5-2015

Extensions of High-order Flux Correction Methods to Flows With Source Terms at Low Speeds

Jonathan L. Thorne
Utah State University

Recommended Citation

Thorne, Jonathan L., "Extensions of High-order Flux Correction Methods to Flows With Source Terms at Low Speeds" (2015). *All Graduate Theses and Dissertations*. 4594.
https://digitalcommons.usu.edu/etd/4594

EXTENSIONS OF HIGH-ORDER FLUX CORRECTION METHODS TO FLOWS WITH SOURCE TERMS AT LOW SPEEDS

by

Jonathan L. Thorne

A thesis submitted in partial fulfillment
of the requirements for the degree

of

MASTER OF SCIENCE

in

Mechanical Engineering

Approved:

Dr. Aaron Katz Major Professor	Dr. Robert E. Spall Committee Member
Dr. Tadd Truscott Committee Member	Dr. Mark R. McLellan Vice President for Research and Dean of the School of Graduate Studies

UTAH STATE UNIVERSITY
Logan, Utah

2015

Abstract

Extensions of High-order Flux Correction Methods to Flows with Source Terms at Low Speeds

by

Jonathan L. Thorne, Master of Science

Utah State University, 2015

Major Professor: Dr. Aaron Katz
Department: Mechanical and Aerospace Engineering

A novel high-order finite volume scheme using flux correction methods in conjunction with structured finite difference schemes is extended to low Mach and incompressible flows on strand grids. Flux correction achieves high-order by explicitly canceling low-order truncation error terms in the finite volume cell. The flux correction method is applied in unstructured layers of the strand grid. The layers are then coupled together using a source term containing the derivatives in the strand direction. Proper source term discretization is verified. Strand-direction derivatives are obtained by using summation-by-parts operators for the first and second derivatives. A preconditioner is used to extend the method to low Mach and incompressible flows. We further extend the method to turbulent flows with the Spalart Allmaras model. We verify high-order accuracy via the method of manufactured solutions, method of exact solutions, and physical problems. Results obtained compare well to analytical solutions, numerical studies, and experimental data. It is foreseen that future application in the Naval field will be possible.

(100 pages)

Public Abstract

Extensions of High-order Flux Correction Methods to Flows with Source Terms at Low Speeds

by

Jonathan L. Thorne, Master of Science

Utah State University, 2015

Major Professor: Dr. Aaron Katz
Department: Mechanical and Aerospace Engineering

Computational Fluid Dynamics (CFD) is an attempt to mimic the physical world of fluids around us. In general, CFD simulations are completed by first making a mesh. Meshes are made of polygons that are arranged in such a way as to create the item in question and the fluid around it. There are many difficulties associated with mesh generation, and it can take days or weeks for a trained professional to produce adequate meshes. The CFD algorithms affect the accuracy of the simulation. Using high-order methods, more accurate results can be achieved than otherwise possible. Flux correction (FC) is a high-order method that uses a simple correction term, upgrading low-order methods to high-order. Some fluids, such as air, are compressible, meaning the density can be changed following a set of relationships between pressure and temperature. Other fluids, such as water, can be considered incompressible, where they do not change density to any large extent. Using a mathematical method called preconditioning, FC has been extended to incompressible flows. Before this can be used generally, it must pass verification and validation tests. Verification test show that the method is working how it should, usually these tests have little to no meaning in the real world. Validation tests show that real world physics can be mimicked. These tests usually are real world problems that have solutions that are either analytical, experimental, or produced by other verified CFD programs. In this work specifics of CFD, meshing, FC, and preconditioning will be explained, and verification and validation will be completed.

Dedicated to my wife, son, and all those who have helped me on my way. Also to God who has been a constant support for my life.

Acknowledgments

Development was performed with the support of the Computational Research and Engineering for Acquisition Tools and Environments (CREATE) Program sponsored by the U.S. Department of Defense HPC Modernization Program Office and by the Office of Naval Research Sea-Based Aviation program directed by Dr. Judah Milgram and Mr. John Kinzer.

Contents

Page

Abstract . . . **iii**

Public Abstract . . . **iv**

Acknowledgments . . . **vi**

List of Tables . . . **ix**

List of Figures . . . **xi**

1 Introduction . . . **1**
- 1.1 Literature Survey . . . 4
  - 1.1.1 Strand Meshing Methods . . . 4
  - 1.1.2 High-order Methods . . . 5
  - 1.1.3 Source Terms . . . 7
  - 1.1.4 Preconditioning . . . 7
  - 1.1.5 Turbulence . . . 10
- 1.2 Overview . . . 12

2 Source Term Discretization Effects on the Accuracy of Finite Volume Schemes: *Jonathan Thorne, Aaron Katz* . . . **13**
- 2.1 Introduction . . . 13
- 2.2 Finite Volume Flux Schemes . . . 15
  - 2.2.1 Constant Reconstruction . . . 17
  - 2.2.2 Linear Reconstruction . . . 19
  - 2.2.3 Flux Correction . . . 20
- 2.3 Source Term Discretization . . . 22
  - 2.3.1 Point Source . . . 22
  - 2.3.2 Galerkin . . . 23
  - 2.3.3 Corrected . . . 24
  - 2.3.4 Combined Source and Flux Truncation Error . . . 25
- 2.4 Computational Studies . . . 28
  - 2.4.1 Exact Euler Solution without Source . . . 29
  - 2.4.2 MMS Source Case . . . 32
  - 2.4.3 Quasi-one-dimensional Flow . . . 37
- 2.5 Conclusions . . . 41

3 High-order Strand Grid Methods for Low Mach and Incompressible Flows: ***Jonathan Thorne, Aaron Katz, Oisin Tong, Yushi Yanagita, Keegan Delaney*** **44**
3.1 Introduction 44
3.2 Numerical Methods 46
3.2.1 High-order Strand Grid Discretization 47
3.2.2 Preconditioning 50
3.3 Results 55
3.3.1 MMS Verification 55
3.3.2 Laminar Flow Validation 56
3.3.3 Turbulent Flow Validation 59
3.3.4 Gertler Series Model 4155 59
3.3.5 Gertler Series Model 4621 60
3.3.6 Body 1 64
3.4 Conclusions 65

4 Conclusion **69**

References **71**

Appendices **77**
A Truncation Error Analysis 78
A.1 Flux Correction Truncation Error 80
A.2 Combined Flux Correction and Corrected Source schemes 81
B Body of Revolution Geometries 83
B.1 Model 4155 83
B.2 Model 4621 84
B.3 Body 1 85
C Measurement Uncertainty Analysis 87

List of Tables

Table Page

2.1 Truncation error analysis results on arbitrary grids 28

2.2 Truncation error analysis results on regular grids 29

2.3 Ringleb flow grid refinement study using constant reconstruction. 30

2.4 Ringleb flow grid refinement study using linear reconstruction. 31

2.5 Ringleb flow grid refinement study using flux correction. 32

2.6 Order of accuracy summary for Ringleb flow grid refinement study. 32

2.7 Order of accuracy results for MMS using constant reconstruction with point source. 33

2.8 Order of accuracy results for MMS using constant reconstruction with Galerkin source. 34

2.9 Order of accuracy results for MMS using constant reconstruction with corrected source. 34

2.10 Time (s) per iteration on 128 by 128 grid using constant reconstruction and various source treatments. 35

2.11 Order of accuracy results for MMS using linear reconstruction with point source. . 35

2.12 Order of accuracy results for MMS using linear reconstruction with Galerkin source. 36

2.13 Order of accuracy results for MMS using linear reconstruction with corrected source. 36

2.14 Time (s) per iteration on 128 by 128 grid using linear reconstruction and various source treatments. 37

2.15 Order of accuracy results for MMS using flux correction with point source. 37

2.16 Order of accuracy results for MMS using flux correction with Galerkin source. . . 38

2.17 Order of accuracy results for MMS using flux correction with corrected source. . . 38

2.18 Time (s) per iteration on 128 by 128 grid using flux correction and various source treatments. 39

3.1 Comparison of preconditioned and non-preconditioned solutions for flow around a sphere, $M = 0.002$, $Re = 40$. 56

B.1 Model 4155 Geometry Coefficient Values 83

B.2 Model 4155 Geometry Particulars 84

B.3 Model 4155 Geometry Coefficient Values 84

B.4 Model 4155 Geometry Particulars 85

B.5 Body 1 Offset Data 86

List of Figures

Figure Page

1.1 Example of Low and High-order Strand Meshes 5

2.1 Node-centered finite volume 15

2.2 Node-centered grid in one dimension 16

2.3 Exact Ringleb flow solution for density. 31

2.4 Results of Ringleb flow grid refinement study. 31

2.5 MMS solution for density. 32

2.6 Convergence results for MMS using constant reconstruction with various source treatments. 34

2.7 Convergence results for MMS using linear reconstruction with various source treatments. 36

2.8 Convergence results for MMS using Flux Correction with various source treatments. 38

2.9 Solution for quasi-one-dimensional flow. 40

2.10 Convergence results for Quasi-one-dimensional flow using the constant reconstruction scheme. 40

2.11 Convergence results for Quasi-one-dimensional flow using the liner reconstruction scheme. 42

2.12 Convergence results for Quasi-one-dimensional flow using the FC scheme. 42

3.1 Transformation between physical and computational domains for a strand grid. . . 48

3.2 Example manufactured solution. 57

3.3 MMS results for laminar and turbulent flow grid convergence studies. 57

3.4 Pressure and streamlines for flow around a sphere, $M = 0.002$, $Re = 40$. 57

3.5 Convergence of u-momentum with preconditioned and non-preconditioned systems for flow around a sphere, $M = 0.002$, $Re = 40$. 58

3.6 Flow characteristics for flow over a sphere over a range of low Reynolds number. . 58

3.7 Model 4155 mesh. 59

3.8 Model 4155 drag computation for various Reynolds numbers. 61

3.9 Model 4155 turbulent viscosity. 61

3.10 Model 4621 mesh. 62

3.11 Normal force coefficient for Model 4621 at various angles of attack at Reynolds number of 4.4 million. 63

3.12 Model 4621 at 18 deg. angle of attack. 63

3.13 Body 1 Meshes. 64

3.14 Body 1 afterbody pressure measurement stations. 65

3.15 Body 1 afterbody surface pressure for last third of the body 66

3.16 Body 1 afterbody pressure coefficient profiles at various stations. 67

Chapter 1
Introduction

Computational Fluid Dynamics (CFD) is a quickly expanding field. As computational tools increase in performance and ability, so does the complexity of the problems that are evaluated with CFD tools. Simulations including compressible and incompressible regions, viscosity, turbulence, chemical reactions, and complex geometries are becoming more and more common in industry. Current issues in the CFD community are high-order mesh generation, the need for high-order solution methods, and overall method scalability for large multiprocessor simulations. Each issue has many complexities that need to be addressed to find a solution.

First, high-order mesh generation has an increasing need for automation. High-order meshes tend to be difficult to generate, but resolve the meshed geometry much more accurately then standard mesh procedures do. This issue has been brought up by Wang as one of the most pressing issues facing modern CFD [1]. Many mesh generation experts need to take days or even weeks to generate a sufficient mesh to achieve the accuracy needed. Often solutions must be run on a mesh, then re-evaluated to see if the mesh quality was sufficient or if further mesh refinement is needed. Better methods of automatic mesh generation are needed to alleviate these tasks.

Second, there are limitations of currently widely used low-order methods on providing adequate solutions on complex unstructured meshes. Research in the use of high-order methods has increased due to the need of providing better results, however, many current CFD users continue the use of second-order or lower methods. This is mainly due to their availability, ease of application, and general stability. Use of low-order methods is sometimes sufficient for the problem being solved. However, the simulation's complexities are often hidden due to reduced accuracy, leading to incorrect results. In addition, many high-order methods are still not up to the production stage,or generally stable, reliably converge, time efficient, and capable of solving most simulations. Because of these reasons, many practitioners still resort to the lower-order methods.

Finally, as CFD simulations continue to increase in size and be spread across larger numbers of processors, the need for scalability becomes clear. Scalability must not only be applied to meshes, but also to the solution methods being used. Limiting the information that must be shared between processing units greatly increases speed and efficiency. However, current common meshing techniques require large amounts of information to be shared among processors, reducing computational power and efficiency.

Many of these problems can be alleviated using strand grids [2, 3]. Strand grids are produced using a surface tessellation of the body. Strands are then emanated off of the body in the normal direction. Along each strand, nodes are placed using a structured distribution. This creates a series of unstructured meshes combined by structured strands. Along the strands a finite difference method can be used, and along the unstructured layers a finite volume or other solution method is used. Strands are then smoothed or clipped to ensure that the strands do not cross, generating poor geometries. The advantage of this is many cad packages in use allow for surface tessellation where the user can specify the quality of the mesh. Because arbitrary polygons can be used for the surface tessellation, this is also effective in generating high-order meshes. In addition, meshes are easily automatically generated, increasing the speed in which meshes can be created and used. The structured distribution allows for great domain connectivity advantages in where each processor can have access to the entire computational domain with little need for cross processor communication.

Flux correction (FC), developed by Katz and Sankaran [4], can provide high-order results with little addition to current low-order methods. FC was developed based off of truncation error analysis on the second-order Galerkin method. The result is a formally third-order method on arbitrary grids in multiple dimensions [5]. Because FC is an extension of a previously widely used method, other techniques, such as limiters and preconditioning schemes, can continue to be used, but the solution will be upgraded to high-order. Currently FC has been developed for the compressible Euler and Navier-Stokes equations [5]. Because of its ability to maintain high-order on arbitrary grids, FC makes a good candidate for solving along strand grids in the unstructured directions.

Strand grids and FC require proper source term discretization or treatment to ensure solution accuracy. Much like other numerical methods, source terms can be changed to incorporate truncation error. However, this truncation error introduced into the equation can be removed by combining it with the truncation error of the flux. An example of this is shown in Appendix A. Proper source terms ensure that the order of accuracy of the method isn't spoiled. Effects of improper source term discretization can reduce the order of accuracy of the solution, and possibly provide inaccurate solutions. Often these source terms are non-intuitive and can be as complex, or more complex, then the derivative terms [6–8].

Any compressible method, such as FC, can be extended incompressible flow by the addition of a preconditioner. Preconditioning works by solving the eigenvalue disparities of the finite volume scheme at low-Mach regions. In other works, it reduces the "stiffness" of the equations, making them simpler to solve for the specific flow. Preconditioning aids in solution convergence as well. It can also be used to introduce the artificial compressibility coefficient developed by Chorin [9]. Preconditioning is applied in the work by using an extension of the arbitrary equation of state preconditioning scheme developed by Merkle [10], as well as using cutoff methods developed by Darmofal and Siu [11] and Venkateswaran and Merkle [12].

By using a combination of these methods, many of the difficulties facing modern CFD methods can be addressed. This leads to the main objective of this work, to extend the applicability of high-order methods, specifically FC. Currently, FC has been limited by not being in a completely production ready state, meaning that it has yet to be applied to many potential simulations, such as incompressible flow. The main advantage of FC over other high-order methods is due to the ability of FC to obtain high-order with a simple correction term to the fluxes. This is advantageous because the basic method that FC is based off of is already in wide use in industry. Many of the same methods used, such as preconditioning and limiters, that are used in current industry applications can be applied to FC as well. Extension of FC to the incompressible flow equations and use of strand grid methods will greatly increase the usability of the high-order method. To achieve this, first, source term discretion must be understood. Next, the application of a preconditioner will extend the method to the incompressible domain. The result will be a more efficient method developed for high-order results in the incompressible domain.

In the following section we will survey the current literature that has been presented on topics that will be addressed in this work. Extra attention will be given to methods used in the preceding two chapters. After the literature survey, the objectives of this work will be outlined followed by two works developed by the author for publication. At the time of writing the first publication, "Source Term Discretization Effects on the Accuracy of Finite Volume Schemes," has been submitted to the Journal of Scientific Computing, and the second publication, "High-order Strand Grid Methods for Low Mach and Incompressible Flows" is targeted to be submitted to the International Journal for Numerical Methods in Fluids.

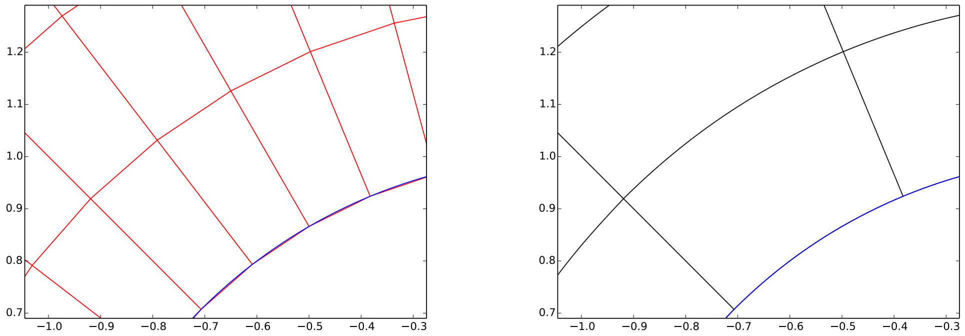
1.1 Literature Survey

1.1.1 Strand Meshing Methods

Strand mesh methods show potential for providing a solution to some of the most difficult problems facing modern CFD. As simulations continue to become larger and increase in complexity, mesh generation consumes more time. Currently it takes mesh experts days or weeks to generate an acceptable unstructured mesh quality. In addition, generation of high-order meshes has been proven to be ever more difficult as well. This has been mentioned as one of the most difficult issues still facing high-order methods' widespread application [1]. Strand methods provide the ability for automatic mesh generation that will greatly reduce the time needed to provide acceptable solutions.

To develop strand meshes, first the surface is tessellated. This tessellation can be triangles, quadrilaterals, or any other arbitrary polygon. Because the surface tessellation can be any arbitrary polygon, high-order meshes can be generated. High-order meshes better represent the body that is tessellated. For example in Figure 1.1 we see two strand meshes around a circle. Both represent the same number of unknowns at each node, however, the low-order mesh clips into the body, where the high-order mesh closely follows the surface.

Next, strands are emanated along normal vectors of the nodes of the polygons. These are then smoothed to ensure they do not cross in concave geometries or leave regions of sparse grids in convex geometries. Often times complexities with concave geometries and sharp corners can be alleviated with the combination of strand clipping and smoothing [2]. Nodes are then placed on the emanating strands. These nodes are commonly spaced using a hyperbolic tangent method as used by



(a) Low-order Mesh

(b) High-order Mesh

Fig. 1.1: Example of Low and High-order Strand Meshes

Katz et al. [3]. The spacing of the nodes is usually dependent on the Reynolds number, for example ensuring a proper mesh spacing perimeter of $y+ = 1$. This provides a one-dimensional structured distribution of nodes. Each level of nodes is connected to generate a complete surface mesh representation. Thus a three-dimensional simulation is reduced into a collection of two-dimensional simulations connected via source terms.

Strand methods also provide an advantage in memory storage and multi-processor applications. Because the meshes can be broken down into separate surfaces, each processor can have a copy of the entire global mesh, ensuring self-satisfying domain connectivity [13]. This is especially useful in moving mesh applications and large multi-processor solutions that are becoming increasingly more common as industry continues to expand into high-performance computing [4].

1.1.2 High-order Methods

A large amount of development in current research has been to improve high-order methods. High-order methods are defined as any solution method that reduces solution error by more than 1/4 for every time the mesh spacing is reduced by half. High-order methods are needed in many cases, such as vortex driven flow and acoustics. In such flows, second-order methods have been shown to be too dissipative and ineffective in capturing the complexity needed for analysis. Wang recently provided an overview of current high-order methods used in research [14], and collaborated with other experts in comparing efficiency of different methods [1]. Vincent and Jameson also have recently provided overviews of the practical use of high-order methods [15, 16]. An exhaustive overview of high-order methods is beyond the extent of this work, however a general overview of the field will be included as well as a brief background on FC.

General Overview

High-order methods are diverse, however, most high-order methods can be placed into one of three groups.

First, finite volume (FV) methods. It is common to break down FV into four distinct steps. First, the selected region is discretized into smaller regions. Next, the governing equations are integrated over all of the finite control volumes of the domain. Then, the resulting integration equations are discretized into a system of algebraic equations. Finally, the solution of the algebraic equations is achieved by use of an iterative method [17]. Among the methods of FV, some of the main high-order methods are K-Exact (k-E) [18, 19] and Essentially Non-Oscillatory and Weighted Essentially Non-Oscillatory (ENO/WENO) [20–22].

Next, finite element (FE) methods. First, the domain is discretized into smaller areas called 'elements.' This is much like the FV methods. The FE method uses a polynomial to approximate the solution inside the elements. Because of this, the order of accuracy is determined by the order of the polynomial used. Hence, the order of these polynomials must be determined prior to solving the Partial Differential Equation (PDE). Then the solution uses an iterative approach to find a solution to the integral form of the PDE [23]. Continuous Galerkin (CG) [24] and Discontinuous Galerkin (DG) [22] methods are high-order FE methods commonly used in research.

The final group is created by using a mixture of FE and FV methodology and ideas. These methods include Spectral Volume methods (SV) [25], Spectral Difference Methods (SD) [26], and Flux Reconstruction (FR) [27].

Flux Correction

Developed by Katz and Sankaran [4], and later extended to the Navier-Stokes equations [5], FC takes traditional FV methodology and incorporates truncation error analysis to improve the solution. The improvements can be added to commonly used methods, such as the second-order

Galerkin method, and upgrades them to high-order. Thus many methods developed for low-order methods, such as limiters and preconditioners, can be applied to FC as well. Unlike other methods, the order of accuracy is dependent on the accuracy of the gradient approximations.

1.1.3 Source Terms

Source terms are commonplace in CFD in both verification and real life solutions. Source terms are found in such equations as quasi-one-dimensional flow, MMS, shallow water, turbulence, and reactive flows to name a few. As an example of source terms, we can examine the one-dimensional heat equation,

$$q_t - kq_{xx} = f(x,t),$$

where q is the heat and f is an arbitrary source term. This could represent a heat addition in the center of a bar, for example. Mainly, what a source term can be defined as is an effect that makes the partial differential equation no longer homogeneous. Discretization of source terms is dependent on the solution method that will be used in addition to it. While in this work we focus mainly on spatial discretization, coupled space-time schemes have been developed as well, such as those by Toro and Titarev [28,29] and Dumbser et al. [30]. In addition, it has been shown to be advantageous at times to include an upwinding treatment as well to the source terms, such as has been done for shallow water equations [31–33]. These ideas are developed to create well-balanced schemes, and often require modifications to the approximate Riemann solvers for hyperbolic systems [34–39].

The overall goal of source term discretization is to reduce error and produce adequate solutions. One method that we employ in this work to discover how source terms do this is through truncation error analysis. In this work we will use truncation error analysis to determine the ‘correctness’ of a source term with a specified flux discretization method.

1.1.4 Preconditioning

Preconditioning is used to reduce the stiffness of PDEs. This is completed by modifications to the eigenvalues of the equations. For example, for the Euler equations it can be found that the eigenvalues are based off of the wave speeds and the particle velocity. For flows that approach

$Mach = 0$ the eigenvalues create large disparities. To counteract this, a preconditioning scheme is added to reduce this disparity. Preconditioners have been shown to also increase convergence rates. Next we will give a general overview of basic preconditioning techniques, followed by a more optimized version that will be implemented.

General Overview

For a general overview of preconditioning techniques, we will be following the description used by Turkel [40]. Generally, preconditioning is used in one of two ways. The first way is in a global sense, the second way is in a local sense. To aid in the discussion, we will first observe the equation

$$\Gamma \mathbf{A}\mathbf{v} = \Gamma \mathbf{S}, \tag{1.1}$$

where Γ is the preconditioning matrix, $\mathbf{A}$ is the discretization, $\mathbf{v}$ is the conserved variables, and S is the source term.

Because of the need to frequently invert the preconditioning matrix, it is advantageous to choose a matrix that is easily invertible. From this several ideas have been presented, first, create a preconditioning matrix that only has values on its diagonal. The advantage of this is the ease of invertibility and the fact that it only affects a single point. This type of preconditioning is commonly referred to as a Jacobi preconditioner and is mainly used to increase convergence rates. Next, the choice of the preconditioning matrix, or the inverse of the preconditioning matrix, has non-zero elements in the same positions as the non-zero elements of the discretization matrix. Finally, ones that don't fit in either category. These are generally global preconditioners, or preconditioners that affect the entire solution rather than a point by point solution.

These preconditioning matrices are developed with one of two main ideas in mind. First, base the preconditioner on a simpler set of equations. Second, the preconditioner is based off of the algebraic properties of the discretization matrix without account of the connection between the discretization and the differential equation being solved. This second method often leads to a global preconditioning scheme.

Because of the many different ideas behind preconditioning, there exist many different schemes, each having its advantages and disadvantages. In general, a preconditioning matrix takes the form of

$$\Gamma = \begin{pmatrix} \frac{1}{\beta^2} & 0 & 0 & \delta \\ \frac{\alpha u}{a\beta^2} & 1 & 0 & 0 \\ \frac{\alpha v}{a\beta^2} & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad \Gamma^{-1} = \begin{pmatrix} \beta^2 & 0 & 0 & -\beta^2\delta \\ -\frac{\alpha u}{a} & 1 & 0 & \frac{\alpha u}{a}\delta \\ -\frac{\alpha v}{a} & 0 & 1 & \frac{\alpha v}{a}\delta \\ 0 & 0 & 0 & 1 \end{pmatrix}, \tag{1.2}$$

where β, α, and δ are parameters to be chosen, u and v are velocity in the x and y direction respectively, and a is a constant. In order to achieve a bounded answer as Mach approaches zero, it is necessary to choose β proportional to the Mach number [40].

From Equation 1.2 we can derive the preconditioning matrices defined by Choi and Merkle [41] where $\alpha = 0$ and $\delta = 1$. This has the disadvantage that it has a discontinuity at $\beta = 1$. Weiss and Smith suggested a preconditioner where $\alpha = 0$ and $\delta = 0$ [42]. Finally, Merkle later suggested a preconditioner that depends on the local Reynolds number Re_Δ and Prandtl number Pr [10]. This formulation, however, does not match the general scheme presented in Equation 1.2.

One challenge of preconditioning is that all preconditioners tend to become singular near stagnation points or as Mach approaches zero in the boundary layer [43]. Because of this difficulty, many preconditioners must be reduced or removed in the boundary layer and near stagnation points. Terkel suggested that this cutoff should be global, however this presents difficulties in the definition of the reference Mach number and what constant to use to properly define the cut-off [44]. Darmofal and Siu suggested a different approach of determining the cut-off based on pressure gradients [11]. Their approach does account for stagnation points, however, it introduces a variable to ensure proper cut-off that can be problem dependent and possibly decrease convergence. Venkateswaran and Merkle also defined a cut-off method for the Navier-Stokes equations that is dependent on the viscosity [12].

Optimization

For use in this work we modify the preconditioning scheme of arbitrary equation of states presented by Merkle [10] that includes the artificial compressibility constant developed by Chorin [9]. We also include the pressure dependent cut-off ratio presented by Darmofal and Siu [11]. The difference of this optimal method is that it is dependent on the optimal definition of the preconditioning Mach number and a preconditioned velocity term. The preconditioned velocity term is defined as

$$V_p = \min\left(\max\left(\sqrt{u^2+v^2}, \sqrt{\frac{|\Delta p|}{\rho}}, \frac{\nu_t}{\Delta x}\right), c\right), \tag{1.3}$$

where p is the pressure, ρ is the density, ν_t is the turbulent viscosity term, and c is the speed of sound. This preconditioned velocity contains the cut-off ratio defined by Darmofal and Siu [11] as well as the boundary layer cut-off for the viscous Navier-Stokes equations defined by Venkateswaran and Merkle [12].

From Equation 1.3, β in Equation 1.2 is defined. Using a common method used by other researchers [43] we define the preconditioning upper cut-off as Mach = .5. This ensures that preconditioning does not ruin the solution around the sonic point, or when Mach approaches one. Using this more optimal definition of preconditioning, the preconditioner should be more stable and applicable over a larger range. This preconditioner also has the advantage that it can easily be extended to multiple types of other equations of state. Preconditioners can also include the artificial compressibility term, used to relax the equations being solved for high eigenvalue disparity regions, thus extend compressible solvers to the incompressible domain.

1.1.5 Turbulence

Turbulence modelling is a large field in CFD. Because of the complexities in modelling a turbulent boundary layer, many different ideas and methodologies have been introduced to achieve accurate results. The best turbulence model often is determined on a problem by problem basis. Here a general overview of different groups of turbulence models will be discussed following the description of Versteeg and Malalasekera [17]. Spalart-Allmaras (SA) turbulence model that is used in this work [45] will be discussed afterwards.

General Overview

One of the more common methods of turbulence models is using the Reynolds averaged Navier-Stokes (RANS) equations. The RANS methods depend on an assumption that the turbulent values fluctuate around a mean value. Before solving, the Navier-Stokes equations are time averaged. Because of this, extra terms are introduced in the flow equations due to the interaction between various turbulent fluctuations. These extra variables are then modelled separately. This methodology includes the k-ε and the Reynolds Stress methods. Using this assumption computational cost can be greatly reduced.

Next, large eddy simulation (LES). With LES the mesh size is greatly reduced in the boundary layer with attempt of more accurately modelling turbulent flow. The mesh size is reduced such that large eddies are captured, meanwhile, small eddies are rejected. To achieve this, a space filtering of the unsteady Navier-Stokes equations is used prior to the computations. The smallest eddies are then included using a sub-grid scale model. This method requires the unsteady Navier-Stokes equations to be solved, so computational costs are higher than the RANS methods. LES is commonly used for determining boundary layers on complex geometries because of its ability to resolve the boundary layer eddies to a better extent.

Direct numerical simulation (DNS) is the most accurate and most time-consuming method. The goal of DNS is to capture the mean velocity values and all turbulent velocity fluctuations. The unsteady Navier-Stokes equations are solved on a spatial grid small enough to capture all eddies with a time step small enough to model the fastest fluctuations. This method is highly expensive computationally. Because of this, this method is not commonly used in industry.

Spalart-Allmaras Model

The Spalart-Allmaras (SA) turbulence model is an improvement on the $k-\varepsilon$ mode [17, 45]. The goal of SA is to reduce the two-equation model into a single equation. This is completed by use of a user defined length scale. This definition has both advantages and disadvantages. The advantage is that this provides a computationally efficient method where only one equation is included to solve for the turbulent viscosity. The disadvantage is, for complex geometries, it is often difficult to define a length scale. SA has been shown to be advantageous in airfoil applications and in flows with adverse pressure gradients, such as a stalled airfoil [17].

There have been few changes in the last years since the turbulence model was introduced [45].

Comparisons between models and the continued applicability of SA was discussed by Spalart [46] in 2000. In addition, work has been done by Allmaras et al. in extending the flow to produce better results for under-resolved grids [47].

1.2 Overview

The following chapters contain two publications discussing FC in connection with source terms, strand grids and preconditioning. In each chapter an introduction is included to provide the reader with relevant background in the subject discussed in the chapter. Each chapter also contains results and conclusions on the research described therein. The first chapter discusses the necessity of proper source term treatment. In Chapter two a preconditioning treatment is discussed, extending the usability of FC and strand grids to low-Mach and incompressible flows. Finally, conclusions on the overall research will be covered as well as future research directions.

Chapter 2
Source Term Discretization Effects on the Accuracy of Finite Volume Schemes[1]

Jonathan Thorne, Aaron Katz[2]

2.1 Introduction

Computational Fluid Dynamics (CFD) has found many areas of application in modern engineering. Currently, CFD is used for calculating complex flow problems with such features as compressibility, viscosity, turbulence, and chemical reactions. Many of these features are mathematically represented by source terms appearing in the governing partial differential equations (PDE). These source terms are often as complex or more complex in nature than derivative terms [6–8] and sometimes require specialized treatment to ensure accurate and stable discretizations [30, 34]. Importantly, source terms also appear in code verification via the method of manufactured solutions (MMS) [48, 49]. Because source terms comprise an essential role in so many applications, and because of the widespread use of finite volume schemes for CFD, it is crucial to understand the compatibility of source discretization schemes with respect to the other aspects of finite volume (FV) discretizations.

In this paper, we employ MMS to explore the compatibility of various source term discretization schemes with various FV flux schemes. However, to verify the order of accuracy of the FV flux schemes alone, with no source terms, we first use the method of exact solutions (MES). The primary purpose of MES in this work is to create a baseline expectation of accuracy. In this way, we are able to directly measure the impact on accuracy of choosing one source term discretization over another. We should note that our past work has made extensive use of both MES and MMS to verify the FV flux methodology discussed in this work, and we refer the reader to that work [4].

[1] Submitted to the Journal of Scientific Computing

[2] Associate Professor, Department of Mechanical and Aerospace Engineering, Utah State University, Logan, UT

We also note that the conclusions drawn in this work stem mainly from observations using MMS because we can ensure that all terms in the governing PDE are exercised, including flux and source terms [49–52].

Because of our reliance on MMS, we briefly review the background of the approach. In short, MMS is the process whereby a solution to a PDE is created *a priori* followed by the addition of a source term which forces the chosen solution to satisfy the PDE (with added source term). Roache provides examples of MMS source terms for several different equation sets [49]. Often, MMS is used in conjunction with a grid refinement study to find the order of accuracy of a particular discretization scheme [49, 51], but it can also be used to understand other effects, such as mesh quality [53, 54], boundary treatments [55], and other mesh features [2]. In the context of an order of accuracy study, if the computed order of accuracy using MMS (with source terms) matches the order of accuracy using MES (without source terms) for a particular FV discretization, then it can be concluded that the particular source term discretization is *compatible*. Therefore, MMS provides a convenient way to identify compatible source term discretization schemes. These results hold not only for source terms generated by the MMS procedure, but also extend to source terms representing physical behavior, such as in turbulent or reacting flows.

The general philosophy we adopt to determine compatibility is to identify the simplest source treatment that preserves accuracy. As we discuss in this work, some source term treatments are non-intuitive. But if a source treatment lacks the required sophistication, then lower order results will inevitably be observed in the grid refinement study, regardless of the sophistication of the overall FV flux scheme. On the other hand, if the source term is unnecessarily sophisticated then the method will require unnecessary computation time. Thus, the simplest source discretization that preserves accuracy is desired.

While the main objective of this paper is to identify compatible source term discretizations for FV schemes, a secondary objective is to examine a relatively new third-order FV flux scheme, known as flux correction (FC), first developed by Katz and Sankaran [4]. This scheme has been shown to obtain third-order accuracy for general non-linear hyperbolic systems of equations, on arbitrary triangular grids. Remarkably, the FC scheme provides increased accuracy without the need for flux quadrature or second derivatives in the reconstruction procedure, such as is needed in other high-order schemes [5, 56]. In this work, we focus on obtaining compatible source discretizations for the FC scheme.

While focusing on compatibility in terms of spatial truncation error for FV schemes, we also acknowledge other important contributions dealing with source term treatments in CFD. In particular, it may be advantageous to consider coupled space-time approaches, such as those of Toro and Titarev [28, 29] and Dumbser et al. [30], which make use of high-order numerical integration of source terms in space and time. Also, other works highlight the need for upwind treatments of source terms for certain equations, such as the shallow water equations [31–33]. These ideas are related to the notion of well-balanced schemes involving modifications to approximate Riemann solvers for hyperbolic systems with source terms [34, 57–61]. While the scope of this paper does not include these concepts directly, future work will focus on understanding our methods in these contexts as well.

The paper is outlined as follows: first, we present three FV flux schemes with their truncation error analysis: constant reconstruction, linear reconstruction, and a third-order FC scheme. Next, three candidate source term discretizations are explored with their corresponding truncation error analysis: point, Galerkin, and a new corrected scheme designed for compatibility with the third-order FC scheme. Then we show the truncation error analysis with combinations of flux and source schemes. Following this, several computational studies are presented using MES (Ringleb flow), MMS, and quasi-one-dimensional flow containing source terms. Finally, conclusions are drawn regarding the compatibility of each source term discretization with each flux scheme.

2.2 Finite Volume Flux Schemes

We first discuss three different flux schemes based on the finite volume method, wherein the

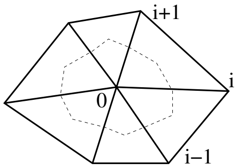


Fig. 2.1: Node-centered finite volume

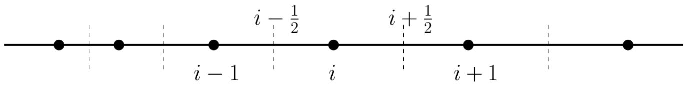


Fig. 2.2: Node-centered grid in one dimension

domain is broken into a number of small volumes or elements. Here, we focus on node-centered volumes, while other configurations based on cell-centered or hybrid schemes are certainly possible. An example of a node-centered FV can be seen in Figure 2.1, where the flux is calculated across the dashed lines representing the median-dual edges of the cell. These dashed lines enclose a small FV with center point 0, surrounded by a number of nearest neighbors, i. The complete derivation of many of the discretization schemes used in this work are found elsewhere [4, 62], but a brief derivation will be given in this section. In addition, truncation error will be calculated to predict the solution error of the schemes. Truncation error will be completed in one-dimension for the constant and linear schemes, and both one and multiple dimensions for the FC method.

We first consider a one dimensional wave equation of the following form,

$$Q_t + F_x = S(x), \tag{2.1}$$

where Q represents the conserved solution variables, F is the one-dimensional flux term, and $S(x)$ is a known source term discretization on a finite set of node-centered cells in one dimension, shown in Figure 2.2. With this discretization we can define the following mesh spacings:

$$\Delta x_i = \frac{1}{2}\left(\Delta x_{i-1/2} + \Delta x_{i+1/2}\right), \quad \Delta x_{i+1/2} = x_{i+1} - x_i. \tag{2.2}$$

Examining the steady-state solution of Equation 2.1, a family of solutions can be formed,

$$F^h_{x,i} = \frac{1}{\Delta x_i}\left(F^h_{1+1/2} - F^h_{i-1/2}\right), \tag{2.3}$$

where the superscript h denotes a discretized approximation. $F^h_{x,i}$ is the numerical flux defining the scheme that is in use in one-dimension.

Next, we consider a conservation law of the following form for multiple dimensions,

$$Q_t + \nabla \cdot \mathbf{F} = S \tag{2.4}$$

where $\mathbf{F}$ is the flux vector in multi-spatial dimensions, and S is the source term. A finite volume discretization of the flux divergence at node 0 is,

$$\nabla^h \cdot \mathbf{F}^h = \frac{1}{V_0} \sum \mathscr{F}^h_{0i} \mathscr{S}_{0i}, \tag{2.5}$$

$\mathscr{S}_{0i}$ is the area of the face separating nodes 0 and i, and $\mathscr{F}^h_{0i}$ is a numerical flux between nodes 0 and i with

$$\mathscr{F} = \mathbf{F} \cdot \mathbf{n}, \tag{2.6}$$

where $\mathbf{n}$ is the normal vector of the volume face.

In this work we consider three different treatments of the numerical flux: constant reconstruction, linear reconstruction, and flux correction. These methods and their corresponding truncation error analysis are described in the subsections that follow.

2.2.1 Constant Reconstruction

For the constant reconstruction scheme in one dimension, the flux at $i+1/2$, and likewise the flux at $i-1/2$ can be calculated as

$$F^h_{i+1/2} = \frac{1}{2}\left(F^h_i + F^h_{i+1}\right) - \frac{1}{2}\left|A^h\left(Q_{R,i+1/2}, Q_{L,i+1/2}\right)\right| D_{i+1/2} \tag{2.7}$$

where $A = dF/dQ$ is the flux Jacobian and is a function of the state variables and contains the absolute values of the eigenvalues enforcing unwinding. Next, D is the artificial dissipation given by

$$D_{i+1/2} = Q_{R,i+1/2} - Q_{L,i+1/2}, \tag{2.8}$$

where Q_R and Q_L are the right and left states of the conserved variables, for constant reconstruction these are taken as

$$Q_{R,i+1/2} = Q_{i+1}, \quad Q_{L,i+1/2} = Q_i. \tag{2.9}$$

For the constant reconstruction method in multiple dimensions, the numerical flux across the boundary is calculated as

$$\mathscr{F}^h_{0i} = \frac{1}{2}\left(\mathscr{F}_0 + \mathscr{F}_i\right) - \frac{1}{2}\left|\mathscr{A}\left(Q_0, Q_i\right)\right|\left(Q_i - Q_0\right) \tag{2.10}$$

where $\mathscr{A} = \partial \mathscr{F}/\partial Q$ is the directed flux Jacobian, computed from two flow states via the method of Roe [63]. Here,

$$|\mathscr{A}| = M|\Lambda|M^{-1}, \tag{2.11}$$

where M contains the eigenvectors of $\mathscr{A}$, and $|\Lambda|$ contains the absolute value of the eigenvalues. The flux terms involving $\mathscr{F}$ produce the flux integral, while the $|\mathscr{A}|$ term produces artificial dissipation added for upwind stability.

Truncation error analysis in one-dimension is begun by substituting Equation 2.7 into Equation 2.3,

$$F^h_{x,i}(Q) = \frac{1}{2\Delta x_i}\left(F_{i+1} - F_{i-1}\right) - \frac{1}{2\Delta x_i}\left(D_{i+1/2} - D_{i-1/2}\right). \tag{2.12}$$

It is convenient to analyze the dissipation terms and flux terms separately. Beginning with the dissipation, by using Taylor series to expand Equation 2.12, and using the left and right states defined in Equation 2.9, we find the truncation error to be at steady state:

$$\frac{1}{2\Delta x_i}\left(D_{i+1/2} - D_{i-1/2}\right) = \frac{1}{4\Delta x_i}\left(\left|A_{i+1/2}\right|\Delta x_{i+1} - \left|A_{i-1/2}\right|\Delta x_{i-1/2}\right) Q_{x,i} + O\left(h^1\right). \tag{2.13}$$

The result is a leading truncation error of order zero for arbitrary meshes, however, it can be shown that there are instances in where $A_{i+1/2} = A_{i-1/2}$. Because of this the zero-order term cancels on regular grids, producing a method of $O(h^1)$. This truncation error comes directly from the right and left states.

Continuing with the flux scheme, using a Taylor series expansion of the flux scheme in Equation 2.7 and substituting it into Equation 2.12 we obtain

$$\frac{1}{2\Delta x_i}\left(F_{i+1}-F_{i-1}\right)=F_{x,i}+\frac{1}{2}\left(\Delta x_{i+1/2}-\Delta x_{i-1/2}\right)F_{2x,i}+ \frac{1}{12\Delta x_i}\left(\Delta x^3_{i+1/2}+\Delta x^3_{i-1/2}\right)F_{3x,i}+O\left(h^3\right). \quad (2.14)$$

Which has a leading truncation error of $O(h^1)$ for arbitrary grids and $O(h^2)$ for regular grids.

Using truncation error analysis the leading term of error occurs in the dissipation from the left and right state assumption. This creates a leading truncation error in the dissipation of zero-order. Constant reconstruction is not commonly used due to the large amount of resulting artificial dissipation, as shown in the truncation error analysis. However, it is included in this study for completeness. This scheme produces first-order results, and may be useful occasionally in practice where extra robustness is required, such as at shocks are regions of poor grid quality.

2.2.2 Linear Reconstruction

In the linear reconstruction scheme, the first-order dissipation terms are modified, producing a formally second-order accurate scheme. This is accomplished via the use of left and right reconstructed states in the dissipation. For one-dimension, the left and right states are changed to

$$\begin{aligned} Q^R_{i+1/2} &= Q_{i+1}+\frac{1}{2}\Delta x_{i+1/2}Q^h_{x,i}, \\ Q^L_{i+1/2} &= Q_i-\frac{1}{2}\Delta x_{i+1/2}Q^h_{x,i+1}. \end{aligned} \quad (2.15)$$

Where the derivative must be at least first-order accurate.

For multiple dimensions, Equation 2.10 is modified to become

$$\mathscr{F}^h_{0i}=\frac{1}{2}\left(\mathscr{F}_0+\mathscr{F}_i\right)-\frac{1}{2}\left|\mathscr{A}\left(Q_L,Q_R\right)\right|\left(Q_R-Q_L\right), \quad (2.16)$$

where the left and right states are constructed as follows:

$$Q_L=Q_0+\frac{1}{2}\Delta\mathbf{r}^T\nabla^h Q_0, \quad Q_R=Q_i-\frac{1}{2}\Delta\mathbf{r}^T\nabla^h Q_i, \quad (2.17)$$

where **r** is the position vector connecting node 0 to node i. The gradient is discretized with a p^{th}-order accurate method, such that

$$\nabla^h = \nabla + O(h^p), \tag{2.18}$$

where $O(h^p)$ denotes the truncation error of the gradient procedure. It is most common in practice to employ methods for which $p = 1$, such as linear least squares or Green-Gauss integration [64].

Truncation error analysis for the flux terms remains the same, however, the change in the dissipation routine changes the resultant truncation error to

$$\begin{aligned}\frac{1}{2\Delta x_i}\left(D_{i+1/2} - D_{i-1/2}\right) &= \\ &- \frac{1}{48\Delta x_i}\left(\Delta x^3_{i+1/2}\left|A_{i+1/2}\right| - \Delta x^3_{i-1/2}\left|A_{i-1/2}\right|\right) Q_{3x,i} + O(h^3) + O(h^p)\end{aligned} \tag{2.19}$$

where $O(h^p)$ is determined from the derivative method used. One common method for one-dimensional analysis is to use is central difference, this leads to a leading truncation error of $O(h^1)$. Setting aside the derivative error, it appears as if the leading truncation error term is $O(h^2)$ for regular grids, however a more detailed analysis shows that the leading term is actually $O(h^3)$. The effects of this will be shown later in the results section. The derivative method tends to be the leading cause of error on arbitrary meshes, but the truncation error magnitude is no worse than the truncation error of the fluxes.

The result of using the reconstructed states is a scheme with greatly reduced artificial dissipation. This is one of the most commonly used methods for practical CFD applications.

2.2.3 Flux Correction

To produce a third-order accurate scheme, Katz and Sankaran developed a novel method to modify Equation 2.16 to further cancel leading truncation error terms [4]. The method was first proposed for hyperbolic systems of equations, but was later extended to the Navier-Stokes equations [65]. Termed flux correction (FC), the only change to Equation 2.7 is the direct reconstruction of the non-linear flux, resulting in

$$F^h_{i+1/2} = \frac{1}{2}\left(F_{L,i+1/2} + F_{R,i+1/2}\right) - \frac{1}{2}\left|A^h_{i+1/2}\left(Q_{R,i+1/2}, Q_{L,i+1/2}\right)\right|\left(Q_{R,i+1/2} - Q_{L,i+1/2}\right). \quad (2.20)$$

Where the left and right flux states are defined as

$$F_{R,i+1/2} = F^h_i + \frac{1}{2}\Delta x_{i+1/2}F^h_{x,i}, \quad F_{L,i+1/2} = F^h_{i+1/2} - \frac{1}{2}\Delta x_{i+1/2}F^h_{x,i+1} \quad (2.21)$$

Note that the actual flux is reconstructed not the dependent variables. The flux and dissipation fluxes, however, must be at least second-order to maintain proper truncation error.

In multiple dimensions, the only change is to Equation 2.16,

$$\mathscr{F}^h_{0i} = \frac{1}{2}\left(\mathscr{F}_L + \mathscr{F}_R\right) - \frac{1}{2}\left|\mathscr{A}\left(Q_L, Q_R\right)\right|\left(Q_R - Q_L\right), \quad (2.22)$$

with the flux being directly reconstructed as follows:

$$\mathscr{F}_L = \mathscr{F}_0 + \frac{1}{2}\Delta\mathbf{r}^T\nabla^h\mathscr{F}_0, \quad \mathscr{F}_R = \mathscr{F}_i - \frac{1}{2}\Delta\mathbf{r}^T\nabla^h\mathscr{F}_i \quad (2.23)$$

Once again, unlike the linear reconstruction scheme, the order of accuracy of the flux and solution gradients must be at least second-order ($p \geq 2$) to maintain consistency.

One dimensional truncation error on the dissipation shows the same results as Equation 2.19, only the introduction of a derivative method of second-order truncation error reduces the leading error to $O(h^2)$ for both arbitrary and regular meshes. The leading flux truncation error is reduced to

$$\frac{1}{\Delta x_i}\left[\frac{1}{2}\left(F_{L,i+1/2} + F_{R,i+1/2}\right) - \frac{1}{2}\left(F_{L,i-1/2} + F_{R,i-1/2}\right)\right] = \\ F_{x,i} - \frac{1}{24\Delta x_i}\left(\Delta x^3_{i+1/2} + \Delta x^3_{i-1/2}\right)F_{3x,i} + O\left(h^3\right) + O\left(h^p\right). \quad (2.24)$$

This is also at least $O(h^2)$ as long as $p \geq 2$.

The advantage of the FC method over other high-order methods is that it bypasses the need for second derivatives in the reconstruction. In addition, it also does not require high-order quadrature.

This makes the method an efficient way to achieve third-order accuracy without excess computational expense or code modification.

2.3 Source Term Discretization

The previous section described methods of approximating the flux divergence, $\nabla \cdot \mathbf{F}$, with three different flux integration schemes. The major goal of this paper is to identify or develop discretely compatible source term treatments for these flux schemes. Three different source term discretization schemes are considered in this work, each with different levels of complexity, and each designed for compatibility with the three flux schemes. Each source discretization scheme below represents the spatial integral of S in Equation 2.4.

2.3.1 Point Source

For the point source scheme, the source term is assumed to be constant over the FV, such that

$$S_i^h = S_i. \tag{2.25}$$

For multiple dimensions, the source term is assumed to be a constant over each node-centered FV cell. The integral of S over each cell then becomes simply

$$S_0^h = V_0 S_0, \tag{2.26}$$

where S_0 and V_0 are the source term and volume evaluated at node 0, respectively.

Because the source is assumed to be known exactly, there is no error introduced due to the source term. Because of this, no truncation error analysis is given. Having no error introduced by the source term sounds optimal, however, without the introduction of error, further truncation errors in the flux routine cannot be removed. This leads to the continuation of truncation error that could otherwise be reduced. This effect will be analyzed at the end of this section.

This is the simplest source term discretization considered in this work, and is commonly used in practical CFD codes.

2.3.2 Galerkin

Because the FV flux scheme of Equation 2.16 can be interpreted as a Galerkin finite element scheme on linear triangles [66], we also consider consistent Galerkin treatment of source terms. Note that the Galerkin interpretation only extends to the central portion of the numerical flux and not the artificial dissipation portion. A full version of Galerkin source term derivation is found elsewhere [5], but is briefly described here. For one dimension, the source is assumed to very linearly through the area. This leads to unique weighting of the source term,

$$S_i^h = \frac{2}{3}S_i + \frac{1}{6\Delta x_i}\left(S_{i-1}\Delta x_{i-1/2} + S_{i+1}\Delta x_{i+1/2}\right). \tag{2.27}$$

For multiple dimensions, following the standard Galerkin approach, it is assumed that the source term varies linearly in each cell, followed by the multiplication by a linearly varying test function and integration over each triangular cell volume. At point 0 surrounded by nearest neighbor nodes i in the triangulation, the resulting contribution is,

$$S_0^h = \sum_i \frac{1}{2}\left(S_0 + S_i\right)V_{0i}, \tag{2.28}$$

where V_{0i} represents a volume associated with edge connecting nodes 0 and i and is computed as

$$V_{0i} = \frac{1}{4}\Delta\mathbf{r}^T\mathbf{n}\mathscr{S}_{0i}, \tag{2.29}$$

where $\mathbf{n}\mathscr{S}_{0i}$ is the median-dual face area vector associated with the edge.

Truncation error analysis shows that error has been added to the source term scheme. Expanding Equation 2.27,

$$S_i^h = S_i + \frac{1}{3}\left(\Delta x_{i-1/2} - \Delta x_{i+1/2}\right)S_{x,i} + \frac{1}{12\Delta x_i}\left(\Delta x_{i-1/2}^3 + \Delta x_{i+1/2}^3\right)S_{2x,i} + O\left(h^3\right). \tag{2.30}$$

The resultant truncation error is order $O(h^1)$. The goal is to use this source to reduce errors introduce by the fluxes.

Because of the analogy with the linear Galerkin scheme, the expectation is that this source treatment will be compatible with the linear reconstruction scheme.

2.3.3 Corrected

The corrected source treatment is designed for compatibility with the third-order FC flux scheme described in Section 2.2.3. An objective of this work is to demonstrate this compatibility numerically. Because the FC flux scheme involves a correction to the linear reconstruction flux scheme, we follow the same approach with the source treatment.

Starting with one dimension, we begin by adding a modification to the Galerkin source in Equation 2.27:

$$S_i^h = \frac{1}{3}S + \frac{1}{3\Delta x_i}\left(\Delta x_{i+1/2}S_{i+1/2} + \Delta x_{i-1/2}S_{i-1/2}\right), \tag{2.31}$$

with $S_{i+1/2}$ and likewise $S_{i-1/2}$ defined as

$$S_{i+1/2} = \frac{1}{2}\left(S_L + S_R\right)_{i+1/2},$$

and

$$\begin{aligned} S_{L,i+1/2} &= S_i - \frac{1}{2}\Delta x_{i+1/2}S_{x,i} - \frac{1}{8}\Delta x_{i+1/2}^2 S_{2x,i}, \\ S_{R,i+1/2} &= S_{i+1} - \frac{1}{2}\Delta x_{i+1/2}S_{x,i+1} - \frac{1}{8}\Delta x_{i+1/2}^2 S_{2x,i+1}. \end{aligned}$$

The higher order derivatives are needed to cancel out higher order truncation error terms.

For multiple dimensions we begin with the linear Galerkin source treatment in Equation 2.28 and modify it slightly to obtain

$$S_0^h = \sum_i \frac{1}{2}\left(S_L + S_R\right)V_{0i}. \tag{2.32}$$

Here, left and right source approximations, S_L and S_R, are defined as

$$S_L = S_0 - \frac{1}{2}\Delta\mathbf{r}^T\nabla^h S_0 - \frac{1}{8}\Delta\mathbf{r}^T\nabla^{h2}S_0\Delta\mathbf{r}, \quad S_R = S_i - \frac{1}{2}\Delta\mathbf{r}^T\nabla^h S_i - \frac{1}{8}\Delta\mathbf{r}^T\nabla^{h2}S_i\Delta\mathbf{r}, \tag{2.33}$$

where ∇^{h2} is the discrete Hessian operator, computed to $(p-1)^{th}$-order accuracy as

$$\nabla^{h2} S_i = \nabla^2 S + O\left(h^{p-1}\right). \tag{2.34}$$

As in the FC flux scheme, we require $p \geq 2$. Note that unlike the FC flux treatment, the source treatment here involves second derivatives of S. However, high-order numerical integration of the source term over each finite volume is not necessary.

Truncation error analysis provides insight to why the source term was constructed in the method it was. Expanding Equation 2.31 using Taylor series we obtain,

$$S_i^h = S_i - \frac{1}{24\Delta x_i}\left(\Delta x_{i+1/2}^3 + \Delta x_{i-1/2}^3\right) S_{2x,i} + O\left(h^3\right). \tag{2.35}$$

We have successfully removed all first-order truncation error, making the only the second derivative of the source term necessary.

We note that using truncation error as the criteria by which to design aspects of FV schemes is rather unique, and led to the original FC scheme of Katz and Sankaran in the first place [4].

2.3.4 Combined Source and Flux Truncation Error

Truncation error analysis has been completed on all the flux schemes and source methods. Although the results of both help us make interesting conclusions, it is also necessary to observe how each flux scheme and source method work to reduce overall truncation error. With this in mind we will first start by using the linear reconstruction flux scheme and combine it with the point source scheme. Next, we will use the linear reconstruction scheme in combination with the Galerkin source term. After, we will combine the FC scheme with the Galerkin source term, then the corrected source term. Analysis with the constant reconstruction is not completed, as the leading error is in the dissipation routine at $O(h^0)$, thus any change to the source term will not affect the leading truncation error. Combination of the FC scheme and point source, as well as the linear reconstruction with the corrected source, are also not presented. This is because the source term would not affect the leading truncation error as well.

For understanding, we will start with the steady-state one-dimensional wave equation for truncation error,

$$F_{x,i}^{h}(Q) - S_i^h = F_{x,i} - S_i + E_t, \tag{2.36}$$

where E_t denotes the truncation error terms.

First, we will combine the linear reconstruction with the point source term. Starting with Equation 2.36 then substituting in the linear reconstruction truncation error, Equations 2.14 and 2.19, and the point source error, Equation 2.25, we obtain,

$$\begin{aligned} E_t =& \frac{1}{2}\left(\Delta x_{i+1/2} - \Delta x_{i+1/2}\right) F_{2x,i} + \frac{1}{12\Delta x_i}\left(\Delta x_{i+1/2}^3 - \Delta x_{i-1/2}^3\right) F_{3x,i} \\ & - \frac{1}{48\Delta x_i}\left(\Delta x_{i+1/2}^3 \left|A_{i-1/2}\right| - \Delta x_{i-1/2}^3 \left|A_{i+1/2}\right|\right) Q_{3x,i} + O\left(h^p\right) + O\left(h^3\right). \end{aligned}$$

On arbitrary grids, the leading truncation error is $O(h^1)$ due to the flux approximation, as well as the gradient approximation. When regular meshes are used, the first-order truncation error and the error in the derivative, assuming central difference was used, also reduces to $O(h^2)$, leading to that truncation error to be the leading cause, as well as the uncanceled flux term. The difference between both regular and arbitrary meshes will be analyzed to see how the truncation error affects the solution error.

Next, we will use linear reconstruction with the Galerkin source term. Evaluating Equation 2.36 using Equations 2.14, 2.19 , and 2.30 we find

$$\begin{aligned} E_t = & \left(\Delta x_{i+1/2} - \Delta x_{i-1/2}\right)\left(\frac{1}{2}F_{2x,i} - \frac{1}{3}S_{x,i}\right) - \\ & \frac{1}{48\Delta x_i}\left(\Delta x_{i+1/2}^3 \left|A_{i-1/2}\right| - \Delta x_{i-1/2}^3 \left|A_{i+1/2}\right|\right) Q_{3x,i} + O\left(h^p\right) + O\left(h^3\right). \end{aligned}$$

For arbitrary meshes the leading truncation error term is still $O(h^1)$, however, all second-order truncation error due to the flux has been removed due to the source term. Notice that we used the relation $F_{2x} = S_{3x}$ exactly. Due to this the truncation error has been reduced to $O(h^2)$ on regular meshes due to the dissipation term.

Continuing with the FC scheme with the Galerkin source term. Substituting Equations 2.24, Equation 2.19, where $p \geq 2$, and Equation 2.30 into Equation 2.36 we obtain

$$\begin{aligned} E_t =& \frac{1}{3}\left(\Delta x_{i-1/2} - \Delta x_{i+1/2}\right) S_{x,i} - \\ & \frac{1}{\Delta x_i}\left(\Delta x_{i+1/2}^3 + \Delta x_{i-1/2}^3\right)\left(\frac{1}{24}F_{3x,i} + \frac{1}{12}S_{2x,i}\right) - \\ & \frac{1}{48\Delta x_i}\left(\Delta x_{i+1/2}^3 \left|A_{i-1/2}\right| - \Delta x_{i-1/2}^3 \left|A_{i+1/2}\right|\right) Q_{3x,i} + O\left(h^3\right) + O\left(h^p\right). \end{aligned}$$

The result is a method that is $O(h^1)$ due to the source term for arbitrary spacing, and $O(h^2)$ due to the flux routine and the source on regular meshes. Because of the mismatch of the source term and flux schemes, error was added to the overall routine rather then being reduced.

Finally we will use the FC scheme with the corrected source term. Using Equations 2.24, 2.19, and 2.35 into Equation 2.36,

$$E_t = \frac{1}{48\Delta x_i}\left(\Delta x_{i+1/2}^3 \left|A_{i-1/2}\right| - \Delta x_{i-1/2}^3 \left|A_{i+1/2}\right|\right) Q_{3x,i} + O\left(h^3\right) + O\left(h^p\right).$$

Using the corrected source term effectively reduced the truncation error on arbitrary grids and regular grids to $O(h^3)$, assuming that $p \geq 3$.

To continue, in multiple dimensions the truncation error can be found by

$$\nabla \cdot \mathscr{F} - S_0 = \nabla^h \cdot \mathscr{F} - S_0^h + E_t$$

Using the truncation error of FC combined with the multi-dimensional corrected source we find the truncation error to be,

$$\begin{aligned} E_t = & \frac{1}{V_0}\sum_i \frac{1}{24}\left(\Delta x_{0i}^3 \mathscr{F}_{3x} + \Delta y_{0i}^3 \mathscr{F}_{3y} + 3\Delta x_{0i}^2 \Delta y_{0i} \mathscr{F}_{2xy} + 3\Delta x_{0i} y_{0i}^2 \mathscr{F}_{x2y}\right) \mathbf{n}_{0i} \mathscr{S}_{0i} \\ & - \frac{1}{V_0}\sum_i \frac{1}{8}\left(\Delta x_{0i}^2 S_{2x} + \Delta y_{0i}^2 S_{2y} + 2\Delta x_{0i} \Delta y_{0i} S_{xy}\right) \mathbf{n}_{0i} \mathscr{S}_{0i} \Delta r + O\left(h^q\right) + O\left(h^3\right). \end{aligned}$$

Once again, if $p \geq 2$ then the method remains with the leading truncation error of $O(h^2)$ for both arbitrary meshes and if $p \geq 3$, $O(h^3)$ regular meshes. The need for truncation errors of this order is needed to preserve accuracy in multiple dimensions. Because of this, the Hessian is needed in the source term.

By using proper combinations of source terms and flux schemes we have shown how truncation error can be reduced. It is interesting to note that the lowering of the truncation error is a necessary, but not always sufficient method to determine solution error. This will be shown in the results section later. From the truncation error analysis, we can conclude that using source term discretizations that are not complex enough we essentially add error to the solution. This will later be verified for several problems presented in the next section. Leading truncation error for all flux and source combinations is given in Tables 2.1 and 2.2.

2.4 Computational Studies

In this section, three computational studies are conducted, and accuracy results using various flux/source combinations are compared. The results in this section serve to identify the simplest source treatment needed to provide consistency with a particular flux treatment. To accomplish this task we employ a combination of exact and manufactured solutions to the Euler equations. The following subsections focus on two exact solutions, and one manufactured solution.

Grid refinement studies are conducted with two different grid types: a triangular grid (both perturbed and regular), and a one-dimensional mesh (used for a quasi-one-dimensional Euler study). The introduction of perturbations lends insight into scheme accuracy for more general meshes, and significant differences in accuracy are often observed when the mesh is perturbed. Though the perturbations are random, a criterion is enforced that keeps the nodes within an acceptable zone to avoid cell overlaps.

Table 2.1: Truncation error analysis results on arbitrary grids

	Constant reconstruction	Linear reconstruction	Flux Correction
Point Source	$O(h^0)$	$O(h^1)$	$O(h^2)$
Galerkin Source	$O(h^0)$	$O(h^1)$	$O(h^1)$
Corrected Source	$O(h^0)$	$O(h^1)$	$O(h^2)$

Table 2.2: Truncation error analysis results on regular grids

	Constant reconstruction	Linear reconstruction	Flux Correction
Point Source	$O(h^1)$	$O(h^1)$	$O(h^2)$
Galerkin Source	$O(h^1)$	$O(h^2)$	$O(h^2)$
Corrected Source	$O(h^1)$	$O(h^1)$	$O(h^3)$

2.4.1 Exact Euler Solution without Source

We first verify the accuracy of the three finite volume flux schemes for regular and perturbed triangular meshes by examining Ringleb flow, which is a smooth flow turning about an infinitely thin plate [67]. The flow includes inviscid and ideal gas assumptions and is an exact solution to the Euler equations, described by Equation 2.4, with

$$Q = \begin{Bmatrix} \rho \\ \rho u \\ \rho v \\ \rho e \end{Bmatrix}, \quad \mathbf{F} = \left(\begin{Bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uh \end{Bmatrix}, \begin{Bmatrix} \rho v \\ \rho vu \\ \rho v^2 + p \\ \rho vh \end{Bmatrix} \right), \quad S = \begin{Bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{Bmatrix}, \tag{2.37}$$

Here ρ is the density of the fluid, a is the speed of sound, u and v are the Cartesian components of velocity, p is the pressure, e is the internal energy per unit mass, and h is the total enthalpy per unit mass. Note that the source term is zero, enabling us to obtain a baseline measure of accuracy for the FV flux schemes.

For this case, we examine a purely subsonic region in the solution domain. The domain is selected such that the solution varies approximately ten percent over the entire domain. The domain chosen for this work extends from $x \in (2, 2.5)$ and $y \in (2, 2.5)$. The exact solution to the flow field can be found by using the hodograph method [68]. The solution at any point (x, y) is expressed as

$$A = \sqrt{1 - \frac{\gamma - 1}{2} q^2}, \quad \rho = a^{\frac{2}{\gamma - 1}}, \quad P = \frac{1}{\gamma} a^{\frac{2\gamma}{\gamma - 1}}, \quad J = \frac{1}{a} + \frac{1}{3a^3} + \frac{1}{5a^5} + \frac{1}{2} \log \frac{1 + a}{1 - a}$$

$$x(q, k) = \frac{1}{2\rho} \left(\frac{2}{k^2} - \frac{1}{q^2} \right) - \frac{J}{2}, \quad y(q, k) = \pm \frac{1}{k \rho q} \sqrt{1 - \left(\frac{q}{k} \right)^2}, \tag{2.38}$$

where q is the local Mach number, k is the local stream constant, γ is the ratio of specific heats, and A and J are intermediate variables for convenience. The solution for density in the domain of interest is shown in Figure 2.3.

To obtain a discrete solution, the domain is initialized with the exact solution. Importantly, the boundary nodes and any nodes with boundary effects in their stencils are held constant. Upon convergence, the discretization error is calculated directly by measuring the difference between the converged solution and the exact solution. Through a grid refinement study, the order of accuracy of each flux method is determined.

Convergence rates and RMS error using the three flux schemes for the Ringleb flow grid refinement study are shown in Figure 2.4 and Tables 2.3-2.6. One finding we wish to highlight is that the linear reconstruction scheme is actually third-order on a regular triangular mesh. This is a consequence of eliminating all boundary effects, which generates fully second-order accurate centered gradients in the reconstructed states, Q_L and Q_R. It is quite remarkable that the linear scheme is capable of producing third-order accuracy under certain conditions. This was also observed in our previous work [4]. We note that with the perturbed mesh the gradient calculations revert to first-order, resulting in second-order discretization error overall. However, the FC flux scheme is able to maintain the third-order accuracy even with the perturbed mesh. These effects are shown in Tables 2.4 and 2.5.

In summary, the computed order of accuracy for each flux scheme listed in Table 2.6 provides a base line of the expected convergence rates. With the baseline established, we are now in a position to examine the effects of the various source term discretization strategies on the order of accuracy. If a particular source treatment results in convergence behavior other than that reported in Table

Table 2.3: Ringleb flow grid refinement study using constant reconstruction.

	Regular Mesh		Perturbed Mesh	
Mesh Size	RMS Error	Order	RMS Error	Order
4	4.708e-5	-	3.128e-4	-
8	3.273e-5	0.556	2.042e-4	0.652
16	1.911e-5	0.799	1.027e-4	1.022
32	1.031e-5	0.905	5.336e-5	0.959
64	5.346e-6	0.954	2.716e-5	0.982
128	2.722e-6	0.977	1.369e-5	0.992

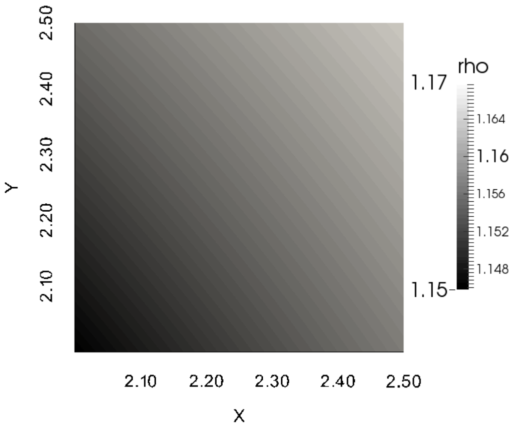


Fig. 2.3: Exact Ringleb flow solution for density.

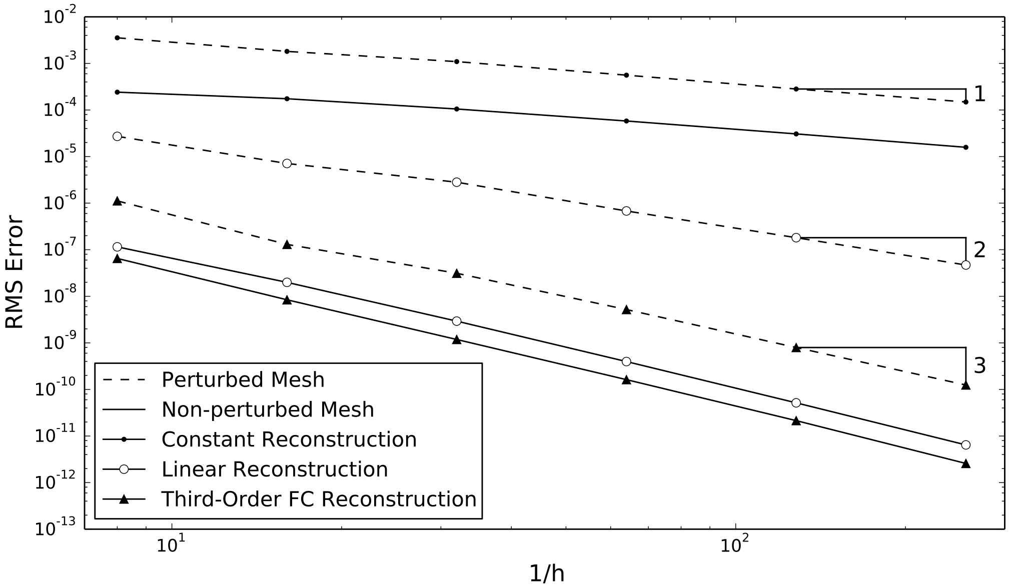


Fig. 2.4: Results of Ringleb flow grid refinement study.

Table 2.4: Ringleb flow grid refinement study using linear reconstruction.

	Regular Mesh		Perturbed Mesh	
Mesh Size	RMS Error	Order	RMS Error	Order
4	4.271e-8	-	2.956e-6	-
8	6.809e-9	2.808	9.547e-7	1.728
16	9.554e-10	2.918	2.027e-7	2.302
32	1.264e-10	2.962	5.058e-8	2.034
64	1.625e-11	2.982	1.294e-8	1.982
128	2.069e-12	2.985	3.239e-9	2.006

Table 2.5: Ringleb flow grid refinement study using flux correction.

	Regular Mesh		Perturbed Mesh	
Mesh Size	RMS Error	Order	RMS Error	Order
4	1.666e-8	-	8.722e-8	-
8	2.328e-9	3.009	1.509e-8	2.683
16	3.062e-10	3.014	1.893e-9	3.084
32	3.924e-11	3.009	2.703e-10	2.850
64	4.963e-12	3.006	3.705e-11	2.889
128	6.299e-13	2.989	4.995e-12	2.902

Table 2.6: Order of accuracy summary for Ringleb flow grid refinement study.

Flux Scheme	Regular Mesh	Perturbed Mesh
Constant Reconstruction	1	1
Linear Reconstruction	3	2
Flux Correction	3	3

2.6, we consider that treatment incompatible with the flux scheme. This study is performed in the following subsection.

2.4.2 MMS Source Case

We now test the three flux methods for compatibility with different source term discretization schemes using MMS. We first generate a manufactured solution for the Euler equations using

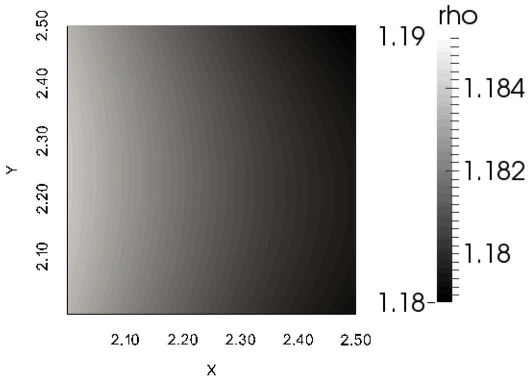


Fig. 2.5: MMS solution for density.

smooth trigonometric functions. For example, density is chosen to be

$$\rho(x,y) = \rho_0 \left(1 + \alpha_x \sin(\beta_x x) + \alpha_y \cos(\beta_y y) + \alpha_{xy} \cos(\beta_{xy} xy)\right), \tag{2.39}$$

where ρ_0 and the α and β values are constants chosen to provide physically realistic values (e.g. positive density), and provide a solution field that varies around ten percent over the chosen domain. Figure 2.5 shows the manufactured solution for density described in Equation 2.39. Following the standard MMS procedure [49], a source term is generated through analytical computation of the governing PDE (Euler equations) applied to the manufactured solution. Observations are made regarding the effect of each source term discretization scheme on each flux discretization scheme.

Constant Reconstruction

Results for the constant reconstruction scheme are shown in Figure 2.6 and Tables 2.7-2.9. Error measurements using MMS indicate that there is very little difference in accuracy using different source treatments. All methods are first-order accurate and match the results of the exact Ringleb flow study. In this sense, all source treatments considered are compatible with constant reconstruction. However, time differences show that increased computational effort is needed for more complex source discretizations, as shown in Table 2.10. The Galerkin source term and the point source term show close relation in computation time, with Galerkin taking slightly more time. However, the corrected source term adds significantly more computation time with no improvement in accuracy. We can therefore conclude that the point source is the best method for the constant reconstruction scheme.

Table 2.7: Order of accuracy results for MMS using constant reconstruction with point source.

	Regular Mesh		Perturbed Mesh	
Mesh Size	RMS Error	Order	RMS Error	Order
4	1.839e-4	-	8.057e-4	-
8	1.270e-4	0.566	3.662e-4	1.206
16	7.398e-5	0.803	1.951e-4	0.935
32	3.991e-5	0.904	9.624e-5	1.035
64	2.076e-5	0.950	4.857e-5	0.994
128	1.063e-5	0.970	2.303e-5	1.080

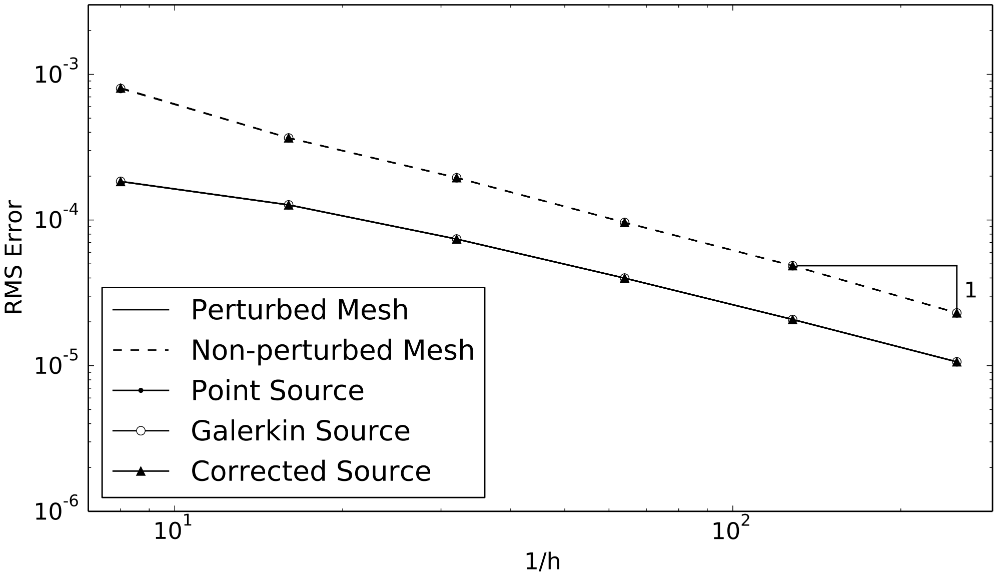


Fig. 2.6: Convergence results for MMS using constant reconstruction with various source treatments.

Table 2.8: Order of accuracy results for MMS using constant reconstruction with Galerkin source.

	Regular Mesh		Perturbed Mesh	
Mesh Size	RMS Error	Order	RMS Error	Order
4	1.842e-4	-	7.989e-4	-
8	1.271e-4	0.567	3.656e-4	1.195
16	7.402e-5	0.803	1.949e-4	0.934
32	3.991e-5	0.904	9.618e-5	1.034
64	2.077e-5	0.950	4.855e-5	0.994
128	1.063e-5	0.970	2.303e-5	1.080

Table 2.9: Order of accuracy results for MMS using constant reconstruction with corrected source.

	Regular Mesh		Perturbed Mesh	
Mesh Size	RMS Error	Order	RMS Error	Order
4	1.837e-4	-	8.052e-4	-
8	1.269e-4	0.565	3.661e-4	1.205
16	7.397e-5	0.802	1.951e-4	0.935
32	3.990e-5	0.904	9.624e-5	1.035
64	2.076e-5	0.950	4.857e-5	0.994
128	1.063e-5	0.969	2.303e-5	1.080

Table 2.10: Time (s) per iteration on 128 by 128 grid using constant reconstruction and various source treatments.

	Regular Mesh		Perturbed Mesh	
Source Term	Time/Iteration	Relative Percent	Time/Iteration	Relative Percent
Point	1.856	0.00%	1.860	0.00%
Galerkin	1.906	2.69%	1.901	2.20%
Corrected	2.469	33.03%	2.489	33.82%

Linear Reconstruction

We now examine various source treatments in conjunction with the linear reconstruction flux method in Equation 2.16. The effect on accuracy using each source treatment can be seen in Figure 2.7 and in Tables 2.11-2.13. It is clear that the only method that correctly gives the same order of accuracy observed with no source terms in Table 2.6 is the Galerkin source discretization. The other methods of source discretization prove to be inadequate and provide larger errors in the overall solution, especially for regular grids. It is interesting that the corrected source treatment actually reduces the accuracy of the linear reconstruction scheme. Unlike constant reconstruction, the Galerkin source treatment is the only compatible discretization with linear reconstruction. This finding has important implications for CFD codes that using linear reconstruction with point source treatment.

The computation time using each source discretization is shown in Table 2.14. Negative relative percent indicates a decrease in iteration time, but this only occurs when the incompatible point source discretization scheme is used. Although using the point source term reduces computation time, the reduction is slight, and does not justify the reduction in accuracy.

Table 2.11: Order of accuracy results for MMS using linear reconstruction with point source.

	Regular Mesh		Perturbed Mesh	
Mesh Size	RMS Error	Order	RMS Error	Order
4	9.310e-7	-	2.407e-5	-
8	3.079e-7	1.692	8.671e-6	1.561
16	8.732e-8	1.872	1.493e-6	2.614
32	2.338e-8	1.930	4.687e-7	1.697
64	6.135e-9	1.944	1.086e-7	2.125
128	1.611e-9	1.936	2.648e-8	2.044

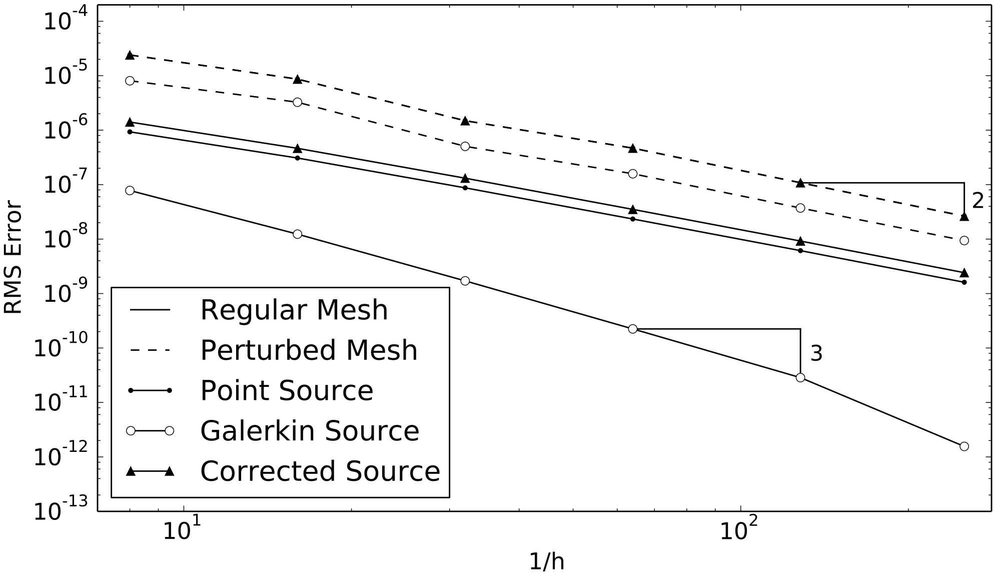


Fig. 2.7: Convergence results for MMS using linear reconstruction with various source treatments.

Table 2.12: Order of accuracy results for MMS using linear reconstruction with Galerkin source.

	Regular Mesh		Perturbed Mesh	
Mesh Size	RMS Error	Order	RMS Error	Order
4	7.800e-8	-	8.073e-6	-
8	1.232e-8	2.822	3.260e-6	1.387
16	1.710e-9	2.934	5.048e-7	2.772
32	2.242e-10	2.975	1.584e-7	1.697
64	2.859e-11	2.994	3.721e-8	2.105
128	1.560e-12	4.212	9.400e-9	1.993

Table 2.13: Order of accuracy results for MMS using linear reconstruction with corrected source.

	Regular Mesh		Perturbed Mesh	
Mesh Size	RMS Error	Order	RMS Error	Order
4	1.409e-6	-	2.399e-5	-
8	4.641e-7	1.699	8.647e-6	1.561
16	1.313e-7	1.876	1.490e-6	2.613
32	3.512e-8	1.932	4.662e-7	1.702
64	9.210e-9	1.945	1.080e-7	2.126
128	2.418e-9	1.937	2.625e-8	2.048

Table 2.14: Time (s) per iteration on 128 by 128 grid using linear reconstruction and various source treatments.

	Regular Mesh		Perturbed Mesh	
Source Term	Time/Iteration	Relative Percent	Time/Iteration	Relative Percent
Point	1.927	-3.02%	1.910	-2.15%
Galerkin	1.987	0.00%	1.952	0.00%
Corrected	2.535	27.58%	2.536	29.92%

Flux Correction

Finally, for the FC flux scheme in Equation 2.22, results with various source treatments are shown in Figure 2.8 and in Tables 2.15-2.17. The error results show that the only source treatment that is able to provide results compatible with the exact solution with no source is the corrected source discretization. This was the expectation since the corrected source was designed to preserve the order of truncation error. This result is critical to maintain the accuracy of the FC scheme in the presence of source terms. Interestingly, The Galerkin source discretization performs worse than the point source discretization in this case.

Timing results in Table 2.18 show that the time needed per iteration is the largest when using the corrected source term discretization scheme. However, this is the price to pay to obtain third-order accuracy and is relatively economical for situations where high levels of accuracy are required.

2.4.3 Quasi-one-dimensional Flow

Now that we have identified compatible source discretizations for each of the three flux schemes, we examine quasi-one-dimensional flow, which features physically meaningful source terms. The quasi-one-dimensional flow equations are a simplified version of the two-dimensional Euler equa-

Table 2.15: Order of accuracy results for MMS using flux correction with point source.

	Regular Mesh		Perturbed Mesh	
Mesh Size	RMS Error	Order	RMS Error	Order
4	1.011e-6	-	1.497e-6	-
8	1.817e-7	2.625	2.407e-7	2.795
16	4.257e-8	2.156	5.051e-8	2.320
32	1.129e-8	1.944	1.218e-8	2.083
64	3.029e-9	1.912	3.271e-9	1.911
128	8.138e-10	1.903	8.775e-10	1.905

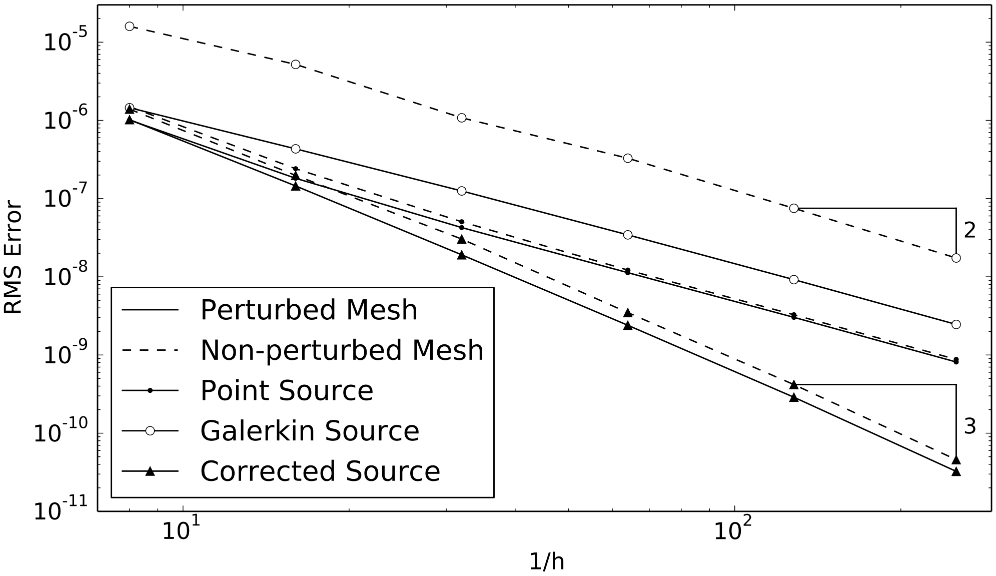


Fig. 2.8: Convergence results for MMS using Flux Correction with various source treatments.

Table 2.16: Order of accuracy results for MMS using flux correction with Galerkin source.

	Regular Mesh		Perturbed Mesh	
Mesh Size	RMS Error	Order	RMS Error	Order
4	1.451e-6	-	1.595e-5	-
8	4.323e-7	1.852	5.195e-6	1.715
16	1.252e-7	1.842	1.078e-6	2.336
32	3.437e-8	1.893	3.280e-7	1.743
64	9.198e-9	1.916	7.509e-8	2.143
128	2.458e-9	1.911	1.739e-8	2.118

Table 2.17: Order of accuracy results for MMS using flux correction with corrected source.

	Regular Mesh		Perturbed Mesh	
Mesh Size	RMS Error	Order	RMS Error	Order
4	1.021e-6	-	1.392e-6	-
8	1.450e-7	2.985	1.979e-7	2.983
16	1.909e-8	3.013	3.036e-8	2.786
32	2.399e-9	3.038	3.482e-9	3.171
64	2.882e-10	3.080	4.185e-10	3.079
128	3.247e-11	3.162	4.580e-11	3.204

Table 2.18: Time (s) per iteration on 128 by 128 grid using flux correction and various source treatments.

	Regular Mesh		Perturbed Mesh	
Source Term	Time/Iteration	Relative Percent	Time/Iteration	Relative Percent
Point	3.307	-15.18%	3.291	-15.27%
Galerkin	3.333	-14.52%	3.318	-14.57%
Corrected	3.899	0.00%	3.884	0.00%

tions and are commonly employed for rocket nozzles and other similar applications. Due to the dependence of the flow on the nozzle area, $a(x)$, the Euler equations become

$$Q = a\left\{\begin{matrix}\rho \\ \rho u \\ \rho e\end{matrix}\right\}, \quad F = a\left\{\begin{matrix}\rho u \\ \rho u^2 + p \\ \rho u h\end{matrix}\right\}, \quad and \quad S = \left\{\begin{matrix}0 \\ p\frac{da}{dx} \\ 0\end{matrix}\right\}. \tag{2.40}$$

Here, the solution variables and flux contain the area, and the source term is dependent on the rate of change of the area in the x direction.

We employ Equation 2.40 for a nozzle that accelerates the flow from Mach = 2 to Mach = 3. The radius of the nozzle expands linearly with distance down the nozzle. Figure 2.9 shows the exact Mach and pressure profiles we expect for this nozzle. As in previous cases, grid refinement studies will be performed to compute the overall order of convergence for each flux method/source treatment combination.

Constant Reconstruction Scheme

Constant reconstruction for the nozzle problem shows an identical trend to that observed with MMS. The differences between the perturbed and regular meshes are smaller, however, due to the restriction to a one-dimensional mesh, as shown in Figure 2.10. Results indicate there is little difference in the source treatment used for the constant reconstruction scheme. While all source treatments are compatible, we advocate the point source treatment with constant reconstruction due to its reduced computational cost.

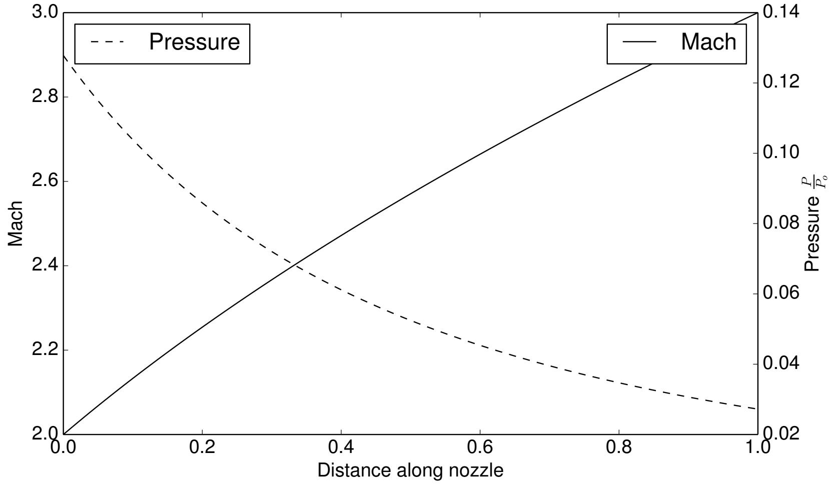


Fig. 2.9: Solution for quasi-one-dimensional flow.

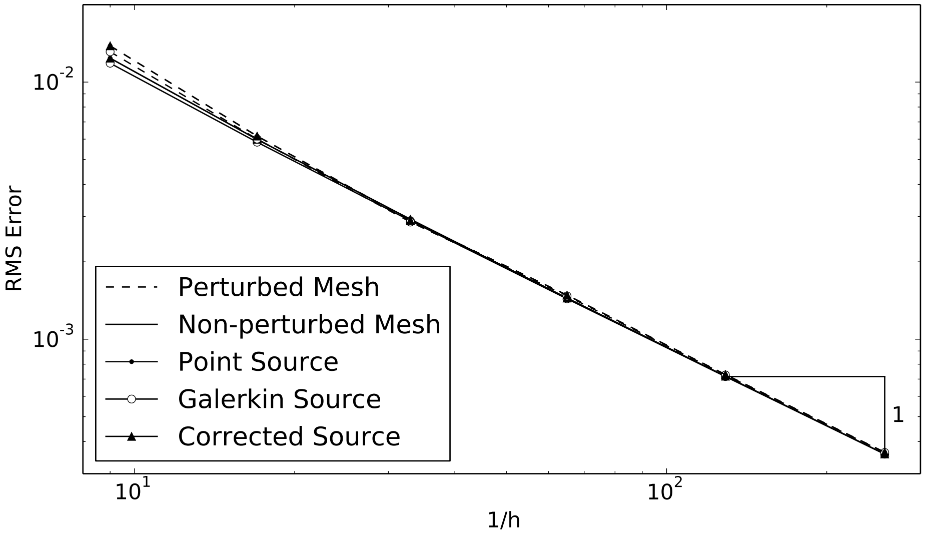


Fig. 2.10: Convergence results for Quasi-one-dimensional flow using the constant reconstruction scheme.

Linear Reconstruction Scheme

Linear reconstruction also shows the same trends as found with MMS. Results show that the method converges at the expected rate of second-order for all schemes, as shown in Figure 2.11. An important difference in this study is that the boundaries are not frozen like in the MMS case. Consequently, the method does not produce a third-order solution with the Galerkin source term on a regular mesh. Essentially, the boundary effects introduce some asymmetry in the gradient stencils, reducing the accuracy to second-order. Results also show that accuracy is lost due to the use of an incompatible source treatment, which is any source term other than Galerkin for linear reconstruction.

Third-order FC Scheme

The third-order FC scheme also behaves as expected from previous studies. We conclude that that the use of incompatible source treatment, which is any treatment other than corrected for the FC scheme, reduces the order of convergence from third-order to second-order. Also like MMS, the Galerkin source produces the greatest errors. These results are shown in Figure 2.12.

2.5 Conclusions

Obtaining source term discretizations that are compatible with finite volume flux schemes is essential to obtaining correct convergence rates. While simple source treatments may reduce solution time, the resulting loss is accuracy can be significant. Even if the truncation error does not reduce the order of convergence, the addition of such increases the solution error. Verification of compatible source discretizations, which are defined as source treatments that preserve the same order of accuracy obtained without source terms, can be effectively accomplished by performing a combination of exact and manufactured solution grid refinement studies in addition to truncation error analysis. Once compatible source treatments are identified, the results extend to equations with physically meaningful source terms.

In this work we formulated three finite volume flux schemes: constant reconstruction, linear reconstruction, and a relatively new scheme called flux correction. With each of these flux schemes,

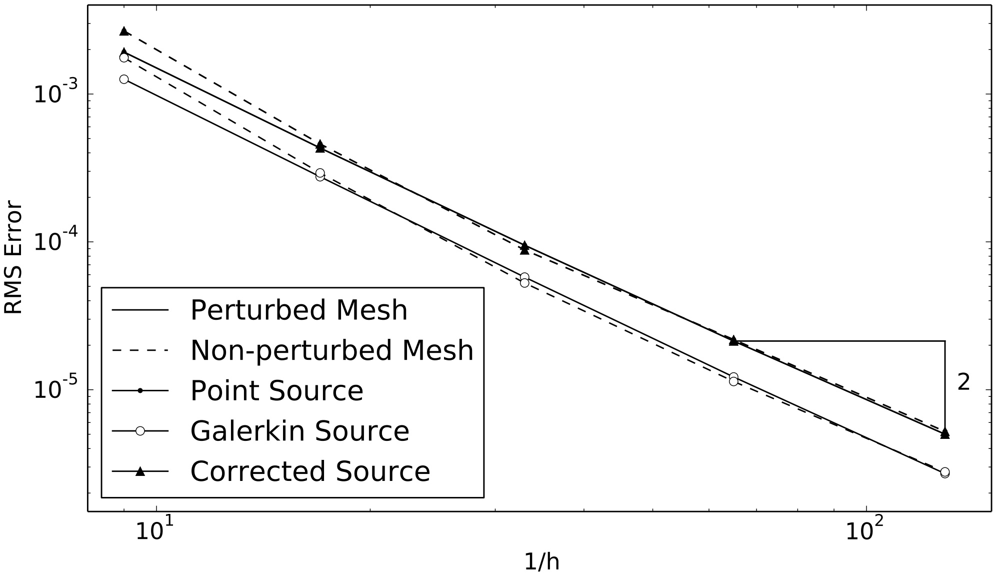


Fig. 2.11: Convergence results for Quasi-one-dimensional flow using the liner reconstruction scheme.

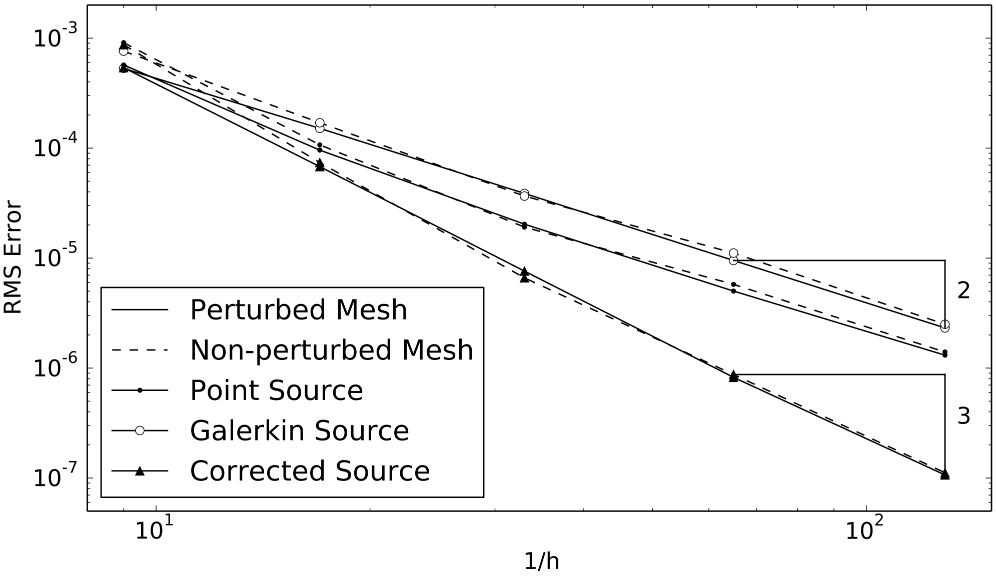


Fig. 2.12: Convergence results for Quasi-one-dimensional flow using the FC scheme.

we tested three different source treatments: point, Galerkin, and corrected. For constant reconstruction, all three source term treatments are compatible and first-order accurate, but we advocate the use of the point source treatment due to reduced computational cost. The only compatible source treatment for the linear reconstruction scheme is Galerkin, which produces second-order accurate results on general meshes, and surprisingly, third-order results on regular meshes in the absence of boundary effects. However, the scheme reverts back to second-order on regular meshes with boundary effects included. The only compatible source treatment for the flux correction scheme is the corrected source discretization, which produces third-order accurate results regardless of mesh perturbations and including boundary effects. The corrected source treatment requires additional computational effort, but the improvements in accuracy are significant. These results conclusions can be drawn both from the truncation error analysis as well as the test results.

In conclusion, a point source discretization is sufficient for constant reconstruction, a Galerkin source discretization is sufficient for linear reconstruction, and a corrected source discretization is sufficient for the flux correction scheme. It is expected that these results will prove valuable for more complex equation sets involving turbulence, multi-phase flow, reacting flow, and other scenarios. Future work will focus on extensions to these types of flows in order to better understand source term discretization effects in these contexts.

Chapter 3
High-order Strand Grid Methods for Low Mach and Incompressible Flows[1]

Jonathan Thorne, Aaron Katz[2], Oisin Tong[3] [4], Yushi Yanagita[5] [6], Keegan Delaney[7] [8]

3.1 Introduction

Low speed flows present challenges for compressible computational fluid dynamics (CFD) algorithms in terms of accuracy and convergence rate. Many of these flows have widely varying particle and acoustic speeds, degrading solution convergence [69]. Poor scaling of artificial dissipation at low Mach number, determined through asymptotic analysis [12], often reduces the accuracy of the solution even when convergence is achieved. For practical flows with complex geometry, mesh generation and grid quality become additional challenges. Meshing can take days or weeks depending on the complexity of the geometry. Progress addressing these challenges will transform many engineering applications, such as underwater vehicle design.

To address these challenges we use a combination of established and new CFD methods, including preconditioning, high-order methods, and strand grids. First, we focus on established preconditioning methods [10, 12]. Preconditioning can extend favorable properties observed for transonic flows to low Mach number regimes, including improved dissipation scaling and fast convergence. Under certain conditions, the preconditioning approach is analogous to the artificial compressibility approach introduced by Chorin [9]. For the sake of generality, in this work we employ

[1] Planned Submission to the International Journal for Numerical Methods in Fluids

[2] Associate Professor, Department of Mechanical and Aerospace Engineering, Utah State University, Logan, UT

[3] PHD Candidate, Department of Mechanical and Aerospace Engineering, Utah State University, Logan, UT

[4] Preconditioned Spalart-Allmaras Solver

[5] Masters Student, Department of Mechanical and Aerospace Engineering, Utah State University, Logan, UT

[6] Body of Revolution Mesh Generation

[7] Computational Hydromechanics Division, Code 5700, Naval Surface Warfare Center Carderock Division

[8] Body of Revolution Experimental Data

the method of Merkle et al. for arbitrary equations of state [10]. This approach handles truly incompressible flows as well as the full range of low Mach through supersonic flows. Additionally, the approach may be extended to low speed reacting flows, in which acoustic effects are significant. This will be a topic of future work.

Second, to reduce excess numerical dissipation, we implement preconditioning schemes within new a relatively new high-order framework. Flux correction (FC) was recently introduced by Katz and Sankaran [4] to provide high-order spatial accuracy while requiring only first derivatives in the reconstruction for finite volume schemes [56,65]. High-order accuracy is achieved by explicitly cancelling low-order truncation error terms arising from numerical flux approximations. This method has been shown to be third-order accurate on arbitrary triangular meshes for inviscid flows [4] and approaches fourth-order accuracy in highly viscous flows [65]. For turbulent flows, we employ the Spalart-Allmaras “negative” model [45, 47] in a fully high-order consistent manner.

Third, to address complex geometry, we explore strand meshing, which fully automates volume grid generation and improves scalability for overset moving-body applications [3, 13]. Strands are automatically generated pointing vectors (determined from a surface tessellation) along which distributions of points are placed, creating layers of unstructured mesh emanating from solid bodies. Strand meshing simplifies the meshing process, increases automation, and provides some structure useful for high-order algorithms. Importantly, compact strand meshing techniques offer self-satisfying domain connectivity (SSDC), whereby each processor in a multi-processor computation has access to the global mesh data, increasing the efficiency of overset domain connectivity algorithms. Previous work has investigated our high-order approach for strand grids [5], but the present work represents the first extension of such methods to a much wider class of flows via a preconditioner.

To cover our multi-faceted approach, this work is outlined as follows: Section 3.2 provides a description of the governing partial differential equations, including the Spalart-Allmaras turbulence model. Also in this section, we describe our strand grid approach, a brief description of FC and finite difference methods, and finally, the implementation of the preconditioner. In Section 3.3 we present a number of verification and validation cases using the methods described in Section 3.2. We verify the algorithms via the method of manufactured solutions, then validate with laminar flow over a sphere and turbulent flow over a number of submersible body-of-revolution test cases [70]. Finally, in Section 3.4 we draw conclusions and outline directions for future work.

3.2 Numerical Methods

In this work we solve the steady Reynolds Averaged Navier-Stokes equations with Spalart-Allmaras turbulence model (RANS-SA), expressed here as

$$\frac{\partial Q}{\partial \tau} + \frac{\partial F_j}{\partial x_j} - \frac{\partial F_j^v}{\partial x_j} = S, \tag{3.1}$$

where Q represents the vector of conserved variables, τ represents the pseudo-time, $F_j = (F, G, H)$ represents the inviscid fluxes, $F_j^v = (F^v, G^v, H^v)$ represents the viscous fluxes, and S represents the source term. The various vectors are defined as follows:

$$Q = \begin{Bmatrix} \rho \\ \rho u_i \\ \rho E \\ \rho \tilde{\nu} \end{Bmatrix}, \quad F_j = \begin{Bmatrix} \rho u_j \\ \rho u_i u_j + p\delta_{ij} \\ \rho h^0 u_j \\ \rho \tilde{\nu} u_j \end{Bmatrix}, \quad F_j^v = \begin{Bmatrix} 0 \\ \sigma_{ij} \\ \sigma_{ij} u_i - q_j \\ \frac{\eta}{\sigma}\frac{\partial \tilde{\nu}}{\partial x_j} \end{Bmatrix}, \quad S = \begin{Bmatrix} 0 \\ 0 \\ 0 \\ \mathscr{P} - \mathscr{D} + c_{b2}\rho \frac{\partial \tilde{\nu}}{\partial x_k}\frac{\partial \tilde{\nu}}{\partial x_k} \end{Bmatrix}. \tag{3.2}$$

Here, ρ is the density, $u_j = (u, v, w)$ is the j^{th} component of velocity, E is the total energy per unit mass, $\tilde{\nu}$ is the turbulence working variable, p is the pressure, h^0 is the total enthalpy per unit mass, $\frac{\eta}{\sigma}$ is the turbulent diffusion coefficient, σ_{ij} is the deviatoric stress tensor, and q_j is the j^{th} component of the heat flux vector. In the source terms, $\mathscr{P}$ is the turbulent production, and $\mathscr{D}$ is the turbulent destruction. Details of the turbulent source, including the well-known production and destruction terms, can be found in the original work by Spalart [45]. Modifications are introduced following Allmaras et al. [47] to accommodate negative values of the turbulence working variable, $\tilde{\nu}$, which may arise when using high-order accurate discretizations of the model.

Additionally, the stress tensor is defined as

$$\sigma_{ij} = 2\mu_t \left(S_{ij} - \frac{1}{3}\frac{\partial u_k}{\partial x_k}\delta_{ij} \right), \tag{3.3}$$

where μ is the dynamic viscosity, and S_{ij} is the rate of strain tensor, defined as

$$S_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right). \tag{3.4}$$

The heat flux is defined as

$$q_j = \left(\frac{\mu}{Pr} + \frac{\mu_t}{Pr_t}\right)\frac{\partial T}{\partial x_j}, \tag{3.5}$$

where μ_t is the turbulent viscosity, Pr is the Prandtl number, Pr_t is the turbulent Prandtl number, and T is the temperature. In the following subsections we discuss the numerical discretization of Equation 3.1 in detail.

3.2.1 High-order Strand Grid Discretization

Strand grids provide a method for automatic mesh generation while providing a compact semi-structured mesh format that is favourable to parallel scalability [3, 13]. Strands are produced from a surface tessellation by extruding straight lines (strands) a short distance in the normal direction. While strands are used to capture near-body effects, overset Cartesian grids are typically used to capture off-body effects. In this work, we focus only on the strand discretization and leave off-body coupling for future work.

Another interpretation of strand grids of relevance here is that strands connect a series of unstructured mesh layers expanding from the surface. Figure 3.1 shows this interpretation, where each node level forms a new layer of unstructured mesh. Interpreting the mesh in this manner leads to our discretization strategy, which is to solve a series of unstructured discretizations in each layer of the strand grid, while coupling the layers through a source term containing derivatives along the strands. This approach was first explored by Katz and Work [5] for compressible flows. Here we provide a brief explanation of this discretization scheme, and then extend the method to low speed and incompressible flows through a preconditioner.

The method begins with a surface mesh consisting of a high-order triangulation from which strands are projected along normal vectors. A quadratic surface element and strand projection is shown in Figure 3.1. After the initial mesh generation, each element of the strand mesh in *xyz* space

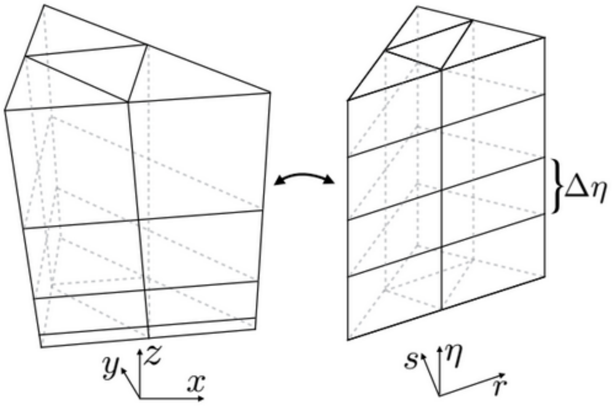


Fig. 3.1: Transformation between physical and computational domains for a strand grid.

is locally transformed to the computational domain, in $rs\eta$ space, as shown in Figure 3.1. Upon transformation to computational space, Equation 3.1 becomes

$$\frac{\partial \hat{Q}}{\partial t}+\frac{\partial \hat{F}}{\partial r}+\frac{\partial \hat{G}}{\partial s}+\frac{\partial \hat{H}}{\partial \eta}-\frac{\partial \hat{F}^v}{\partial r}-\frac{\partial \hat{G}^v}{\partial s}-\frac{\partial \hat{H}^v}{\partial \eta}=\hat{S} \tag{3.6}$$

where the transformed variables, fluxes, and source term are

$$\hat{Q}\equiv JQ, \quad \hat{S}\equiv JS,$$

$$\hat{F}\equiv J\left(r_xF+r_yG+r_zH\right), \quad \hat{F}^v\equiv J\left(r_xF^v+r_yG^v+r_zH^v\right),$$

$$\hat{G}\equiv J\left(s_xF+s_yG+s_zH\right), \quad \hat{G}^v\equiv J\left(s_xF^v+s_yG^v+s_zH^v\right),$$

$$\hat{H}\equiv J\left(\eta_xF+\eta_yG+\eta_zH\right), \quad \hat{H}^v\equiv J\left(\eta_xF^v+\eta_yG^v+\eta_zH^v\right),$$

$$\begin{pmatrix} r_x & s_x & \eta_x \\ r_y & s_y & \eta_y \\ r_z & s_z & \eta_z \end{pmatrix} = \frac{1}{J}\begin{pmatrix} y_sz_\eta - z_sy_\eta & z_ry_\eta - y_rz_\eta & y_rz_s - z_ry_s \\ z_sx_\eta - x_sy_\eta & x_rz_\eta - z_rx_\eta & z_rx_s - x_rz_s \\ x_sy_\eta - y_sx_\eta & y_rx_\eta - x_ry_\eta & x_ry_s - y_rx_s \end{pmatrix},$$

$$J = x_\eta\left(y_rz_s - z_ry_s\right)+y_\eta\left(z_rx_s - x_rz_s\right)+z_\eta\left(x_ry_s - y_rx_s\right).$$

Here, J is the Jacobian of the transformation, and partial differentiation is denoted with a subscript (e.g. $\partial x/\partial s = x_s$).

After transformation, high-order flux balances are calculated in each unstructured layer (r-s plane) and are then coupled together using high-order finite differences based on summation-by-parts (SBP) along strands (η-direction). This is accomplished by moving all η-derivatives to the right-hand side of Equation 3.6, resulting in

$$\frac{\partial \hat{Q}}{\partial \tau}+\frac{\partial \hat{F}}{\partial r}+\frac{\partial \hat{G}}{\partial s}-\frac{\partial \hat{F}^{v}}{\partial r}-\frac{\partial \hat{G}^{v}}{\partial s}=\tilde{S}, \tag{3.7}$$

where

$$\tilde{S}\equiv \hat{S}-\frac{\partial \hat{Q}}{\partial t}-\frac{\partial \hat{H}}{\partial \eta}+\frac{\partial \hat{H}^{v}}{\partial \eta}.$$

The interpretation of Equation 3.7 is now essentially a two dimensional (r-s) flux balance with source (η) terms. Additionally, this interpretation becomes convenient when the method is parallelized, requiring partitioning of the surface grid only. In this manner, each processor takes responsibility for a partition of strands.

For the source (η) terms, stable summation-by-parts (SBP) operators [71–75] approximate flux derivatives along strands. We impose boundary conditions weakly through simultaneous approximation terms (SAT) [76] added as penalties at boundaries [77, 78]. Much of the SBP theory has been established now for many years, and we refer readers to the references provided.

A recently introduced method, flux correction (FC) [4, 65], is employed in the r-s plane. The key aspect of the FC method is the cancellation of leading truncation error terms appearing in the finite volume flux balance. Considering the inviscid portion of Equation 3.7, the numerical flux at a face between nodes i and j in a layer of the strand mesh is formulated as

$$\mathscr{F}_{ij}^{h}=\frac{1}{2}\left(\mathscr{F}_{L}+\mathscr{F}_{R}\right)-\frac{1}{2}\left|\mathscr{A}\left(Q_{R},Q_{L}\right)\right|\left(Q_{R}-Q_{L}\right), \tag{3.8}$$

where $\mathscr{F}=\boldsymbol{F}\cdot\boldsymbol{n}$ is the inviscid flux in the face normal direction, $\mathscr{A}=\partial\mathscr{F}/\partial Q$ is the directed flux Jacobian, and Q_R and Q_L are the conserved variables at the left and right reconstructed states. Following the usual convention [63], the directed flux Jacobian is decomposed into the left and right eigenvectors and the eigenvalues,

$$|\mathscr{A}| = M\,|\Lambda|\,M^{-1}, \tag{3.9}$$

where the eigenvalues are

$$\begin{aligned}
&\Lambda = \text{diag}(\lambda_i) \\
&\lambda_{1,2} = u_n \pm c, \quad \lambda_{3,4,5,6} = u_n.
\end{aligned} \tag{3.10}$$

A critical aspect of the method is the reconstruction of the fully non-linear flux as well as the conserved variables

$$\begin{aligned}
F_L = \mathscr{F}_i + \frac{1}{2}\Delta\mathbf{r}^T\nabla^h\mathscr{F}_i, \quad F_R = \mathscr{F}_j - \frac{1}{2}\Delta\mathbf{r}^T\nabla^h\mathscr{F}_j, \\
Q_L = Q_i + \frac{1}{2}\Delta\mathbf{r}^T\nabla^h Q_i, \quad Q_R = Q_j - \frac{1}{2}\Delta\mathbf{r}^T\nabla^h Q_j,
\end{aligned} \tag{3.11}$$

where $\mathbf{r}$ is the position vector along the edge connecting node i to node j. Importantly, the gradient is discretized with p^{th}-order accurate method such that,

$$\nabla^h = \nabla + O(h^p), \tag{3.12}$$

where $O(h^p)$ denotes the truncation error of the gradient procedure. For FC, the order of accuracy of the flux and solution gradients must be at least second-order ($p \geq 2$) to maintain consistency. Such gradient approximations may be formulated in a variety of ways such as least squares procedures or Green-Gauss integration [56]. Following our previous work [5], we use high-order finite element-type approximations to construct the required flux and solution gradients.

The advantage of flux correction over other high-order methods is that it bypasses the need for second derivatives in the reconstruction and does not require high-order quadrature. This makes the method an efficient way to achieve high-order accuracy without excess computational expense.

3.2.2 Preconditioning

In order to apply the high-order strand grid methodology to a much wider class of flows, we explore preconditioning techniques. In particular, numerous preconditioning methods based on matrix dissipation and other approaches have been utilized successfully by other researchers to improve performance of compressible CFD methods in low speed and incompressible regimes [10, 12, 69]. We first present a traditional framework for preconditioning based on the work of Merkle et al. [10], then modify the framework to generate an optimal form that we use.

We begin with a conversion of Equation 3.1 from conserved variables to primitive variables,

$$\Gamma\frac{\partial Q_p}{\partial \tau}+\frac{\partial F_j}{\partial x_j}-\frac{\partial F_j^v}{\partial x_j}=S, \tag{3.13}$$

where

$$Q_p=\begin{Bmatrix} p \\ u_j \\ T \\ \tilde{v} \end{Bmatrix}, \tag{3.14}$$

which include pressure, velocity, temperature, and the turbulent working variable. This is done to allow for arbitrary equations of state, including incompressible flow ($\rho=$ constant) investigated in this work. Conversion between the primitive variables and the conserved variables in Equation 3.13 is accomplished via a Jacobian matrix,

$$\Gamma\equiv\frac{\partial Q}{\partial Q_p}=\begin{pmatrix} \rho_p & 0 & 0 & 0 & \rho_T & 0 \\ u\rho_p & \rho & 0 & 0 & u\rho_T & 0 \\ v\rho_p & 0 & \rho & 0 & v\rho_T & 0 \\ w\rho_p & 0 & 0 & \rho & w\rho_T & 0 \\ h^0\rho_p+\rho h_p-1 & \rho u & \rho v & \rho w & h^0\rho_T+\rho h_T & 0 \\ \tilde{v}\rho_p & 0 & 0 & 0 & \tilde{v}\rho_T & \rho \end{pmatrix}, \tag{3.15}$$

where subscripts p and T denote partial differentiation with respect to pressure and temperature (e.g. $\rho_p=\frac{\partial\rho}{\partial p}$).

The goal of the preconditioner is to decrease the condition number of the system, and allow for arbitrary equations of state. This is done by replacing the Jacobian matrix, Γ, in the pseudo-time derivative term with a preconditioning matrix, Γ_p, defined as

$$\Gamma_p = \begin{pmatrix} \rho'_p & 0 & 0 & 0 & \rho_T & 0 \\ u\rho'_p & \rho & 0 & 0 & u\rho_T & 0 \\ v\rho'_p & 0 & \rho & 0 & v\rho_T & 0 \\ w\rho'_p & 0 & 0 & \rho & w\rho_T & 0 \\ h^0\rho'_p + \rho h_p - 1 & \rho u & \rho v & \rho w & h^0\rho_T + \rho h_T & 0 \\ \tilde{v}\rho'_p & 0 & 0 & 0 & \tilde{v}\rho_T & \rho \end{pmatrix}. \tag{3.16}$$

Here, Γ_p is identical to Γ except for ρ_p is replaced by ρ'_p.

The selection of ρ'_p is accomplished by choosing an acoustic wave speed to be the same order of magnitude as the particle speeds. Sankaran [12] and Merkle, et al. [10] take this approach by defining a preconditioned speed of sound,

$$V_p^2 \equiv \frac{\rho h_t}{d'}, \tag{3.17}$$

where d' is defined as

$$d' \equiv \rho h_T \rho'_p + \rho_T \left(1 - \rho h_p\right). \tag{3.18}$$

Once V_p is chosen, then ρ'_p can be determined via Equations 3.17 and 3.18. In this manner, the convergence and dissipation scaling properties of the preconditioner are dependent primarily on the preconditioned speed of sound chosen. Traditionally this is chosen with a comparison of several different terms:

$$V_p = \min\left(\max\left(\sqrt{u^2+v^2+w^2}, \sqrt{\frac{|\Delta p|}{\rho}}, \frac{\nu}{\Delta x}\right), c\right), \tag{3.19}$$

where c is the local speed of sound. In Equation 3.19, the preconditioned sound speed for subsonic flows is chosen to be the particle velocity unless pressure perturbations (Δp) or viscous diffusion ($\nu/\Delta x$) are dominant. Such scenarios include large pressure changes at stagnations points, or inside viscous boundary layers where the mesh spacing, Δx, is small.

Using the preconditioned Mach number notation, $M_p \equiv V_p/c$, the inviscid eigenvalues of the new dissipation matrix, $\Gamma_p^{-1}\mathscr{A} = \hat{M}\lambda\hat{M}^{-1}$, are expressed as

$$\lambda_{1,2} = \frac{1}{2}\left(u_n\left(1+M_p^2\right) \pm \sqrt{u_n^2\left(M_p^2-1\right)^2+4V_p^2}\right), \quad \lambda_{3,4,5,6} = u_n, \tag{3.20}$$

where $u_n \equiv \boldsymbol{u}\cdot\boldsymbol{n}$, and $\boldsymbol{n}$ is the unit normal of the corresponding finite volume face or finite difference strand. From Equation 3.20, it is clear that the pseudo-acoustic speeds, $\lambda_{1,2}$, are of the same order of magnitude as the particle velocity, u_n. The new right and left eigenvectors are expressed as

$$\begin{gathered}
\hat{M} = \begin{pmatrix}
\rho\left(\lambda_1-u_n\right) & \rho\left(\lambda_2-u_n\right) & 0 & 0 & 0 & 0 \\
n_x & n_x & l_1 & m_1 & 0 & 0 \\
n_y & n_y & l_2 & m_2 & 0 & 0 \\
n_z & n_z & l_3 & m_3 & 0 & 0 \\
\frac{1-\rho h_p}{\rho h_T}\left(\lambda_1-u_n\right) & \frac{1-\rho h_p}{\rho h_T}\left(\lambda_2-u_n\right) & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1
\end{pmatrix} \\
\hat{M}^{-1} = \begin{pmatrix}
-\frac{1}{\rho(\lambda_2-\lambda_1)} & \frac{\lambda_2-u_n}{\lambda_2-\lambda_1}n_x & \frac{\lambda_2-u_n}{\lambda_2-\lambda_1}n_y & \frac{\lambda_2-u_n}{\lambda_2-\lambda_1}n_z & 0 & 0 \\
\frac{1}{\rho(\lambda_2-\lambda_1)} & -\frac{\lambda_1-u_n}{\lambda_2-\lambda_1}n_x & -\frac{\lambda_1-u_n}{\lambda_2-\lambda_1}n_y & -\frac{\lambda_1-u_n}{\lambda_2-\lambda_1}n_z & 0 & 0 \\
0 & l_1 & l_2 & l_3 & 0 & 0 \\
n_y & m_1 & m_2 & m_2 & 0 & 0 \\
\frac{1-\rho h_p}{\rho h_T} & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1
\end{pmatrix}
\end{gathered} \tag{3.21}$$

where l_i and m_i are given by

$$\begin{aligned}
l_1 &= \eta_1 \left(n_y - n_z\right), \\
l_2 &= \eta_1 \left(n_z - n_x\right), \\
l_3 &= \eta_1 \left(n_x - n_y\right), \\
m_1 &= \eta_1 \left(\eta_2 n_x - 1\right), \\
m_2 &= \eta_1 \left(\eta_2 n_y - 1\right), \\
m_3 &= \eta_1 \left(\eta_2 n_z - 1\right), \\
\eta_1 &= \frac{1}{\sqrt{2}\sqrt{1 - n_x n_y - n_x n_z - n_y n_z}}, \\
\eta_2 &= n_x + n_y + n_z.
\end{aligned} \tag{3.22}$$

At this point, we adapt the framework of Merkle et al. [10] to the present high-order strand framework with a few modifications. First, a closer examination of the Euler equations using asymptotic analysis [79], leads to an "optimal" preconditioned speed of sound, $V_p = M_p c$, where

$$M_p = \begin{cases} \sqrt{\frac{2M^2}{1-2M^2}} & M < .5 \\ 1 & M \geq .5 \end{cases}, \tag{3.23}$$

where M is the local Mach number. When $M > .5$ then the standard compressible eigenvalues are returned. Otherwise, the preconditioner becomes active at low speeds.

Second, we take advantage of the strand grid structure to help determine when to turn the preconditioner off in the presence of large pressure perturbations or viscous effects. Because pressure is nearly constant normal to the wall in boundary layer regions, we expect large pressure perturbations will typically appear in $r-s$ planes only. Therefore, we measure pressure perturbations along edges in $r-s$ planes. At an edge connecting nodes i and j in an $r-s$ plane, we compute the pressure perturbation as

$$\frac{\left|p_i - p_j\right|}{\frac{1}{2}\left(\rho_i + \rho_j\right)}. \tag{3.24}$$

Third, along the same line of reasoning, we expect viscous diffusion to be dominant along strands (η-direction). Therefore, we estimate the viscous diffusion velocity with

$$\frac{\nu}{\Delta\eta}\frac{\sqrt{\eta_x^2+\eta_y^2+\eta_z^2}}{J}. \tag{3.25}$$

This quantity scales such that it becomes large for the high aspect ratio cells used near wall boundaries.

Combining Equations 3.23-3.25, we compute the preconditioned sound speed within the strand grid as

$$V_p = \begin{cases} \max\left(\sqrt{\frac{2M^2}{1-2M^2}}, \sqrt{\frac{\left|p_i-p_j\right|}{\frac{1}{2}\left(\rho_i+\rho_j\right)}}, \frac{\nu}{\Delta\eta}\frac{\sqrt{\eta_x^2+\eta_y^2+\eta_z^2}}{J}\right) & M < .5 \\ c & M \geq .5 \end{cases}, \tag{3.26}$$

where M is the local Mach number. We have experienced reasonably good convergence and accuracy for the test cases examined, which we discuss in the next section.

3.3 Results

We now investigate the high-order preconditioned strand grid methodology described in the previous section with some representative cases. First, we verify the order of accuracy of the method via the method of manufactured solutions (MMS) [49]. Next, we provide a laminar flow validation case for steady flow over a sphere. Finally, we present a number of turbulent validation cases involving bodies of revolution (BOR), which are representative submersible configurations.

3.3.1 MMS Verification

The method of manufactured solutions is an effective method for solution verification [49, 51] of complex equations for which exact solutions are difficult or impossible to obtain. With MMS, a solution is determined *a priori* and forced via source terms. An example manufactured solution based on smooth trigonometric functions is shown in Figure 3.2. The degree to which the discrete solution departs from the exact solution as the mesh is refined may then be used to verify the order of accuracy of the scheme. Using this method, proper implementation of additional features, such as a preconditioner, can be verified. Verification is a necessary step prior to validation, which we subsequently explore.

Using the solution and geometry in Figure 3.2, we solve the governing equations with and without the turbulence model using four meshes with 36,864, 73,728, 147,456, and 294,912 nodes. In each case, we run the study with and without the viscous fluxes. The results, shown in Figure 3.3, indicate third-order accuracy when using the FC scheme with high-order SBP operators. These results match those presented in earlier works [4, 5, 65].

3.3.2 Laminar Flow Validation

We next examine the accuracy of laminar flow predictions compared to experimental data and other established numerical solutions. We examine steady flow over a sphere at low speed and low Reynolds number. For this test we use a sphere with 256 fourth-order surface elements, resulting in 2304 surface nodes. We extend the strand length 20 diameters to the far field and use 128 nodes along each strand. The resulting grid contains roughly 300,000 nodes total. Low speed flow at $M = 0.002$ is imposed to determine the effectiveness of the preconditioner in terms of accuracy and convergence. The incompressible equation of state (ρ = constant) is used.

Both preconditioned and non-preconditioned results are shown in Figure 3.4 for $Re = 40$. It is evident that the non-preconditioned solution in Figure 3.4(b) possesses non-physical numerical artefacts in the pressure profile. This is due to the poor scaling of the standard compressible pressure dissipation at low Mach number. The resulting velocity profile is also overly diffuse, as indicated by the streamlines in the figure. In Table 3.1, we observe that only the preconditioned system can correctly predict separation angle and recirculation length at low Mach number. In fact, without preconditioning there is no recirculation region predicted at all. Convergence in pseudo-time is also notably faster using the preconditioned system as shown in Figure 3.5.

Table 3.1: Comparison of preconditioned and non-preconditioned solutions for flow around a sphere, $M = 0.002$, $Re = 40$.

	Preconditioned	Non-preconditioned	Experimental
Recirculation length	0.285	0.00	0.29
Angle of Separation (deg)	144.0	NA	144.1

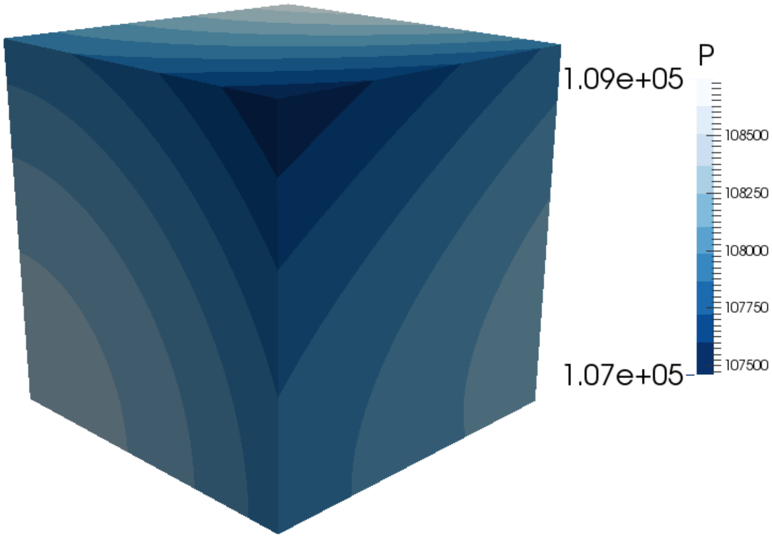
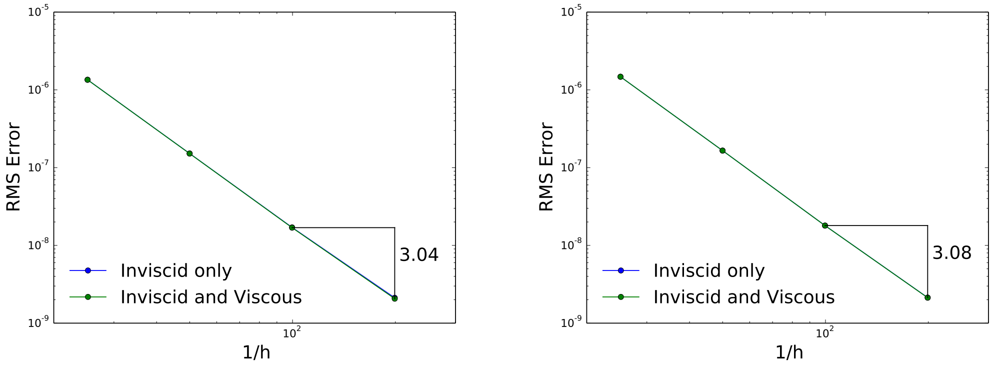
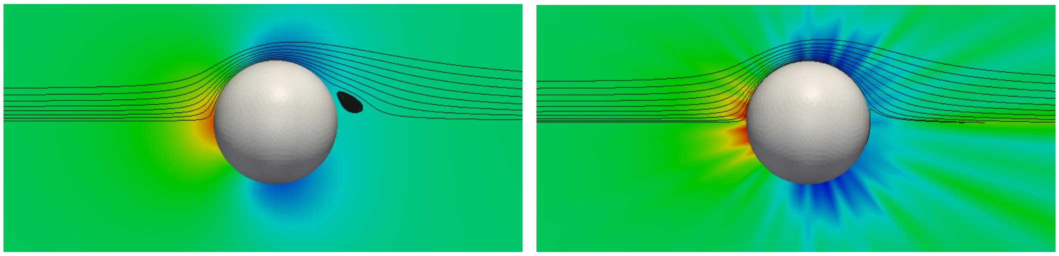


Fig. 3.2: Example manufactured solution.



(a) MMS results for laminar preconditioned FC (b) MMS results for turbulent SA preconditioned FC

Fig. 3.3: MMS results for laminar and turbulent flow grid convergence studies.



(a) Preconditioned (b) Non-preconditioned

Fig. 3.4: Pressure and streamlines for flow around a sphere, $M = 0.002$, $Re = 40$.

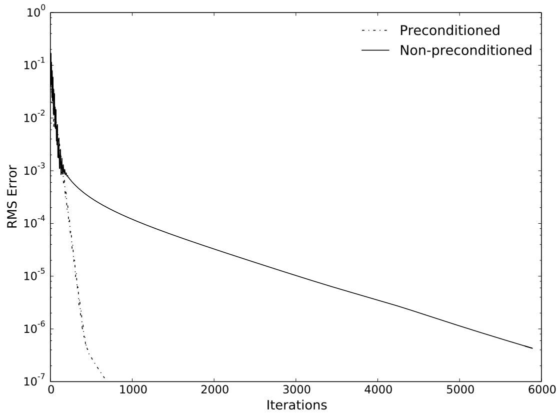
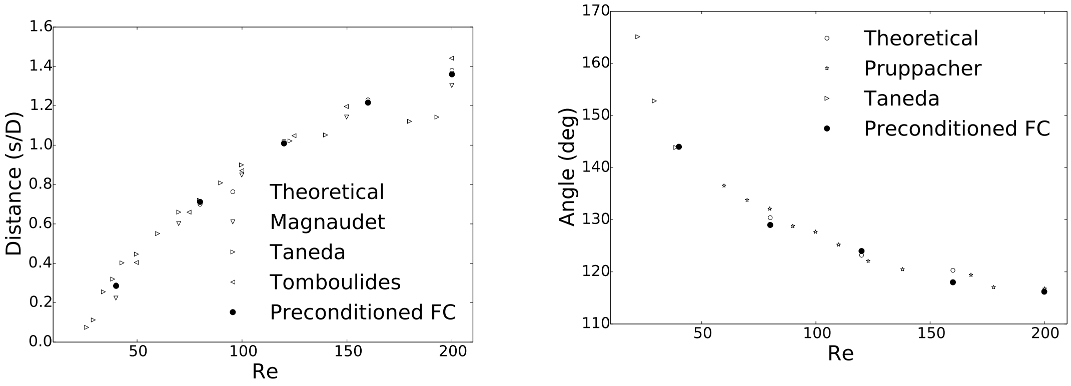


Fig. 3.5: Convergence of u-momentum with preconditioned and non-preconditioned systems for flow around a sphere, $M = 0.002$, $Re = 40$.



(a) Recirculation length

(b) Angle of separation

Fig. 3.6: Flow characteristics for flow over a sphere over a range of low Reynolds number.

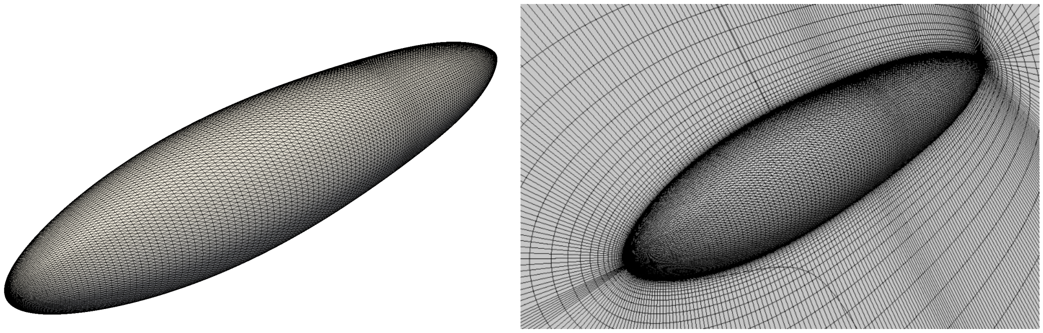
We extend this case to a series of low Reynolds numbers for which we have a large amount of numerical and experimental results for comparison. Again, we focus on recirculation length and angle of separation. We examine results at $Re = 40, 80, 120, 160, 200$. We then compare our results to the computational results of Magnaudet [80] and Tomboulides [81], and to the experimental results of Pruppacher [82] and Taneda [83]. The resulting comparison is shown in Figure 3.6, which displays good correlation with the other experimental and numerical results.

3.3.3 Turbulent Flow Validation

To validate the preconditioned solver capabilities to model turbulence, we simulate low Mach number flow around three bodies of revolution (BOR), which are representative submersible configurations. These bodies provide good candidates for application as separation is minimal, thus the results of FC will not be dwarfed by the errors in the turbulence model. The three geometries were obtained from the work of Gertler [70], Van Randwijck and Feldman [84], and Huang [85]. Each configuration highlights a particular solver capability: (1) Model 4155 for straight ahead flow, (2) Model 4621 for flow at angles, and (3) Body 1 for after-body pressure prediction.

3.3.4 Gertler Series Model 4155

Model 4155 is a configuration from work by Gertler [70]. The submersible-shaped body has a length of $2.7432m$ and a maximum radius of $0.27432m$. Flow velocities were varied to achieve a sweep of Reynolds numbers from 5 million to 20 million. The experimental results were obtained



(a) Surface mesh

(b) Volume mesh

Fig. 3.7: Model 4155 mesh.

in water at relatively low speeds, and so we impose an incompressible flow condition. The surface mesh and volume mesh, shown in Figure 3.7, consists of 11,760 fourth-order elements with 64 nodes along each strand, resulting in around 3 million cells total. Strand extend a distance 20m away from the body surface.

The drag coefficient at each Reynolds number was calculated with

$$C_d = \frac{F_d}{\frac{1}{2}\rho V^2 S}, \tag{3.27}$$

where F_d is the total drag force, V is the free stream velocity, and S is the wetted surface area of the body. Both second-order and third-order results are shown. The third-order results are obtained with the FC method and high-order SBP strand operators described in this work. The second-order results are obtained with a conventional finite volume (FV) method with the high-order flux reconstruction turned off, as well as second-order finite differences along strands.

Generally, the computational results agree with the experimental results of Gertler, who presents drag coefficients with and without a small patch of surface sand near the BOR nose to ensure fast transition to turbulence, as shown in Figure 3.8. The discrepancies between the FV and FC schemes and the computational and experimental results are likely due to differences in laminar to turbulent transition along the smooth body. The Spalart-Allmaras model used does not attempt to model transition and makes use of fully turbulent inflow conditions. The experimental results, on the other hand, likely consist of a combination of laminar and turbulent flow regimes along the body, with the sand case providing a more definite turbulence trip location. Modelling transition continues to be a challenge to CFD algorithms in general. Even so, the trends obtained computationally match quite well with the experimental measurements.

3.3.5 Gertler Series Model 4621

Model 4621 was presented in the work of Van Randwijck and Feldman [84], and is a more slender BOR than the previous Model 4155 case. The body length is 4.572m with a maximum diameter of 0.623m. This body was experimentally tested at a Reynolds number of 4.7 million at angles of attack up to 18 degrees. Again, the experiment was performed in water, which we assume

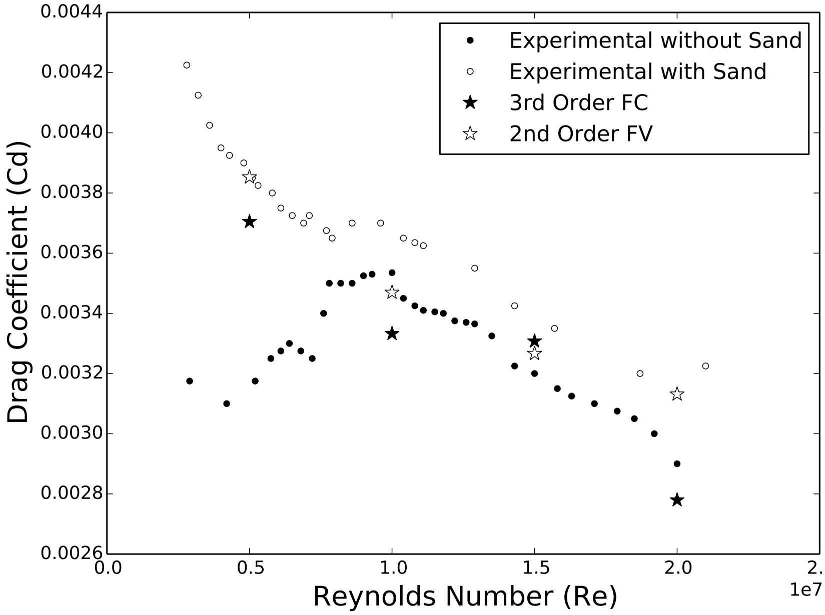
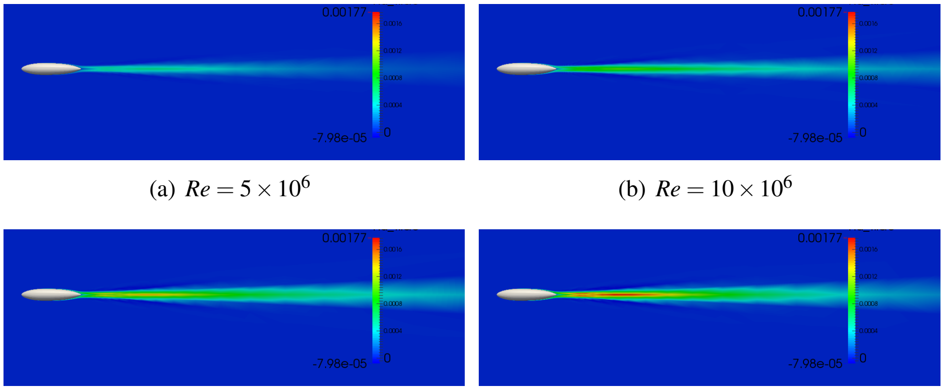


Fig. 3.8: Model 4155 drag computation for various Reynolds numbers.



(a) $Re = 5 \times 10^6$

(b) $Re = 10 \times 10^6$

(c) $Re = 15 \times 10^6$

(d) $Re = 20 \times 10^6$

Fig. 3.9: Model 4155 turbulent viscosity.

is incompressible in the simulations. Unlike the Model 4155 case, the large angles of attack for this case lead to large regions of separation near the aft portion of the BOR.

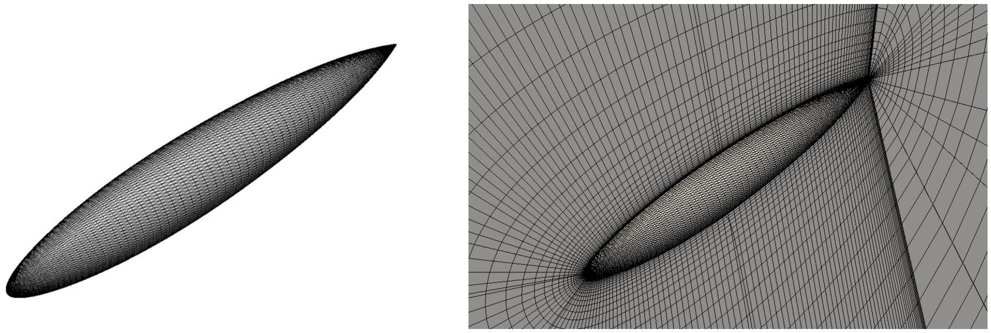
The mesh for Model 4621 consists of 3,840 fourth-order elements on the surface with 64 nodes along each strand, leading to around 1 million cells total. The surface and volume mesh is shown in Figure 3.10.

The normal coefficient was calculated for angles of attack 2,4,10, and 18 deg. as

$$Z = \frac{F_n}{\frac{1}{2}\rho V^2 S},$$

where S is the body length squared, and F_n is the component of the force normal to the free stream direction. Computational results compared to the experiment are shown in Figure 3.11. In each case, the normal force magnitude is under predicted, with errors increasing as the angle of attack increases. A slightly better prediction is obtained at higher angles of attack with the FC scheme compared to the FV scheme.

While the trend in Figure 3.11 is consistent with the experimental data, the predictions at high angles of attack are error-prone. Upon examination of the pressure and velocity fields at 18 deg., shown in Figure 3.12, it is evident that a large region of separation is present near the aft end of the BOR. It is well-known that most turbulence models, and certainly the Spalart-Allmaras model, do not perform reliably in the presence of large regions of separation [86], especially occurring along smooth bodies. Future work will focus on testing other turbulence models including a variety of



(a) Surface mesh

(b) Volume mesh

Fig. 3.10: Model 4621 mesh.

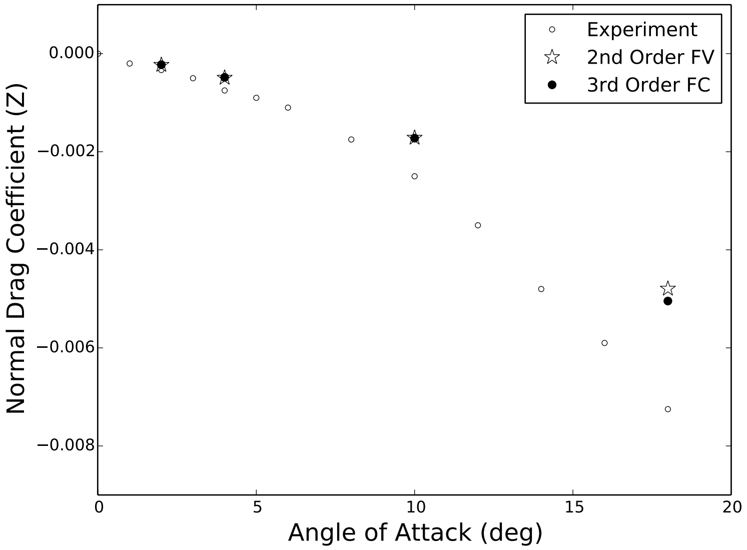
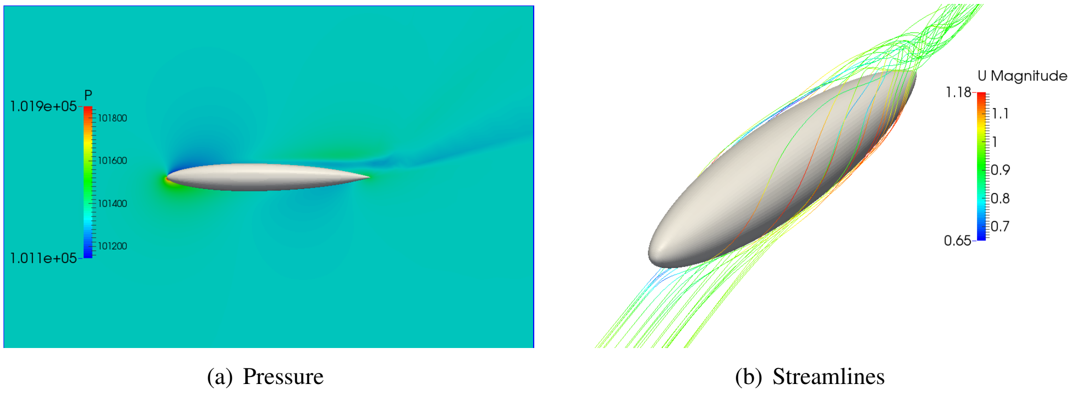


Fig. 3.11: Normal force coefficient for Model 4621 at various angles of attack at Reynolds number of 4.4 million.



(a) Pressure

(b) Streamlines

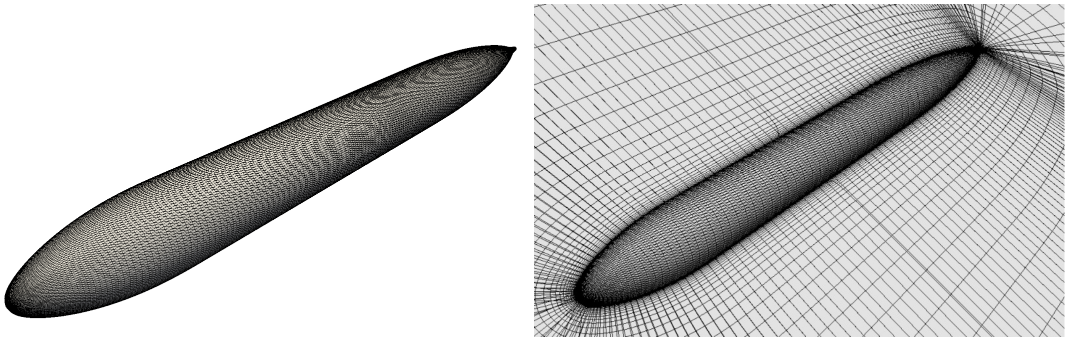
Fig. 3.12: Model 4621 at 18 deg. angle of attack.

two-equation models.

3.3.6 Body 1

The geometry known as "Body 1" as detailed by Huang [85], consists of a cylindrical middle body, with axisymmetric afterbody and forebody attachments. The total length of the entire body is 3.066m, with a maximum diameter of 0.2794m. The surface mesh used consists of 3,840 of fourth-order elements with 64 nodes along each strand in a hyperbolic tangent distribution, leading to around 1 million cells total. The surface and volume meshes are shown in Figure 3.13.

The original experiments of Huang [85] were performed in a wind tunnel at $M = 0.09$ and $Re = 6.6e6$, and the flow can be assumed to be incompressible. Pitot tube pressure measurements were obtained along the afterbody and near wake region at distances of 2.316m, 2.594m, 2.804m, and 3.241m measured from the body nose, as shown in Figure 3.14. The last measurement is completely in the wake region of the body. Pressure was measured radially outward from the body a distance R/R_{max} where R_{max} is the maximum radius of the body. In addition, pressure was measured on the surface of the afterbody for the last third of the body length. Measurement errors for the pressure on the surface of the body were reportedly less than 0.5% of the dynamic pressure, and off the body the measurements were reportedly less than .8% of the dynamic pressure. The velocity was reportedly measured within ±.15 m/s. Location of the pitot tube was measured within 0.01mm. With these uncertainties the pressure coefficient uncertainty is estimated to be .9%, uncertainty



(a) Surface Mesh

(b) Volume Mesh

Fig. 3.13: Body 1 Meshes.

analysis can be found in Appendix C. The pressure coefficient is determined from

$$C_p = \frac{P - P_\infty}{\frac{1}{2}\rho V^2}, \tag{3.28}$$

where P_∞ is the ambient pressure and P is the local pressure.

Figure 3.15 shows the computed surface pressure coefficient along the afterbody along with the experimental data with uncertainty bands. Excellent agreement is observed along most of the afterbody, with some oscillation near the trailing edge, which is more pronounced with the third-order FC scheme. Pressure coefficient extracted radially at the stations shown in Figure 3.14 is displayed in Figure 3.16. Experimental uncertainty bands are also shown. Generally good agreement in trends are observed with the data. We expect further improvements in accuracy near the wake region with the addition of off-body adaptive Cartesian grids. Previous work has revealed the importance of fine wake resolution through the use of Cartesian grids because the strands tend to spread out quite fast near the trailing tip [2].

3.4 Conclusions

In this paper, we address certain challenges associated with the computation of low speed and incompressible flows through the use of automated strand grid generation, unique high-order methods, and preconditioning. Strand grids provide fully automatic volume grid generation from a

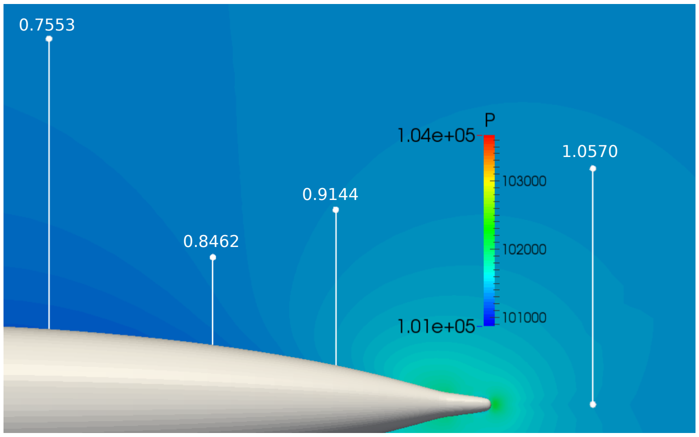


Fig. 3.14: Body 1 afterbody pressure measurement stations.

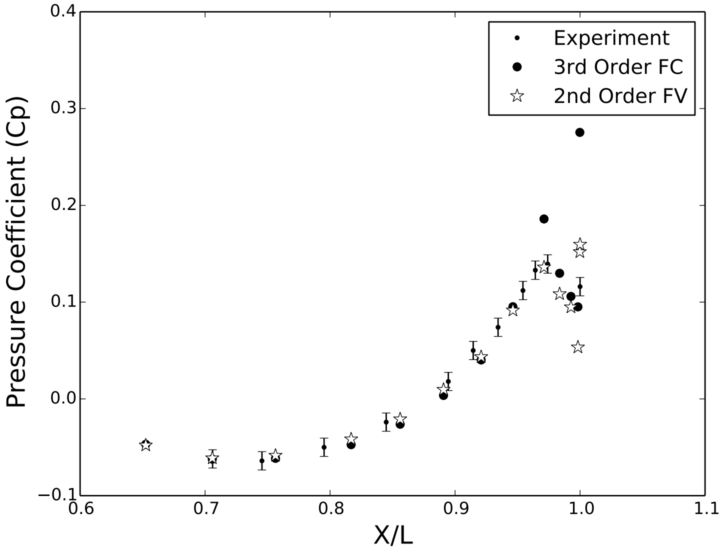


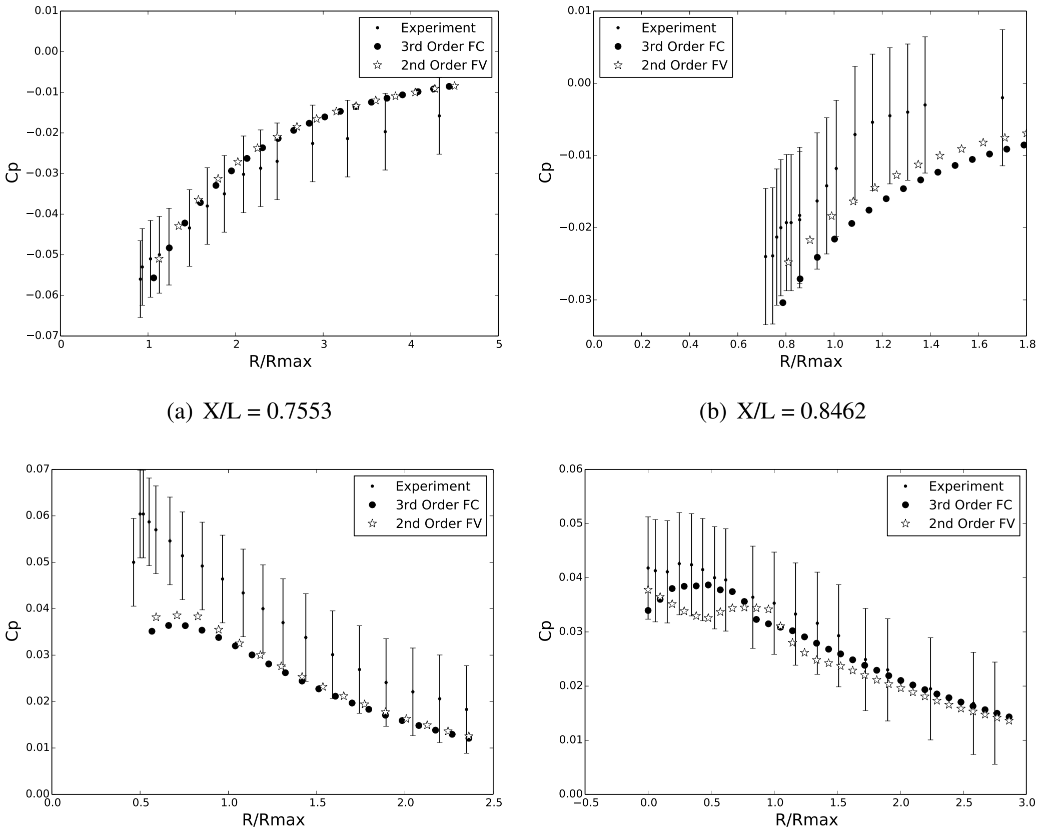
Fig. 3.15: Body 1 afterbody surface pressure for last third of the body

surface mesh and provide a framework for massive scalability due to a compact grid representation. We formulated a unique high-order algorithm for strand grids using flux correction finite volume balances in combination with high-order summation-by-parts operators along strands. To extend the method to low speed and incompressible flows, we explore preconditioning techniques which take advantage of the strand grid structure.

Using the method of manufactured solutions, we verify the correct implementation of the preconditioned high-order strand method. Grid refinement studies indicate third-order accuracy for both inviscid and viscous terms, with and without the Spalart-Allmaras turbulence model.

Validation is first performed for steady laminar flow over a sphere. This case highlights the improvements in accuracy and convergence using the preconditioner. Without the preconditioner, spurious pressure oscillations and poor velocity prediction are observed. With the preconditioner, separation angle and reattachment length are predicted very accurately.

Validation is next performed for incompressible turbulent flow over a series of bodies of revolution. In all cases, trends are captured accurately. However, discrepancies sometimes appear in cases involving transition and smooth-body separation. In these cases, the improved order of accuracy does not provide much benefit, indicating that dominant errors may be attributed to turbulence



(a) X/L = 0.7553

(b) X/L = 0.8462

(c) X/L = 0.9144

(d) X/L = 1.0570

Fig. 3.16: Body 1 afterbody pressure coefficient profiles at various stations.

modeling of these flow features. These results highlight the fact that the order of accuracy is only one of many factors impacting overall fidelity of solutions. Future work will focus on validating similar cases with other turbulence models to improve algorithm performance.

Chapter 4
Conclusion

High-order methods provide many advantages. They provide better solution accuracy and dissipate solutions less than low-order methods. However, they also can take longer to converge and are less stable than their counterparts. In this work we looked at FC in particular, which obtains high-order results by modifying the truncation error of the second-order Galerkin method. This provides a simple correction term that can be implemented to currently existing codes. By analyzing the truncation error of the method in conjunction with source terms we can conclude the proper form of the source terms. We have observed that FC can be extended to incompressible flows by use of preconditioning schemes.

Truncation error analysis shows that FC has second-order truncation error on arbitrary grids and third-order truncation error on regular grids. This leads to FC being formally third-order on arbitrary meshes. However, proper source term discretization must be utilized to retain the high-order solution error. Solution error can be reduced and computational efficiency retained by using proper source term discretization.

Although strand grids are useful for meshing curved surfaces, they lack the ability to produce adequate mesh quality for the body of revolutions. Because of this, it is suggested that a Cartesian mesh be combined with the strand mesh to produce more stable solutions for FC in particular. The strand mesh provides great resolution close to the body, while the simplicity of a Cartesian off-body mesh provides resolution in the off-body. We speculate that the combination of the two methods produce better solution convergence and more accurate answers.

There are many challenges facing modern CFD methods. Among the list given by Wang [1] is the need for better methods to produce high-order meshing. Strand-Cartesian grids show potential in alleviating the task of producing high-order meshes. A high-order method has been shown to be effective in low-Mach and incompressible flow. This extension was enabled by the addition of a preconditioning scheme. In addition, a couple SA turbulence model was coupled with the solution variables. These methods showed viability by providing adequate results for a series of bodies of revolution.

Because of the addition of a preconditioner, many new avenues have been opened to future research for FC. Using the arbitrary equation of state preconditioning scheme has enabled the addition of chemical reactions and combustion simulations. In addition, the extension of FC to incompressible flows allows application to many more real world problems, such as shown with the bodies of revolution. The ability to couple turbulence models into the equations also provides advantages over the duel-step methods. Further turbulence models can also be applied in this way, increasing accuracy for different types of simulations. By understanding source terms and their effects on FV schemes, such as FC, and by the addition of a preconditioning scheme, FC has become a more viable option for future research and implementation.

References

[1] Wang, Z., Fidkowski, K., Abgrall, R., Bassi, F., Caraeni, D., Cary, A., Deconinck, H., Hartmann, R., Hillewaert, K., and Huynh, H., "High-order CFD methods: current status and perspective," *International Journal for Numerical Methods in Fluids*, Vol. 72, No. 8, 2013, pp. 811–845.

[2] Work, D., Tong, O., Workman, R., Katz, A., and Wissink, A., "Strand Grid Solution Procedures for Sharp Corners," *AIAA Journal*, Vol. 52, No. 7, 2014, pp. 1528–1541.

[3] Katz, A., Wissink, A., Sankaran, V., Meakin, R., and Sitaraman, J., "Application of Strand Meshes to Complex Aerodynamic Flow Fields," *Journal of Computational Physics*, Vol. 230, 2011, pp. 6512–6530.

[4] Katz, A. and Sankaran, V., "An Efficient Correction Method to Obtain a Formally Third-Order Accurate Flow Solver for Node-Centered Unstructured Grids," *Journal of Scientific Computing*, Vol. 51, No. 2, 2012, pp. 375–393.

[5] Katz, A. and Work, D., "High-Order Flux Correction/Finite Difference Schemes for Strand Grids," *Journal of Computational Physics*, Vol. 282, 2015, pp. 360–380.

[6] LeVeque, R., *Numerical Methods for Conservation Laws*, Birkhauser, Basel/Boston/Berlin, 1992, pp. 144-145.

[7] Godlewski, E. and Raviart, P. A., *Numerical Approximation of Hyperbolic Systems of Conservation Laws, of Applied Mathematical Sciences*, Springer-Verlag, New York, 1996.

[8] Toro, E., *Riemann Solvers and Numerical Methods for Fluid Dynamics. A Practical Approach*, Springer-Verlag, New York, 1997.

[9] Chorin, A., "A Numerical Method for Solving Incompressible Viscous Flow Problems," *Journal of Computational Physics*, Vol. 2, 1967, pp. 12–26.

[10] Charles L. Merkle, Jennifer Y. Sullivan, P. E. O. B. and Venkateswaran, S., "Computation of Flows with Arbitrary Equations of State," *AIAA Journal*, Vol. 36, No. 4, 1998, pp. 512–521.

[11] Darmofal, D. and Siu, K., "A Robust Multigrid Algorithm for the Euler Equations with Local Preconditioning and Semi-coarsening," *Journal of Computational Physics*, Vol. 151, No. 2, 1999, pp. 728 – 756.

[12] Venkateswaran, S. and Merkle, C. L., "Analysis of Preconditioning Methods for the Euler and Navier-Stokes Equations," Von Karman Institute lecture series, University of Tenessee Space Institute, Tullahoma, TN, 1999.

[13] Meakin, R., Wissink, A., Chan, W., Pandya, S., and Sitaraman, J., "On Strand Grids for Complex Flows," *AIAA paper* 2007-3834, AIAA 18th Computational Fluid Dynamics Conference, Miami, FL, June 2007.

[14] Wang, Z. J., "High-order Methods for the Euler and Navier-Stokes Equations on Unstructured Grids," *Progress in Aerospace Sciences*, Vol. 43, 2007, pp. 1–41.

[15] Vincent, P. and Jameson, A., "FaciDynamics the Adoption of Unstructured High-order Methods Amongst a Wider Community of Fluid Dynamicists," *Mathematical Modelling of Natural Phenomena*, Vol. 6, No. 3, 2011, pp. 97–140.

[16] Jameson, A., "Advances in Bringing High-order Methods to Practical Applications in Computational Fluid Dynamics," *20th AIAA Computational Fluid Dynamics Conference*, Vol. AIAA 2011-, No. 3226, June 2011.

[17] Versteeg, H. K. and Malalasekera, W., *An Introduction to Computational Fluid Dynamics: The Finite Volume Method*, Pearson Education Ltd, 2nd ed., 2007, ISBN 0-582-21884-5.

[18] Barth, T. J., "Recent Developments in High Order k-exact Reconstruction on Unstructured Meshes," *AIAA paper*, Vol. 93, No. 993, 1993, pp. 0668.

[19] Li, W. and Ren, Y.-X., "High-order k-exact WENO finite volume schemes for solving gas dynamic Euler equations on unstructured grids," *International Journal for Numerical Methods in Fluids*, Vol. 70, 2011, pp. 742–763.

[20] Abgrall, R., "On Essentially Non-oscillatory Schemes on Unstructured Meshes: Analysis and Implementation," Tech. rep., ICASE Report 92-74, National Aeronautics and Space Administration, December 1992.

[21] Jiang, G.-S. and Shu, C.-W., "Efficient Implementation of Weighted ENO Schemes," *Journal of Computational Physics*, Vol. 126, No. 0130, June 1996, pp. 202–228.

[22] Shu, C.-W., "High-order Finite Difference and Finite Volume WENO Schemes and Discontinuous Galerkin Methods for CFD," *International Journal of Computational Fluid Dynamics*, Vol. 17, No. 2, September 2003, pp. 107–118.

[23] Dick, E., "Introduction to Finite Element Methods in Computational Fluid Dynamics," *Computational Fluid Dynamics*, edited by J. Wendt, Springer Berlin Heidelberg, 2009, pp. 235–274.

[24] Sherwin, S. J. and Karniadakis, G. E., "A New Triangular and Tetrahedral Basis for High-order (hp) Finite Element Methods," *International Journal for Numerical Methods in Engineering*, Vol. 38, No. 22, November 1995, pp. 3775–3802.

[25] Liu, Y., Vinokur, M., and Wang, Z., "Spectral (finite) volume method for conservation laws on unstructured grids V: Extension to three-dimensional systems," *Journal of Computational Physics*, Vol. 212, No. 2, 2006, pp. 454 – 472.

[26] Sun, Y., Wang, Z. J., and Liu, Y., "High-order Multidomain Spectral Difference Method for the Navier-Stokes Equations on Unstructured Hexahedral Grids," *Communications in Computational Physics*, Vol. 2, No. 2, 2007, pp. 310–333.

[27] Huynh, H. T., "A Reconstruction Approach to High-order Schemes Including Discontinuous Galerkin Methods," *18th AIAA Computational Fluid Dynamics Conference*, American Institute of Aeronautics and Astronautics, January 2007.

[28] Toro, E. and Titarev, V., “ADER Schemes for Scalar Hyperbolic Conservation Laws with Source Terms in Three Space Dimensions,” *Journal of Computational Physics*, Vol. 202, No. 1, January 2005, pp. 196–215.

[29] Toro, E. and Titarev, V., “Derivative Riemann Solvers for Systems of Conservation Laws and ADER Methods,” *Journal of Computational Physics*, Vol. 212, No. 1, February 2006, pp. 150–165.

[30] Dumbser, M., Enaux, C., and Toro, E., “Finite Volume Schemes of Very High Order of Accuracy for Stiff Hyperbolic Balance Laws,” *Journal of Computational Physics*, Vol. 227, No. 8, 2008, pp. 3971–4001.

[31] Vazquez-Cendon, M., “Improved Treatment of Source Terms in Upwind Schemes for the Shallow Water Equations in Channels with Irregular Geometry,” *Journal of Computational Physics*, Vol. 148, No. 2, January 1999, pp. 497–526.

[32] Bermudez, A. and Vazquez, M. E., “Upwind Methods for Hyperbolic Conservation Laws with Source Terms,” *Computers & Fluids*, Vol. 23, No. 8, 1994, pp. 1049.

[33] Bermudez, A., Dervieux, A., Desideri, J. A., and Vazquez, M. E., “Upwind Schemes for the Two-dimensional Shallow Water Equations with Variable Depth using Unstructured Meshes,” *Computational Methods Applied to Mechanical Engineering*, Vol. 155, No. 49, March 1998, pp. 49–72.

[34] Murillo, J. and Garcia-Navarro, P., “Weak Solutions for Partial Differential Equations with Source Terms: Application to the Shallow Water Equations,” *Journal of Computational Physics*, Vol. 229, No. 11, June 2010, pp. 4327–4368.

[35] Leveque, R. J., *Finite Volume Methods for Hyperbolic Problems*, Cambridge University Press, New York, 2002.

[36] Liang, Q. and Marche, F., “Numerical resolution of well-balanced shallow water equations with complex source terms,” *Adv. Water Resour.*, Vol. 32, 2009, pp. 873–884.

[37] Rogers, B., Borthwick, A., and Taylor, P., “Mathematical balancing of flux gradient and source terms prior to using Roe’s approximate Riemann solver,” *J. Comput. Phys.*, Vol. 168, 2003, pp. 422–451.

[38] Toro, E., *Shock-Capturing Methods for Free-Surface Shallow Flows*, Wiley, New York, 2001.

[39] Zhou, J., Causon, D., Mingham, C., and Ingram, D., “The Surface Gradient Method for the Treatment of Source Terms in the Shallow-water Equations,” *J. Comput. Phys.*, Vol. 168, 2001, pp. 1–25.

[40] Turkel, E., “Preconditioning Techniques in Computational Fluid Dynamics,” *Annual Review of Fluid Mechanics*, Vol. 31, No. 1, 1999, pp. 385–416.

[41] Choi, Y.-H. and Merkle, C., “The Application of Preconditioning in Viscous Flows,” *Journal of Computational Physics*, Vol. 105, No. 2, 1993, pp. 207 – 223.

[42] Weiss, J. and Smith, W., "Preconditioning applied to variable and constant density flows," *AIAA Journal*, Vol. 33, No. 11, November 1995, pp. 2050–2057.

[43] Colin, Y., Deniau, H., and Boussuge, J.-F., "A robust low speed preconditioning formulation for viscous flow computations," *Computers & Fluids*, Vol. 47, No. 1, 2011, pp. 1 – 15.

[44] Turkel, E., "Preconditioned Methods for Solving the Incompressible and Low Speed Compressible Equations," *Journal of Computational Physics*, Vol. 72, No. 2, October 1987, pp. 277–298.

[45] Spalart, P. and Allmaras, S., "A One-equation Turbulence Model for Aerodynamic Flows," *Recherche Aerospatiale*, Vol. 1, No. 1, 1994, pp. 5–21.

[46] Spalart, P., "Strategies for Turbulence Modelling and Simulations," *International Journal of Heat and Fluid Flow*, Vol. 21, No. 3, 2000, pp. 252 – 263.

[47] Allmaras, S., Johnson, F., and Spalart, P., "Modifications and Clarifications for the Implementation of the Spalart-Allmaras Turbulence Model," Tech. Rep. ICCFD7-1902, 7th International Conference on Computational Fluid Dynamics, July 2012.

[48] Roache, P., "Verification of Codes and Calculations," *AIAA Journal*, Vol. 36, No. 0001-1452, May 1998, pp. 696–702.

[49] Roache, P., "Code Verification by the Method of Manufactured Solutions," *Transactions of the ASME*, Vol. 124, 2002, pp. 4–10.

[50] Salari, K. and Knupp, P., "Code Verification by the Method of Manufactured Solutions," Tech. Rep. SAND2000-1444, Sandia National Laboratories, June 2000.

[51] Roy, C. J., "Review of Code and Solution Verification Procedures for Computational Simulation," *Journal of Computational Physics*, Vol. 205, 2005, pp. 131–156.

[52] Eca, L., Hoekstra, M., Hay, A., and D., P., "A Manufactured Solution for a Two-dimensional Steady Wall-bounded Incompressible Turbulent Flow," *International Journal of Computational Fluid Dynamics*, Vol. 21, 2007, pp. 175–188.

[53] Katz, A. and Sankaran, V., "Mesh Quality Effects on the Accuracy of Euler and Navier-Stokes Solutions on Unstructured Meshes," *Journal of Computational Physics*, Vol. 230, No. 20, 2011, pp. 7670–7686.

[54] Katz, A. and Sankaran, V., "High Aspect Ratio Grid Effects on the Accuracy of Navier-Stokes Solutions on Unstructured Meshes," *Computers and Fluids*, Vol. 65, 2012, pp. 67–79.

[55] Folkner, D., Katz, A., and Sankaran, V., "Design and Verification Methodology of Boundary Conditions for Finite Volume Schemes," *Computers and Fluids*, Vol. 96, 2014, pp. 264–275.

[56] Delanaye, M. and Liu, Y., "Quadratic Reconstruction Finite Volume Schemes on 3D Arbitrary Unstructured Polyhedral Grids," *AIAA paper* 1995-3259, AIAA 14th CFD Conference, Norfolk, June 1999.

[57] Leveque, R. J., *Finite Volume Methods for Hyperbolic Problems*, Cambridge University Press, New York, 2002.

[58] Liang, Q. and Marche, F., “Numerical resolution of well-balanced shallow water equations with complex source terms,” *Adv. Water Resour.*, Vol. 32, 2009, pp. 873–884.

[59] Rogers, B., Borthwick, A., and Taylor, P., “Mathematical balancing of flux gradient and source terms prior to using Roe’s approximate Riemann solver,” *J. Comput. Phys.*, Vol. 168, 2003, pp. 422–451.

[60] Toro, E., *Shock-Capturing Methods for Free-Surface Shallow Flows*, Wiley, New York, 2001.

[61] Zhou, J., Causon, D., Mingham, C., and Ingram, D., “The surface gradient method for the treatment of source terms in the shallow-water equations,” *J. Comput. Phys.*, Vol. 168, 2001, pp. 1–25.

[62] Barth, T. J., “Numerical Aspects of Computing Viscous High Reynolds Number Flows on Unstructured Meshes,” *AIAA paper* 1991-0721, AIAA 29th ASM, Reno, Jan. 1991.

[63] Roe, P. L., “Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes,” *Journal of Computational Physics*, Vol. 43, 1981, pp. 357–372.

[64] Diskin, B. and Thomas, J., “Comparison of Node-Centered and Cell-Centered Unstructured Finite-Volume Discretizations: Inviscid Fluxes,” *AIAA paper* 2010-1079, AIAA 48th Aerospace Sciences Meeting, Orlando, FL, January 2010.

[65] Pincock, B. and Katz, A., “High-Order Flux Correction for Viscous Flows on Arbitrary Unstructured Grids,” *Journal of Scientific Computing*, Vol. 61, No. 2, 2014, pp. 454–476.

[66] Jameson, A., Baker, T., and Weatherill, N., “Calculation of Inviscid Transonic Flow over a Complete Aircraft,” *AIAA paper* 86-0103, AIAA 24th Aerospace Sciences Meeting, Reno, January 1986.

[67] Shapiro, A. H., *The Dynamics and Thermodynamics of Compressible Fluid Flow*, Vol. 2, The Ronald Press Company, 1954.

[68] Chiocchia, G., “Exact Solutions to Transonic and Supersonic Flows,” Tech. Rep. AR-211, AGARD, 1985.

[69] Folkner, D., Katz, A., and Sankaran, V., “An Unsteady Preconditioning Scheme Based on Convective-upwind Split-pressure Artificial Dissipation,” *International Journal for Numerical Methods in Fluids*, Vol. 78, No. 1, January 2015, pp. 1–16.

[70] Gertler, M., “Resustance Experiments on a Systematic Series of Streamlined Bodies of Revolution - for Application to the Design of High-speed Submarines,” Technical Report Report C-297, Navy Department, The David W. Taylor Model Basin, 1950.

[71] Kreiss, H. and Scherer, G., “Finite element and finite difference methods for hyperbolic partial differential equations,” *Mathematical Aspects of Finite Elements in Partial Differential Equations*, edited by C. D. Boor, Academic Press, 1974.

[72] Strand, B., “Summation by parts for finite difference approximation for d/dx,” *Journal of Computational Physics*, Vol. 110, 1994, pp. 47–67.

[73] Carpenter, M. H., Gottlieb, D., and Abarbanel, S., "The Stability of Numerical Boundary Treatments for Compact High-Order Finite-Difference Schemes," *Journal of Computational Physics*, Vol. 108, No. 2, 1993, pp. 272 – 295.

[74] Mattsson, K., "Summation by Parts Operators for Finite Difference Approximations of Second-Derivatives with Variable Coefficients," *Journal of Scientific Computing*, Vol. 51, 2012, pp. 650–682.

[75] Fernandez, D. C. D. R. and Zingg, D., "High-Order Compact-Stencil Summation-By-Parts Operators for the Second Derivative with Variable Coefficients," Tech. Rep. ICCFD7-2803, 7th International Conference on Computational Fluid Dynamics (ICCFD7), Big Island, HI, July 2012.

[76] Carpenter, M., Gottlieb, D., and Abarbanel, S., "Time-Stagle Boundary Conditions for Finite-Difference Schemes Solving Hyperbolic Systems: Methodology and Application to High-Order Compact Schemes," ICASE Report 93-9, ICASE, Hampton, VA, March 1993.

[77] Svärd, M., Carpenter, M., and Nordström, J., "A Stable High-order Finite Difference Scheme for the Compressible NavierStokes Equations, Far-field Boundary Conditions," *Journal of Computational Physics*, Vol. 225, 2007, pp. 1020–1038.

[78] Svärd, M. and Nordström, J., "A Stable High-order Finite Difference Scheme for the Compressible NavierStokes Equations, No-slip Wall Boundary Conditions," *Journal of Computational Physics*, Vol. 227, 2008, pp. 4805–4824.

[79] Venkateswaran, S., "Private Communication," 2013.

[80] Magnaudet, J., Riviero, M., and Fabre, J., "Accelerated flows past a sphere or spherical bubble. part 1: Steady streaming flow," *Journal of Fluid Mechanics*, Vol. 284, 1995, pp. 97–135.

[81] Tomboulides, A., Orszag, S., and Karniadakis, G., "Direct and Large Eddy Simulations of Axisymmetric Wakes," *AIAA paper*, No. 93-0546, 1993.

[82] Pruppacher, H., Clair, B., and Hamielec, A., "Some Relations Between Drag and Flow Pattern of Viscous Flow Past a Sphere and Cylinder at Low and Intermediate Reynolds Numbers," *Journal of Fluid Mechanics*, Vol. 44, 1970, pp. 781–791.

[83] Taneda, S., "Experimental Investigation of Wake Behind a Sphere at low Reynolds Numbers," *J. Physical Society of Japan*, Vol. 11, 1956, pp. 1104–1108.

[84] Van Randwijck, E. F. and Feldman, J. P., "Results of Experiments With a Segmented Model to Investigate the Distribution of the Hydrodynamic Forces and Moments on a Streamlined Body of Revolution at an Angle of Attack or with a Pitching Angular Velocity," Tech. rep., Carderock Division Naval Surface Warfare Center, 2000.

[85] Huang, T., Santelli, N., and Belt, G., "Stern boundary-layer flow on axisymmetric bodies," Tech. Rep. u211033, David W. Taylor Naval Ship Research and Development Center, Bethesda, Maryland 20084 USA, June 1978.

[86] Rumsey, C., "NASA Langley Turbulence Modeling Resource (http://turbmodels.larc.nasa.gov)," http://turbmodels.larc.nasa.gov, 2015 Dec 4.

Appendices

Appendix A

Truncation Error Analysis

Truncation error analysis of Flux Correction and the corrected source term in multiple dimensions. Some useful identities can be found using the summation in connection with the normals,

$$
\begin{aligned}
&\frac{1}{2}\sum_i \Delta x_{0i} n_{x0i} \mathscr{S}_{0i} = V_0 \\
&\frac{1}{2}\sum_i \Delta x_{0i} n_{y0i} \mathscr{S}_{0i} = 0 \\
&\frac{1}{2}\sum_i \Delta y_{0i} n_{x0i} \mathscr{S}_{0i} = 0 \\
&\frac{1}{2}\sum_i \Delta y_{0i} n_{y0i} \mathscr{S}_{0i} = V_0 \\
&\sum_i n_{x0i} \mathscr{S}_{0i} = \sum_i n_{y0i} \mathscr{S}_{0i} = 0
\end{aligned}
$$

$$
S_0^h = \frac{1}{V_0}\sum_i \frac{1}{2}\left(S_L + S_R\right)\mathbf{n}_{0i}\mathscr{S}_{0i}\Delta r \tag{A.1}
$$

$$
\begin{aligned}
S_L &= S_0 - \frac{1}{2}\Delta r^T \nabla^h S_0 - \frac{1}{8}\Delta r^T \nabla^{h2} S_0 \Delta r \\
S_R &= S_i - \frac{1}{2}\Delta r^T \nabla^h S_i - \frac{1}{8}\Delta r^T \nabla^{h2} S_i \Delta r
\end{aligned}
$$

Using Taylor series expansion to find S_i,

$$
\begin{aligned}
S_i = S_0 &+ \Delta x_{0i} S_{0x} + \Delta y_{0i} S_{0y} + \frac{\Delta x_{0i}^2}{2} S_{02x} + \Delta x_{0i}\Delta y_{0i} S_{0xy} + \frac{\Delta y_{0i}^2}{2} S_{02y} \\
&+ \frac{\Delta x_{0i}^3}{6} S_{03x} + \frac{\Delta x_{0i}^2 \Delta y_{0i}}{2} S_{02xy} + \frac{\Delta x_{0i}\Delta y_{0i}^2}{2} S_{0x2y} + \frac{\Delta y_{0i}^3}{6} S_{03y} + O\left(h^4\right)
\end{aligned}
$$

Breaking down S_R line by line to get the truncation error leads to,

$$\begin{aligned}\frac{1}{2}\Delta r^T\nabla^h S_i = \frac{\Delta x_{0i}}{2}S_{0x} + \frac{\Delta y_{0i}}{2}S_{0y} + \frac{\Delta x_{0i}^2}{2}S_{02x} \\ + \frac{\Delta y_{0i}^2}{2}S_{02y} + \Delta x_{0i}\Delta y_{0i}S_{0xy} + O\left(h^q\right) + O\left(h^3\right)\end{aligned}$$

$$\frac{1}{8}\Delta r^T\nabla^2 S_i\Delta r = \frac{\Delta x_{0i}^2}{8}S_{0,2x} + \frac{\Delta x_{0i}\Delta y_{0i}}{4}S_{0,xy} + \frac{\Delta y_{0i}^2}{8}S_{0,2y} + O\left(h^q\right) + O\left(h^3\right)$$

Evaluating Equation A.1 yields,

$$S_0^h + E_t = \frac{1}{V_0}\sum_i S_0 \cancelto{V_0}{\frac{1}{2}\mathbf{n}_{0i}\mathscr{S}_{0i}}\Delta r + O\left(h^1\right)$$

The proper source term is achieved, but further cancellation in the truncation error can be found,

$$\begin{aligned}E_t = \frac{1}{2V_0}\sum_i \Bigg(& -\cancel{\frac{\Delta x_{0i}}{2}}S_x - \cancel{\frac{\Delta y_{0i}}{2}}S_y - \frac{\Delta x_{0i}^2}{8}S_{2x} - \frac{\Delta y_{0i}^2}{8}S_{2y} - \frac{\Delta x_{0i}\Delta y_{0i}}{4}S_{xy} \\ & \cancel{\Delta x_{0i}S_x} + \cancel{\Delta y_{0i}S_y} + \cancel{\frac{\Delta x_{0i}^2}{2}}S_{2x} + \cancel{\frac{\Delta y_{0i}^2}{2}}S_{2y} + \cancel{\Delta x_{0i}\Delta y_{0i}S_{xy}} \\ & -\cancel{\frac{\Delta x_{0i}}{2}}S_x - \cancel{\frac{\Delta y_{0i}}{2}}S_y - \cancel{\frac{\Delta x_{0i}^2}{2}}S_{2x} - \cancel{\frac{\Delta y_{0i}^2}{2}}S_{2y} - \cancel{\Delta x_{0i}\Delta y_{0i}S_{xy}} \\ & -\frac{\Delta x_{0i}^2}{8}S_{2x} - \frac{\Delta x_{0i}^2}{8} - \frac{\Delta x_{0i}\Delta y_{0i}}{4}S_{xy}\Bigg)\mathbf{n}_{0i}\mathscr{S}_{0i}\Delta r + O\left(h^3\right) + O\left(h^q\right)\end{aligned}$$

Resulting in a truncation error term of

$$E_t = \frac{1}{V_0}\sum_i\left(-\frac{\Delta x_{0i}^2}{8}S_{2x} - \frac{\Delta y_{0i}^2}{8}S_{2y} - \frac{\Delta x_{0i}\Delta y_{0i}}{4}S_{xy}\right)\mathbf{n}_{0i}\mathscr{S}_{0i}\Delta r + O\left(h^3\right) + O\left(h^q\right). \tag{A.2}$$

Leading to a truncation error term that is $O\left(h^2\right)$ on arbitrary meshes as long as $q \geq 2$. An astute observer would notice that the Hessian adds back terms that would of otherwise been canceled in the source term error. This is so that consistency can be maintained when the method is combined with the fluxes.

A.1 Flux Correction Truncation Error

Flux Correction is formally second-order in truncation error terms. The following is a breakdown of the truncation error analysis. Starting with the general steady state solution,

$$\nabla \cdot \mathscr{F} = \frac{1}{V_0} \sum_i \frac{1}{2} \left(\mathscr{F}_L + \mathscr{F}_R \right) \mathbf{n}_{0i} \mathscr{S}_{0i}. \tag{A.3}$$

With,

$$\begin{aligned} \mathscr{F}_L &= \mathscr{F}_0 + \frac{1}{2} \Delta r^T \nabla^h \mathscr{F}_0 \\ \mathscr{F}_R &= \mathscr{F}_i - \frac{1}{2} \Delta r^T \nabla^h \mathscr{F}_i. \end{aligned}$$

Beginning the truncation error analysis, we start with Equation A.3 and split the fluxes into their components, leading to,

$$\begin{aligned} \nabla^h \cdot \mathscr{F} = \frac{1}{V_0} \sum_i \frac{1}{2} \Bigg(& f_0 n_{x0i} + g_0 n_{y0i} + f_i n_{x0i} + g_i n_{y0i} \\ & + \frac{1}{2} \Delta r^T \nabla^h \left(f_0 n_{x0i} + g_0 n_{y0i} - f_i n_{x0i} - g_i n_{y0i} \right) \Bigg) \mathscr{S}_{0i}. \end{aligned}$$

Now a multi-dimensional Taylor series expansion on both terms is completed on the flux,

$$\begin{aligned} f_i = f_0 + \Delta x_{0i} f_{0x} + \Delta y_{0i} f_{0y} + \frac{\Delta x_{0i}^2}{2} f_{02x} + \Delta x_{0i} \Delta y_{0i} f_{0xy} + \frac{\Delta y_{0i}^2}{2} f_{02y} \\ + \frac{\Delta x_{0i}^3}{6} f_{03x} + \frac{\Delta x_{0i}^2 \Delta y_{0i}}{2} f_{02xy} + \frac{\Delta x_{0i} \Delta y_{0i}^2}{2} f_{0x2y} + \frac{\Delta y_{0i}^3}{6} f_{03y} + O\left(h^4\right), \end{aligned}$$

$$\begin{aligned} g_i = g_0 + \Delta x_{0i} g_{0x} + \Delta y_{0i} g_{0y} + \frac{\Delta x_{0i}^2}{2} g_{02x} + \Delta x_{0i} \Delta y_{0i} g_{0xy} + \frac{\Delta y_{0i}^2}{2} g_{02y} \\ + \frac{\Delta x_{0i}^3}{6} g_{03x} + \frac{\Delta x_{0i}^2 \Delta y_{0i}}{2} g_{02xy} + \frac{\Delta x_{0i} \Delta y_{0i}^2}{2} g_{0x2y} + \frac{\Delta y_{0i}^3}{6} g_{03y} + O\left(h^4\right). \end{aligned}$$

To perform the truncation error we will first show that the necessary values are obtained. Substitution into Equation A.3 and applying the proper relationships yields,

$$\begin{aligned} \nabla^h \cdot \mathscr{F} = \frac{1}{V_0} \sum_i \Big(& f_0 \cancelto{0}{n_{x0i} \mathscr{S}_{0i}} + g_0 \cancelto{0}{n_{y0i} \mathscr{S}_{0i}} + f_{0x} \cancelto{V_0}{\frac{1}{2} \Delta x_{0i} n_{x0i} \mathscr{S}_{0i}} + g_{0x} \cancelto{0}{\frac{1}{2} \Delta x_{0i} n_{y0i} \mathscr{S}_{0i}} \\ & + f_{0y} \cancelto{0}{\frac{1}{2} \Delta y_{0i} n_{x0i} \mathscr{S}_{0i}} + g_{0y} \cancelto{V_0}{\frac{1}{2} \Delta y_{0i} n_{y0i} \mathscr{S}_{0i}} \Big) + O\left(h^1\right) \end{aligned}$$

Where $O\left(h^1\right)$ represents the truncation error. Simplification shows

$$\nabla^h \cdot \mathscr{F} + E_t = \nabla \cdot \mathscr{F} + O\left(h^1\right).$$

Thus the needed flux terms are still included where needed. Next we will expand the $O\left(h^1\right)$ terms to find the truncation error is actually $O\left(h^2\right)$. Further expansion shows,

$$\begin{aligned}
E_t = \frac{1}{2V_0}\sum_i \Bigg(& \cancel{\frac{\Delta x_{0i}^2}{2}(\mathscr{F}_{2x})} + \cancel{\Delta x_{0i}\Delta y_{0i}(\mathscr{F}_{xy})} + \cancel{\frac{\Delta y_{0i}^2}{2}(\mathscr{F}_{2y})} \\
& + \frac{\Delta x_{0i}^3}{6}(\mathscr{F}_{3x}) + \frac{\Delta x_{0i}^2 \Delta y_{0i}}{2}(\mathscr{F}_{2xy}) \\
& + \frac{\Delta x_{0i}\Delta y_{0i}^2}{2}(\mathscr{F}_{x2y}) + \frac{\Delta y_{0i}^3}{6}(\mathscr{F}_{3y}) \\
& - \cancel{\frac{\Delta x_{0i}^2}{2}(\mathscr{F}_{2x})} - \cancel{\Delta x_{0i}\Delta y_{0i}(\mathscr{F}_{xy})} - \cancel{\frac{\Delta y_{0i}^2}{2}(\mathscr{F}_{2y})} \\
& - \frac{\Delta x_{0i}^3}{4}(\mathscr{F}_{3x}) - \frac{3\Delta x_{0i}^2 \Delta y_{0i}}{4}(\mathscr{F}_{2xy}) \\
& - \frac{3\Delta x_{0i}\Delta y_{0i}^2}{4}(\mathscr{F}_{x2y}) - \frac{\Delta y_{0i}^3}{4}(\mathscr{F}_{3y}) + O\left(h^{q+1}\right) + O\left(h^4\right) \Bigg) \mathbf{n}_{0i}\mathscr{S}_{0i}.
\end{aligned}$$

Resulting in a truncation error term of

$$\begin{aligned}
E_t = -\frac{1}{2V_0}\sum_i \Bigg(\frac{1}{12}\Bigg(& \Delta x_{0i}^3(\mathscr{F}_{3x}) + \Delta y_{0i}^3(\mathscr{F}_{3y}) \\
& + 3\Delta x_{0i}^2\Delta y_{0i}(\mathscr{F}_{2xy}) + 3\Delta x_{0i}\Delta y_{0i}^2(\mathscr{F}_{x2y}) \Bigg)\Bigg) \mathbf{n}_{oi}\mathscr{S}_{0i} + O(h^q) + O\left(h^3\right).
\end{aligned} \tag{A.4}$$

This leads to a method that is formally $O\left(h^2\right)$ on arbitrary grids, as long as $q \geq 2$.

A.2 Combined Flux Correction and Corrected Source schemes

Now that truncation error analysis has been completed on both the corrected source and FC schemes, now the methods will be used in combination. The goal is to create a method that is consistent with that of the two dimensional version. So on an arbitrary grid the truncation error will be $O\left(h^2\right)$ and on a regular grid the expected truncation error will be $O\left(h^3\right)$. A few useful relationships will be used to determine the truncation error, on a regular grid in two dimensions

$$\sum_i \Delta x_{0i}^3 \mathscr{S}_{0i} \mathbf{n}_{0i} = \sum_i \Delta y_{0i}^3 \mathscr{S}_{0i} \mathbf{n}_{0i} = 0,$$

in addition, the relationship between fluxes and source terms is given by

$$f_x + g_y = S.$$

This is differentiated to obtain three equations that will be useful,

$$\begin{aligned} f_{3x} + g_{2xy} &= S_{2x} \\ f_{2xy} + g_{x2y} &= S_{xy} \\ f_{x2y} + g_{3y} &= S_{2y}. \end{aligned} \tag{A.5}$$

Appendix B

Body of Revolution Geometries

B.1 Model 4155

Model 4155 geometry was given by Gertler [70] in Imperial units. The basic profile was given by the polynomial equation,

$$y^2 = a_1x + a_2x^2 + a_3x^3 + a_4x^4 + a_5x^5 + a_6x^6 \tag{B.1}$$

where the coefficient a values are given in Table B.1.

Table B.1: Model 4155 Geometry Coefficient Values

a_1	1.000000
a_2	2.149653
a_3	-17.773496
a_4	36.716580
a_5	-33.511285
a_6	11.418548

Leading to the following model particulars in Table B.2,

Table B.2: Model 4155 Geometry Particulars

Length (ft)	9.000
Diameter (ft)	1.800
Nose radius (ft)	0.180
Tail radius (ft)	0.036
Wetted surface (ft^2)	39.75
Volume (ft^3)	14.89
Longitudinal center of buoyancy (ft)	4.180

B.2 Model 4621

Model 4621 defined by Van Randwijck and Feldman [84]. This model also uses Equation B.1 and Imperial units. The basic profile is given by the coefficient values in Table B.3.

Table B.3: Model 4155 Geometry Coefficient Values

a_1	1.000000
a_2	1.137153
a_3	-10.774885
a_4	19.784286
a_5	-16.792534
a_6	5.645977

Leading to the following model particulars in Table B.4,

Table B.4: Model 4155 Geometry Particulars

Length (ft)	15.00
Diameter (ft)	2.044
Nose radius (ft)	0.1392
Tail radius (ft)	0.0
Wetted surface (ft^2)	70.55
Volume (ft^3)	29.53
Longitudinal center of buoyancy (ft)	6.684

B.3 Body 1

Body 1 defined by Huang [85] is based off of the offset points given in Table B.5.

Table B.5: Body 1 Offset Data

x/L	y/L	x/L	y/L	x/L	y/L	x/L	y/L	x/L	y/L	x/L	y/L
0.000	0.000	.1193	.0441	.2684	.0456	.5070	.0456	.7363	.0427	.9612	.0095
.0050	.0100	.1243	.0444	.2783	.0456	.5169	.0456	.7477	.0421	.9642	.0087
.0099	.0142	.1292	.0447	.2883	.0456	.5268	.0456	.7553	.0416	.9662	.0081
.0149	.0175	.1342	.0450	.2982	.0456	.5368	.0456	.7666	.0408	.9682	.0076
.0199	.0202	.1392	.0452	.3082	.0456	.5467	.0456	.7780	.0399	.9692	.0074
.0298	.0248	.1441	.0453	.3280	.0456	.5567	.0456	.7970	.0382	.9702	.0072
.0348	.0268	.1491	.0454	.3380	.0456	.5666	.0456	.8045	.0375	.9722	.0068
.0398	.0287	.1541	.0455	.3479	.0456	.5765	.0456	.8159	.0363	.9732	.0066
.0447	.0303	.1590	.0448	.3579	.0456	.5865	.0456	.8273	.0350	.9751	.0063
.0497	.0319	.1640	.0456	.3678	.0456	.5964	.0456	.8349	.0326	.9771	.0062
.0547	.0333	.1690	.0456	.3777	.0456	.6064	.0456	.8462	.0326	.9791	.0059
.0596	.0347	.1740	.0456	.3877	.0456	.6188	.0456	.8576	.0310	.9811	.0056
.0646	.0359	.1789	.0456	.3976	.0456	.6264	.0455	.8652	.0299	.9831	.0053
.0696	.0370	.1839	.0456	.4076	.0456	.6378	.0455	.8765	.0281	.9851	.0050
.0746	.0381	.1889	.0456	.4175	.0456	.6454	.0455	.8841	.0268	.9871	.0048
.0795	.0390	.1938	.0456	.4274	.0456	.6567	.0453	.8955	.0248	.9881	.0046
.0845	.0399	.1988	.0456	.4374	.0456	.6681	.0452	.9069	.0226	.9901	.0043
.0895	.0407	.2087	.0456	.4473	.0456	.6757	.0450	.9144	.0211	.9920	.0040
.0944	.0414	.2187	.0456	.4573	.0456	.6871	.0448	.9245	.0189	.9940	.0036
.0994	.0421	.2286	.0456	.4672	.0456	.6984	.0444	.9344	.0166	.9960	.0028
.1044	.0427	.2366	.0456	.4771	.0456	.7060	.0441	.9443	.0140	.9980	.0019
.1093	.0432	.2485	.0456	.4871	.0456	.7174	.0437	.9513	.0122	1.000	0.000
.1143	.0437	.2584	.0456	.4970	.0456	.7250	.0433	.9563	.0108		

Appendix C

Measurement Uncertainty Analysis

The original experiments of Huang [85] were performed in a wind tunnel at $M = 0.09$ and $Re = 6.6e6$, and the flow can be assumed to be incompressible. Pitot tube pressure measurements were obtained along the afterbody and near wake region at distances of $2.316m$, $2.594m$, $2.804m$, and $3.241m$ measured from the body nose, as shown in Figure 3.14 The last measurement is completely in the wake region of the body. Pressure was measured radially outward from the body a short distance. In addition, pressure was measured on the surface of the afterbody. Measurement errors for the pressure on the surface of the body were reportedly less than 0.5% of the dynamic pressure, and off the body the measurements were reportedly less than .9% of the dynamic pressure. The velocity was reportedly measured within ±.15 m/s. Location of the pitot tube was measured within $0.01mm$. With these uncertainties the pressure coefficient uncertainty is estimated to be 1.15%. The pressure coefficient is determined from

$$C_p = \frac{P - P_\infty}{\frac{1}{2}\rho V^2}, \tag{C.1}$$

where P_∞ is the ambient pressure and P is the local pressure.

Beginning the analysis, Equation C.1 is partially differentiated with respect to both velocity and pressure:

$$\delta C_p = \frac{1}{\frac{1}{2}\rho V^2}\delta P, \tag{C.2}$$

$$\delta C_p = -\frac{P - P_\infty}{\frac{1}{4}\rho V^3}\delta V.$$

Next, we determine the dynamic pressure. Dynamic pressure is assumed to be “perfect” as all the measurement errors are dependent on it. Using the values used in the original test we obtain,

$$\frac{1}{2}\rho V^2 = \frac{1}{2}1.2754\left(30.15\right)^2 = 579.7Pa. \tag{C.3}$$

Thus pressure measurements varies ± 2.9 Pa, pitot tube misalignment measurements varies ± 4.6 Pa, and using the root mean square (RMS) method the total error is in the range ± 5.4 Pa. Given in the paper, velocity varies ±.15 m/s. Applying this to the partial differential equations in Equation C.2, we obtain one constant error value and one slightly varying error value dependent on local pressure difference.

$$u_{C_{p_P}} = \frac{1}{\frac{1}{2}\rho V^2} u_P = 0.0094,$$

$$u_{C_{p_V}} = -\frac{P - P_\infty}{\frac{1}{4}\rho V^3} u_V = \frac{0.000017}{Pa} \left(P_\infty - P\right).$$

These values are then combined using RMS. The final error bands can be seen in the figures presented in this work.