.5,\ \alpha = 2^\circ\ (k = 3)$

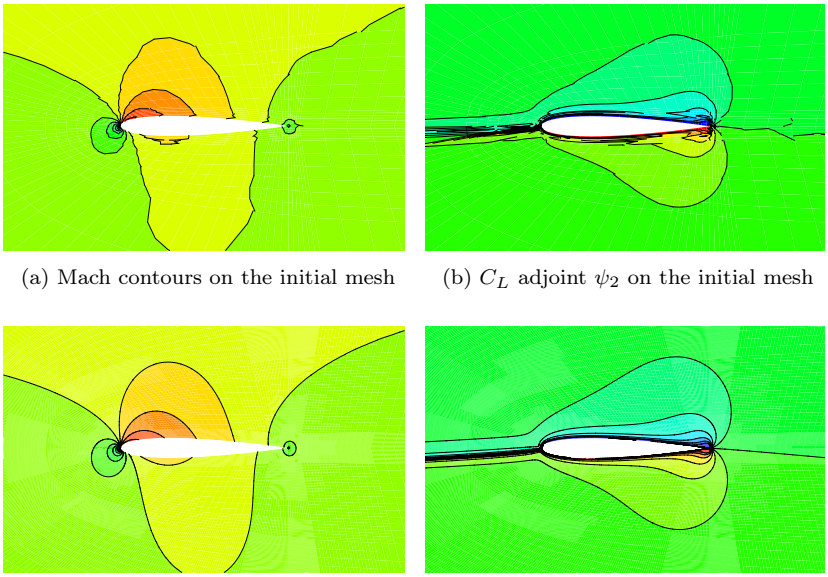
surface are refined using either the lift or drag adjoint. It is well-known that the trailing edge singularity can generate spurious entropy. Therefore, refinement near the trailing edge is very important to predict an accurate drag value. Adaptations using lift-adjoint added some degrees of freedom on the stagnation streamlines, where the lift adjoint is singular and oscillatory. Figure 13 displays the Mach and adjount solution distributions on the initial and adapted meshes. Note the significant improvement in solution smoothness in both the Mach and lift-adjoint contours from the initial mesh to the adapted mesh. This improvement was achieved because the present adaptation framework considers both of the primal and adjoint solutions.

The convergence histories of the lift and drag coefficients are shown in Figure 14. The corrected outputs are computed using the adjoint-based error estimate. The results show that the corrected values converge much faster than the uncorrected ones, and all converge to the same value. The estimated error at the last adaptation stage is around 10^{-10}. So the truth C_L and C_D in this section are the values from the final adapted mesh. Figure 15 shows the output error of the adaptation with the uniform refinement results for comparison. An effective convergence rate of at least 6th order was achived for both C_L and C_D with h-adaptation. It is clear that the adjoint based h-adaptation framework reduced the computational cost by orders of magnitude in term of the number of DOFs.

2.4.2 Laminar Flow over the NACA0012 Airfoil

In this case, we consider subsonic laminar flow over a NACA0012 airfoil with freestream $M_0 = 0.5$ and angle of attack $\alpha = 1^\circ$. The Reynolds number based on the chord length of the airfoil is $Re = 5000$. The same initial mesh for the inviscid test case was used in this test. The drag and lift coefficients are again the outputs of interest. Adjoint-based h-adaptation with $k = 3$ are driven by the output-based error indicator. Additionally, uniform h-refinement is performed to compare those adaptation strategies.

Figure 16 compares the Mach contours on the initial mesh and the final adapted mesh



(a) Mach contours on the initial mesh

(b) C_L adjoint ψ_2 on the initial mesh

(c) Mach contours on the adapted mesh

(d) C_L adjoint ψ_2 on the adapted mesh

Figure 13: Primal and adjoint solution of the adjoint-based h-adaptation for a inviscid NACA 0012 airfoil at $M_0 = 0.5,\ \alpha = 2^\circ$ $(k = 3)$

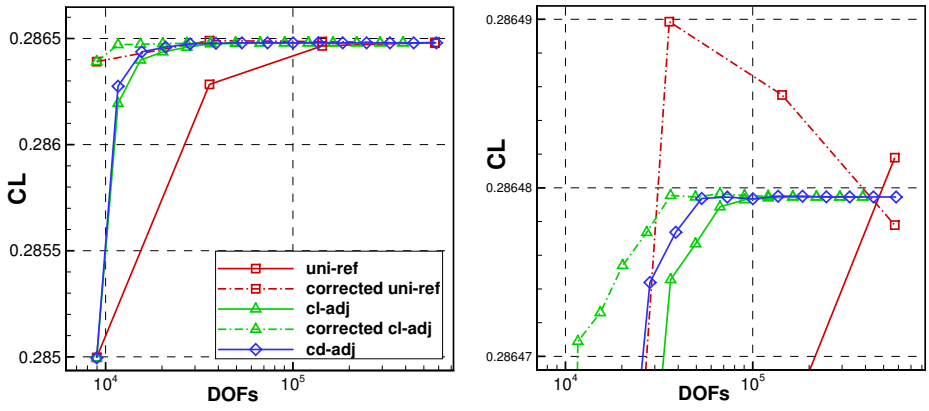
and presents the adapted meshes from different adaptation strategies. Note that regions near the stagnation streamlines and in the boundary layer were targeted for refinements. The trailing edge was also refined repeatedly to reduce the effect of the singularity.

Figure 17 displays the convergence of the lift and drag coefficients using adjoint-based error estimate in terms of nDOFs. It is obvious that both the lift and drag coefficients, from all adaptation strategies, converge to the same value. Note that the corrected outputs converge much faster than the uncorrected ones. The truth outputs are chosen from the output-based h-adaptive simulations with $k = 3$, whose estimated error is less than 10^{-8} at the final stage. Figure 18 compares the C_L and C_D error of all tested adaptation strategies with results from uniform h-refinements. With h-adaptation, an effective convergence rate of 6th order was achieved for both the C_L and C_D, as shown in the figure. Again it is shown that the ajoint based h-adaptation approach can reduce the computational cost by orders of magnitude.

2.4.3 Inviscid Flow over a Sphere

In this case, we consider subsonic inviscid flow over a sphere. The p3 hexahedral mesh is used for this simulation. Figure 19 shows the outline of the computational domain and the initial surface mesh on the sphere. The inflow Mach number is set to be 0.3 with an angle of attack $\alpha = 0^\circ$.

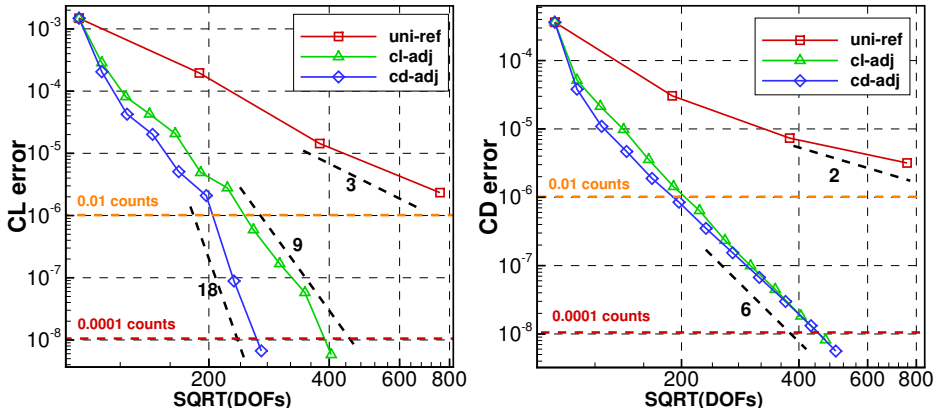
First, to demonstrate the super convergence of the output using a dual-consistent CPR formulation, a uniform refinement study is performed. As shown in Figure 20, a super convergence of $2k+1$ is obtained for the C_D error with $k = 1$ and $k = 2$. For the adaptive simulation, a relative coarse mesh which has 480 p3 hexahedral elements is used as the initial grid and the 3^{rd} order CPR scheme with the Gauss points as the SPs/FPs and the LP approach is tested. The adaptation is driven by the adjoint-based error indicator with drag as the output of interest. On each adaptation level, 10% of the current elements with



(a) C_L (b) C_L zoom in

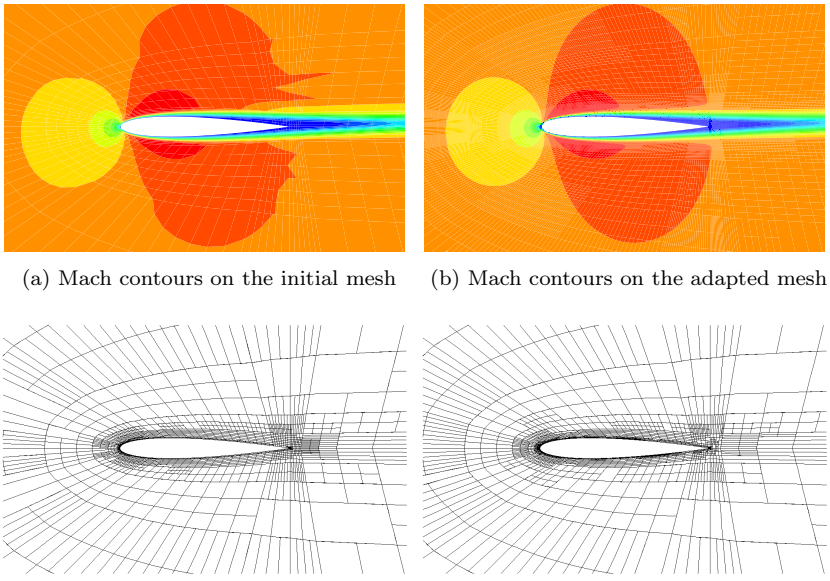
(c) C_D (d) C_D zoom in

Figure 14: C_L and C_D convergence of the adjoint-based h-adaptation for a NACA 0012 airfoil at $M_0 = 0.5,\ \alpha = 2^\circ\ (k = 3)$



(a) C_L Error (b) C_D Error

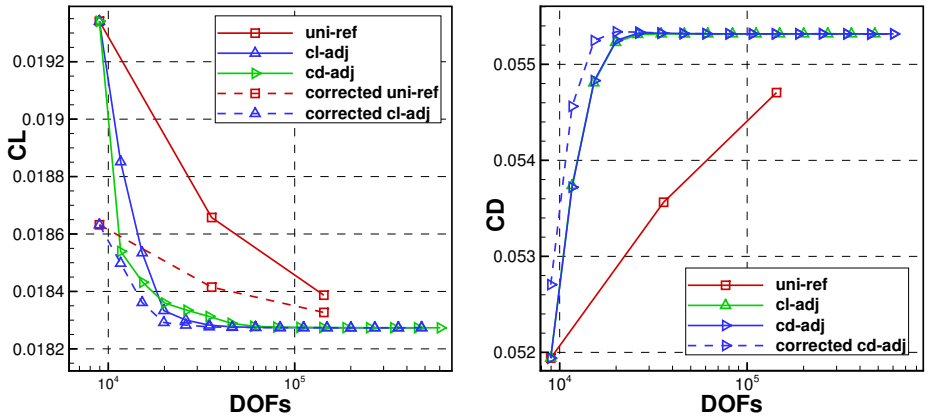
Figure 15: C_L and C_D error of the adjoint-based h-adaptation for a NACA 0012 airfoil at $M_0 = 0.5,\ \alpha = 2^\circ$ $(k = 3)$



(a) Mach contours on the initial mesh (b) Mach contours on the adapted mesh

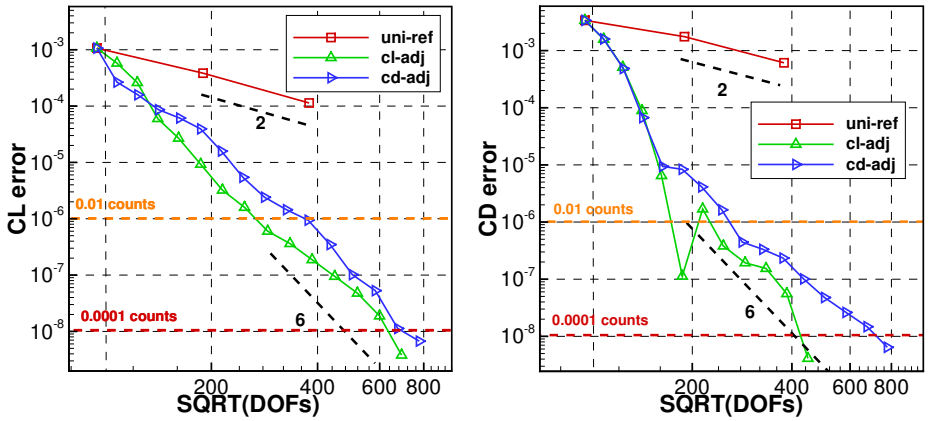
(c) C_L adjoint (d) C_D adjoint

Figure 16: Adjoint-based h-adapted mesh for a NACA 0012 airfoil at $M_0 = 0.5,\ \alpha = 1^\circ, Re = 5000$ $(k = 3)$



(a) C_L Error (b) C_D Error

Figure 17: C_L and C_D convergence for a NACA 0012 airfoil at $M_0 = 0.5,\ \alpha = 1°, Re = 5000$ $(k = 3)$



(a) C_L Error (b) C_D Error

Figure 18: C_L and C_D error for a NACA 0012 airfoil at $M_0 = 0.5,\ \alpha = 1°, Re = 5000$ $(k = 3)$

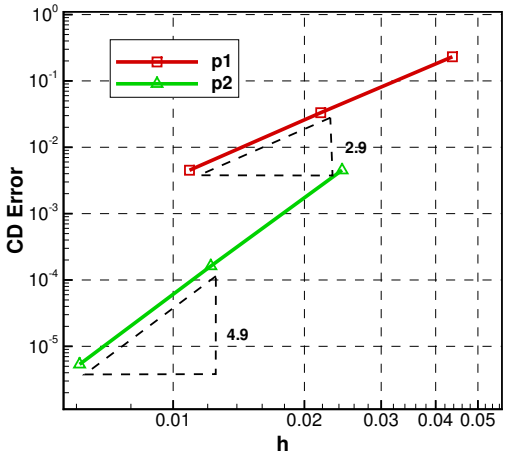


Figure 20: C_D error for the inviscid flow over a sphere problem.

the largest error are marked to be refined. The adapted mesh and the Mach contours are shown in Figure 21. Regions around the sphere surface are refined persistently. Figure 22 compares the drag coefficient error of the adaptive simulations with the results from the uniform h-refinements. It is clear to see that the current adaptive method could produce much more efficient error reductions in terms of the number of the DOFs. An effective convergence order of 6 is obtained through the adaptation, which is much faster than the uniform refinements. This preliminary adaptation results demonstrates the effectiveness of the present adaptive method for a 3D problem.

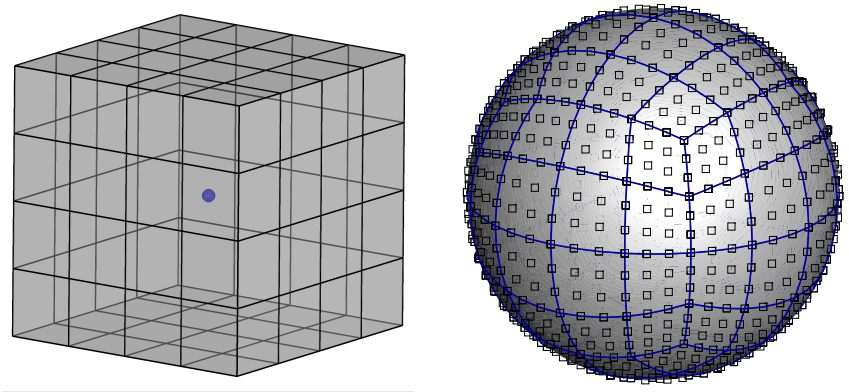
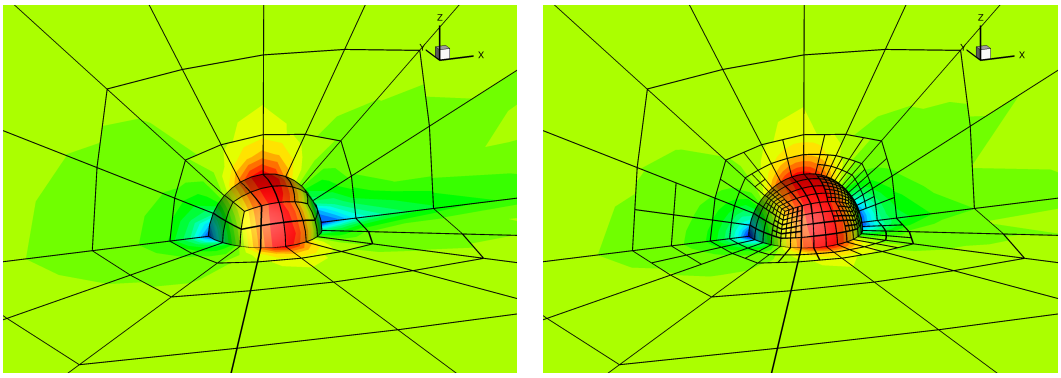


Figure 19: The initial $p3$ curved hexahedral mesh for the inviscid flow over a sphere problem.



(a) Adaptation level 1 (b) Adaptation level 4

Figure 21: Mach contours on the adapted mesh for the inviscid flow over a sphere at $M_0 = 0.3$, $\alpha = 2^\circ$ $(k = 2)$.

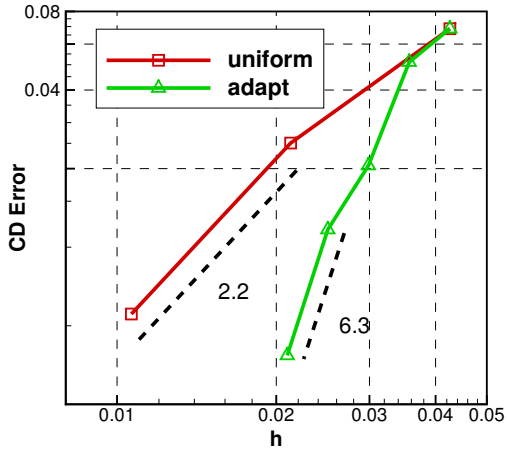


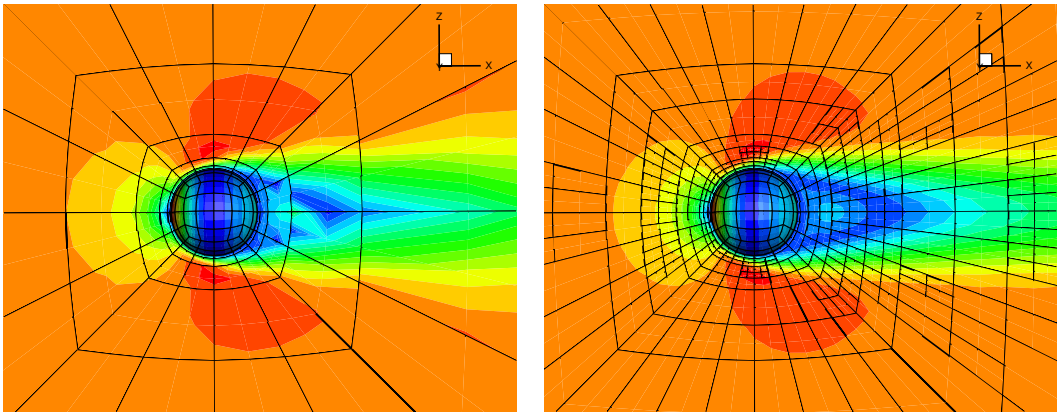
Figure 22: CD errors of the inviscid flow over the sphere at $M_0 = 0.3$, $\alpha = 2^\circ (k = 2)$.

2.4.4 Laminar Flow over a Sphere

Next, we consider steady viscous flow over a sphere. The same settings from the Ref. [94, 95] is utilized for the comparison purpose. The Reynolds number based on the sphere diameter is chosen to be 118. The inflow Mach number is 0.2535 with an angle of attack $\alpha = 0^\circ$. The 3^{rd} order CPR scheme with the Gauss points as the SPs/FPs and the LP approach is tested. Figure 23a shows the initial mesh which has 480 p3 hexahedral elements and the corresponding Mach number contours.

The adaptation is driven by the adjoint-based error indicator with drag as the output of interest. On each adaptation level, 10% of the current elements with the largest error are marked to be refined. The adapted mesh and the Mach contours on each adaptation level are shown in Figure 23. Regions around the sphere surface are refined persistently. The reference $C_D = 1.0162$ is chosen from Ref. [95, 19]. Figure 24 compares the drag

coefficient error of the adaptation with the result from the uniform h-refinements. A super convergence rate of $2k$ is obtained for the uniform h-refinement. The adjoint-based adaptation shows a faster convergence, whose effective order of accuracy is 5.9.



(a) The initial mesh (b) The adapted mesh, level 3

Figure 23: Mach contours and the adapted mesh for the viscous flow over a sphere problem at $M_0 = 0.2535$, $Re = 118$ and $\alpha = 0^\circ$ $(k = 2)$.

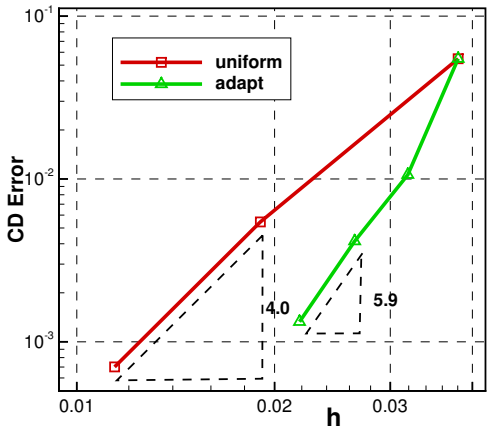
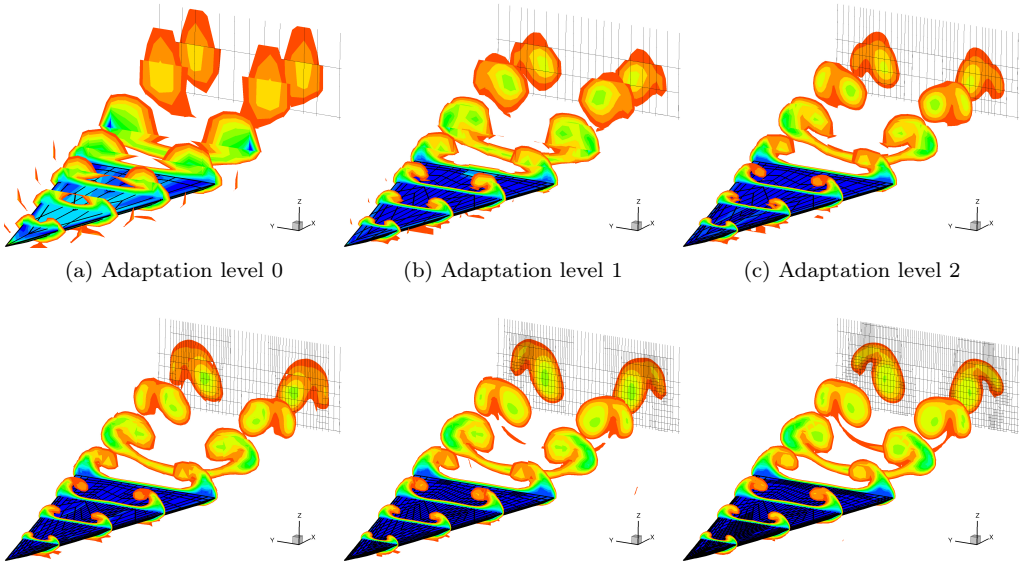


Figure 24: CD errors of the viscous flow over a sphere at $M_0 = 0.2535$, $Re = 118$, $\alpha = 0^\circ$ $(k = 2)$.

2.4.5 Laminar Flow over a Delta Wing

In this test case, we consider laminar flow over a delta wing at Mach number $M_0 = 0.3$ with a high angle of attack $\alpha = 12.5^\circ$. This case is a part of the International workshop on high-order methods (HOW) and the EU ADIGMA project. The Reynolds number based on the root chord is 4000. The Prandtl number is set to 0.72 and the constant viscosity



(a) Adaptation level 0 (b) Adaptation level 1 (c) Adaptation level 2

(d) Adaptation level 3 (e) Adaptation level 4 (f) Adaptation level 5

Figure 25: Mach number and the adapted mesh slices on different adaptation stages for the delta wing case.

is used. The delta wing has a swept sharp leading-edge and a blunt trailing edge. The Mach number contours of the flow around the delta wing at different adaptation stages are shown in Figure 25. Both of the singularity along the leading edge and the regions around the smooth vortices are targeted to refine. The Mach number distribution is much smoother after several adaptation levels. Figure 26 shows the residual history of the whole adaptation procedure. For each adaptation stage, the residual norm drops 10 order. Figure 27 compares the CD error and the CPU time of the adaptation with the result from the uniform refinements.

2.4.6 Laminar Flow over an Analytic 3D Body

As the final case, we consider laminar flow over a streamlined analytic 3D body. This test case is a part of the International workshop on high-order methods (HOW) and the EU ADIGMA project. The inflow Mach number is set to 0.5 at an angle of attack $\alpha = 1°$. The Reynolds number is 5000 with adiabatic no-slip wall boundary condition enforced on the body surface. The viscosity is assumed to be a constant and Prandtl number is set to 0.72. The reference area is 0.05 and the reference drag coefficient $C_D = 0.06317$ from the HOW results is used. The coarsest mesh from the HOW website is used as the initial mesh, which consists of 768 p4 curved hexahedral elements. The drag adjoint is used to drive the mesh adaptation procedures. The adapted mesh and the Mach contours are shown in Figure 28b. The refinements are mainly performed around the leading edge and around the body surface. Figure 29 display the residual history and the C_D error.

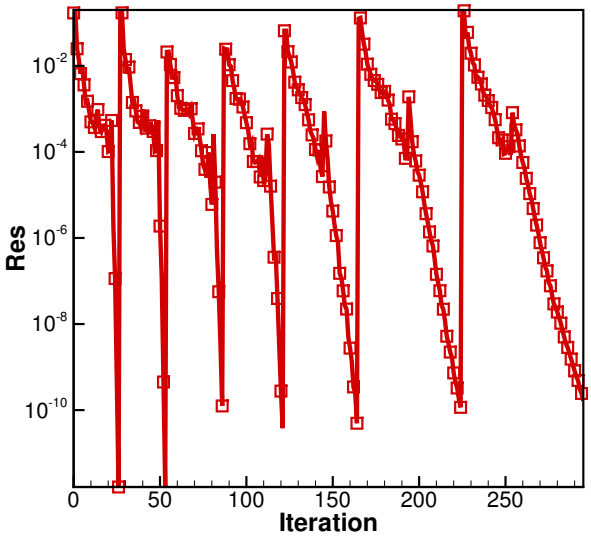


Figure 26: The residual history of the whole adaptation procedure for the delta wing case.

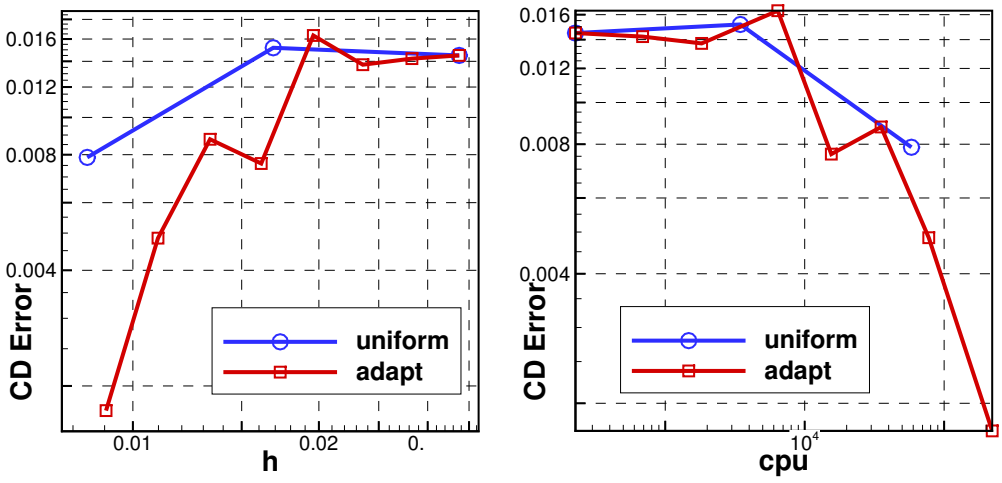
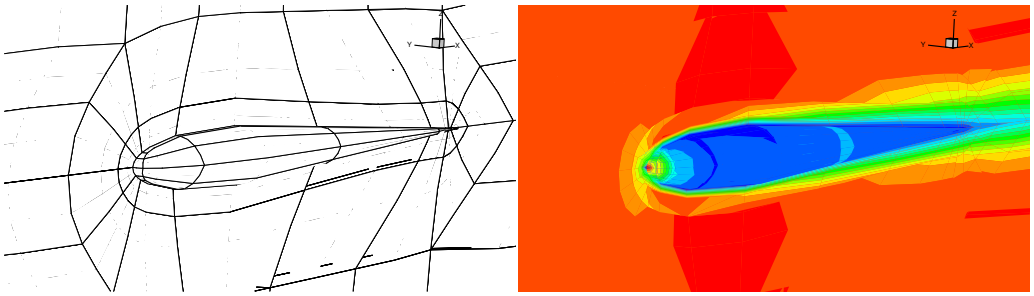
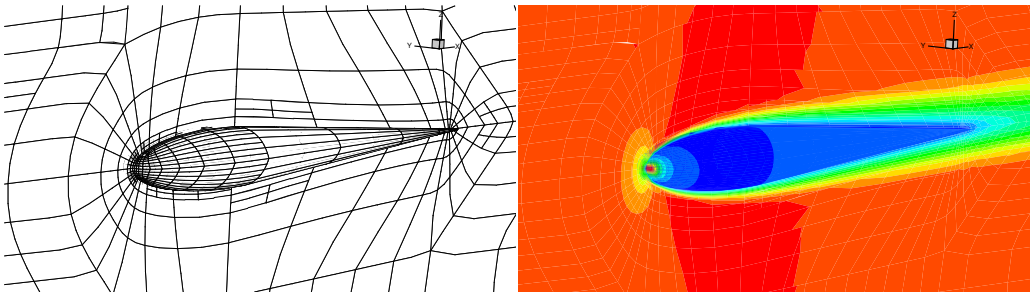


Figure 27: C_D error and the CPU time of the delta wing case.

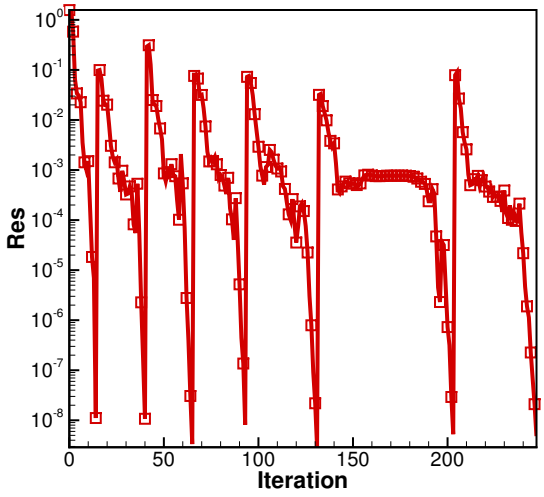


(a) The initial mesh and Mach contours

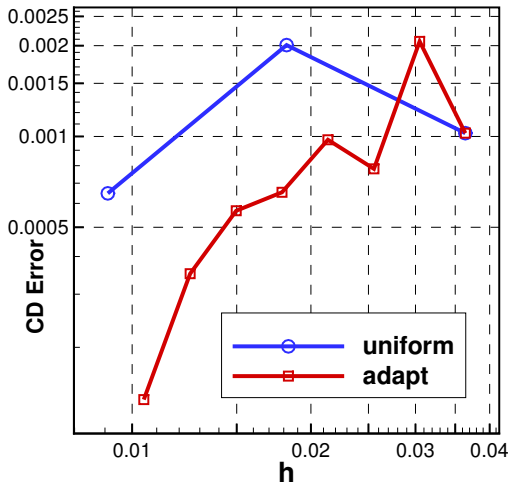


(b) The adapted mesh and Mach contours

Figure 28: Laminar flow over an analytic 3D body at $M_0 = 0.5$, $\alpha = 1^\circ$ and $Re = 5000$.



(a) Residual history



(b) C_D Error

Figure 29: Results of the laminar flow over an analytic 3D body.

3 Results and Discussion on Subgrid-scale Stress Model Evaluation

3.1 Governing Equation and SGS Models

The governing equations for three dimensional turbulent flows are the three dimensional Navier-Stokes equations. As a simpler counterpart, we consider the one dimensional Burgers' equation,

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \nu\frac{\partial^2 u}{\partial^2 x}, x \in [-1, 1], \tag{36}$$

where u is the state variable such as velocity, ν is a constant viscosity. In the present study, $\nu = 8E-05$ is chosen to imitate a high Reynolds number flow problem. To derive the LES governing equation, we apply a low-pass spatial filter, $G_\Delta(x, \xi)$ satisfying the following conservative property

$$\int_{-\infty}^{\infty} G_\Delta(x, \xi)d\xi = 1, \tag{37}$$

where Δ denotes the filter width. A typical filter, the box filter, is defined below,

$$G_\Delta(x, \xi) = \begin{cases} \frac{1}{\Delta} & |x - \xi| \leq \frac{\Delta}{2}, \\ 0 & otherwise \end{cases} \tag{38}$$

The filtering process is defined mathematically in the physical space as a convolution product. The filtered variable $\hat{\phi}(x, t)$ of a space-time variable $\phi(x, t)$ in 1D is defined as

$$\hat{\phi}(x, t) = \int_{-\infty}^{\infty} G(x, \xi)\phi(x, t)d\xi. \tag{39}$$

The filtering process is linear, i.e. $\phi \hat{+} \varphi = \hat{\phi} + \hat{phi}$. If the filter width is constant, the differential and the filter operators commute, i.e. $\frac{\hat{\partial\phi}}{\partial x} = \frac{\partial \hat{phi}}{\partial x}$. In the present study, all the filtering processes are done with the box filter. After applying the filter to Eq.36, we obtain

$$\frac{\partial \hat{u}}{\partial t} + \hat{u}\frac{\partial \hat{u}}{\partial x} = \nu\frac{\partial^2 \hat{u}}{\partial^2 x} - \frac{\partial(\frac{1}{2}\hat{uu} - \frac{1}{2}\hat{u}\hat{u})}{\partial x}. \tag{40}$$

The unclosed term arises due to the filtering of the nonlinear convection term

$$\tau^{SGS} = \frac{1}{2}\hat{uu} - \frac{1}{2}\hat{u}\hat{u}. \tag{41}$$

This is the SGS of the Burgers' equation. SGS models act as the closure of the governing equation. In this section we review some of the ideas and translate them to work for the one dimensional Burgers' equation.

3.1.1 Static Smagorinsky Model

The SS is in the eddy viscosity form. For 3D incompressible flow, the SGS stress is defined as,

$$\tau_{ij}^{SGS} = -2\nu_{SGS}\hat{S_{ij}}, \tag{42}$$

where $\hat{S}_{i,j}$ is the resolved rate of strain tensor, and

$$\hat{S}_{i,j} = \frac{1}{2}(\partial_i \hat{u}_j + \partial_j \hat{u}_i). \tag{43}$$

The SGS viscosity, ν_{SGS} is modeled following the mixing length idea

$$\nu_{SGS} = (c_s\Delta)^2\sqrt{2|\hat{S}|^2}, \tag{44}$$

where $|\hat{S}|^2 = \hat{S}_{ij}\hat{S}_{ji}$, c_s is the prescribed coefficient. By comparing the mean SGS dissipation from DNS data and the modeled SGS dissipation, c_s can be determined. Lilly used this procedure for isotropic turbulence to obtain $c_s = 0.16$. The SS was described by Moin and Kim Moin P [81], Rogallo and Moin Rogallo RS [86], Lesieur and Metais Lesieur M [66] and Pope SB [88]. The deficiency of this model first showed up in the comparison of the modeled SGS stress and the true SGS stress computed from the DNS solution by Clark et al. Clark RA [22], McMillan and Ferziger McMillan OJ [79], and Bardina et al. Bardina J [6]. The comparisons imply that the model does not capture the SGS adequately. In Meneveau C [80], Meneveau et al. gave an explanation of this problem. Another weakness of this model is that it gives non-zero eddy viscosity in laminar-flow regions. Therefore a wall function is needed to damp the SGS viscosity in a wall-bounded flow. Next, we derive its 1D formulation. The rate of strain in 1D is

$$\nu_{SGS} = (c_s\Delta)^2|\partial_x\hat{u}|. \tag{45}$$

Therefore the SGS stress becomes

$$\tau^{SGS} = -\nu_{SGS}\hat{S} \tag{46}$$

3.1.2 Dynamic Smagorinsky Model

The coefficient, c_s, in SS is prescribed. However, it is found empirically that c_s depends on the flow, being 0.1 for plane channel flow and 0.2 for isotropic turbulence Durbin P. A [28]. The DS makes it a variable spatially and temporally. It introduces a test filter to the resolved scales and uses the assumption of scale invariance to compute the model coefficient. As the model for three dimensional turbulence is readily available, we derive it for the 1D Burgers' equation next.

Following Eq. 40, we consider the 2^{nd} filter with width $\hat{\Delta}$, defined as $\hat{\Delta} = \gamma\Delta$. By applying this filter to the SGS stress, we obtain

$$\hat{\tau}^{SGS} = \frac{1}{2}\widehat{u\tilde{u}} - \frac{1}{2}\hat{\tilde{u}}\hat{\tilde{u}}. \tag{47}$$

By applying the filter to the LES solution, we obtain the resolved stress,

$$L = \frac{1}{2}\hat{\tilde{u}}\hat{\tilde{u}} - \frac{1}{2}\tilde{\hat{u}}\tilde{\hat{u}}. \tag{48}$$

The Germano identity can be written as

$$T = \tilde{\tau}^{SGS} + L \tag{49}$$

where $T = \frac{1}{2}\widehat{u\tilde{u}} - \frac{1}{2}\tilde{\hat{u}}\tilde{\hat{u}}$. We apply the SS to both T and τ nd assume they share the same coefficient, c_s,

$$-(c_s\tilde{\Delta})^2|\partial_x\tilde{\hat{u}}|\partial_x\tilde{\hat{u}} = -(c_s\Delta)^2|\partial_x\hat{\tilde{u}}|\partial_x\hat{\tilde{u}} + L. \tag{50}$$

We define

$$M = \Delta^2|\partial_x\hat{\tilde{u}}|\partial_x\hat{\tilde{u}} - \tilde{\Delta}^2|\partial_x\tilde{\hat{u}}|\partial_x\tilde{\hat{u}}. \tag{51}$$

Thus $c_s^2 = \frac{L}{M}$. It is assumed that c_s is spatially uniform so that it can be extracted from the test-filtering operation (Ghosal et al 1995)Ghosal S [39]. In the 1D test, we take the

most common choice of $\gamma = 2$. In three dimensions, this is an over-determined system. To minimize the square error, Lilly used the following approach

$$c_s^2 = \frac{\langle L_{ij} M_{ij} \rangle}{\langle M_{ij} M_{ij} \rangle}, \tag{52}$$

where $\langle \cdot \rangle$ means averaging along the homogeneous direction. The DS gives a highly variable eddy viscosity field Germano M and WH [38] including negative values which makes the simulation unstable. Averaging over homogeneous directions was used by Germano et al. Germano M and WH [38] to prevent this problem. Ghosal et al. Ghosal S [39] showed that this procedure minimizes the total error in the homogeneous region over which the averaging is performed. With these modifications, the eddy viscosity still can be negative. So the value of c_s^2 is clipped to be non-negative. In 1D, we don't have these problems. Thus we don't use a least square averaging operation. But we still require c_s^2 to be non-negative.

3.1.3 Scale-Similarity model

The SSM was first introduced by Bardina et al. Bardina J [6]. It assumes scale invariance between the computable stress L and the SGS stress τ^{SGS}. This assumption was verified with empirical band pass-filtered PIV measurements by Liu et al. Liu S [73]. It suggests that is similar to a stress constructed from the resolved scales,

$$\tau^{SGS} = c_{ssm} L, \tag{53}$$

where L s the resolved stress, which is given in Eq. 48. Many different second filter widths were suggested by various researchers. The Bardina's original model uses the same filter width for the two filters, i.e. $\Delta = \hat{\Delta}$ and $\gamma = 1$. Liu et al used $\gamma = 2$ and Akhavan et al use $\gamma = \frac{4}{3}$ Akhavan R [2]. The coefficient c_{ssm} is empirical and found to be close to 1. In the 1D test, c_{ssm} is adjusted to be 0.25 with , based on an analysis performed in Z.J. Wang [120]. In Bardina J [6], the true and modeled stresses showed a high degree of correlation in Bardina et al's a priori tests, and the SSM allowed for energy backscatter. However, this model was found to be not sufficiently dissipative. Energy accumulated at small scales and finally led to numerical instability. In the present study, we will duplicate this result with a non-dissipative numerical scheme, and will show that the phenomenon does not occur with the dissipative FR/CPR method.

3.1.4 Mixed Model

To resolve the above-mentioned problem of the SSM, the DS is included in the formulation to add extra dissipation. In three dimensions, the mixed model (MM) is

$$\tau_{ij}^{SGS} = c_{ssm} L_{ij} - 2\nu_{SGS} \hat{S}_{ij}. \tag{54}$$

Liu et al. showed that the magnitude of the similarity term is much larger than that of the dissipative DS term. Hence, the high correlation of the SSM is not degraded by the extra viscosity. Zang et al. used this model for recirculating flows with $\gamma = 1$. Wu and Squires applied this model successfully with Lagrangian averaging in simulations of 3D boundary layers Wu X [114]. There are dynamic ways to determine c_{ssm} as well. Vreman et al proposed a two-parameter dynamic MM in which c_s and c_{ssm} are both calculated dynamically with $\gamma = 1$. In the present one dimensional study, the values $c_{ssm} = 0.25$ and $\gamma = 2$ are used. The SGS stress is defined as

$$\tau = c_{ssm}(\frac{1}{2}\tilde{\hat{u}}\tilde{\hat{u}} - \frac{1}{2}\tilde{\hat{u}}\tilde{\hat{u}}) - \nu_{SGS}\hat{S}. \tag{55}$$

As will be shown later, the numerical instability of the SSM model is not a problem for numerical schemes with embedded numerical dissipation. The study will be described and discussed in Section 4.

3.1.5 Linear Unified RANS-LES Model

The wall-bounded turbulent flows at high Reynolds number are a significant challenge for LES. The near wall region requires a high resolution grid to resolve the small energetic scales. The linear unified RANS-LES model (LUM) combines RANS with LES to solve this problem. The model equations for incompressible flows are

$$\begin{cases} \frac{\tilde{D}\tilde{U}_i}{\tilde{D}t} = -\frac{\partial(\frac{\langle p\rangle}{\rho}+\frac{2k_t}{3})}{\partial x_i} + 2\frac{\partial(\nu+\nu_t)\tilde{S}_{ik}}{\partial x_k} \\ \frac{\tilde{D}k_t}{\tilde{D}t} = -\frac{\partial((\nu+\nu_t)\frac{\partial k_t}{\partial x_i})}{\partial x_i} + \nu_t S^2 - 2\frac{(1-c_0)k_t}{\tau_L} \\ \frac{\tilde{D}\omega}{\tilde{D}t} = C_{\omega 1}\frac{\omega}{k}nu_t S^2 - \frac{C_{\omega 2}}{C_k k}\omega^2 + \frac{\partial((\nu+\frac{\nu_t}{\sigma_{omega}})\frac{\partial\omega}{partialx_j})}{\partial x_j} + \frac{C_\omega}{k}(\nu+\nu_t)\frac{\partial k_t}{\partial x_j}\frac{\partial\omega}{\partial x_j} \end{cases}$$

where $C_{\omega 1}, C_{\omega 2}, C_\omega, C_k, c_0 and \sigma_\omega$ are all model constants, $\hat{U}_i$ is the filtered velocity, k_t is the turbulent kinetic energy, ω is the specific dissipation, τ_L is the time scale and ν_t is the modeled viscosity. In this work, we only focus on the LES aspect of the model. Therefore τ_L is calculated with $\tau_L = l_{ast}\Delta/k^{\frac{1}{2}}$, where $l_* = \frac{1}{3}$. In summary, for the one-dimensional Burgers' equation, we solve

$$\begin{cases} \frac{\hat{D}\hat{u}}{\hat{D}t} = \frac{\partial(\nu+\nu_t)\frac{\partial\hat{u}}{\partial x}}{\partial x} \\ \frac{\hat{D}k_t}{\hat{D}t} = -\frac{\partial((\nu+\nu_t)\frac{\partial k_t}{\partial x})}{\partial x} + 2\nu_t(\frac{\partial\hat{u}}{\partial x})^2 - 2\frac{(1-c_0)k_t}{\tau_L} \end{cases}$$

where $\nu_t = \frac{k_t\tau_L}{3}$, $\tau_L = l_*\Delta/k_t^{\frac{1}{2}}$.

3.2 Numeric Methods

3.2.1 The High-order CPR/FR Method for the 1D Burger's Equation

To give a complete picture of the LES, in this section, we briefly review the FR/CPR method's formula for the 1D Burgers' equation. Huynh Huynh [54] developed a high-order FR/CPR formulation, which was later employed for the Navier-Stokes equations on hybrid 3D meshes T. Haga and Wang [94]. It has been used for 1D, 2D and 3D laminar and turbulent flows. Validations and successful applications can be found in Gao and Wang [35] Gao and Wang [36] Shi and Wang [92] Shi and Wang [91] Yu et al. [118] Li and Wang [69]. In this study, we apply the 3^{rd} order FR/CPR scheme to the 1D Burgers' equation and evaluate its performance with various SGS models. The FR/CPR formulation for the inviscid 1D Burgers' equation is given as

$$\frac{\partial u_{i,j}}{\partial t} + \Pi(\frac{\partial F(u_i)}{\partial x}) + \frac{1}{\Delta x_i}(\alpha_{R,j}[F^n]_i + 1/2 alpha_{L,j}[F^n]_i - 1/2) = 0, \tag{56}$$

where $u_{i,j}$ is the solution at solution point j of element i, u_i is the solution polynomial for element i, $\Pi(\frac{\partial F(u_i)}{\partial x})$ denotes the projected flux derivative at the solution point, $Deltax_i$ is the length of element i, $[F^n]_{i+1/2}$ and $[F^n]_{i-1/2}$ are the differences between the local flux and the common Riemann flux at the right and left interfaces of element i, $\alpha_{R,j}$ and $\alpha_{L,j}$ are the correction coefficients independent of the solution variables. For the viscous term on the right hand side of Burgers' equation, we follow the BR2 approach Bassi and Rebay [10]. The 1D version is described below. First we introduce a new variable $R = \frac{\partial u}{\partial x}$. The corrected gradient is then

$$R_{i,j} = \frac{\partial u_{i,j}}{\partial x} + \frac{1}{\Delta x_i}(\alpha_{R,j}[u^{com} - u_i]_{i+1/2} + \alpha_{L,j}[u^{com} - u_i]_{i-1/2}), \tag{57}$$

where $[u^{com}]_{i+1/2}$ and $[u^{com}]_{i-1/2}$ are the common solutions at interfaces, $[u_i]_{i+1/2}$ and $[u_i]_{i-1/2}$ are the solutions from element i at the interfaces. The common solution is defined as

$$u^{com}_{i+1/2} = \frac{u^-_{i+1/2} + u^+_{i+1/2}}{2}, \tag{58}$$

where $u^-_{i+1/2} = [u_i]_{i+1/2}$ and $u^+_{i+1/2} = [u_{i+1}]_{i+1/2}$ are the solutions at the left and right sides of interface $i + 1/2$. Next, the viscous flux, $F^\nu = \nu\frac{\partial u}{\partial x}$, at solution points can be calculated by

$$F^\nu_{i,j} = F^\nu(R_{i,j}). \tag{59}$$

Then $\frac{\partial F^\nu}{\partial x}$ can be obtained by using the Lagrange polynomial approach. The common viscous flux at the interface is needed to correct $\frac{\partial F^\nu}{\partial x}$ at solution points,

$$F^{\nu,com}_{i+1/2} = \nu(\frac{\partial u}{\partial x})^{com}_{i+1/2}. \tag{60}$$

For the common gradient,

$$(\frac{\partial u}{\partial x})^{com}_{i+1/2} = \frac{1}{2}((\frac{\partial u}{\partial x})^-_{i+1/2} + r^-_{i+1/2} + (\frac{\partial u}{\partial x})^+_{i+1/2} + r^+_{i+1/2}). \tag{61}$$

where $(\frac{\partial u}{\partial x})^-_{i+1/2}$ and $(\frac{\partial u}{\lceil}\partial x])^+_{i+1/2}$ are the gradients of the solution of the left and right cells with no correction, $r^-_{i+1/2}$ and $r^+_{i+1/2}$ are the corrections to the gradients due to the common solution at the interface. More specifically, the corrections are,

$$r^-_{i+1/2} = \frac{1}{\Delta x_i}(\alpha^-[u^{com} - u^-]_{i+1/2}), \tag{62}$$

$$r^+_{i+1/2} = \frac{1}{\Delta x_{i+1}}(\alpha^+[u_{com} - u^+]_{i+1/2}), \tag{63}$$

where $alpha^-$ and α^+ are the interface correction coefficients.

3.2.2 Temporal Discretization

The explicit SSP three-stage 3^{rd} order Runge-Kutta scheme Gottlieb and Shu [43] is used as the temporal discretization. Here we give a brief description. Rewrite the discretized Burgers' equation as

$$\frac{\partial U}{\partial t} = Res(U), Ut_0 = U_0, \tag{64}$$

where $Res(U)$ is a function of solution U and t. Given solution U^n, we obtain solution U^{n+1} using

$$U^{(1)} = U^n + \Delta t Res(U^n), \tag{65}$$

$$U^{(2)} = \frac{3}{4}U^n + \frac{1}{4}U^{(1)} + \frac{1}{4}\Delta t Res(U^{(1)}), \tag{66}$$

$$U^{n+1} = \frac{1}{3}U^n + \frac{2}{3}U^{(2)} + \frac{2}{3}\Delta t Res(U^{(2)}). \tag{67}$$

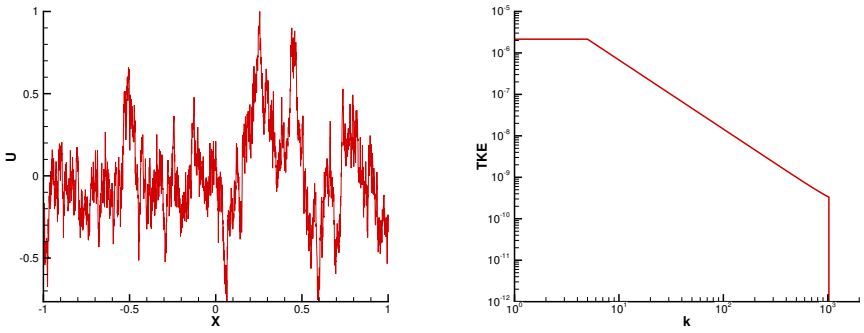


Figure 30: Initial condition (left) and the initial energy spectrum (right)

3.3 Initial and Boundary Conditions for 1D Burgers' Equation

To imitate turbulence, the initial energy spectrum is given in the Fourier space k. In the present work, the following initial spectrum is used

$$E_0(k) = \begin{cases} A5^{-\frac{5}{3}} & 1 \leq k \leq 5, \\ Ak^{-\frac{5}{3}} & k > 5, \end{cases} \tag{68}$$

where k is an integer varying from 1 to 1280. For each k, the velocity u has a random phase angle, β, in $[-\pi, \pi]$.

$$u(x) = \sum_{i}^{n} (2E_0(k_i))^{\frac{1}{2}} sin(k_i x + \beta_i) + 1 \tag{69}$$

A is a constant to make the turbulence intensity $\frac{u'}{\bar{u}} = 0.7\%$, where $u' = \sqrt{(\frac{\sum_{i=1}^{N}(u_i - \bar{u})^2}{N})}$, $\bar{u} = 1$. The two boundaries are set to be periodic due to the periodicity of the initial condition.

3.4 Grid and Spatial filter

The computational domain is $[-1, 1]$. A mesh refinement study indicated that 8,192 cells with the 3^{rd} order FR/CPR method are required to resolve all the scales. Figure 31 shows the energy spectrum at for the linear wave propagation with the same initial condition. There is no visible decay at even the highest frequency. We consider the simulation at this resolution a DNS, and denote $\Delta x_{DNS} = \frac{2}{8192}$. In the a priori study, the DNS solution is filtered with a box filter with $\Delta = 32\Delta x_{DNS}$. The filtered solution is obtained

$$\hat{u}_{i,j} = \sum_{n=1}^{N} \sum_{m=1}^{K+1} u_{n,m} * w_{n,m,i,j}, \tag{70}$$

where N is the number of cells in the filtering stencil of the current degree of freedom and K is the degree of the polynomial of the solution. In each cell, a Gauss quadrature rule was implemented and w is the weighting coefficient. Then the filtered solution on the DNS grid is projected to the (coarse) LES grid if necessary to serve as the LES solution. In the current LES cell, the projected solution at each solution point j is calculated using

$$\int_{\Delta x_{LES}} L_k \sum_{j}^{K+1} L_j \hat{u}_{i,j} dx = \sum_{n=1}^{N} \int_{\Delta x_{DNS}} L_k \sum_{j}^{K+1} l_j \hat{u}_{n,j} dx, \tag{71}$$

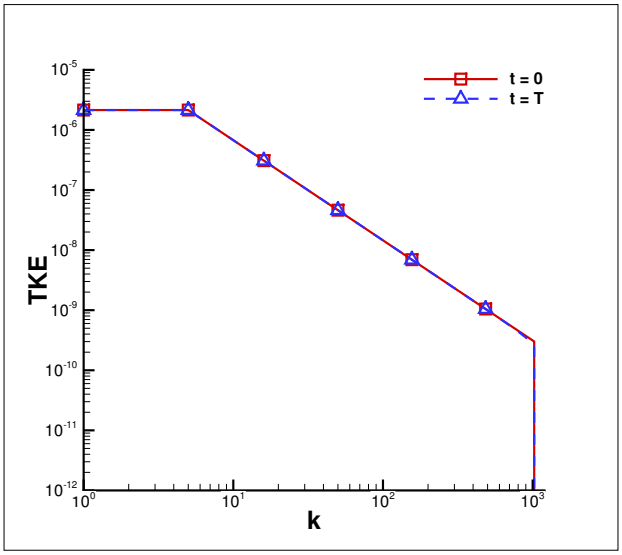


Figure 31: Turbulent energy spectrum at $t = 0$ and $t = T$

Table 5: Correlation coefficients of a priori test

$\frac{\Delta x_{LES}}{\Delta x_{DNS}}$	SS	DS	SSM	Mix	LUM
1	-0.10	0.6	0.95	0.89	-0.09
2	-0.10	0.6	0.95	0.89	-0.09
4	-0.10	0.6	0.95	0.89	-0.09
8	-0.10	0.6	0.95	0.89	-0.09
16	-0.10	0.6	0.95	0.89	-0.09
32	-0.09	0.59	0.95	0.88	0.04

where L_j is the shape function defined based on the solution points of the LES cell, l_j is the shape function based on the solution points of the DNS cell. In the a posteriori study, we do the same thing to the DNS initial condition to generate the LES initial condition.

In Figure 32, $\Delta = 8\Delta x_{DNS}$, $\Delta x_{LES} = 4\Delta x_{DNS}$ are used to demonstrate the filtering operation. Different cell sizes for LES were tested to evaluate the influence of the truncation error and the SGS modeling error. We call the filter used on the initial condition the first filter and the filter used in deciding the coefficient of the dynamic model or computing the resolved SGS stress the test filter. The test filter width is 2 times the width of the first filter, which makes $\gamma = 2$.

3.5 Numerical Results and Discussions

In this section, the results for the a priori and a posteriori tests are presented. Due to the nonlinear convection term, shock waves start to appear after a certain time. Thus all results are obtained at a time $T = 0.1$, when the solution is still smooth. Figure 33 shows the energy spectrum at $t = 0$ and $t = T$ of the DNS. We can see that the high frequencies are damped out by the physical viscosity while the lower frequencies remain.

3.5.1 A Priori Tests

Figure 34 shows the SGS stress computed using different models based on the filtered-DNS data at $t = T$ with various mesh resolutions and a fixed filter width of $\Delta = 32\Delta x_{DNS}$.

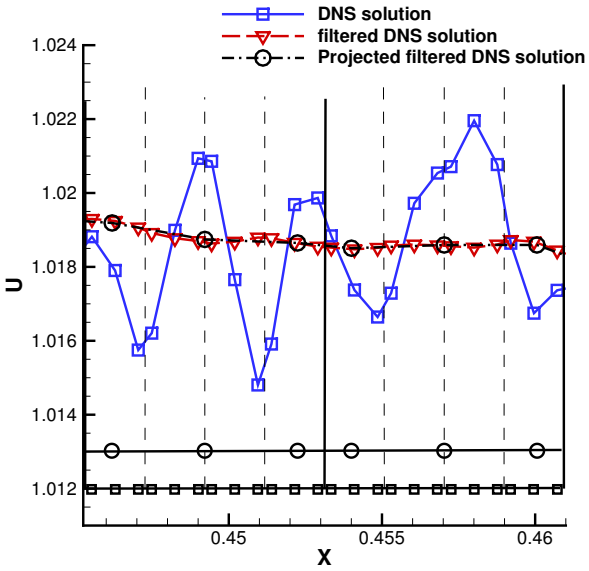


Figure 32: Comparison of various solutions(square: fine mesh solution points; circle: coarse mesh solution points)

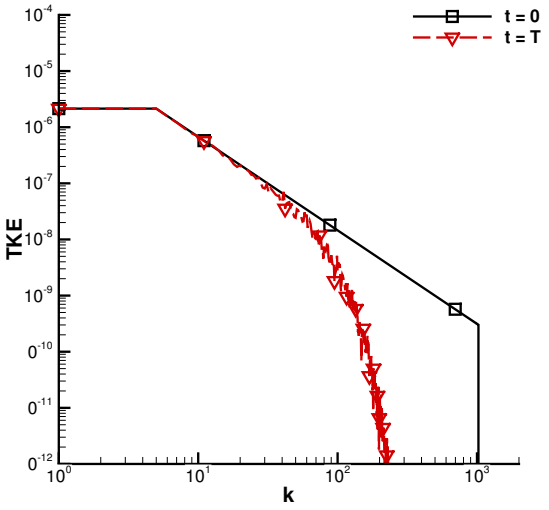


Figure 33: The energy spectrum at two different times

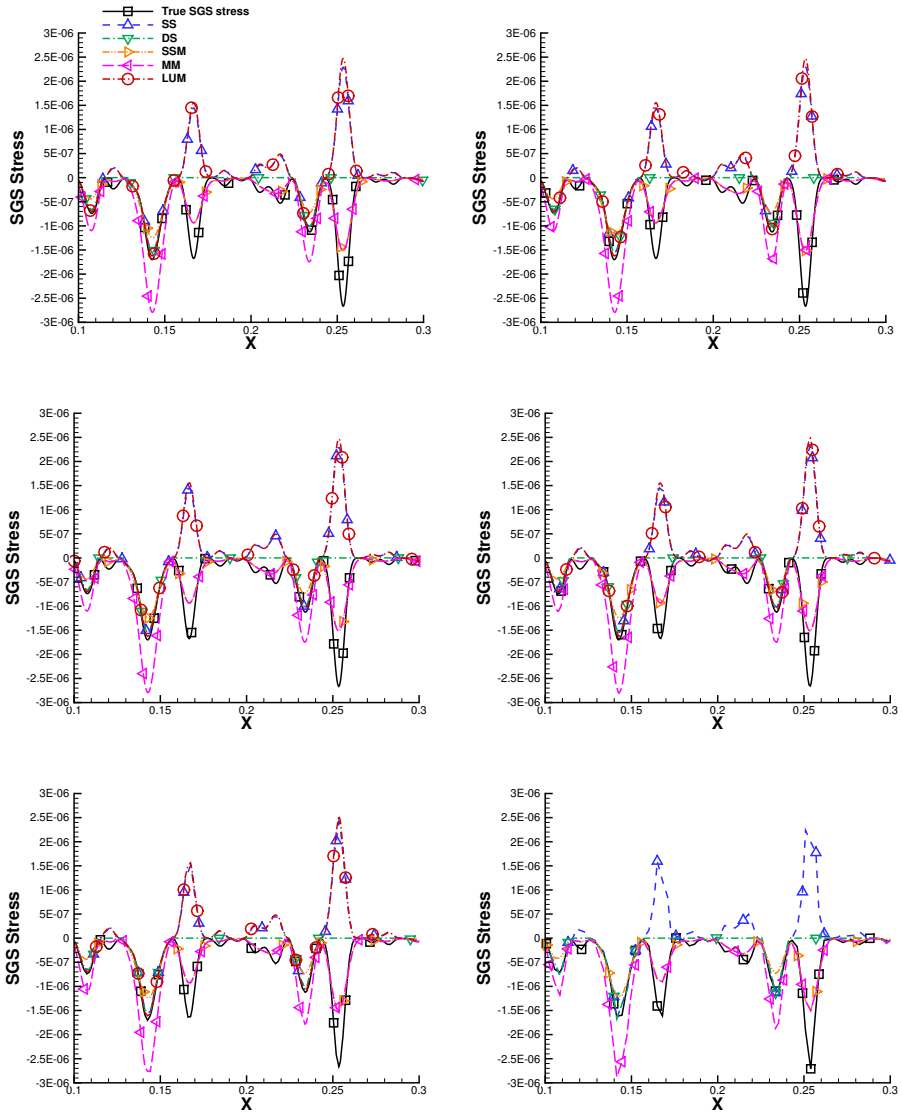


Figure 34: The SGS stress comparison in the a priori tests

Table 6: Correlation coefficients of a posteriori test

$\frac{\Delta x_{LES}}{\Delta x_{DNS}}$	SS	DS	SSM	Mix	LUM
1	-0.08	0.6	0.95	0.89	-0.06
2	-0.08	0.6	0.95	0.89	-0.07
4	-0.08	0.6	0.95	0.89	-0.07
8	-0.08	0.6	0.95	0.89	-0.07
16	-0.08	0.6	0.95	0.89	–
32	-0.10	0.57	0.92	0.85	–

The ratio between the cell size of LES and DNS is (a) 1, (b) 2, (c) 4, (d) 8, (e) 16, (f) 32. For the SS model, c_s is set to the default value of 0.2 for all of the comparisons. For ILES, the SGS stress is 0 everywhere. From Figure 34, we can make some general observation regardless of mesh resolutions:

- No models are able to predict the true stress in both amplitude and phase (peaks and valleys).
- Both the SSM and MM always correctly predict the phase of the true stress.
- SS correctly predicts the phase of the true stress about half the time, and 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 SGS prediction with enough grid resolution, but diverges for the coarsest mesh.

Obviously the good phase prediction capability of the MM is due to the dominant SSM term. Next we examine the correlation of the modeled stress with the true stress. Table 5 presents the correlation coefficients between the true SGS and the ones computed with the models. Clearly the SSM and the MM models perform the best. The mesh resolution Δx_{LES} does not have any significant influence on the model behavior, except for LUM. To further evaluate the behavior of these models in an actual computation, we perform a posteriori tests next.

3.5.2 A Posteriori Tests

In this test, the filtered 1D Burgers' equation is solved with different models on different meshes with a fixed filter size $\Delta = 32\Delta x_{DNS}$. The results at the same physical time $t = T$ are compared. Figure 35 shows the SGS stress computed using different models with various mesh resolution.

In Figure 35, we can see that the results are very similar to those in the a priori test. We can draw the same conclusions here. Table 6 shows the correlation coefficients for all the a posteriori tests. The SGS stresses computed by SSM and the MM always show high correlations with the true SGS stress. The DS comes the second. The SS and the LUM models yield very low correlation with the true SGS stress. LUM diverged for some cases and the correlation is not available. The LES mesh resolution does not have a significant influence on the model behavior, except for LUM.

Figure 36 shows the comparison of the solution, $\hat{u}$, for $\frac{\Delta x_{LES}}{\Delta x_{DNS}}$. The solution computed with the true SGS stress is right on top of the filtered DNS solution. The solutions computed with models and ILES all show differences with the filtered DNS solution. Table 7 shows the L2 norm error comparison. When the LES mesh is sufficiently fine, it is clear that the SSM and the MM produced the best solutions. This is because both

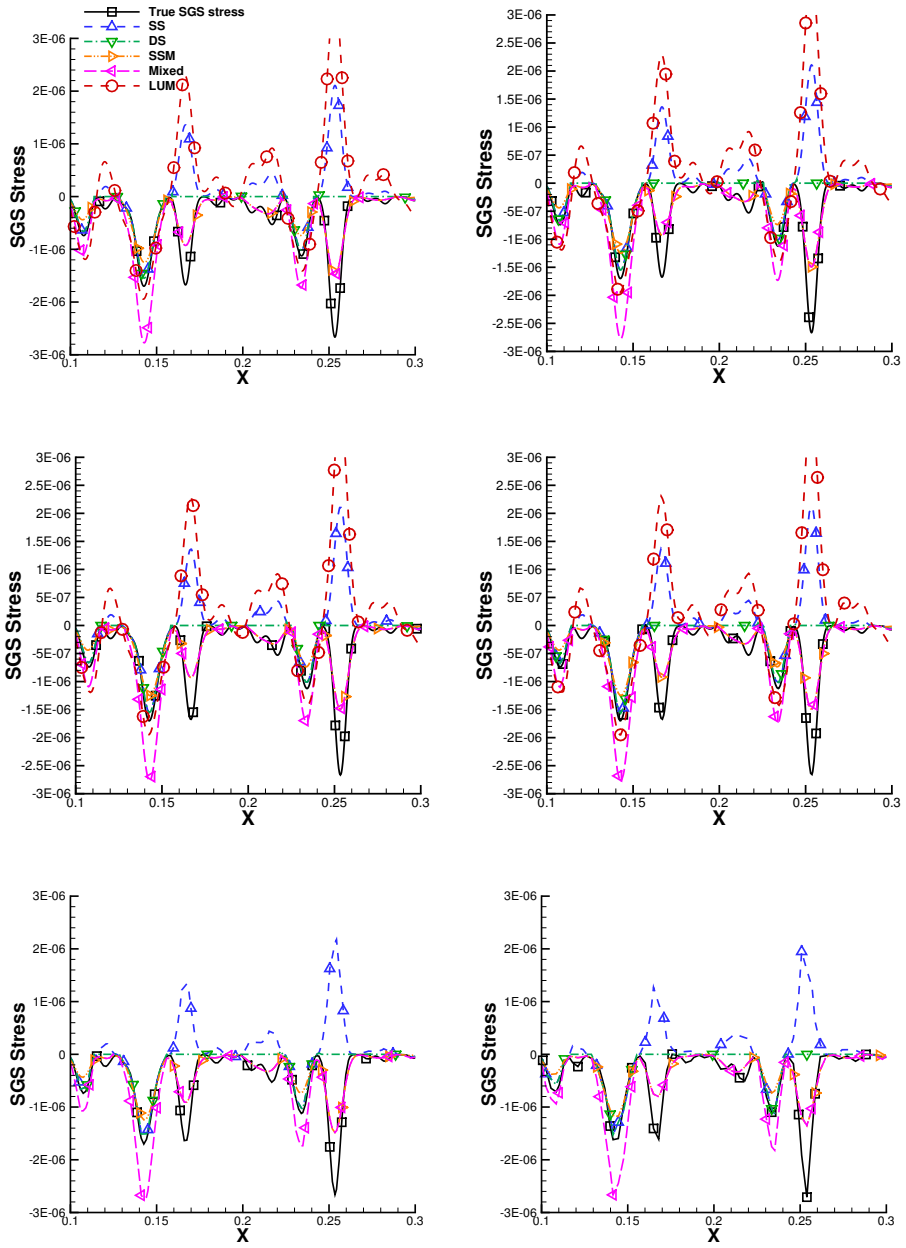


Figure 35: The SGS stress comparison in the a posteriori tests

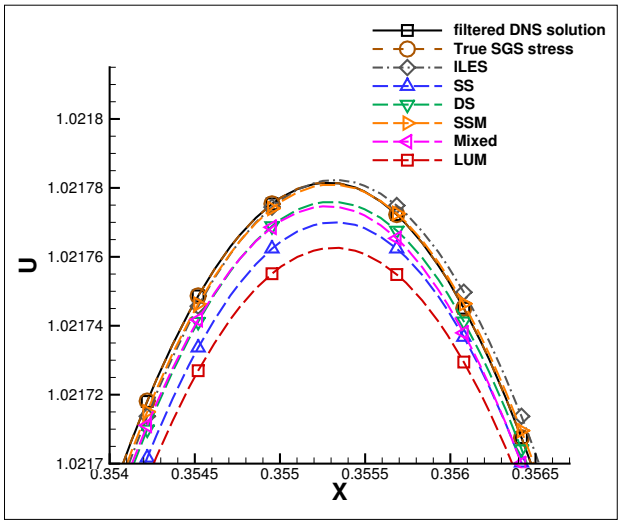


Figure 36: Solution Comparison

Table 7: L2 norm error of the solution

$\frac{\Delta x_{LES}}{\Delta x_{DNS}}$	True stress	ILES	SS	DS	SSM	Mix	LUM
1	2.03E-08	1.14E-05	1.57E-05	8.88E-06	7.02E-06	6.19E-06	1.80E-05
2	2.13E-08	1.14E-05	1.57E-05	8.88E-06	7.02E-06	6.18E-06	3.02E-05
4	4.60E-08	1.14E-05	1.57E-05	8.88E-06	7.01E-06	6.17E-06	5.62E-05
8	4.91E-07	1.14E-05	1.57E-05	8.97E-06	6.94E-06	6.25E-06	1.07E-04
16	1.09E-05	1.46E-05	2.01E-05	1.49E-05	1.18E-05	1.35E-05	–
32	1.38E-04	1.38E-04	1.41E-04	1.39E-04	1.37E-04	1.39E-04	–

models show the best correlation with the true SGS stress. When the LES mesh is coarse, the truncation error is dominant. The results with any model and with the true SGS stress are comparable. Clearly ILES is the best choice because it costs the least.

3.5.3 Sensitivity of the Models to the Mesh Resolution

Given the fixed filter width, we compare models' behavior on different mesh resolution. Figure 37 shows different modeled SGS comparison with respect to different. The $ratio = \frac{\Delta x_{LES}}{\Delta x_{DNS}}$. We can see that both in the a priori and the a posteriori tests, all the models shows no sensitivity to Δx_{LES} except for the LUM.

3.5.4 Effects of Truncation Error vs. SGS Model Error

n large eddy simulations, the numerical results depend on many factors, including the flow condition, the initial and boundary conditions, the numerical method, the computational mesh, the filter and the SGS model. Some of the factors are physical and others are numerical, and they intertwine together to produce the final solution. At the most fundamental level, the filter width Δ in a LES is perhaps the most critical parameter, and Pope discussed the importance of the filter width in SB [88]. The true LES solution can be obtained by filtering the DNS solution using this Δ. In reality, however, the filter width is often implicitly tied with the mesh size. In such cases, mesh refinement convergence studies become impossible to perform because the filter size is always a variable. One can only see convergence when the mesh size approaches that required of a DNS simulation. Generally speaking, we want to accurately predict the SGS stress using the numerical solution at the "resolved scale". The filtered solution $\hat{u}$ is always taken to be the solution at the "resolved scale". Let's consider the box filter here. When a solution is filtered with a width Δ , we often state that waves with shorter wavelengths than Δ are filtered out. In fact, waves of wavelengths of 2Δ and 4Δ are heavily damped out too. Based on our analysis, we can see that the amplitudes of 2Δ and 4Δ waves are reduced by 36% and 10% respectively Z.J. Wang [120]. If we accept 36% filtering error as acceptable, the "resolved scale" should be 2Δ instead of Δ. In addition, numerical methods also have limited resolution depending on the "points per wave" (PPW) or "degrees of freedom per wave" (DOFPW). Let's assume that for the present 3rd order FR/CPR scheme, 9 DOFPW is required to resolve a wave. In other words, 3 elements are needed for a wave since there are 3 DOFs in one element. A truly resolved scale must meet the accuracy requirement from both the filtering operator and the numerical scheme. In this particular case, the resolved scale is

$$S_R = max(3\Delta x_{LES}, 2\Delta), \tag{72}$$

In order to have the resolved scale determined by the given filter width, Δx_{LES} should satisfy the following requirement

$$\Delta x_{LES} \leq \frac{2\Delta}{3} \tag{73}$$

In the case of second-order finite volume methods, each element has 1 solution unknown. If one requires 20 PPW for accuracy, the resolved scale is then

$$S_R = max(20\Delta x_{LES}, 2\Delta). \tag{74}$$

If one chooses Δx_{LES} as the filter width, the resolved scale is 20 times larger than the filter width because of the accuracy requirement. In other words, the numerical truncation error is dominant in the LES results. This is the reason why we see smaller and smaller differences between the ILES and LES with SGS models with the Δx_{LES} increase.

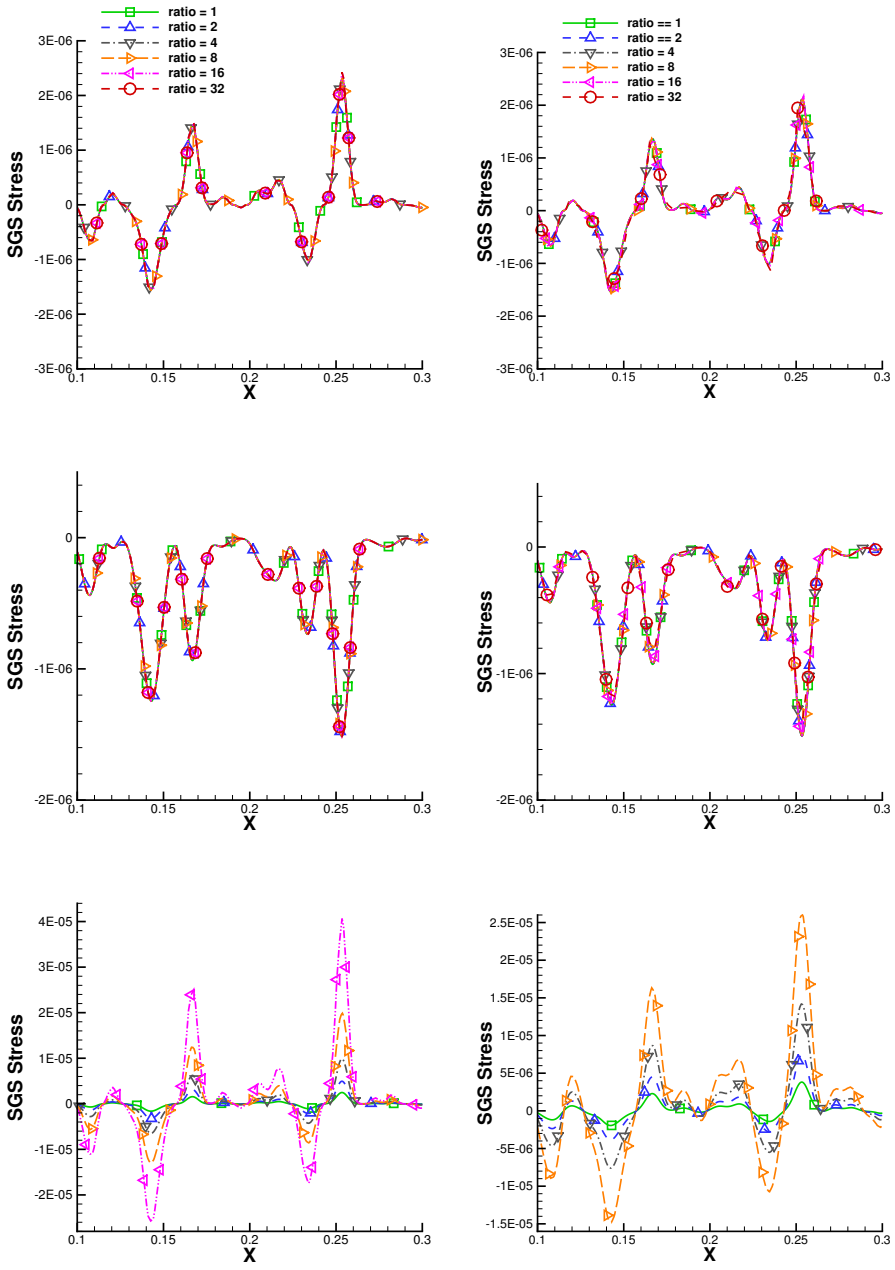


Figure 37: The modeled SGS with different mesh resolution

4 Investigation of Scale Similarity

From the comparison in the previous section, we see that the stress computed with the scale similarity model shows the highest correlation with the true SGS stress. And we discovered that they are related by a factor of 4, i.e,

$$\frac{L}{\tau_{true}} = 4. \tag{75}$$

When we first saw the relation 75, we suspected that this was a bug. After we tested different initial conditions, we were convinced that it was not a bug. The fact that relation 75 is true for arbitrarily-generated random initial conditions prompted us to look for a deeper reason resulting in the following analysis.

4.1 Analysis of Scale Similarity with a Single Fourier Mode

For the sake of simplicity without loss of generality, we consider periodic data $u(x)$ at a given time on domain $[-\pi, \pi]$. The solution can be decomposed into the following Fourier modes

$$u(x) = \sum_{n=0}^{inf} a_n e^{inx}, \tag{76}$$

where $i = \sqrt{-1}$, and n is the wave number. To illustrate the basic idea, we first consider a single Fourier mode, i.e., $u(x) = e^{inx}$ and the top hat filter. The filtered solution is then

$$\hat{u}(x) = \frac{1}{\Delta}\int_{x-\frac{\Delta}{2}}^{x+\frac{\Delta}{2}} e^{in\xi} d\xi = sinc(\frac{n\Delta}{2}) \cdot e^{inx}, \tag{77}$$

where $sinc(\frac{n\Delta}{2}) = \frac{sin\frac{n\Delta}{2}}{\frac{n\Delta}{2}}$. Obviously the filter only changes the magnitude of the solution, but not the phase. In addition, we have

$$\hat{uu}(x) = \frac{1}{\Delta}\int_{x-\frac{\Delta}{2}}^{x+\frac{\Delta}{2}} e^{i2n\xi} d\xi = sinc(n\Delta) \cdot e^{i2nx}. \tag{78}$$

The SGS stress is then

$$\begin{aligned} \tau = \hat{uu} - \hat{u}\hat{u} &= sinc(n\Delta) \cdot e^{i2nx} - sinc^2(\frac{n\Delta}{2}) \cdot e^{i2nx} \\ &= [sinc(n\Delta) - sinc^2(\frac{n\Delta}{2})]e^{i2nx}. \end{aligned} \tag{79}$$

Next we apply a second filter with a width of $\Delta_2 = \gamma\Delta$ to the resolved variable to obtain

$$\tilde{\hat{u}}(x) = \frac{1}{\gamma\Delta}\int_{x-\frac{\gamma\Delta}{2}}^{x+\frac{\gamma\Delta}{2}} \hat{u}(\xi) d\xi = sinc(\frac{n\Delta}{2}) sinc(\frac{\gamma n\Delta}{2}) e^{inx}, \tag{80}$$

and

$$\tilde{\hat{u}\hat{u}}(x) = \frac{1}{\gamma\Delta}\int_{x-\frac{\gamma\Delta}{2}}^{x+\frac{\gamma\Delta}{2}} \hat{u}(\xi)\hat{u}(\xi) d\xi = sinc^2(\frac{n\Delta}{2}) sinc(\frac{\gamma n\Delta}{2}) e^{i2nx}. \tag{81}$$

The SGS stress of the resolved scale is then

$$L = \tilde{\hat{u}\hat{u}} - \tilde{\hat{u}}\tilde{\hat{u}} = sinc^2(\frac{n\Delta}{2})(sinc(\gamma n\Delta) - sinc^2(\frac{\gamma n\Delta}{2}))e^{i2nx}. \tag{82}$$

From Eqn.79 and Eqn. 82, we obtain

$$\frac{L}{\tau} = \frac{sinc^2(\frac{n\Delta}{2})(sinc(\gamma n\Delta) - sinc^2(\frac{\gamma n\Delta}{2}))}{sinc(n\Delta) - sinc^2(\frac{n\Delta}{2})}. \tag{83}$$

In the limit of small $n\Delta$, we have

$$sinc(n\Delta) = 1 - \frac{(n\Delta)^2}{6} + O(n\Delta)^4. \tag{84}$$

Therefore, we obtain

$$\frac{L}{\tau} = \frac{[1 - \frac{(n\Delta)^2}{12} + O(n\Delta)^4][-\frac{(\gamma n\Delta)^2}{6} + \frac{(\gamma n\Delta)^2}{12} + O(n\Delta)^4]}{-\frac{(n\Delta)^2}{6} + \frac{(n\Delta)^2}{12} + O(n\Delta)^4} = \gamma^2 + O(n\Delta)^2. \tag{85}$$

Note that the error term is quadratic. In the special case of $\gamma = 2$, $L = 4\tau$. As it turns out this result is also true for the Gaussian filter. The filtered solution with a Gaussian filter is

$$\hat{u}(x) = \int_{-\inf}^{\inf} \sqrt{(\frac{6}{\pi\Delta^2})} e^{-\frac{6(x-\xi)^2}{\Delta^2}} ein\xi d\xi = sqrt(\frac{6}{\pi\Delta^2}) \int_{-\inf}^{\inf} e^{-\frac{6(x-\xi)^2}{\Delta^2}} ein\xi d\xi. \tag{86}$$

Set $X = \frac{\sqrt{6}(x-\xi)}{\Delta}$, so that $\xi = x - \frac{\Delta}{\sqrt{6}}X$, $d\xi = -\frac{\Delta}{\sqrt{6}}dX$. Thus, we have

$$\begin{aligned} \hat{u}(x) &= \sqrt{\frac{6}{\pi\Delta^2}} \cdot \frac{\Delta}{\sqrt{6}} \int_{-\inf}^{\inf} e^{-X^2 + in(x - \frac{\Delta}{\sqrt{6}}X)} dX \\ &= \sqrt{\frac{1}{pi}} e^{inx} \int_{-inf}^{inf} e^{-X^2 - in\frac{\Delta}{\sqrt{6}}X} dX = e^{inx} e^{-\frac{(n\Delta 2}{24}}. \end{aligned} \tag{87}$$

Similarly we can derive the following result

$$\begin{aligned} \hat{uu}(x) &= \int_{-inf}^{inf} \sqrt{\frac{6}{\pi\Delta^2}} e^{-\frac{6(x-\xi)^2}{\Delta^2}} e^{i2nx} d\xi \\ &= \sqrt{\frac{1}{pi}} e^{i2nx} \int_{-inf}^{inf} e^{-X^2 - i2n\frac{\Delta}{\sqrt{6}}X} dX = e^{i2nx} e^{-\frac{(n\Delta)^2}{6}}. \end{aligned} \tag{88}$$

The SGS stress is them

$$\tau = \hat{uu} - \hat{u}\hat{u} = e^{i2nx} e^{-\frac{(n\Delta)^2}{6}} - e^{i2nx} e^{-\frac{(n\Delta)^2}{12}} = e^{i2nx} e^{-\frac{(n\Delta)^2}{6}} [1 - e^{-\frac{(n\Delta)^2}{12}}]. \tag{89}$$

Again we apply a second filter with a width of $\Delta_2 = \gamma\Delta$ to the resolved field to obtain

$$\tilde{\hat{u}}(x) = e^{-\frac{(n\Delta)^2}{24}} \tilde{u}(x) = e^{-\frac{(n\Delta)^2}{24}} e^{-\frac{(\gamma n\Delta)^2}{24}} e^{inx}. \tag{90}$$

and

$$\tilde{\hat{uu}}(x) = e^{-\frac{(n\Delta)^2}{12}} \tilde{uu}(x) = e^{-\frac{(n\Delta)^2}{12}} e^{-\frac{(\gamma n\Delta)^2}{6}} e^{i2nx}. \tag{91}$$

The SGS stress of the resolved scale is

$$L = \tilde{\hat{uu}} - \tilde{\hat{u}}\tilde{\hat{u}} = e^{i2nx} e^{-\frac{(1+2\gamma^2)(n\Delta)^2}{12}} [1 - e^{\frac{(\gamma n\Delta)^2}{12}}]. \tag{92}$$

From 89 and 92, we obtain

$$\frac{L}{\tau} = e^{\frac{(1-2\gamma^2)(n\Delta)^2}{12}} \frac{1 - e^{\frac{(\gamma n\Delta)^2}{12}}}{1 - e^{\frac{(n\Delta)^2}{12}}}. \tag{93}$$

In the limit of small $n\Delta$, we have

$$\frac{L}{\tau} = (1 + O(n\Delta)^2) \frac{1 - 1 - \frac{(\gamma n\Delta)^2}{12} + O(n\Delta)^4}{1 - 1 - \frac{(n\Delta)^2}{12} + O(n\Delta)^4} = \gamma^2 + O(n\Delta)^2. \tag{94}$$

4.2 Analysis of Scale Similarity with All Fourier Modes

Next we consider a solution with all the Fourier modes, i.e.,

$$u(x) = \sum_{n=0}^{inf} a_n e^{inx}. \tag{95}$$

With the top hat filter, we obtain the following filtered solution

$$\hat{u}(x) = \sum_{n=0}^{inf} a_n sinc(\frac{n\Delta}{2}) \cdot e^{inx}. \tag{96}$$

In addition, we have

$$\hat{u}\hat{u}(x) = \sum_{n=0}^{inf} \sum_{m=0}^{inf} a_n a_m sinc \frac{(n+m)\Delta}{2} e^{i(n+m)x}. \tag{97}$$

The SGS stress is then

$$\tau = \hat{u}\hat{u} - \hat{u}\hat{u} = \sum_{n=0}^{inf} \sum_{m=0}^{inf} a_n a_m [sinc \frac{(n+m)\Delta}{2} - sinc(\frac{n\Delta}{2}) sinc(\frac{m\Delta}{2})] e^{i(n+m)x}. \tag{98}$$

Next we apply a second filter with a width of $\gamma\Delta$ to the resolved variable

$$\tilde{\hat{u}} = \sum_{n=0}^{inf} a_n sinc(\frac{n\Delta}{2}) sinc(\frac{\gamma n\Delta}{2}) e^{inx}, \tag{99}$$

and

$$\tilde{\hat{u}\hat{u}} = \sum_{n=0}^{inf} \sum_{m=0}^{inf} a_n a_m sinc(\frac{n\Delta}{2}) sinc(\frac{m\Delta}{2}) sinc(\frac{\gamma(n+m)\Delta}{2}) e^{i(n+m)x}. \tag{100}$$

The SGS stress of the resolved scale is then

$$\begin{aligned} L &= \tilde{\hat{u}\hat{u}} - \tilde{\hat{u}}\tilde{\hat{u}} \\ &= \sum_{n=0}^{inf} \sum_{m=0}^{inf} a_n a_m sinc(\frac{n\Delta}{2}) sinc(\frac{m\Delta}{2}) (sinc(\frac{\gamma(n+m)\Delta}{2}) \\ &- sinc(\frac{\gamma n\Delta}{2}) sinc(\frac{\gamma m\Delta}{2})) e^{i(n+m)x}. \end{aligned} \tag{101}$$

Now let's consider each term in Eqn. 98 and 101. It is obvious that

$$\frac{L_{nm}}{\tau_{nm}} = \frac{sinc(\frac{n\Delta}{2})sinc(\frac{m\Delta}{2})(sinc(\frac{\gamma(n+m)\Delta}{2}) - sinc(\frac{\gamma n\Delta}{2})sinc(\frac{\gamma m\Delta}{2}))}{sinc\frac{(n+m)\Delta}{2} - sinc(\frac{n\Delta}{2})sinc(\frac{m\Delta}{2})}$$
$$= \gamma^2 + O[(n+m)\Delta]^2. \quad (102)$$

Therefore, we have

$$\frac{L}{\tau} = \gamma^2 + O[(n+m)\Delta]^2. \quad (103)$$

n the same limit. The analysis with the Gaussian filter is similar and is not repeated here.

4.3 Analysis of Scale Similarity in 2D

In two dimensions, we only perform a single mode analysis with the top hat filter. Consider the following two dimensional velocity field

$$u(x,y) = e^{inx}e^{imy}, v(x,y) = e^{ipx}e^{iqy}. \quad (104)$$

The filtered solution is then

$$\hat{u}(x,y) = \frac{1}{\Delta^2}\int_{x-\frac{\Delta}{2}}^{x+\frac{\Delta}{2}}\int_{y-\frac{\Delta}{2}}^{y+\frac{\Delta}{2}} e^{in\xi}e^{im\eta}d\eta d\xi = e^{inx}e^{imy}\cdot sinc(\frac{n\Delta}{2})sinc(\frac{m\Delta}{2}), \quad (105)$$

$$\hat{v}(x,y) = \frac{1}{\Delta^2}\int_{x-\frac{\Delta}{2}}^{x+\frac{\Delta}{2}}\int_{y-\frac{\Delta}{2}}^{y+\frac{\Delta}{2}} e^{ip\xi}e^{iq\eta}d\eta d\xi = e^{ipx}e^{iqy}\cdot sinc(\frac{p\Delta}{2})sinc(\frac{q\Delta}{2}). \quad (106)$$

In addition, we have

$$\hat{uv}(x,y) = \frac{1}{\Delta^2}\int_{x-\frac{\Delta}{2}}^{x+\frac{\Delta}{2}}\int_{y-\frac{\Delta}{2}}^{y+\frac{\Delta}{2}} e^{i(n+p)\xi}e^{i(m+q)\eta}d\eta d\xi$$
$$= e^{i(n+p)x}e^{i(m+q)y}\cdot sinc(\frac{(n+p)\Delta}{2})sinc(\frac{(m+q)\Delta}{2}). \quad (107)$$

The SGS stress is then

$$\tau = \hat{uv} - \hat{u}\hat{v}$$
$$= e^{i(n+p)x}e^{i(m+q)y}[sinc(\frac{(n+p)\Delta}{2})sinc(\frac{(m+q)\Delta}{2})$$
$$- sinc(\frac{n\Delta}{2})sinc(\frac{m\Delta}{2})sinc(\frac{p\Delta}{2})sinc(\frac{q\Delta}{2})]. \quad (108)$$

Applying a second filter with a width $\gamma\Delta$ to the resolved variable, we obtain

$$\tilde{\hat{u}} = e^{inx}e^{imy}\cdot sinc(\frac{n\Delta}{2})sinc(\frac{m\Delta}{2})sinc(\frac{\gamma n\Delta}{2})sinc(\frac{\gamma m\Delta}{2}), \quad (109)$$

and

$$\tilde{\hat{v}} = e^{ipx}e^{iqy}\cdot sinc(\frac{p\Delta}{2})sinc(\frac{q\Delta}{2})sinc(\frac{\gamma p\Delta}{2})sinc(\frac{\gamma q\Delta}{2}), \quad (110)$$

Denote $\alpha = sinc(\frac{n\Delta}{2})sinc(\frac{m\Delta}{2})sinc(\frac{p\Delta}{2})sinc(\frac{q\Delta}{2})$. Then we have

$$\tilde{\hat{u}}\hat{v} = \alpha\tilde{uv}, \tilde{\hat{u}}\tilde{\hat{v}} = \alpha\tilde{u}\tilde{v}. \quad (111)$$

The SGS stress of the resolved scale is then

$$\begin{aligned} L &= \hat{\tilde{u}}\hat{\tilde{v}} - \hat{\tilde{u}}\hat{\tilde{v}} = \alpha(\tilde{uv} - \tilde{u}\tilde{v}) \\ &= \alpha e^{i(n+p)x} e^{i(m+q)y} [sinc(\frac{\gamma(n+p)\Delta}{2}) sinc(\frac{\gamma(m+q)\Delta}{2}) \\ &- sinc(\frac{\gamma n\Delta}{2}) sinc(\frac{\gamma m\Delta}{2}) sinc(\frac{\gamma p\Delta}{2}) sinc(\frac{\gamma q\Delta}{2})]. \end{aligned} \tag{112}$$

From 108 and 112, we obtain

$$\frac{L}{\tau} = \alpha \frac{sinc(\frac{\gamma(n+p)\Delta}{2}) sinc(\frac{\gamma(m+q)\Delta}{2}) - sinc(\frac{\gamma n\Delta}{2}) sinc(\frac{\gamma m\Delta}{2}) sinc(\frac{\gamma p\Delta}{2}) sinc(\frac{\gamma q\Delta}{2})}{sinc(\frac{(n+p)\Delta}{2}) sinc(\frac{(m+q)\Delta}{2}) - sinc(\frac{n\Delta}{2}) sinc(\frac{m\Delta}{2}) sinc(\frac{p\Delta}{2}) sinc(\frac{q\Delta}{2})}. \tag{113}$$

In the limit of small $(n+m+p+q)\Delta$, we have $alpha \approx 1$, and

$$sinc(A+B) \cdot sinc(C+D) - sincA \cdot sincB \cdot sincC \cdot sincD = -\frac{AB+CD}{3} + HOT. \tag{114}$$

Finally, we derive the following result using Eqn. 114

$$\frac{L}{\tau} \approx \gamma^2. \tag{115}$$

4.4 Implications for Large Eddy Simulation

The present analysis shows that perfect scale similarity exists for arbitrary (periodic) data including turbulence under the assumption that the spectrum contains relatively low frequency contents with respect to the filter width, regardless of amplitude and phase angle of each mode. Obviously for an arbitrary spectrum including both high and low frequency contents, the present analysis is not valid. This is easily seen in Figure 38, which displays the modeled and true SGS stress based on the full spectrum shown in Figure 30, using the same filter width which is $16\Delta_{DNS}$. The correlation between the modeled and true stresses is quite low.

Next let's examine whether Eqn. 115 is true in an actual LES. The promise of the SSM is that the SGS stress is highly correlated with the stress computed based on the resolved scale, taken to be $\hat{u}$. Take the top hat filter for example. Modes of smaller wavelength than D corresponding to the cutoff wavenumber k_Δ are filtered out. In LES, it is believed that the SGS stress from higher modes close to the cutoff wave number k_Δ plays an important role. In the next test, we therefore include modes between k_Δ and $2k_\Delta$ using a filter width D/2 to filter the spectrum shown in Figure 30. The filtered solution is then treated as DNS data, which is used to obtain the true stress. This true stress is also compared with the stresses computed using the SSM based on the resolved scale, i.e., $\hat{u}$. Two test filter widths are used corresponding to $\gamma = 1$ and 2. The results are displayed in Figure 38. Note that there is a reasonably high level of correlation between the stresses.

The ratio between the true and modeled stresses are computed using simple averages

$$\frac{L}{\tau} = \frac{\langle L \rangle}{\langle \tau \rangle}. \tag{116}$$

The correlation coefficients and the average stress ratios from 10 realizations are summarized in Table 5. The table confirms that the true stress shows a quite high correlation with the modeled stress, with an average correlation coefficients of 0.88 and 0.69 for $\gamma = 1$ and 2, respectively. In addition, $\gamma = 1$ demonstrates consistently higher correlation coefficients than $\gamma = 1$. This may indicate that one should use the same filter width for the

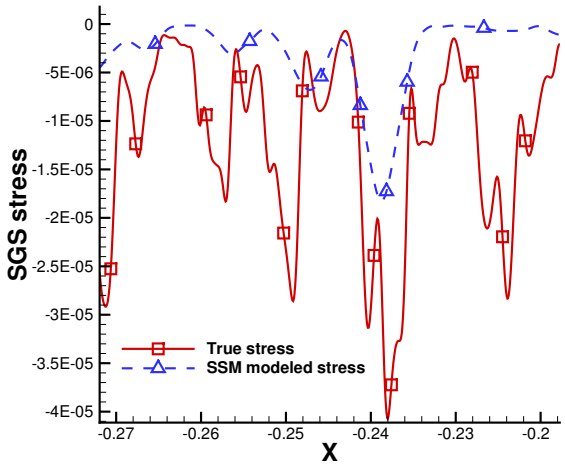


Figure 38: The true stress and the modeled stress for the full spectrum

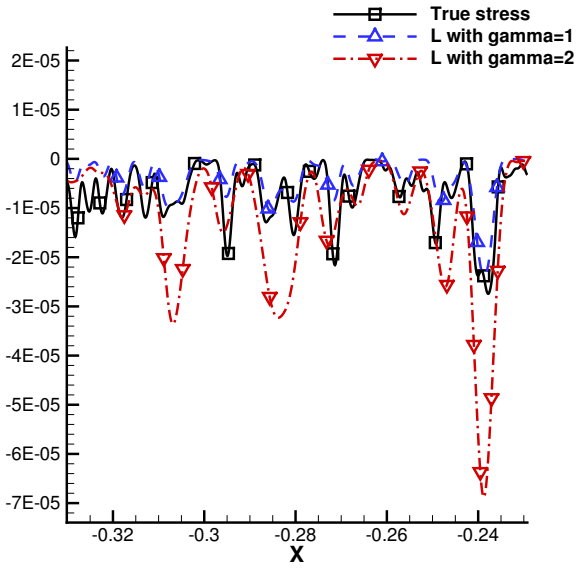


Figure 39: Comparison between the true stress computed with the SGS between k_Δ and $2k_\Delta$ and the modeled stresses computed using $\hat{u}$

second filter in an SSM implementation. Furthermore, the ratio of the averaged stresses remains a constant with different realizations, indicating that this ratio is only dependent on the spectrum. However, the ratio is much smaller than γ^2. This result appears to agree well with others in the literature AW [4] Liu S [73].

4.5 Investigation of Stability of Scale Similarity Model

In Bardina's original paper, the SSM was found unstable in some simulations when used with a central finite difference scheme. To remedy the instability, a MM with the DS model was developed to stabilize the simulations. In this section, we attempt to show that the extra dissipation added by the MM is not necessary for the FR/CPR method which has embedded numerical dissipation to automatically damp high frequency modes. We first demonstrate that there is indeed a pile-up of high frequency modes with a central difference scheme in solving nonlinear equations such as the Burgers' equation, while there is no such pile-up with a dissipative high-order FR/CPR scheme. For this purpose, we conduct a numerical study with the initial condition of a single Fourier mode,

$$u(x) = 2(E_0(1))^{1/2} sin(\pi x) + 1. \tag{117}$$

where $= 2(E_0(1))^{1/2} = 0.012$. The 1D inviscid Burgers' equation is employed to mimic very high Reynolds number problems. We run the simulation until $t = 26$ when it is right before a shock wave develops. First, the upwind flux and the central flux are employed in the 3rd order FR/CPR scheme to compare their behaviors. Figure 40 shows the energy spectrum at $t = 26$ with different mesh resolutions. On the finest mesh, both the central and upwind schemes produced a converged solution within the visible energy spectrum in the figure. On the two coarser meshes, we can see clearly that energy is piling up at high frequencies on those meshes for the simulation with the central flux. But the upwind flux is able to smoothly damp out the high frequency modes so that they are never accumulated to cause stability problems. Next we test the influence of the SGS models on the energy spectrum. Figure 41 shows the spectrum comparison of the simulations with and without the SSM and MM. The filter width equals to the cell size. We can see that with the central flux, the SSM neither damps out all the energy accumulated at high frequencies nor accumulates more energy there. Thus the extra dissipation, i.e. the DS, is necessary to stabilize the simulation. It is worth noting that the extra dissipation, in the MM, also damps out the energy at some lower frequencies, which does harm to the resolved large scales.

We also verify that a central difference finite difference scheme behaves similarly with the CPR scheme with a central flux. Figure 42 indeed shows that the 4th order central finite difference method has a similar performance to the 3rd order FR/CPR scheme with the central flux. This means that for schemes that are not dissipative, more dissipation may be necessary to stabilize the turbulent flow simulations with the SSM model. But for the dissipative ones, such as the FR/CPR method with an upwind flux, no extra dissipation is needed.

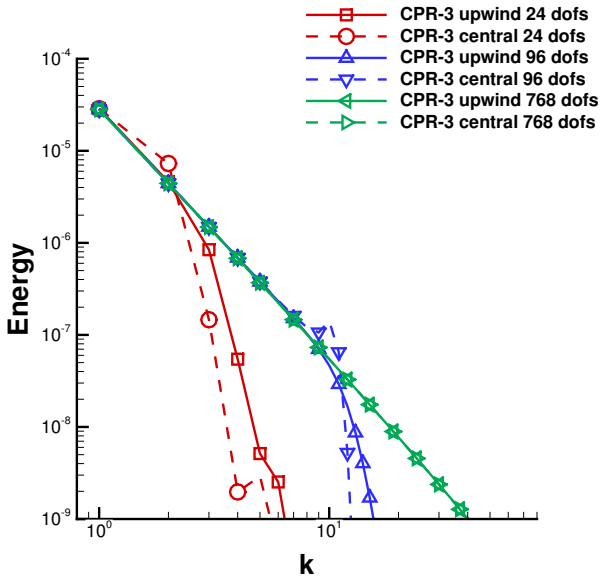


Figure 40: The spectrum of the upwind flux and the central flux with 3^{rd} order FR/CPR scheme at $t = 26$

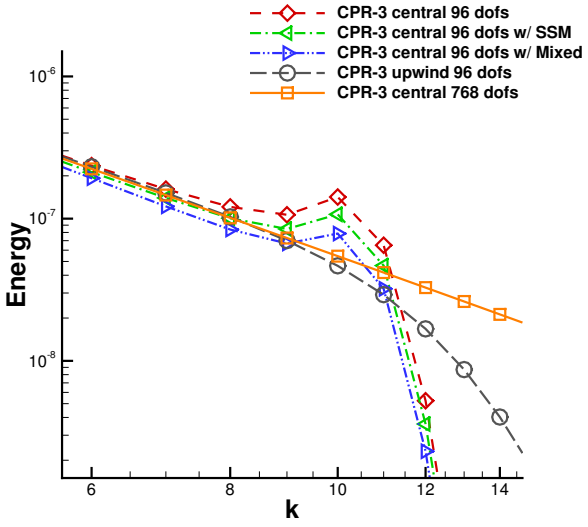


Figure 41: The spectrum of different models with 3^{rd} order FR/CPR scheme at $t = 26$

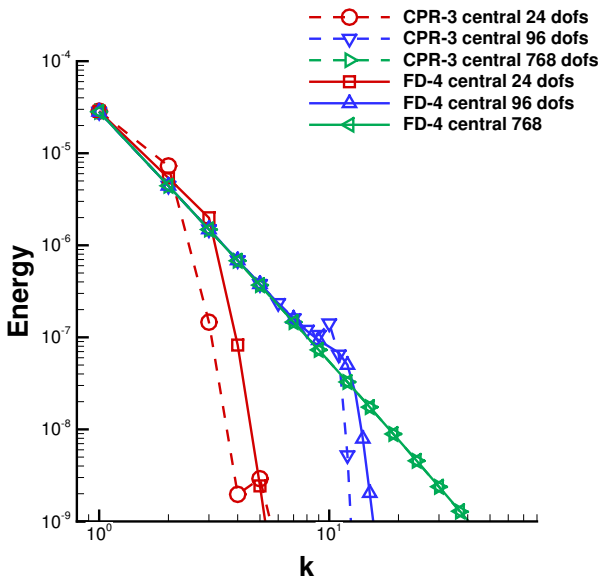


Figure 42: The spectrum of the 3^{rd} order FR/CPR scheme and the 4^{th} order finite difference scheme with the central flux at $t = 26$

5 Results and Discussion on MeshCurve: An Automated Low-Order to High-Order Mesh Converter

5.1 Low versus High-Order Meshes: An Illustrative Example

Figure 43 illustrates the difference between a low-order mesh and a high-order mesh with a side-by-side comparison of two circular meshes. The key difference is the number of nodes, with the low-order mesh, shown on the left, having fewer than the high-order mesh, shown on the right. The extra nodes enhance simulation accuracy by serving as secondary interpolation points. Also, they enhance geometric accuracy by tracing the arc of curved edges.

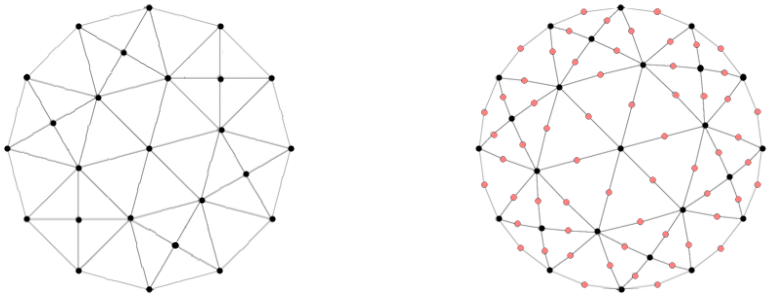
The goal of meshCurve is to transform a mesh similar to the one on the left of figure 43 into a mesh similar to the one on the right, but in 3D. Figure 44 is a before-and-after image from meshCurve. The additional nodes are not visible in the high-order mesh, but the surface curvature is clearly discernible.

5.2 The Design of meshCurve

5.2.1 Feature Criteria

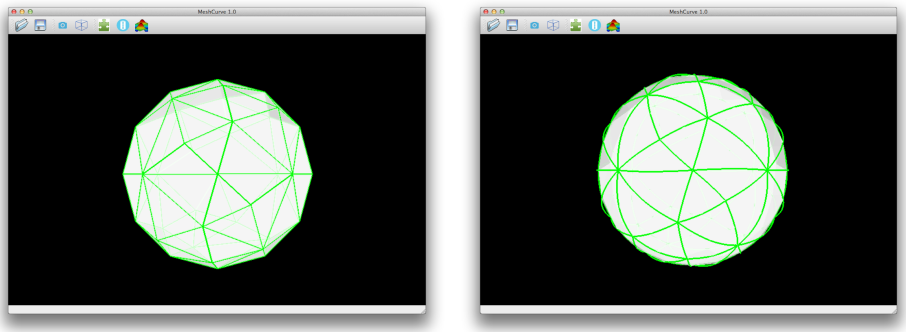
From our requirements, we assembled the following list of features for meshCurve.

- Full support for 3D unstructured CGNS meshes—any cell type in any combination.
- Support for multi-zone, multi-patch CGNS meshes, including the ability to selectively reconstruct interior patches without affecting far-field boundaries.
- Mesh processing without CAD geometry files.
- Mesh processing in the face of complex edge arrangements and sharp corners.



(a) a low-order mesh (b) a high-order mesh

Figure 43: A low-order mesh contrasted with a high-order mesh. The key difference is that the high-order mesh has more nodes than the low-order mesh. The extra nodes enhance the geometric accuracy by tracing the path of curved edges.



(a) a low-order mesh (b) a high-order mesh

Figure 44: Before-and-after comparison of a mesh upgraded with meshCurve. The additional nodes in the high-order mesh are not visible but the surface curvature is visible.

- Easy-to-use graphical interface, requiring minimal effort to learn and navigate.
- Interactive 3D graphics to show the mesh and to provide visual feedback after the upgrade.
- Cross-platform support—Linux, Windows, and Mac.
- Solid code base with minimal bugs, designed for maintainability and future upgrades.
- Reasonably low memory footprint and fast operation on a desktop computer.
- Minimal reliance on outside software libraries to increase control of the code and simplify design.

5.2.2 Design Decisions

To satisfy requirements, we made the following decisions.

- Programming would be in C++ due to the speed and flexibility of the language. We opted for C++11, the most recent version of the language, because of newly introduced optimizations and platform-independent multithreading. With multithreading, it would be possible to utilize the multiple cores of modern CPUs.
- To support our desired cross-platform graphical-user-interface (GUI), we selected the popular Qt application framework.[1] The Qt framework is a platform-independent windowing library which includes multiple user interface widgets and support libraries. Applications built with Qt adapt to the target operating system. Consequently, meshCurve appears to be an X11 application when run on Linux, a Windows application when run on Microsoft's OS, and a Macintosh application when run on OS X. Figure 45 shows an annotated view of the meshCurve user interface as it appears on Microsoft Windows. The figures elsewhere show the interface as it appears on Linux (Ubuntu) and Mac OS X.
- For interactive 3D visualization, we chose the Visualization Toolkit (VTK), an open source, cross-platform 3D visualization system created by Kitware.[89] VTK wraps the underlying graphics APIs, simplifying cross-platform development while providing a capable visualization environment.
- For fast linear algebra, we decided to use the high-speed Armadillo C++ library.[87] Armadillo interfaces platform-specific LAPACK and BLAS libraries to provide vectorized mathematics and fast matrix routines. Its syntax is based on Matlab, making mathematical programming both simple and efficient.
- We constructed our own mesh database. Building our own database provided freedom to optimize for specific needs, while also granting flexibility to design for future extensibility.

5.3 User Workflow

Figure 46 diagrams the mesh-processing workflow from the user's perspective. The first step of the workflow is the loading of a mesh file, either CGNS or Gmsh format. The loaded mesh displays as a 3D view—rotatable, zoomable, and translatable via the mouse. The view can optionally be set to wireframe or opaque solid, and separate patches within

[1] http://www.qt.io

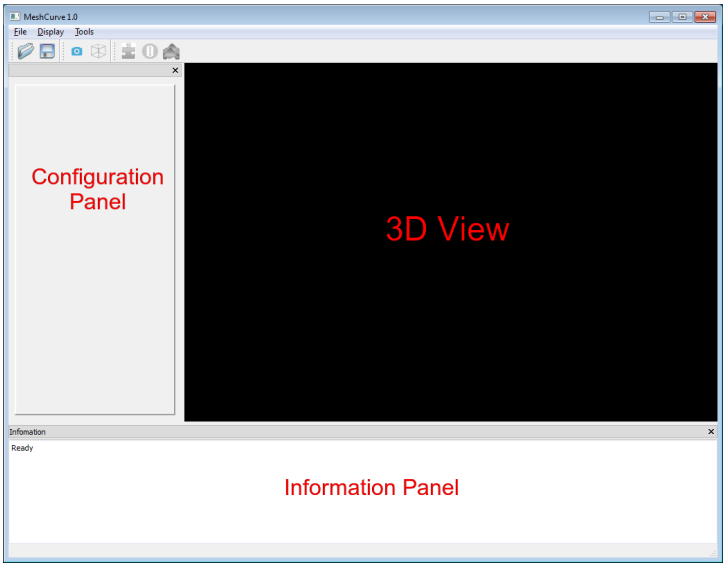


Figure 45: The meshCurve graphical interface, as it appears on Microsoft Windows 7. By default, three panels are visible: 1) 3D viewing, 2) configuration setting, and 3) information display. The information panel and the configuration panel can be hidden to increase the size of the 3D viewport. A row of toolbar buttons and menus along the top provide options to the user.

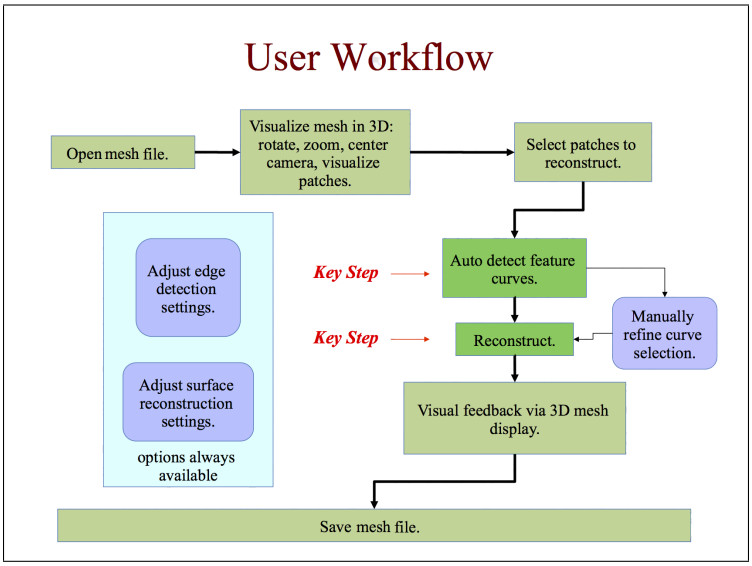


Figure 46: The mesh upgrade workflow from the user's perspective.

the mesh can be drawn individually. Also, surface nodes can be displayed on the mesh. Viewing the nodes is especially useful as a final verification after the upgrade has completed.

The next step is to select patches to process. During the upgrade, the selected patches will be given curvature. For efficiency, unselected patches will not be given curvature although they will be given high-order nodes. Figure 47 illustrates patch selection, showing a fighter jet mesh with wing and tail patches selected. Green highlight marks the selection.

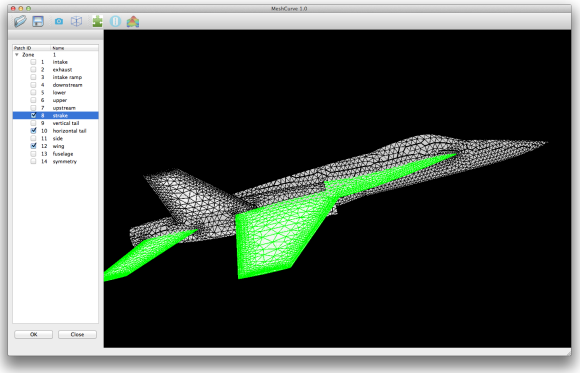


Figure 47: Screenshot of the program meshCurve, illustrating patch selection. The green portions of the mesh are the patches that have been selected for processing.

Next comes the identification of feature curves, such as the rim of an airplane canopy or the sharp edge of a box. Feature curves define the geometry, so it is important that meshCurve know the location of the curves within the mesh. Every feature curve may be manually flagged edge-by-edge, via the mouse. Alternatively, the curves can be flagged automatically by way of meshCurve's own feature-curve detection engine. Employing discontinuity detection algorithms developed by Jiao and Bayyana,[59] the feature-curve detection engine can identify curves even when they cross nearly flat portions of the mesh. After auto-detection, the mouse can be used to fine tune the result.

Once feature curves have been flagged, upgrading can commence. During the upgrade, surface geometry is approximated by a continuous 2D polynomial, which is used to position high-order nodes on the surface, giving the surface curvature. If the curvature is too great, the cells abutting the boundary will intersect. To prevent intersections, meshCurve can bend the edges of interior cells to coincide with the boundary. This bending action is applied by default but may be turned off by a checkbox in the configuration panel for surface reconstruction, as a way to speed calculation. Not all meshes need to have their interior deformed in this way.

At the completion of the upgrade, the 3D view updates to show the new high-order mesh. The user can then save the mesh as a new CGNS file.

5.4 Case Studies

Three case studies are presented to illustrate the capability of meshCurve. For each case study, a mesh is upgraded from linear to 2nd order (i.e. low to high-order). A picture of the mesh is presented, rendered with meshCurve. A table follows, summarizing key information to demonstrate performance. The table lists:

1. the total number of cells
2. the total number of nodes
3. the fraction of patches given curvature
4. the time it took to process the mesh

5. whether or not interior deformation was activated

6. minimum and maximum Jacobians of the original low-order mesh

7. minimum and maximum Jacobians of the new high-order mesh

Processing was done on a Macintosh computer with a 2.7 GHz Intel i5 processor (quad core).

The Jacobians are a check for the presence of intersecting and/or inverted cells, both of which could be problematic for simulations. Negative Jacobians indicate trouble. Only one of the case studies encountered a negative Jacobian, and this occurrence was remedied by interior deformation. For the other case studies, interior deformation had little affect. Since the timing calculations show that interior deformation can be the most time consuming step, it is best to apply interior deformation only when it is truly needed. meshCurve applies it by default, assuming the worst, but interior deformation can be easily unselected. It should be noted that the interior deformation algorithm isn't guaranteed to prevent negative Jacobians; it merely reduces their likelihood. For example, negative Jacobians can occur if boundary cells are borderline degenerate. The most recent version of meshCurve has the ability to calculate the Jacobians, alerting the user to potential problems.

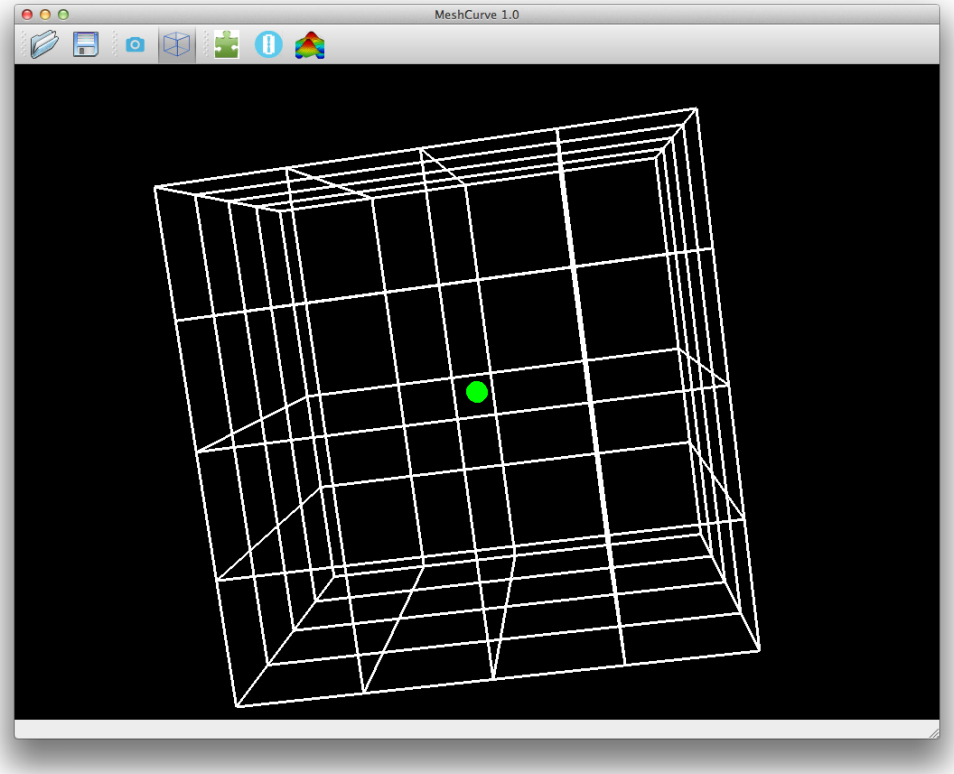


Figure 48: Case study: meshCurve was used to process the above mesh, upgrading it from linear to 2nd order. The mesh represents a sphere inside a square shaped domain. There are two patches in the mesh, one representing the surface of the sphere (highlighted green) and the second representing the outer surface of the domain (shown but not highlighted). The green areas of the mesh were given curvature during the upgrade. Table 8 presents statistics for the before and after mesh.

Table 8: Statistics for figure 48 mesh: meshCurve was used to upgrade the originally linear mesh to 2nd order. The program was executed on a Macintosh computer with a 2.7 GHz Intel Core i5 processor running Mac OS X.

# cells	# nodes	patches: recon./total	interior defor.	time	original Jacobian [low — high]	new Jacobian [low — high]
480	588	1 / 2	no	0.08 sec	[0.0131343 – 192.609]	[0.0129203 – 192.613]
480	588	1 / 2	yes	0.11 sec	[0.0131343 – 192.609]	[0.0129203 – 192.613]

5.5 Internal Architecture Overview

The mesh database is the core of meshCurve. It manages all information associated with the mesh, serving as both the information storage facility and the conduit through which the various components of meshCurve communicate. There are four external classes which interface the database to provide additional functionality. These classes perform the following tasks: 1) file input/output, 2) feature curve detection, 3) surface reconstruction/interior deformation, and 4) 3D visualization. The code for the user interface wraps around the database and accompanying classes. It exposes the internal switches for the user to manipulate. The modular structure of the code base and its object-oriented design make the program easy to edit. Figure 51 illustrates the flow of data.

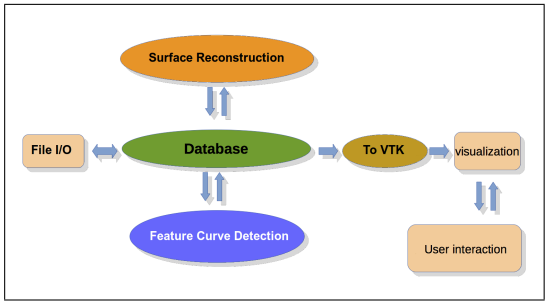


Figure 51: The internal data flow for the program meshCurve. The database is the core of the program, serving as storage facility and communication hub. Four external classes interface the database to accomplish the four tasks 1) File input/output, 2) 3D visualization, 3) feature curve detection, and 4) surface reconstruction.

The mesh database is built for maximum flexibility. It is designed with no assumption about the cell types or node orderings stored within. These definitions are placed elsewhere, as a configuration table, which makes updating the definitions extremely easy. A second example of flexibility is the storage scheme for element connectivities, which uses both forward and backward references to not only answer questions such as "Which nodes occupy this cell?" but also the inverse, "Which cells contain this node?" Furthermore, the database provides a mechanism to cycle through elements as a list, as if the data were stored as an array. The database also provides a mechanism to jump through elements neighbor-to-neighbor, as if the mesh were stored as a connected graph.

5.5.1 Key Algorithms: feature curve detection

To find feature curves, meshCurve employs an algorithm proposed by Jiao and Bayyana, which identifies the curves probabilistically, using a combination of several topological metrics.[59] The algorithm proceeds through several steps. Surface nodes and edges are flagged as candidate discontinuities if the local dihedral angle, ridge direction, and angle defect fall within a predefined range. Then the candidate discontinuities are linked into chains. The list of chains is filtered, and each chain receives a classification based on the properties of its starting and ending nodes. As the algorithm progresses, the list is narrowed until only the most likely candidates remain. In the end, the remaining chains are classified as feature curves. Figure 52 illustrates the capabilities of the method, implemented in meshCurve. Blue lines represent detected curves; orange dots represent detected corners.

5.5.2 Key Algorithms: surface reconstruction

For high-order surface reconstruction, meshCurve employs a variation of Weighted Averaging of Local Fittings (WALF), developed by Jiao and Wang.[60] WALF reconstructs a mesh surface by averaging a set of locally fitted 2D Taylor polynomials. In the original

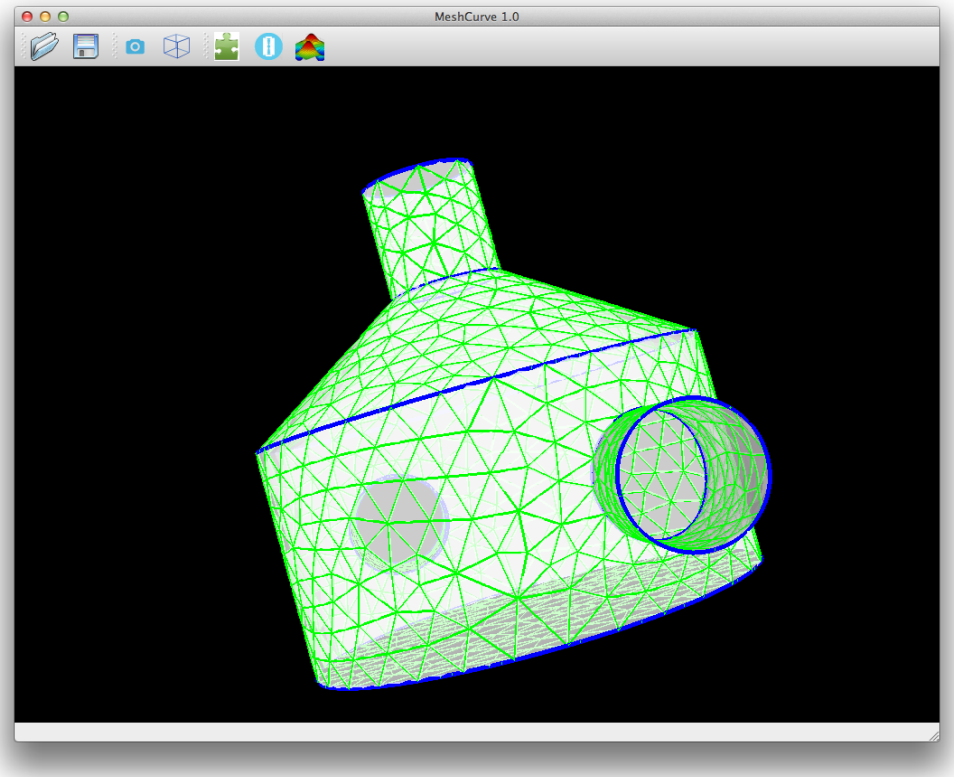


Figure 49: Case study: meshCurve was used to process the above mesh, upgrading it from linear to 2nd order. The mesh represents a static mixer. There are four patches in the mesh, one of which is shown (highlighted green). The other three patches have been hidden from view to make the visualization easier to interpret. Curvature imparted to the high-order mesh is clearly discernible around the rim of the inlet spout. Table **??** presents statistics for the before and after mesh.

Table 9: Statistics for figure 49 mesh: meshCurve was used to upgrade the originally linear mesh to 2nd order. The program was executed on a Macintosh computer with a 2.7 GHz Intel Core i5 processor running Mac OS X.

# cells	# nodes	patches: recon./total	interior defor.	time	original Jacobian [low — high]	new Jacobian [low — high]
13,761	2,786	4 / 4	no	15.3 sec	[0.00164224 – 0.0418735]	[0.000572914 – 0.043354]
13,761	2,786	4 / 4	yes	16.1 sec	[0.00164224 – 0.0418735]	[0.000572914 – 0.0429421]

Table 10: Statistics for figure 50 mesh, representing an airfoil: meshCurve was used to upgrade the originally linear mesh to 2nd order. The program was executed on a Macintosh computer with a 2.7 GHz Intel Core i5 processor running Mac OS X.

# cells	# nodes	patches: recon./total	interior defor.	time	original Jacobian [low — high]	new Jacobian [low — high]
455,988	478,840	1 / 7	no	14.2 sec	[3.00E-016 – 1.31E-008]	[-4.54E-018 – 1.31E-008]
455,988	478,840	1 / 7	yes	6.6 min	[3.00E-016 – 1.31E-008]	[3.00E-016 – 1.31E-008]

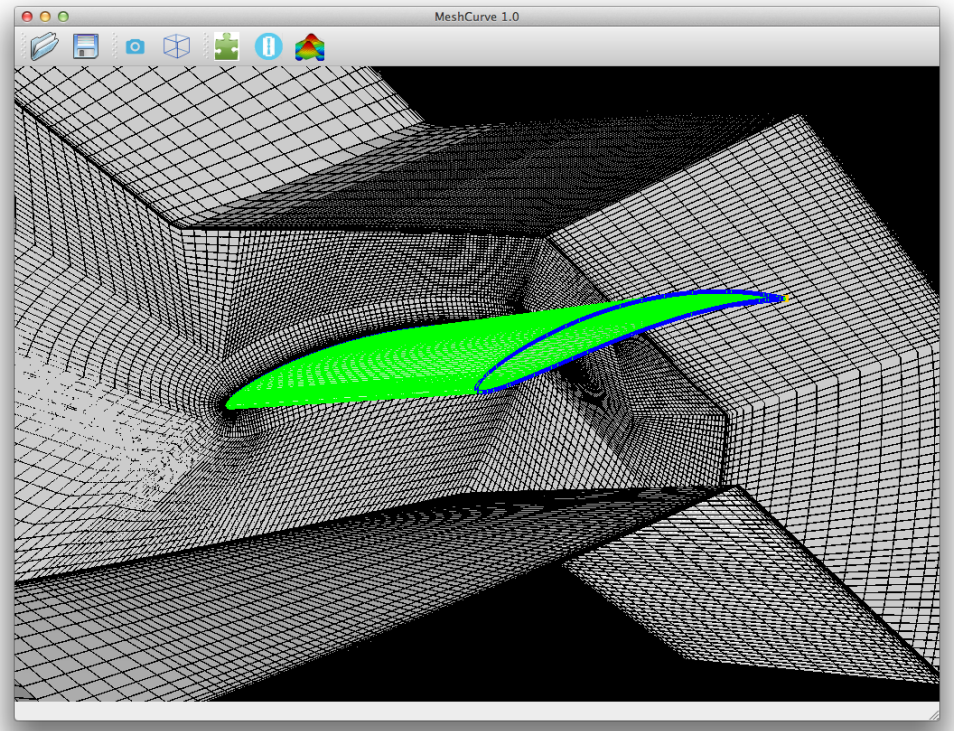
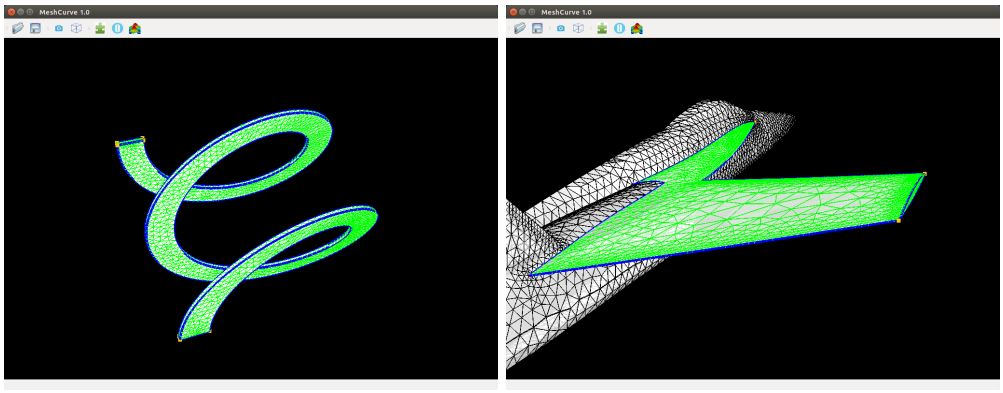


Figure 50: Case study: meshCurve was used to process the above mesh, upgrading it from linear to 2nd order. The mesh represents an airfoil. There are seven patches in the mesh, one representing the surface of the airfoil (highlighted green) and six representing the outer surfaces of the simulation domain. In the above view, the front patch has been hidden to reveal the interior airfoil. Table **??** presents statistics for the before and after mesh.



(a) (b)

Figure 52: Screenshots of the program meshCurve, illustrating automatic feature curve detection. Blue lines mark feature curves identified by the program. In the left subfigure, the edges of a spring have been identified as feature curves. In the right subfigure, feature detection has been limited to the portion of the mesh isolated for processing. The mesh represents a fighter jet with a single wing selected. Curve detection has marked the wing's boundary as a feature curve. Interior curves and corners have also been identified.

algorithm, the polynomials are fit to every node on the surface. In our version, the polynomials are fit to every *face*. The use of faces as the location-of-fit reduces the required number of averages. Also the use of face-centered-fits facilitates the treatment of feature curves as boundaries, thereby preventing the feature curves from being smoothed away. An iterative solver, initialized to zero, helps reduce oscillations.

5.5.3 Key Algorithms: interior deformation

To deform the interior elements, meshCurve employs an interpolation scheme developed by Luke, Collins, and Blades originally designed for the simulation of fluid structure interaction.[76] According to this method, each interior node moves in accordance with a weighted average of the boundary node motion. Every boundary node is weighted inverse to its distance and in proportion to the area it commands.

6 Conclusions

In the present project, a robust and efficient high-order CFD software suite for the compressible Navier-Stokes equations that can provide engineering accuracy for real world industry problems is developed. The following points summarize the key accomplishments and significant conclusions that can be drawn from the current program:

To estimate the error of an engineering output, we extended the dual-weighted residual method originally developed in the variational framework to the high-order CPR method which is in the differential form. A dual-consistent CPR formulation of hyperbolic conservation laws is developed and its dual consistency is analyzed. Super-convergent functional and error estimate for the output with the CPR method are obtained. Factors affecting the dual consistency, such as the solution point distribution, correction functions, boundary conditions and the discretization approach for the non-linear flux divergence term, are studied.

Next, we developed a parallel adjoint-based adaptive CPR solver with the capability of handling any element-based error estimate and arbitrary discretization orders for mixed grids. The current method have been applied to aerodynamic flows and challenging engineering applications. Numerical tests show that significant savings in the number of DOFs can be achieved through the adjoint-based adaptation.

Five SGS models are evaluated with the 1D Burgers' equation discretized with the CPR method. Different LES cell sizes were tested with a fixed filter width. In both the a priori and a posteriori tests on a fine LES mesh, the SSM and the MM showed excellent correlation with the true SGS, while the other models do not predict the SGS stress satisfactorily. However, as the LES cell size increases, numerical truncation error is dominant in the results. In this case, none of the models shows any benefits over ILES. The analysis of scale similarity shows that perfect scale similarity exists for arbitrary (periodic) data including turbulence under the assumption that the spectrum contains relatively low frequency contents with respect to the filter width, regardless of amplitude and phase angle of each mode. In an actual large eddy simulation, in which both large and sub-grid scales exist, the present result on the ratio of the resolved scale stress and the SGS stress may be the upper limit. Test results with data including higher modes near the grid cutoff demonstrate that there is a high level of correlation between the modelled and SGS stresses. Furthermore, $\gamma = 1$ demonstrates consistently higher correlation coefficients than $\gamma = 2$. This may indicate that $\gamma = 1$ is preferred in a SSM implementation. The stability of the SSM is also investigated. The study shows that it is the central flux rather than the SSM that causes energy accumulating at high frequencies, which may lead to the instability of a simulation. In this case, extra dissipation other than SSM, the DS, for example, is necessary. However, the schemes with upwind flux smoothly damps out the energy at high frequencies. Thus no extra dissipation is needed to stabilize the simulation with the SSM.

Finally, we created a user-friendly GUI-based software tool, called meshCurve, to convert linear unstructured meshes to curved high-order meshes. Based on a geometry reconstruction procedure, the software infers boundary curvature from the mesh itself, without need of a CAD geometry file. The software aims to be a cross-platform tool for mesh upgrading, useful wherever high-order meshes are needed.

References

[1] Ainsworth, M., Oden, J., 1993. A unified approach to a posteriori error estimation using element residual methods. Numerische Mathematik 65 (1), 23–50.

[2] Akhavan R, Ansari A, K. S., 2000. Subgrid-scale interactions in a numerically simulated planar turbulent jet and implications for modeling. J. Fluid Mech. 408, 83–120.

[3] Akselvoll K, M. P., 1996. Large-eddy simulation of turbulent confined coannular jets. J. Fluid Mech 315, 387–411.

[4] AW, C., 1997. Determination of the constant coefficient in scale similarity models of turbulence. Phys. Fluids 9, 1485–1487.

[5] Baker, T. J., 1997. Mesh adaptation strategies for problems in fluid dynamics. Finite Elements in Analysis and Design 25 (3-4), 243–273.

[6] Bardina J, Ferziger JH, R. W., 1980. Improved subgrid scale models for large eddy simulation. Am. Inst. Aeronaut. Astronaut, 80–1357.

[7] Barth, T., 1998. Numerical methods for gasdynamic systems on unstructured meshes. An Introduction to Recent Developments in Theory and Numerics for Conservation Laws 5, 195–285.

[8] Barth, T., Frederickson, P., 1990. Higher order solution of the Euler equations on unstructured grids using quadratic reconstruction. AIAA paper 1990-0013.

[9] Bassi, F., Rebay, S., 1997. High-order accurate discontinuous finite element solution of the 2D Euler equations. Journal of Computational Physics 138 (2), 251–285.

[10] Bassi, F., Rebay, S., 2000. Gmres discontinuous Galerkin solution of the compressible Navier-Stokes equations. Lecture Notes in Computational Science and Engineering 11, 197–208.

[11] Baumann, C., Oden, J., 1999. A discontinuous hp- finite element method for the Euler and Navier-Stokes equations. International Journal for Numerical Methods in Fluids 31 (1), 79–95.

[12] Becker, R., Rannacher, R., 1996. A feed-back approach to error control in finite element methods: Basic analysis and examples. East-West J. Numer. Math 4, 237–264.

[13] Becker, R., Rannacher, R., 2001. An optimal control approach to a posteriori error estimation in finite element methods. Acta Numerica 10, 1–102.

[14] Berg, J., Nordström, J., 2014. Duality based boundary conditions and dual consistent finite difference discretizations of the Navier-Stokes and Euler equations. Journal of Computational Physics 259 (0), 135 – 153.

[15] Berger, M., Colella, P., 1989. Local adaptive mesh refinement for shock hydrodynamics. Journal of computational Physics 82 (1), 64–84.

[16] Burgess, N., 2011. An adaptive discontinuous Galerkin solver for aerodynamic flows. Ph.D. thesis, University of Wyoming.

[17] Cagnone, J., Nadarajah, S., 2012. A stable interface element scheme for the p-adaptive lifting collocation penalty formulation. Journal of Computational Physics 231 (4), 1615 – 1634.

[18] Cagnone, J., Vermeire, B., Nadarajah, S., 2013. A p-adaptive LCP formulation for the compressible Navier-Stokes equations. Journal of Computational Physics 233, 324 – 338.

[19] Castonguay, P., 2012. High-order energy stable flux reconstruction schemes for fluid flow simulations on unstructured grids. Ph.D. thesis, STANFORD.

[20] Castro-Diaz, M. J., Hecht, F., Mohammadi, B., Pironneau, O., 1997. Anisotropic unstructured mesh adaption for flow simulations. International Journal for Numerical Methods in Fluids 25 (4), 475–491.

[21] Ceze, M., Fidkowski, K. J., 2012. Anisotropic hp-adaptation framework for functional prediction. AIAA Journal 51 (2), 492–509.

[22] Clark RA, Ferziger JH, R. W., 1979. Evaluation of subgrid-scale models using an accurately simulated turbulent flow. J.Fluid Mech 91, 1–16.

[23] Cockburn, B., Lin, S., Shu, C., 1989. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: One-dimensional systems. Journal of Computational Physics 84 (1), 90–113.

[24] Cockburn, B., Shu, C., 1998. The Runge-Kutta discontinuous Galerkin method for conservation laws V: Multidimensional systems. Journal of Computational Physics 141 (2), 199–224.

[25] Devloo, P., Tinsley Oden, J., Pattani, P., 1988. An hp-adaptive finite element method for the numerical simulation of compressible flow. Computer methods in applied mechanics and engineering 70 (2), 203–235.

[26] DK, L., 1992. A propose modification of the germano subgrid-sclae closure method. Phys. Fluids A 4, 633–635.

[27] Dompierre, J., Vallet, M.-G., Bourgault, Y., Fortin, M., Habashi, W. G., 2002. Anisotropic mesh adaptation: towards user-independent, mesh-independent and solver-independent CFD. part III. unstructured meshes. International Journal for Numerical Methods in Fluids 39 (8), 675–702.

[28] Durbin P. A, P. R. B. A., 2011. Statistical Theory and Modeling for Turbulent Flows. A John Wiley and Sons.

[29] Ekaterinaris, J., 2005. High-order accurate, low numerical diffusion methods for aerodynamics. Progress in Aerospace Sciences 41 (3-4), 192–300.

[30] Fidkowski, K., 2011. Review of output-based error estimation and mesh adaptation in computational fluid dynamics. AIAA Journal 49 (4), 673–694.

[31] Fidkowski, K., Darmofal, D., 2007. A triangular cut-cell adaptive method for high-order discretizations of the compressible Navier-Stokes equations. Journal of Computational Physics 225 (2), 1653–1672.

[32] Fidkowski, K., Roe, P., 2009. Entropy-based mesh refinement, I: The entropy adjoint approach.

[33] Fidkowski, K., Roe, P., 2010. An entropy adjoint approach to mesh refinement. SIAM Journal on Scientific Computing 32 (3), 1261–1287.

[34] Gao, H., Wang, Z. J., 2011. A residual-based procedure for hp-adaptation on 2D hybrid meshes. AIAA Paper 2011-492.

[35] Gao, H., Wang, Z. J., Jan. 2013. A conservative correction procedure via reconstruction formulation with the chain-rule divergence evaluation. Journal of Computational Physics 232, 7–13.

[36] Gao, H., Wang, Z. J., 2013. Differential formulation of discontinuous Galerkin and related methods for the Navier-Stokes equations. Commun. Comput. Phys. 13, 1013–1044.

[37] Georgoulis, E. H., Hall, E., Houston, P., Sep. 2009. Discontinuous Galerkin methods on hp-anisotropic meshes II: a posteriori error analysis and adaptivity. Appl. Numer. Math. 59 (9), 2179–2194.

[38] Germano M, Piomelli U, M. P., WH, C., 1991. A dynamic subgrid-scale eddy viscosity model A 3, 1760–1765.

[39] Ghosal S, L. T., 1995. A dynamic localization model for large eddy simulation of turbulent flows. J. Fluid Mech 286, 229–255.

[40] Ghosal S, R. M., 1997. A numerical study of self-similarity in a turbulent plane wake using large-eddy simulation. Phys. Fluids 9, 1729–1739.

[41] Giles, M., Pierce, N., 1997. Adjoint equations in CFD: duality, boundary conditions and solution behaviour. AIAA paper 97-1850.

[42] Giles, M., Pierce, N., 2003. Adjoint error correction for integral outputs. In: Barth, T., Deconinck, H. (Eds.), Error Estimation and Adaptive Discretization Methods in Computational Fluid Dynamics. Vol. 25 of Lecture Notes in Computational Science and Engineering. Springer Berlin Heidelberg, pp. 47–95.

[43] Gottlieb, S., Shu, C.-W., 2011. Strong stability-preserving high-order time discretization methods. Scociety for Industrial and Applied Mathematics 43, 89–112.

[44] Grinstein F. F., M. L. G., 2011. Implicit large eddy simulation:computing turbulent fluid dynamics. Cambridge University Press.

[45] Harish Gopalan, S. H., 2013. A unified rans-les model:computational development, accuracy and cost. J. Computational Physics 249, 249–274.

[46] Harris, R., Wang, Z. J., 2009. High-order adaptive quadrature-free spectral volume method on unstructured grids. Computers & Fluids 38 (10), 2006–2025.

[47] Hartmann, R., 2007. Adjoint consistency analysis of discontinuous Galerkin discretizations. SIAM J. Numer. Anal. 45 (6), 2671–2696.

[48] Hartmann, R., Houston, P., 2002. Adaptive discontinuous Galerkin finite element methods for the compressible Euler equations. Journal of Computational Physics 183 (2), 508 – 532.

[49] Hesthaven, J., Warburton, T., 2008. Nodal Discontinuous Galerkin Methods: Algorithms, Analysis and Applications. Vol. 54. Springer-Verlag New York Inc.

[50] Hicken, J., 2012. Output error estimation for summation-by-parts finite-difference schemes. Journal of Computational Physics 231 (9), 3828 – 3848.

[51] Hicken, J., Zingg, D., 2013. Summation-by-parts operators and high-order quadrature. Journal of Computational and Applied Mathematics 237 (1), 111 – 125.

[52] Houston, P., Süli, E., 2001. Hp-adaptive discontinuous Galerkin finite element methods for first-order hyperbolic problems. SIAM Journal on Scientific Computing 23 (4), 1226–1252.

[53] Huang, W., Russell, R. D., 2010. Adaptive Moving Mesh Methods. Vol. 174. Springer.

[54] Huynh, H. T., 2007. A flux reconstruction approach to high-order schemes including discontinuous Galerkin methods. AIAA Paper 2007-4079.

[55] Huynh, H. T., 2011. High-order methods by correction procedures using reconstructions. Adaptive High-Order Methods in Computational Fluid Dynamics 2, 391–422.

[56] Huynh, H. T., Z. J. W., Vincent, P. E., 2014. High-order methods for computational fluid dynamics: A brief review of compact differential formulations on unstructured grids. Computers & Fluids 98 (0), 209–220.

[57] J., S., 1963. General circulation experiments with the primitive equations. The basic experiment, Weather Rev, 91–99.

[58] Jameson, A., Vincent, P. E., Castonguay, P., Feb. 2012. On the non-linear stability of flux reconstruction schemes. J. Sci. Comput. 50 (2), 434–445.

[59] Jiao, X., Bayyana, N. R., 2008. Identification of c1 and c2 discontinuities for surface meshes in cad. Computer-Aided Design 40 (2), 160–175.

[60] Jiao, X., Wang, D., 2012. Reconstructing high-order surfaces for meshing. Engineering with Computers 28 (4), 361–373.

[61] Johnson, C., 1998. Adaptive finite element methods for conservation laws. Advanced numerical approximation of nonlinear hyperbolic equations, 269–323.

[62] K. Duraisamy, J. J. Alonso, F. P., Chandrashekar, P., 2010. Error estimation for high speed flows using continuous and discrete adjoints. AIAA Paper 2010–128.

[63] Karniadakis, G., Sherwin, S., 1999. Spectral/hp Element Methods for CFD. Oxford University Press, USA.

[64] Kopriva, D., Kolias, J., 1996. A conservative staggered-grid chebyshev multidomain method for compressible flows. Journal of computational physics 125 (1), 244–261.

[65] Leicht, T., Hartmann, R., Sep. 2010. Error estimation and anisotropic mesh refinement for 3D laminar aerodynamic flow simulations. Journal of Computational Physics 229 (19), 7344–7360.

[66] Lesieur M, M. O., 1984. New trends in large-eddy simulations of turbulence. Annu. Rev. Fluid Mech 16, 99–137.

[67] Li, Y., Allaneau, Y., Jameson, A., 2010. Continuous adjoint approach for adaptive mesh refinement. AIAA Paper 2010-3982.

[68] Li, Y., Premasuthan, S., Jameson, A., 2010. Comparison of h- and p-adaptations for spectral difference methods. AIAA Paper 2010-4435.

[69] Li, Y., Wang, Z. J., 2013. Evaluation of optimized CPR schemes for computational aeroacoustics benchmark problems. AIAA Paper 2013-2689.

[70] Liang, C., Jameson, A., Wang, Z. J., May 2009. Spectral difference method for compressible flow on unstructured grids with mixed elements. Journal of Computational Physics 228 (8), 2847–2858.

[71] Liu, Y., Vinokur, M., Wang, Z. J., 2006. Discontinuous spectral difference method for conservation laws on unstructured grids. Computational Fluid Dynamics 2004, 449–454.

[72] Liu, Y., Vinokur, M., Wang, Z. J., 2006. Spectral finite volume method for conservation laws on unstructured grids V: Extension to three-dimensional systems. Journal of Computational Physics 212 (2), 454–472.

[73] Liu S, M. C., 1994. On the properties of similarity subgrid-scale models as deduced from measurements in a turbulent jet. J. Fluid Mech 275, 83–119.

[74] Lohner, R., Morgan, K., Peraire, J., Vahdati, M., 1987. Finite element flux-corrected transport for the Euler and Navier-Stokes equations. International Journal for Numerical Methods in Fluids 7 (10), 1093–1109.

[75] Lu, J. C.-C., 2005. An a posteriori error control framework for adaptive precision optimization using discontinuous Galerkin finite element method. Ph.D. thesis, Massachusetts Institute of Technology.

[76] Luke, E., Collins, E., Blades, E., 2012. A fast mesh deformation method using explicit interpolation. Journal of Computational Physics 231 (2), 586–601.

[77] Luo, X., Shephard, M. S., Remacle, J.-F., 2001. The influence of geometric approximation on the accuracy of high order methods. Rensselaer SCOREC report 1.

[78] May, G., Jameson, A., 2006. A spectral difference method for the Euler and Navier-Stokes equations on unstructured meshes. AIAA paper 2006-304.

[79] McMillan OJ, Ferzinger JH, R. R., 1982. Tests of new subgrid scale models in strained turbulence. Am. Inst. Aeronaut. Astronaut. 80, 13–39.

[80] Meneveau C, K. J., 2000. Scale-invariance and turbulence models for large-eddy simulation. Annu. Rev. Fluid Mech., 1–32.

[81] Moin P, K. J., 1982. Numerical investigation of turbulent channel flow. J. Fluid Mech 118, 341–77.

[82] Park, M., 2002. Adjoint-based three-dimensional error prediction and grid adaptation. AIAA paper 2002-3286.

[83] Peraire, J., Persson, P., 2007. The compact discontinuous Galerkin CDG method for elliptic problems. Arxiv preprint math/0702353.

[84] Piomelli U, L. J., 1995. Large-eddy simulation of rotating channel flows using a localized dynamic model. Phys.Fluids 7, 839–848.

[85] Reed, W. H., Hill, T. R., 1973. Triangular Mesh Methods for the Neutron Transport Equation.

[86] Rogallo RS, M. P., 1984. Numerical simulation of turbulent flows. Annu. Rev. Fluid Mech 16, 99–137.

[87] Sanderson, C., 2010. Armadillo: An open source c++ linear algebra library for fast prototyping and computationally intensive experiments.

[88] SB, P., 2013. Turbulent Flows. Cambridge Univ. Press, Cambridge UK.

[89] Schroeder, W., Martin, K., Lorensen, B., 2006. The Visualization Toolkit, 4th Edition. Kitware, Inc., United States of America.

[90] Schütz, J., May, G., 2013. An adjoint consistency analysis for a class of hybrid mixed methods. IMA Journal of Numerical Analysis 34, 1222–1239.

[91] Shi, L., Wang, Z. J., 2013. Adjoint based anisotropic mesh adaptation for the cpr method. AIAA Paper 2013-2869.

[92] Shi, L., Wang, Z. J., 2015. Adjoint-based Error Estimation and Mesh Adaptation for the Correction Procedure via Reconstruction Method. Journal of Computational Physics 295, 261–284.

[93] Shih, T., Qin, Y., 2007. A posteriori method for estimating and correcting grid-induced errors in CFD solutions-part 1: Theory and method. AIAA Paper 2007-100.

[94] T. Haga, H. G., Wang, Z. J., 2011. A high-order unifying discontinuous formulation for the Navier-Stokes equations on 3D mixed grids. Math. Model. Nat. Phenom. 6 (03), 28–56.

[95] Taneda, S., Oct. 1956. Experimental Investigation of the Wake behind a Sphere at Low Reynolds Numbers. Journal of the Physical Society of Japan 11, 1104.

[96] V. Heuveline, R. R., 2003. Duality-based adaptivity in the hp-finite element method. Journal of Numerical Mathematics (2), 95–113.

[97] Van den Abeele, K., Lacor, C., Wang, Z. J., 2008. On the stability and accuracy of the spectral difference method. Journal of Scientific Computing 37 (2), 162–188.

[98] Venditti, D., Darmofal, D., 2003. Anisotropic grid adaptation for functional outputs: application to two-dimensional viscous flows. Journal of Computational Physics 187 (1), 22–46.

[99] Venditti, D. A., Darmofal, D. L., Oct. 2000. Adjoint error estimation and grid adaptation for functional outputs: application to quasi-one-dimensional flow. Journal of Computational Physics 164 (1), 204–227.

[100] Venditti, D. A., Darmofal, D. L., May 2003. Anisotropic grid adaptation for functional outputs: application to two-dimensional viscous flows. Journal of Computational Physics 187 (1), 22–46.

[101] Wang, L., Mavriplis, D., 2009. Adjoint-based hp-adaptive discontinuous Galerkin methods for the 2D compressible Euler equations. Journal of Computational Physics 228 (20), 7643–7661.

[102] Wang, L., Mavriplis, D. J., Nov. 2009. Adjoint-based hp-adaptive discontinuous Galerkin methods for the 2D compressible Euler equations. Journal of Computational Physics 228 (20), 7643–7661.

[103] Wang, Z., 2007. High-order methods for the euler and navier–stokes equations on unstructured grids. Progress in Aerospace Sciences 43 (1), 1–41.

[104] Wang, Z. J., 2002. Spectral (finite) volume method for conservation laws on unstructured grids: basic formulation. Journal of Computational Physics 178 (1), 210–251.

[105] Wang, Z. J., 2007. High-order methods for the Euler and Navier-Stokes equations on unstructured grids. Progress in Aerospace Sciences 43 (1-3), 1–41.

[106] Wang, Z. J., May 2011. Adaptive High-Order Methods in Computational Fluid Dynamics. World Scientific Publishing.

[107] Wang, Z. J., Fidkowski, K., Abgrall, R., Bassi, F., Caraeni, D., Cary, A., Deconinck, H., Hartmann, R., Hillewaert, K., Huynh, H. T., Kroll, N., May, G., Persson, P.-O., van Leer, B., Visbal, M., Jul. 2013. High-order CFD Methods: Current Status and Perspective. International Journal for Numerical Methods in Fluids 72, 811–845.

[108] Wang, Z. J., Gao, H., 2009. A unifying lifting collocation penalty formulation including the discontinuous Galerkin, spectral volume/difference methods for conservation laws on mixed grids. Journal of Computational Physics 228, 8161–8186.

[109] Wang, Z. J., Gao, H., Haga, T., 2011. A unifying discontinuous formulation for hybrid meshes. Adaptive High-Order Methods in Computational Fluid Dynamics, 423–453.

[110] Wang, Z. J., Liu, Y., 2002. Spectral (finite) volume method for conservation laws on unstructured grids II. extension to two-dimensional scalar equation. Journal of Computational Physics 179 (2), 665–697.

[111] Warburton, T., 2006. An explicit construction of interpolation nodes on the simplex. Journal of engineering mathematics 56 (3), 247–262.

[112] Warren, G., Anderson, W., Thomas, J., Krist, S., 1991. Grid convergence for adaptive methods. In: 10th AIAA Computational Fluid Dynamics Conference.

[113] Wu X, S. K., 1997. Large eddy simulation of an equilibrium three dimensional turbulent boundary layer. Am. Inst. Aeronaut. Astronaut. J. 35, 67–74.

[114] Wu X, S. K., 1998. Numerical investigation of the turbulent boundary layer over a bump. J. Fluid Mech. 362, 229–271.

[115] Xie, Z. Q., Sevilla, R., Hassan, O., Morgan, K., 2013. The generation of arbitrary order curved meshes for 3d finite element analysis. Computational Mechanics 51 (3), 361–374.

[116] Yang, X., Huang, W., Qiu, J., 2012. A moving mesh weno method for one-dimensional conservation laws. SIAM Journal on Scientific Computing 34 (4), A2317–A2343.

[117] Yano, M., Darmofal, D. L., Sep. 2012. An optimization-based framework for anisotropic simplex mesh adaptation. Journal of Computational Physics 231 (22), 7626–7649.

[118] Yu, M., Wang, Z. J., Liu, Y., 2014. On the accuracy and efficiency of discontinuous Galerkin, spectral difference and correction procedure via reconstruction methods. Journal of Computational Physics 259, 70–95.

[119] Zhang, X., Vallet, M.-G., Dompierre, J., Labbe, P., Pelletier, D., Trepanier, J.-Y., 2001. Mesh adaptation using different error indicators for the Euler equations. AIAA Paper 2001-2549.

[120] Z.J. Wang, Y. L., 2015. An analysis of scale similarity and its implications for large eddy simulation. Commun. Comput. Phys.Submitted.

1.

1. Report Type

Final Report

Primary Contact E-mail

Contact email if there is a problem with the report.

zjw@ku.edu

Primary Contact Phone Number

Contact phone number if there is a problem with the report

7858642440

Organization / Institution name

University of Kansas

Grant/Contract Title

The full title of the funded effort.

The Development of High-Order Methods for Real World Applications

Grant/Contract Number

AFOSR assigned control number. It must begin with "FA9550" or "F49620" or "FA2386".

FA9550-12-1-0286

Principal Investigator Name

The full name of the principal investigator on the grant or contract.

ZJ Wang

Program Manager

The AFOSR Program Manager currently assigned to the award

Jean-Luc Cambier

Reporting Period Start Date

09/01/2012

Reporting Period End Date

08/31/2015

Abstract

With increased computational power and progress in numerical methods over the past several decades, Computational Fluid Dynamics (CFD) is now used routinely as a powerful tool in the design of aircraft. Current production CFD codes used in the aerospace industry are usually second order accurate. High-order methods have the potential to achieve higher accuracy at less cost than low-order methods. This potential has been demonstrated conclusively for smooth problems in the latest International Workshops on High-Order Methods. For non-smooth problems, solution based hp-adaptation offers the best promise. The primary objective of the present study is to develop robust and efficient high-order CFD methods and tools for the compressible Navier-Stokes equations that can provide engineering accuracy for real world industry problems. Several pacing items are addressed, which include hp-adaptations, sub-grid stress models for large eddy simulations, and high-order mesh generation.

Distribution Statement

This is block 12 on the SF298 form.

Distribution A - Approved for Public Release

Explanation for Distribution Statement

If this is not approved for public release, please provide a short explanation. E.g., contains proprietary information.

SF298 Form

Please attach your SF298 form. A blank SF298 can be found here. Please do not password protect or secure the PDF The maximum file size for an SF298 is 50MB.

AFD-070820-035.pdf

Upload the Report Document. File must be a PDF. Please do not password protect or secure the PDF . The maximum file size for the Report Document is 50MB.

airforce_report_r2.pdf

Upload a Report Document, if any. The maximum file size for the Report Document is 50MB.

Archival Publications (published) during reporting period:

1. B.J. Zimmerman and Z.J. Wang, "The efficient implementation of correction procedure via reconstruction with graphics processing unit computing," Computers & Fluids, 101 (2014) 263–272.
2. Z.J. Wang, "High-order CFD Tools for Aircraft Design", Philosophical Transactions A of the Royal Society, 372, 20130318, July 2014.
3. H.T. Huynh, Z.J. Wang and P.E. Vincent, "High-Order Methods for Computational Fluid Dynamics: A Brief Review of Compact Differential Formulations on Unstructured Grids, Computers and Fluids Volume 98, 2 July 2014, Pages 209–220.
4. M.L. Yu, Z. J. Wang and Y. Liu, "On the accuracy and efficiency of discontinuous Galerkin, spectral difference and correction procedure via reconstruction methods," Journal of Computational Physics Volume 259, 15 February 2014, Pages 70–95.
5. M.L. Yu, Z. J. Wang, H. Hu, Formation of Bifurcated Wakes Behind Finite Span Flapping Wings, AIAA Journal 51 (No. 8), 2040-2044, 2013.
6. Z.J. Wang, K.J. Fidkowski, R. Abgrall, F. Bassi, D. Caraeni, A. Cary, H. Deconinck, R. Hartmann, K. Hillewaert, H.T. Huynh, N. Kroll, G. May, P-O. Persson, B. van Leer, and M. Visbal. "High-Order CFD Methods: Current Status and Perspective," International Journal for Numerical Methods in Fluids, 72, 811-845, (2013).
7. H. Gao and Z.J. Wang, "A Conservative Correction Procedure via Reconstruction Formulation with the Chain-Rule Divergence Evaluation", J. Computational Physics 232, 7–13 (2013).
8. Y. Li and Z.J. Wang, "An Optimized Correction Procedure via Reconstruction Formulation for Broadband Wave Computation", Communications in Computational Physics, Vol. 13, No. 5, pp. 1265-1291 (2013).
9. H. Gao, Z.J. Wang and H.T. Huynh, "Differential Formulation of Discontinuous Galerkin and Related Methods for the Navier-Stokes Equations", Communications in Computational Physics 13, No. 4, 1013-1044 (2013).
10. M.L. Yu, Z.J. Wang, "On the Connection Between the Correction and Weighting Functions in the Correction Procedure via Reconstruction Method," J Sci Comput 54, 227–244 (2013).

Changes in research objectives (if any):

None

Change in AFOSR Program Manager, if any:

From Dr. Fariba Fahroo to Dr. Jean-Luc Cambier

Extensions granted or milestones slipped, if any:

None

AFOSR LRIR Number

LRIR Title

Reporting Period

Laboratory Task Manager

Program Officer

Research Objectives

Technical Summary

Funding Summary by Cost Category (by FY, $K)

	Starting FY	FY+1	FY+2
Salary			
Equipment/Facilities			
Supplies			
Total			

Report Document

Report Document - Text Analysis

Report Document - Text Analysis

Appendix Documents

2. Thank You

E-mail user

Nov 24, 2015 19:44:27 Success: Email Sent to: zjw@ku.edu