

ESTIMATION OF DISCRETIZATION ERROR FOR THREE DIMENSIONAL CFD  
SIMULATIONS USING A TAYLOR SERIES MODIFIED EQUATION ANALYSIS

A Thesis

by

AARON MARTIN KRUEGER

Submitted to the Office of Graduate and Professional Studies of  
Texas A&M University  
in partial fulfillment of the requirements for the degree of  
MASTER OF SCIENCE

Chair of Committee, Yassin Hassan  
Committee Members, Hamn-Ching Chen  
Rodolfo Vaghetto  
Vincent Mousseau  
Head of Department, Yassin Hassan

December 2017

Major Subject: Nuclear Engineering

Copyright 2017 Aaron Martin Krueger

## ABSTRACT

The Consortium for Advanced Simulation of Light Water Reactors (CASL) is working towards developing a virtual reactor called the Virtual Environment for Reactor Application (VERA). As part of this work, computational fluid dynamics (CFD) simulations are being made to inform lower fidelity models to predict heat transfer and fluid flow through a light water reactor core. A 5x5 fuel rod assembly with mixing vanes was chosen to represent a 17x17 fuel rod assembly. Even with this simplified geometry, it is estimated that hundreds of millions of cells are needed for a solution to be close to the asymptotic region. The large number of cells is an issue when completing solution verification studies because of computational cost.

Solution verification studies traditionally involve the use of Roache's grid convergence index (GCI) to estimate the error, but require the solution to be in the asymptotic region. This is a severely limiting restriction for simulations with large range of length scales as is the case with the 5x5 fuel rod assembly with mixing vanes. Unfortunately, GCI does not perform well when the solution is outside the asymptotic region. However, a new method called the robust multi-regression (RMR) solution verification method has the potential to produce good results, even when the solution is outside the asymptotic region.

This study builds a software framework that improves the RMR solution verification analysis by improving the error model used to estimate the discretization error. Previous RMR work used a power function to model the error, which was the same function used in the Richardson extrapolation. The power function form is a result of a Taylor series expansion on a uniform grid for simple numerical schemes and physics. It can be improved by completing a Taylor series expansion for the numerical scheme, boundary conditions, and physics that are being employed in the simulation of interest. This framework was

shown to improve the ability for the error model to estimate the discretization error and uncertainty. The improved error model was able to predict error on a refined grid within the uncertainty bounds, while the standard error model did not. In addition, the method of manufactured modified equation analysis solutions (MMMEAS) was developed and applied to justify the use of a down selection method for terms in the error model.

## DEDICATION

I would like to dedicate this work to my parents. Without their constant love, support, and encouragement, I would not be the person I am today.

## ACKNOWLEDGMENTS

I would like to thank Professor Yassin Hassan for advising me throughout this process and the research opportunities he has provided me over the years. I would also like to thank Dr. Vincent Mousseau for the enthusiasm he has brought to this project. His countless hours of explaining V&V-relevant concepts and reviewing my work has been critical to the success of this project. I would like to thank my other committee members, Professor Hamn-Ching Chen and Professor Rodolfo Vaghetto, for technical support and guidance through the project. Additionally, I would like to thank my colleagues at Sandia National Laboratories, especially Dr. Christopher Jones, Victor Figueroa, Lindsay Gilkey, Natalie Gordon, and Dr. William Rider for their numerous comments and suggestions. I would thank Nathan Porter of North Carolina State University for his technical and editorial comments of this thesis. Lastly, I would like to thank my wife Jem for her support and encouragement. I couldn't have finished this without her.

## CONTRIBUTORS AND FUNDING SOURCES

### **Contributors**

This work was supported by a thesis committee consisting of Professor Yassin Hassan (advisor) and Professor Rodolfo Vaghetto of the Department of Nuclear Engineering, Professor Hamn-Ching Chen of the Department of Civil Engineering, and Dr. Vincent Mousseau of Sandia National Laboratories.

All work conducted for the thesis was completed by the student, under the advisement of Professor Yassin Hassan of the Department of Nuclear Engineering and Dr. Vincent Mousseau of Sandia National Laboratories.

### **Funding Sources**

This research was supported by the Consortium for Advanced Simulation of Light Water Reactors ([www.casl.gov](http://www.casl.gov)), an Energy Innovation Hub (<http://www.energy.gov/hubs>) for Modeling and Simulation of Nuclear Reactors under U.S. Department of Energy Contract No. DE-AC05-00OR22725. This project would not have been possible without Sandia National Laboratories for their personnel and equipment support. Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525. In addition, portions of this research were conducted with the advanced computing resources provided by Texas A&M High Performance Research Computing.

## NOMENCLATURE

### Abbreviations

CASL	Consortium for Advanced Simulation of Light Water Reactors
CFD	Computational Fluid Dynamics
GCI	Grid Convergence Index
LASSO	Least Absolute Shrinkage and Selection Operator
MAD	Median Absolute Difference
MEA	Modified Equation Analysis
MMMEAS	Method of Manufactured Modified Equation Analysis Solutions
MMS	Method of Manufactured Solutions
MSE	Mean Squared Error
QoI	Quantity of Interest
Re	Reynold's Number
RMR	Robust Multi-Regression
SQA	Software Quality Assurance
VERA	Virtual Environment for Reactor Applications

### Parameter Notation

$A$	Channel Area
$A_k$	QoI on the $k^{th}$ Grid

$F_s$	Factor of Safety
$H$	Half Channel Height
$L$	Channel Length
$N$	Total Number of LASSO Variables
$N_c$	Total Number of Cells in the Domain of Interest
$U$	Velocity
$V$	Total Volume
$W$	Channel Width
$b$	Base Size
$c$	Coefficient of Leading Error Term
$f$	Solution on a Particular Grid
$h$	Average Cell Size Estimate
$h_k$	Average Cell Size Estimate on the $k^{th}$ Grid
$i$	Index in the X Direction
$j$	Index in the Y Direction
$m$	LASSO Cross-Validation Reference Index
$n$	LASSO Variable Index
$p$	Pressure
$p_{obs}$	Observed Order of Accuracy
$p_{th}$	Formal Order of Accuracy
$r$	Refinement Factor
$t$	Time
$u$	X Velocity
$v$	Y Velocity



$\alpha$	Vector Differencing Method Iteration
$\beta$	LASSO Variable Coefficient
$\Delta x_1$	First Cell in Domain
$\lambda$	LASSO Coefficients
$\phi$	Arbitrary CFD Output Variable
$\rho$	Density
$\tau$	Shear Stress
$\mu$	Viscosity

## TABLE OF CONTENTS

	Page
ABSTRACT . . . . .	ii
DEDICATION . . . . .	iv
ACKNOWLEDGMENTS . . . . .	v
CONTRIBUTORS AND FUNDING SOURCES . . . . .	vi
NOMENCLATURE . . . . .	vii
TABLE OF CONTENTS . . . . .	x
LIST OF FIGURES . . . . .	xiii
LIST OF TABLES . . . . .	xv
1. INTRODUCTION . . . . .	1
1.1 Verification and Validation . . . . .	1
1.1.1 Roache’s Grid Convergence Index . . . . .	3
1.1.2 Robust Multi-Regression Solution Verification . . . . .	3
1.1.3 Modified Equation Analysis . . . . .	4
1.2 Objectives . . . . .	4
2. METHODOLOGY . . . . .	6
2.1 Generation Error Terms . . . . .	6
2.2 Generation of Data . . . . .	6
2.2.1 Geometric Sensitivities . . . . .	7
2.2.2 Derivation of Analytic Solution . . . . .	7
2.3 LASSO Analysis . . . . .	9
2.3.1 Data Normalization . . . . .	10
2.3.2 Expert Opinion Check . . . . .	10
2.3.3 Homogeneity Check . . . . .	11
2.4 Coefficient Optimization . . . . .	11
2.5 Uncertainty Quantification . . . . .	11
2.6 Computational Efficiency . . . . .	12

2.7	Comparison Between Solution Verification Methods . . . . .	12
2.8	Teacher-Student Analysis . . . . .	12
2.8.1	One Dimensional MEA . . . . .	13
2.8.2	Solution Form and Local Error Equation . . . . .	13
2.8.3	Grid Refinement Methods . . . . .	13
2.8.4	Grid Comparison and LASSO Analysis . . . . .	14
2.8.5	Teacher-Student Error Comparison . . . . .	14
3.	RESULTS . . . . .	15
3.1	STAR-CCM+ Verification Case Setup . . . . .	15
3.1.1	Geometry and Boundary Conditions . . . . .	15
3.1.2	STAR-CCM+ Grid . . . . .	16
3.1.3	Flow Description . . . . .	17
3.1.4	Numerical Scheme . . . . .	17
3.1.5	Geometric Sensitivities . . . . .	17
3.2	Generation of Error Terms . . . . .	19
3.2.1	Continuity MEA . . . . .	20
3.2.2	Conservation of Momentum MEA . . . . .	25
3.2.3	Cell Size Estimation Terms . . . . .	31
3.2.4	Grid Quality Terms . . . . .	32
3.3	LASSO Analysis . . . . .	38
3.3.1	Data Normalization . . . . .	39
3.3.2	Expert Opinion Check . . . . .	40
3.3.3	Homogeneity Check . . . . .	41
3.3.4	LASSO Results . . . . .	41
3.4	Coefficient Optimization . . . . .	42
3.4.1	Coefficient Optimization Results . . . . .	43
3.5	Uncertainty Quantification . . . . .	44
3.6	Computational Effectiveness . . . . .	45
3.7	Comparison Between Solution Verification Methods . . . . .	45
3.8	Teacher-Student Analysis . . . . .	47
3.8.1	One Dimensional MEA for Steady Continuity Equation . . . . .	47
3.8.2	Solution Equation and Local Error Equation (Teacher Equation) . . . . .	49
3.8.3	Grid Refinement Methods . . . . .	51
3.8.4	Grid Comparison and LASSO Analysis - Local Error . . . . .	52
3.8.5	Grid Comparison and LASSO Analysis - Global Error . . . . .	58
3.8.6	Teacher-Student Data Comparison . . . . .	58
4.	CONCLUSION . . . . .	71
4.1	STAR-CCM+ Analysis . . . . .	71
4.2	Teacher-Student Analysis . . . . .	71

4.3 Future Work . . . . .	72
REFERENCES . . . . .	73

## LIST OF FIGURES

FIGURE	Page
3.1 Test Case Geometry with Boundary Conditions . . . . .	16
3.2 Section View of Grid Detail at the Inlet . . . . .	17
3.3 Top View of Section Geometry with Probe Locations . . . . .	18
3.4 Indexing Used for MEA . . . . .	21
3.5 Cell Aspect Ratio Example . . . . .	33
3.6 Face Validity Example . . . . .	34
3.7 Cell Quality Example . . . . .	34
3.8 Volume Change Example . . . . .	35
3.9 Cell Skewness Example . . . . .	36
3.10 Chevron Cell Example . . . . .	36
3.11 Least Squares Quality Example . . . . .	37
3.12 Cell Warpage Example . . . . .	38
3.13 Raw Data with Large Iterations . . . . .	40
3.14 Error Prediction Using LASSO Coefficients . . . . .	42
3.15 Error Model Results . . . . .	44
3.16 Uncertainty Estimate . . . . .	45
3.17 Uncertainty Due to Sample Size . . . . .	46
3.18 Uncertainty Estimate Comparison . . . . .	46
3.19 Teacher-Student Study 1 Grid 1 Local Error Comparison Plot . . . . .	60
3.20 Teacher-Student Study 1 Grid 2 Local Error Comparison Plot . . . . .	60

3.21	Teacher-Student Study 1 Grid 3 Local Error Comparison Plot . . . . .	61
3.22	Teacher-Student Study 1 Grid 4 Local Error Comparison Plot . . . . .	61
3.23	Teacher-Student Study 2 Grid 1 Local Error Comparison Plot . . . . .	62
3.24	Teacher-Student Study 2 Grid 2 Local Error Comparison Plot . . . . .	62
3.25	Teacher-Student Study 2 Grid 3 Local Error Comparison Plot . . . . .	63
3.26	Teacher-Student Study 2 Grid 4 Local Error Comparison Plot . . . . .	63
3.27	Teacher-Student Study 3 Grid 1 Local Error Comparison Plot . . . . .	64
3.28	Teacher-Student Study 3 Grid 2 Local Error Comparison Plot . . . . .	64
3.29	Teacher-Student Study 3 Grid 3 Local Error Comparison Plot . . . . .	65
3.30	Teacher-Student Study 3 Grid 4 Local Error Comparison Plot . . . . .	65
3.31	Teacher-Student Study 4 Grid 1 Local Error Comparison Plot . . . . .	66
3.32	Teacher-Student Study 4 Grid 2 Local Error Comparison Plot . . . . .	66
3.33	Teacher-Student Study 4 Grid 3 Local Error Comparison Plot . . . . .	67
3.34	Teacher-Student Study 4 Grid 4 Local Error Comparison Plot . . . . .	67
3.35	Teacher-Student Study 1 Grid 1 Global Error Comparison Plot . . . . .	68
3.36	Teacher-Student Study 1 Grid 2 Global Error Comparison Plot . . . . .	68
3.37	Teacher-Student Study 1 Grid 3 Global Error Comparison Plot . . . . .	69
3.38	Teacher-Student Study 1 Grid 4 Global Error Comparison Plot . . . . .	69

## LIST OF TABLES

TABLE	Page
3.1 Fully-Developed Sensitivity Results . . . . .	18
3.2 Symmetry Sensitivity Results . . . . .	19
3.3 LASSO Coefficients . . . . .	41
3.4 Optimization and Residual Data . . . . .	43
3.5 Teacher Equation Coefficients . . . . .	51
3.6 Teacher-Student Local LASSO Results (Study 1) . . . . .	54
3.7 Teacher-Student Local LASSO Results (Study 2) . . . . .	55
3.8 Teacher-Student Local LASSO Results (Study 3) . . . . .	56
3.9 Teacher-Student Local LASSO Results (Study 4) . . . . .	57
3.10 Teacher-Student Global LASSO Results . . . . .	59

## 1. INTRODUCTION

In the last few decades, the amount of computational power available to scientists and engineers has drastically increased the use of computational fluid dynamics (CFD) simulation to solve engineering problems. As CFD simulations grow in complexity, the ability to determine if the CFD simulation is functioning properly becomes even harder. As a result, a subset within the CFD community has pushed for more rigorous methods to check if the equations in the CFD software are solved correctly and if the equations being solved were correctly chosen for the problem the CFD user was trying to solve [1]. This is the primary purpose of the verification and validation (V&V) community.

### 1.1 Verification and Validation

During the 1960's and the 1970's, the V&V community began from the modeling and simulation researchers who dealt with extremely complex industrial production models, business or government organizations, marketing models, economic models, and military models [2]. These researchers were the first to define some of the terminology within the V&V field. One of the main advancements was to define the methods for checking if the equations in the model were being solved correctly, which was referred to as verification, and if the equations were the correctly represented reality, which was referred to as validation. One of the other advancements was the order of the two. The V&V community identified that validation methods would be useless if there was an error in the equations. Therefore, verification methods were required to be performed first.

Verification methods are split up into two different parts: code verification and solution verification. Code verification methods are methods that aim to reduce the number of coding errors that effect either the solution or the order of accuracy. Examples of this are method of exact solution [3], method of manufactured solutions (MMS) [3][4], method



of manufactured universes [5], and formal order of accuracy calculations [3]. In addition, software quality assurance (SQA) procedures are also part of code verification [2]. Solution verification methods are methods that aim to estimate the error for a particular simulation of interest. Errors of concern fall under the following categories: iterative error, round off error, statistical sampling error, and discretization error. Discretization error is impacted by the conservation equations, boundary conditions, and source terms. Most CFD users that perform solution verification estimate discretization error either using *a priori* or *a posteriori* methods.

*a priori* discretization error estimation methods, such as recovery-based methods and residual-based methods, estimate the error during the course of the simulation. By adding error equations to the solution matrix, the CFD solver can accurately estimate the discretization error. Unfortunately, these methods require source code changes and are often difficult to devise for an industrial CFD code. Because of their complexity and recent development, these methods are not available in current commercial CFD codes.

*a posteriori* discretization error estimation methods are methods that estimate the error after the simulation has completed. For finite volume solvers, extrapolation methods are almost exclusively used. Extrapolation methods are either H method, which examines QoIs as the grid size changes, or P method, which examines QoIs as the order of the numerical scheme changes. Since higher-order finite volume numerical schemes are difficult to program, H methods are typically used. H methods have been performed ever since Lewis Richardson performed his Richardson extrapolation to predict stresses on a dam [6]. While using the Richardson extrapolation is convenient to predict the error, there is inherent uncertainty in the calculation. In the mid-1990's, Patrick Roach developed a method called the Grid Convergence Index (GCI) that accounts for the uncertainty within the discretization error estimation.

### 1.1.1 Roache's Grid Convergence Index

GCI was a method developed to capture the uncertainties within the discretization error estimation by making the assumption that the Richardson extrapolation is the 50<sup>th</sup> percentile estimate. By multiplying a factor of safety to the extrapolated solution, the resultant would give the 95<sup>th</sup> percentile estimate. Eq. 1.1 is formulation of the standard GCI

$$GCI = \frac{F_s}{r^p - 1} |f_2 - f_1| \quad (1.1)$$

where  $p$  is the order of accuracy,  $f_2$  is the QoI calculated on the coarse grid, and  $f_1$  is the QoI calculated on the fine grid. Two different factor of safeties are available to use. When solutions are available on three different grids and the observed order of accuracy matches the formal order of accuracy,  $F_s$  is 1.25 and  $p$  is the observed order of accuracy, but when solutions are only available on two different grids or the observed order of accuracy does not match the formal order of accuracy,  $F_s$  is 3.0 and  $p$  is the formal order of accuracy. When simulations are not in the asymptotic range, the factor of safety is large and QoIs have been know to be unpredictable [2]. A major current research topic in the V&V field is discretization error estimation methods that work outside the asymptotic range, where the asymptotic range is defined as the range of  $\delta x$  where the leading error term dominates all higher order error terms. One method that has the potential to work slightly outside the asymptotic range is the RMR solution verification method.

### 1.1.2 Robust Multi-Regression Solution Verification

The RMR solution verification method is a method that uses expert judgment as well as robust statistics to determine optimal coefficients of an error model to estimate discretization error [7]. This method shares the same form of the discretization error as GCI, but differs in the calculation of the model coefficients. GCI uses QoI calculated on three

grids, while the RMR method uses QoI calculated on more than three grids to compute the observed order of accuracy. This means that GCI is a fully-determined system of equations, while the RMR method is an overly-determined system of equations. By taking the median of all observed order of accuracy values, outlier data points have less impact. This provides a robust method to calculate the error model coefficients, which is beneficial in the non-asymptotic region. One improvement to the RMR method is to use a more complex error model by completing a modified equation analysis (MEA).

### **1.1.3 Modified Equation Analysis**

MEA was first used as a stability analyzer to determine the point at which a numerical scheme is guaranteed to be stable. Work completed by C. Hirt [8] and R.F. Warming and B.J. Hyett [9] were some of the first to use MEA on a finite difference problem to analyze the stability of a numerical scheme. While a stability analysis was not in the scope of this project, the method used to complete the stability analysis was used to analytically calculate the local error.

## **1.2 Objectives**

The primary objective of this research is to improve discretization error and uncertainty estimation for three dimensional CFD simulations slightly outside the asymptotic region. Traditional discretization error and uncertainty estimation methods, such as GCI, require the simulation to be in the asymptotic region because they only account for the leading error term. A new method called the RMR solution verification has shown the potential to perform better than GCI when the data is not refined uniformly [7], but improvements will be made to the assumed form of the error for better performance.

The RMR solution verification method with an improved error model was implemented for this study. Since the current method relies on a simple form of the error, complex error terms cannot be represented. By performing a semi-rigorous MEA, the

form of the error will be better understood. The term "semi-rigorous" is used because the unique terms are important to calculate correctly, while the coefficients are considered not important. To accomplish this objective, a list of possible error terms will be generated to describe the form of the error. Next, a least absolute shrinkage and selection operator (LASSO) analysis will reduce the number of possible error terms. QoI data will be calculated on a large number of coarse grids and the optimization routine will use the QoI data to generate optimal values of the error model coefficients. This will produce an error model that has the potential to work slightly outside the asymptotic range and with poor grid refinement data.

## 2. METHODOLOGY

This chapter outlines the methods and equations used to obtain the results. The chapter is split into seven parts: generation of error terms, generation of QoI data (including analytic solution derivation), LASSO analysis for down selection of terms, coefficient optimization method, uncertainty quantification, computational efficiency, and the teacher-student equation analysis. All analyses were completed using STAR-CCM+ as the CFD software, Python for data generation, Mathematica for symbolic manipulation of the MEA, Dakota for the optimization tool, and MATLAB for data analysis and data visualization.

### 2.1 Generation Error Terms

Since a complete MEA was not possible for even simple three dimensional simulations, a semi-rigorous MEA was able to determine most terms that were part of the truncation error. There were two error terms this study focused on: terms that appeared from a MEA of the conservation equations as discretized in STAR-CCM+ using Mathematica and STAR-CCM+'s internal grid metric terms to describe less-than-ideal grid.

### 2.2 Generation of Data

To determine if the new method of estimating discretization error was viable, a problem which avoids added complexities that would hinder a comparison between the numerical solution and analytical derivation was required. A simple flow problem is one where the physics of the simulation avoids added complexities that would hinder a comparison between the numerical solution and the analytic derivation. Examples of added simulation complexities include turbulence modeling and heat transfer. One of the simplest CFD simulations to compute is laminar flow between two infinitely long plates. This was used as a test case for the study because the simulation did not have to model turbulence or heat

transfer.

The QoI was maximum velocity. Maximum velocity was chosen because of its common use in engineering applications and the ease of calculating an analytic solution. In addition, maximum velocity was not a system variable, which was solved to be within iteration error and was not affected by the discretization of the fluid domain. To make sure that the geometry or boundary conditions were not influencing the simulation, sensitivity analyses were completed, which are detailed in 2.2.1. In addition, the derivation of maximum velocity using a Reynold's number ( $Re$ ) of 500 is detailed in Section 2.2.2.

### **2.2.1 Geometric Sensitivities**

Two different geometric sensitivities were evaluated to verify that the geometry had no impact on the results: if the flow was fully developed and if the location between the symmetric wall had no impact on the flow profile, the geometry did not impact the results. If either conditions were not met, the geometry was modified until there was no impact.

### **2.2.2 Derivation of Analytic Solution**

It is important to be able to compare the estimate of the discretization error with the exact discretization error. This test case is simple enough to have an analytic solution for the velocity profile assuming fully-developed conditions. An analytic solution provided a measure of how well the error model performed and was not directly used to aid in error model development.

### **Velocity Profile**

The analytic incompressible, steady state, fully-developed velocity profile was derived using the steady state continuity equation, which is shown in Eq. 2.1 and conservation of momentum in Eq. 2.2 [10].

$$\nabla \cdot (\rho U) = 0 \quad (2.1)$$

$$\nabla \cdot (\rho U U) = -\nabla p - \nabla \cdot \tau + \rho g \quad (2.2)$$

Eq. 2.3 and Eq. 2.4 arose from assuming incompressibility, negligible effect of gravity, and steady state for Eq. 2.1 and Eq. 2.2.

$$\nabla \cdot U = 0 \quad (2.3)$$

$$\rho \nabla \cdot (U U) = -\nabla p - \nabla \cdot \tau \quad (2.4)$$

Substituting Eq. 2.1 into the convective term in Eq. 2.4, lead to Eq. 2.5.

$$\nabla p = -\nabla \cdot \tau \quad (2.5)$$

For laminar flow between two plates in the x (downstream) direction, the right hand side became Eq. 2.6.

$$\nabla \cdot \tau = -\mu \frac{\partial^2 u}{\partial y^2} \quad (2.6)$$

Eq. 2.7 was made by substituting Eq. 2.6 into Eq. 2.5, which shows the relationship between the second derivative of velocity in the downstream direction and the pressure gradient. With integration and the boundary conditions of zero velocity at both sides of the plate, the velocity profile equation was determined to have a dependence on the pressure gradient and the distance between the plates. This is shown in Eq. 2.8.

$$\nabla p = \mu \frac{\partial^2 u}{\partial y^2} \quad (2.7)$$

$$u(y) = -\frac{H^2}{2\mu} \frac{dp}{dx} \left[ 1 - \left( \frac{y}{H} \right)^2 \right] \quad (2.8)$$

Using conservation of mass, the pressure gradient term was replaced with the mean velocity by making sure the mass flow rate is conserved, which is shown in Eq. 2.9.

$$u_{ave} = \frac{1}{\rho A} \int_A \rho u dA = \frac{1}{\rho (2H) W} \int_0^W \int_{-H}^H \rho u dy dz = \frac{1}{H} \int_0^H \frac{H^2}{2\mu} \frac{dp}{dx} \left[ \left( \frac{y}{H} \right)^2 - 1 \right] dy \quad (2.9)$$

Integrating Eq. 2.9 with respect to  $y$  produces Eq. 2.10.

$$u_{ave} = \frac{H^2}{2\mu} \frac{dp}{dx} \left( \frac{1}{3} - 1 \right) = -\frac{H^2}{3\mu} \frac{dp}{dx} \quad (2.10)$$

Eq. 2.10 was rearranged for the pressure gradient term as a function of mean velocity as seen in Eq. 2.11, which was substituted into Eq. 2.8 to form Eq. 2.12.

$$\frac{dp}{dx} = -\frac{3\mu}{H^2} u_{ave} \quad (2.11)$$

$$u(y) = \frac{3}{2} u_{ave} \left[ 1 - \left( \frac{y}{H} \right)^2 \right] \quad (2.12)$$

### 2.3 LASSO Analysis

To reduce the computational cost of the optimization study, a LASSO analysis was used as a down selection method [11]. The LASSO function is able to calculate coefficients that relate the predictor to the response. When doing this, it uses the lowest number



of predictors to fully represent the response. When the predictor and response values are normalized, the coefficients represent the relative importance of the terms. This was used to determine the dominant terms in the error model. LASSO does this by minimizing Eq. 2.13 [12]

$$\min_{\beta_0, \beta} \left( \frac{1}{2N} \sum_{i=1}^N (y_i - \beta_0 - x_i^T \beta)^2 + \lambda \sum_{j=1}^P |\beta_j| \right) \quad (2.13)$$

where  $N$  is the number of observations,  $y_i$  is the response at observation  $i$ ,  $x_i$  is a vector of data of  $P$  values at observation  $i$ ,  $\lambda$  is a non-negative regularization parameter corresponding to one value of  $\lambda$ , and  $\beta_0$  and  $\beta$  are a scalar and a vector of length  $p$ . The coefficients of interest was the  $\beta$  values when  $\lambda$  was within one standard error of the minimum mean squared error (MSE).

### 2.3.1 Data Normalization

As stated before, the predictor and the response data were normalized. This was to avoid different units associated with the coefficients. It also guarantees that the LASSO-created coefficients share the first data point. The normalization was done in a way that scales the data between zero and one, which is shown in Eq. 2.14.

$$\phi_{norm} = \frac{\phi - \min(\phi)}{\max(\phi) - \min(\phi)} \quad (2.14)$$

### 2.3.2 Expert Opinion Check

To avoid using poor quality data, plotting the MEA-identified variables as a function of QoI provided visual correlation feedback. Poor data often is found on grids with poor grid metrics and large cell sizes. If this step is quantified, using a rank correlation such as Pearson, Kendall's tau, or Spearman quantifies the correlation between the predictor and the response.

### 2.3.3 Homogeneity Check

To make sure there was enough QoI data, a Homogeneity check was completed. This verified that the LASSO algorithm has enough data to produce the same results within an acceptable tolerance, even if more QoI data was added or replaced. If the data produced by STAR-CCM+ was heterogeneous, then the variability was quantified as an uncertainty.

## 2.4 Coefficient Optimization

The optimization scheme was designed to work with potentially non-smooth solution space, while being computational efficient. A hybrid method using both sampling and gradient methods was implemented to obtain optimization characteristics that sampled the solution space well and was computationally efficient. The optimization method chosen optimized the coefficients within each term in Eq. 2.15

$$\arg \min_k |\Delta A_{k,k+1} + Term(h_k)_i - Term(h_{k+1})_i \dots| \quad (2.15)$$

where  $k$  is the  $k^{th}$  grid,  $\Delta A_{k,k+1}$  is the difference between the QoI on the  $k^{th}$  and  $k^{th} + 1$  grid, and  $i$  is the  $i^{th}$  term in the error model.

## 2.5 Uncertainty Quantification

When determining the uncertainties within the coefficient optimization, calculations that provided slightly different results by using a different normalization method or data sets were quantified as uncertainties. An example of this was using multiple differencing methods for determining the optimized coefficient. Using many differencing methods as part of the optimization scheme provided slightly different optimized coefficients without a clear way of knowing which differencing method provided the best coefficients. By quantifying the variability in the data, not only were coefficient estimates produced, but the uncertainty of the coefficients were produced. This provided an uncertainty estimate

of the discretization error.

## **2.6 Computational Efficiency**

Since a variety of grids were used, it was important to weigh the accuracy of the error estimate with computational power necessary to achieve the error estimate. Since calculating QoI on fine grids were more computationally expensive than on a coarse grids, it was beneficial to exclusively use coarse grids. Error model estimates using a set of QoI data from fine and coarse grids and QoI data from exclusively coarse grids using the same computational power were compared to determine if coarse grids were a better use of computational power.

## **2.7 Comparison Between Solution Verification Methods**

To determine if the improved error model improved estimation capabilities, error estimation results from RMR with the improved error model were compared to the results from RMR with the standard error model. This comparison was used as justification for continued work with improving the error model.

## **2.8 Teacher-Student Analysis**

A teacher-student equation analysis is a method to recover the coefficient of the terms in the teacher equation by only using a generic form of the teacher equation (the student equation) and data generated by the teacher equation. The teacher equation represents the analytic error equation that the CFD software produces, which for this analysis was the error equation and solution form from a method of manufactured modified equation analysis solutions (MMMEAS) of the one dimensional steady continuity equation using an upwind discretization scheme. MMMEAS is an analysis, similar to MMS, that uses the MEA to generate synthetic data and produces an exact error equation. In addition to knowing the analytic error equation, the analytic equations for QoI were known so an

exact error at each cell location can be calculated. Using the teacher equations with grids of different refinement levels, grid comparison data were generated in a similar fashion as with STAR-CCM+. A LASSO analysis in combination with the student equation was then able to determine the coefficient values of the teacher equation. While using STAR-CCM+ for the creation of data is the end goal, a teacher-student equation analysis was of benefit to confirm that the framework is working correctly. In addition, the teacher-student equation analysis showed a clear origin of local error, how the local error is analyzed in a global manner, and how the coefficients of the error terms are recovered.

### **2.8.1 One Dimensional MEA**

The one dimensional steady continuity equation with constant velocity was chosen as a simplification of the full three dimensional unsteady continuity equation. Since the process was the same as with the MEA for the STAR-CCM+ analysis, the simplification from three dimensions to one dimension was valid and made the explanation of the MMMEAS process easier to follow.

### **2.8.2 Solution Form and Local Error Equation**

Once the local error equation was known, an assumed solution was chosen for use with MMMEAS. For this particular analysis, the form of the solution was an equation that had a length scale attributed to it. This allows for a study inside, around, and outside the asymptotic range to be completed. Next, derivatives of the solution equation were taken. Using the local error equation determined from the MEA and the solution equation and its derivatives, local error were calculated on a grid compatible with the MEA.

### **2.8.3 Grid Refinement Methods**

Four different grid refinement methods were implemented to show the effect on global error as a function of refinement for each refinement method. For a one dimensional grid

and allowing a constant gradient in the grid size, the variables that control the grid spacing were number of cells in the domain, first cell size, and the value of the constant grid refinement gradient. For the grid sizing of Eq. 2.16, the variables relationship are given in Eq. 2.17 [13] .

$$L = \Delta x_1 + \Delta x_1 r + \Delta x_1 r^2 + \dots + \Delta x_1 r^{N_c-1} \quad (2.16)$$

$$\Delta x_1 r^{N_c} - rL - \Delta x_1 + L = 0 \quad (2.17)$$

The four refinement methods were different combinations of varying two parameters and solving for the third. Note that since  $N_c$  is an integer, the permutation of solving for  $N_c$  is much more complex and was avoided.

#### **2.8.4 Grid Comparison and LASSO Analysis**

The grid comparison data was collected by solving the teacher equation on each grid, calculating both the local error at each cell and the global error norm for each grid, and calculating a  $\Delta_{error}$  in the same way as calculating  $\Delta A_{k,k-1}$  for the STAR-CCM+ data. The  $\Delta_{error}$  along with the input parameters and the student equation were then used with the LASSO algorithm to predict the coefficient of the teacher equation.

#### **2.8.5 Teacher-Student Error Comparison**

The last portion of the teacher student analysis was to compare the exact error of the teacher equation to the error determined by the student equation calculated by the LASSO algorithm. If the LASSO algorithm was able to determine the error within 30% of the actual value, the LASSO algorithm has shown that it is sufficient to down select parameters for future studies and provide a guide for future studies using STAR-CCM+ instead of the teacher equation and the known QoI function.

### 3. RESULTS

To improve the form of the error model, dominant truncation error terms were taken into account. A method to determine the truncation error terms was to perform an MEA on the discrete conservation equations. The MEA was performed on the discrete equations implemented within STAR-CCM+ [14], a commercial CFD code. Additional terms arise from low quality grids. STAR-CCM+ provides various metrics to determine if the generated grid was of low quality.

#### 3.1 STAR-CCM+ Verification Case Setup

Verification cases were used to test the adequacy of the error model, which was based on the unique terms found in the truncation error analysis of the conservation equations and grid metrics. Simulating flow between infinite parallel plates was used as a test case to ensure RMR with the improved error model was implemented correctly. Below is a description of the simulation setup.

##### 3.1.1 Geometry and Boundary Conditions

To model two infinite plates, the infinite domain was simulated by using symmetric boundary conditions. The section was 50 hydraulic diameters long (6.666 meters) to create fully-developed conditions and the width was 1.5 hydraulic diameters wide (0.2 meters) to make sure the symmetric boundaries were implemented correctly. The distance between the plates was 0.75 hydraulic diameters high (0.1 meters). Sensitivity analyses were also used to ensure the length and width used did not violate the fully developed and infinite plate assumptions, which is detailed below. The boundary conditions used for the primary inlet and outlet surface were uniform velocity inlet and a uniform pressure outlet. The velocity inlet was set so the Reynolds number was 500 (where the hydraulic diameter was

the characteristic length) and the pressure outlet was set to 0.0 Pascals. Gravity was not included in the simulation to simplify the velocity profile derivation. Figure 3.1 shows the dimensions and boundary conditions used for the simulation.

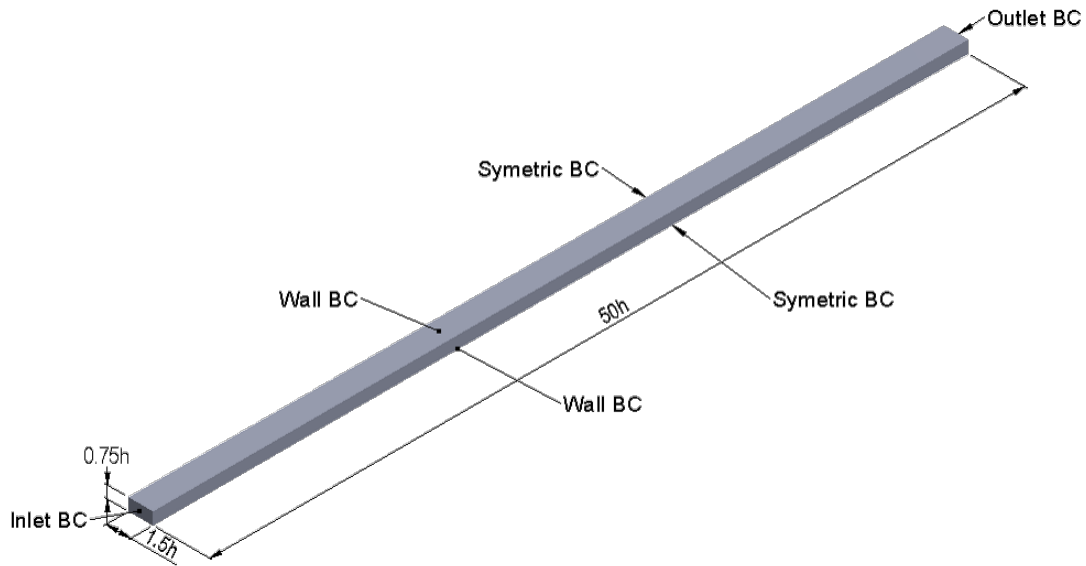


Figure 3.1: Test Case Geometry with Dimensions and Boundary Conditions

### 3.1.2 STAR-CCM+ Grid

Three hundred sixty six STAR-CCM+ grids were built by specifying the average cell size to be between one tenth and one half the distance between the plates ( $0.01$  meters and  $0.05$  meters respectively), which is near the start of the asymptotic range (see Figure 3.2). In addition to being near or outside of the asymptotic range, the simulations were relatively short. Total CPU time ranged from 18 seconds to 28 mins. 166 grids with large iterations for the cell size were removed from the data set. A detailed description of the data removal is described in Section 3.3.2.

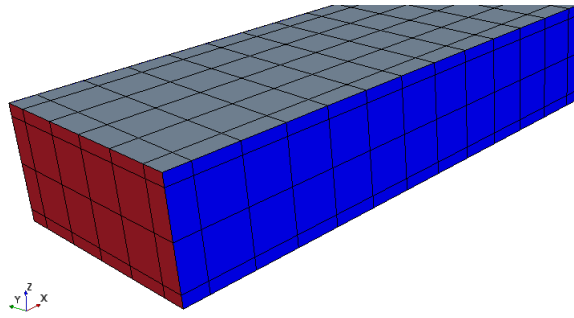


Figure 3.2: Section View of Grid Detail at the Inlet

### 3.1.3 Flow Description

The three-dimensional channel flow simulation was set up to be steady (derivatives in time are zero), constant density (derivatives of density are zero), and laminar.

### 3.1.4 Numerical Scheme

The momentum and continuity equations were solved using a second-order upwind SIMPLE algorithm [15]. Gradients were calculated using a second-order hybrid Gauss least squares method. Initial conditions for pressure and velocity were 0.0 Pascals and 0.0 meters per second, respectively.

### 3.1.5 Geometric Sensitivities

To make sure the results from STAR-CCM+ were not being influenced by the modeling assumptions (domain width and length), two different sensitivities were analyzed. The first was to confirm that the flow was fully developed where the data was being collected and the second was to confirm that the symmetric boundary conditions were not influencing the results. To do this, probes were placed upstream and towards the symmetry wall and compared to the probe in the center and end of the domain. The probe locations are shown in Fig. 3.3.



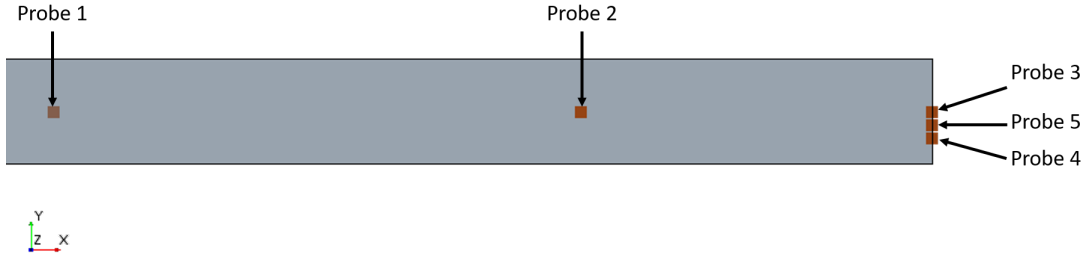


Figure 3.3: Top View of Section Geometry with Probe Locations

### Fully-Developed Sensitivity

Since there is a transition in the channel between uniform and fully-developed flow and the need to collect data in the fully-developed region, a fully-developed sensitivity study was completed. Line probes were placed in the center of the flow domain at various points downstream to extract the velocity profile. The difference between the velocity profiles at different line probe locations were calculated and normalized. Eq. 3.1 shows the calculation of the data and Table 3.1 shows the data collected on the coarsest grid (0.5 m).

$$Velocity_{normalized} = \frac{mean(\vec{V}_X - \vec{V}_Y)}{V_{ave}} \quad (3.1)$$

where  $\vec{V}$  is a vector of velocity values along a probe line and  $X$  and  $Y$  are different combinations of 15 Dh Downstream (probe 1), 26.25 Dh Downstream (probe 2), and 30 Dh Downstream (probe 3), which are shown in Fig 3.3.

Table 3.1: Fully-Developed Sensitivity Results

Simulation Base Size	Normalized Average Between Probe 1 and 2	Normalized Average Between Probe 1 and 3	Normalized Average Between Probe 2 and 3
0.5m	$6.987 * 10^{-4}$	$7.915 * 10^{-4}$	$8.2811 * 10^{-5}$

The comparison between probe 2 and probe 3, which was the difference between the velocity profiles near the outlet was about four magnitudes less than the average velocity. This shows that the velocity profile was effectively constant and therefore the flow was fully-developed at probe 3’s location.

### Symmetric Wall Sensitivity

To model the infinite plate, symmetry boundary conditions were used. To make sure the geometry used was wide enough that the symmetry boundary conditions did not influence the results in the center of the channel, three different probes were used. One in the center of the flow (probe 3), one halfway between probe 3 and the side (probe 4), and one between probe 3 and probe 4 (probe 5), which are shown in Fig 3.3. The results were normalized using Eq. 3.1 where  $X$  and  $Y$  are different combinations of probe 3, probe 4, and probe 5. The results of the velocity profile comparisons are shown in Table 3.2.

Table 3.2: Symmetry Sensitivity Results

Simulation Base Size	Normalized Average Between Probe 3 and 4	Normalized Average Between Probe 3 and 5	Normalized Average Between Probe 4 and 5
0.5m	$-3.096 * 10^{-5}$	$-9.234 * 10^{-6}$	$2.172 * 10^{-5}$

The comparison between probe 3, probe 4, and probe 5 are all relatively small. This confirms that the effect of the symmetry was negligible and that the geometry used was wide enough.

### 3.2 Generation of Error Terms

The list of error terms used in this study includes terms from an MEA of the discrete conservation equation as implemented in STAR-CCM+ and grid quality metrics that were

internally calculated by STAR-CCM+. While not used in this study, additional functions could be added to this list based on engineering judgment. The MEA was performed on a first-order upwind discretization of the conservation equations, even though the numerical scheme chosen in STAR-CCM+ was second-order upwind. This provided sufficient generation of terms that were applicable to both first or second-order schemes. Since this was an isothermal study, conservation of energy was not included in the MEA.

The approach used in this study was to calculate the local error between the discretized equations and the exact equations as a Taylor series. When the Taylor series is in the asymptotic region (i.e. the grid is resolving the dominant flow physics), the leading term in the Taylor series is approximately the local error. For this analysis, the form of the local error was used as a global error estimator. It is also important to note that when the Taylor series is not in the asymptotic region (i.e. the grid isn't resolving the dominant flow physics), higher order terms are necessary to describe the local error. Therefore, the LASSO analysis included one higher order term than the leading term in the Taylor series in case the grid does not produce a solution in the asymptotic range. Below is the MEA for continuity and conservation of momentum. The symbolic manipulation was completed using Mathematica to avoid mistakes.

### **3.2.1 Continuity MEA**

An MEA of the two dimensional, steady, constant velocity continuity equation with non-constant cell sizing was used. The terms found in this analysis aided in determining the error terms that were used in the LASSO analysis. The derivation used the notation in Figure 3.4 for indexing.

To start, Eq. 3.2 is the 2D continuity equation and Eq. 3.3 is the upwind discretization of the 2D continuity equation with non-constant cell size. Eq. 3.3's terms are replaced by Taylor series expansions.

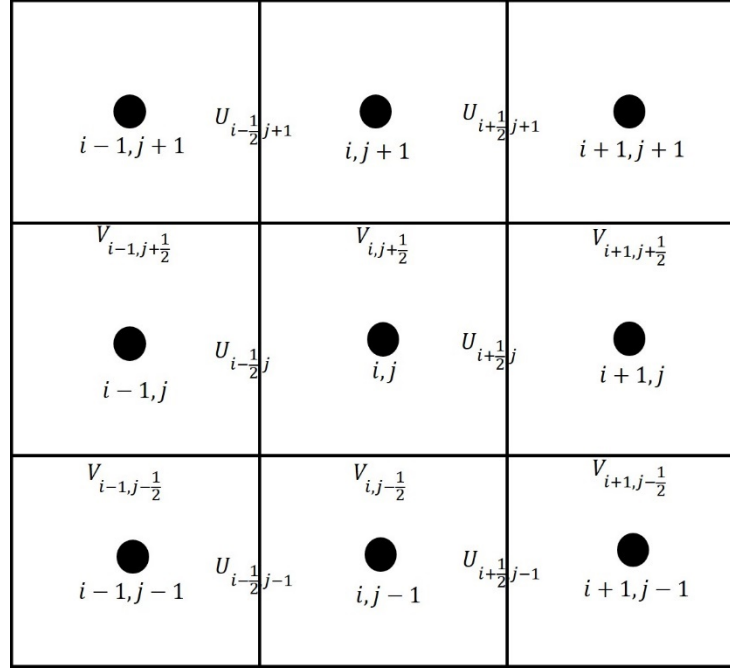


Figure 3.4: Indexing used for MEA

$$\nabla \rho u = 0 = u \left( \frac{\partial \rho}{\partial x} + \frac{\partial \rho}{\partial y} \right) + \rho \left( \frac{\partial u}{\partial x} + \frac{\partial u}{\partial y} \right) \quad (3.2)$$

$$u_{i,j} \left( \frac{\rho_{i,j} - \rho_{i-1,j}}{\Delta x_i} + \frac{\rho_{i,j} - \rho_{i,j-1}}{\Delta y_j} \right) + \rho_{i,j} \left( \frac{u_{i,j} - u_{i-1,j}}{\left( \frac{\Delta x_i + \Delta x_{i-1}}{2} \right)} + \frac{u_{i,j} - u_{i,j-1}}{\left( \frac{\Delta y_j + \Delta y_{j-1}}{2} \right)} \right) = 0 \quad (3.3)$$

where:  $u_{i,j} = \frac{1}{2} \left( u_{i+\frac{1}{2},j} + u_{i-\frac{1}{2},j} \right)$

$$u_{i-1,j} = \frac{1}{2} \left( u_{i-\frac{1}{2},j} + u_{i-\frac{3}{2},j} \right)$$

$$u_{i,j-1} = \frac{1}{2} \left( u_{i+\frac{1}{2},j-1} + u_{i-\frac{1}{2},j-1} \right)$$

$$\Delta x_i = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}$$

$$\Delta x_{i-1} = x_{i-\frac{1}{2}} - x_{i-\frac{3}{2}}$$

$$\Delta y_j = y_{j+\frac{1}{2}} - y_{j-\frac{1}{2}}$$

$$\Delta y_{j-1} = y_{j-\frac{1}{2}} - y_{j-\frac{3}{2}}$$

The density terms will be expanded around  $\rho_{i,j}$ , which is shown in Eq. 3.4 and Eq. 3.5.

$$\rho_{i-1,j} = \rho_{i,j} + \frac{\left(\frac{-\Delta x_i - \Delta x_{i-1}}{2}\right)}{1!} \frac{\partial \rho}{\partial x} \Big|_{i,j} + \frac{\left(\frac{-\Delta x_i - \Delta x_{i-1}}{2}\right)^2}{2!} \frac{\partial^2 \rho}{\partial x^2} \Big|_{i,j} + HOT \quad (3.4)$$

$$\rho_{i,j-1} = \rho_{i,j} + \frac{\left(\frac{-\Delta y_j - \Delta y_{j-1}}{2}\right)}{1!} \frac{\partial \rho}{\partial y} \Big|_{i,j} + \frac{\left(\frac{-\Delta y_j - \Delta y_{j-1}}{2}\right)^2}{2!} \frac{\partial^2 \rho}{\partial y^2} \Big|_{i,j} + HOT \quad (3.5)$$

The velocity in the x direction terms are expanded around  $u_{i,j}$ , which is shown in Eq. 3.6 though Eq. 3.10.

$$u_{i+\frac{1}{2},j} = u_{i,j} + \frac{\left(\frac{\Delta x_i}{2}\right)}{1!} \frac{\partial u}{\partial x} \Big|_{i,j} + \frac{\left(\frac{\Delta x_i}{2}\right)^2}{2!} \frac{\partial^2 u}{\partial x^2} \Big|_{i,j} + HOT \quad (3.6)$$

$$u_{i-\frac{1}{2},j} = u_{i,j} + \frac{\left(\frac{-\Delta x_i}{2}\right)}{1!} \frac{\partial u}{\partial x} \Big|_{i,j} + \frac{\left(\frac{-\Delta x_i}{2}\right)^2}{2!} \frac{\partial^2 u}{\partial x^2} \Big|_{i,j} + HOT \quad (3.7)$$

$$u_{i-\frac{3}{2},j} = u_{i,j} + \frac{\left(\frac{-\Delta x_i - \Delta x_{i-1}}{2}\right)}{1!} \frac{\partial u}{\partial x} \Big|_{i,j} + \frac{\left(\frac{-\Delta x_i - \Delta x_{i-1}}{2}\right)^2}{2!} \frac{\partial^2 u}{\partial x^2} \Big|_{i,j} + HOT \quad (3.8)$$

$$\begin{aligned} u_{i+\frac{1}{2},j-1} = & u_{i,j} + \frac{\left(\frac{\Delta x_i}{2}\right)}{1!} \frac{\partial u}{\partial y} \Big|_{i,j} + \frac{\left(\frac{\Delta x_i}{2}\right)^2}{2!} \frac{\partial^2 u}{\partial y^2} \Big|_{i,j} \\ & + \frac{\left(\frac{-\Delta y_j - \Delta y_{j-1}}{2}\right)}{1!} \frac{\partial u}{\partial y} \Big|_{i,j} + \frac{\left(\frac{-\Delta y_j - \Delta y_{j-1}}{2}\right)^2}{2!} \frac{\partial^2 u}{\partial y^2} \Big|_{i,j} + HOT \end{aligned} \quad (3.9)$$

$$\begin{aligned} u_{i-\frac{1}{2},j-1} = & u_{i,j} + \frac{\left(\frac{-\Delta y_j - \Delta y_{j-1}}{2}\right)}{1!} \frac{\partial u}{\partial y} \Big|_{i,j} + \frac{\left(\frac{-\Delta y_j - \Delta y_{j-1}}{2}\right)^2}{2!} \frac{\partial^2 u}{\partial y^2} \Big|_{i,j} \\ & + \frac{\left(\frac{-\Delta x_i}{2}\right)}{1!} \frac{\partial u}{\partial y} \Big|_{i,j} + \frac{\left(\frac{-\Delta x_i}{2}\right)^2}{2!} \frac{\partial^2 u}{\partial y^2} \Big|_{i,j} + HOT \end{aligned} \quad (3.10)$$

The averaged terms from Eq. 3.3 are calculated in Eq. 3.11 through Eq. 3.13.

$$\begin{aligned}
u_{i,j} &= \frac{1}{2}(u_{i+\frac{1}{2},j} + u_{i-\frac{1}{2},j}) \\
&= u_{i,j} + \frac{1}{8}\Delta x_i^2 \left. \frac{\partial^2 u}{\partial x^2} \right|_{i,j} + HOT
\end{aligned} \tag{3.11}$$

$$\begin{aligned}
u_{i-1,j} &= \frac{1}{2}(u_{i-\frac{1}{2},j} + u_{i-\frac{3}{2},j}) \\
&= u_{i,j} - \frac{1}{2}\Delta x_i \left. \frac{\partial u}{\partial x} \right|_{i,j} - \frac{1}{2}\Delta x_{i-1} \left. \frac{\partial u}{\partial x} \right|_{i,j} + \frac{1}{8}\Delta x_i^2 \left. \frac{\partial^2 u}{\partial x^2} \right|_{i,j} \\
&\quad + \frac{1}{4}\Delta x_i \Delta x_{i-1} \left. \frac{\partial^2 u}{\partial x^2} \right|_{i,j} + \frac{1}{4}\Delta x_{i-1}^2 \left. \frac{\partial^2 u}{\partial x^2} \right|_{i,j} + HOT
\end{aligned} \tag{3.12}$$

$$\begin{aligned}
u_{i,j-1} &= \frac{1}{2}(u_{i+\frac{1}{2},j-1} + u_{i-\frac{1}{2},j-1}) \\
&= u_{i,j} - \frac{1}{2}\Delta y_j \left. \frac{\partial u}{\partial y} \right|_{i,j} - \frac{1}{2}\Delta y_{j-1} \left. \frac{\partial u}{\partial y} \right|_{i,j} + \frac{1}{8}\Delta y_j^2 \left. \frac{\partial^2 u}{\partial y^2} \right|_{i,j} \\
&\quad + \frac{1}{4}\Delta y_j \Delta y_{j-1} \left. \frac{\partial^2 u}{\partial y^2} \right|_{i,j} + \frac{1}{4}\Delta y_{j-1}^2 \left. \frac{\partial^2 u}{\partial y^2} \right|_{i,j} + \frac{1}{8}\Delta x_i^2 \left. \frac{\partial^2 u}{\partial x^2} \right|_{i,j} + HOT
\end{aligned} \tag{3.13}$$

Next, Eq. 3.4, Eq. 3.5, and Eq. 3.11 through 3.13 are substituted back into Eq. 3.3, which is shown in Eq. 3.14.

$$\begin{aligned}
& u_{i,j} \left( \frac{\rho_{i,j} - \left( \rho_{i,j} + \frac{(-\Delta x_i - \Delta x_{i-1})}{1!} \frac{\partial \rho}{\partial x} \Big|_{i,j} + \frac{(-\Delta x_i - \Delta x_{i-1})^2}{2!} \frac{\partial^2 \rho}{\partial x^2} \Big|_{i,j} + HOT \right)}{\Delta x_i} \right. \\
& \left. + \frac{\rho_{i,j} - \left( \rho_{i,j} + \frac{(-\Delta y_j - \Delta y_{j-1})}{1!} \frac{\partial \rho}{\partial y} \Big|_{i,j} + \frac{(-\Delta y_j - \Delta y_{j-1})^2}{2!} \frac{\partial^2 \rho}{\partial y^2} \Big|_{i,j} + HOT \right)}{\Delta y_j} \right) \\
& + \frac{\rho_{i,j}}{\left( \frac{\Delta x_i + \Delta x_{i-1}}{2} \right)} \left[ \left( u_{i,j} + \frac{1}{8} \Delta x_i^2 \frac{\partial^2 u}{\partial x^2} \Big|_{i,j} + HOT \right) - \left( u_{i,j} - \frac{1}{2} \Delta x_i \frac{\partial u}{\partial x} \Big|_{i,j} \right. \right. \\
& \left. \left. - \frac{1}{2} \Delta x_{i-1} \frac{\partial u}{\partial x} \Big|_{i,j} + \frac{1}{8} \Delta x_i^2 \frac{\partial^2 u}{\partial x^2} \Big|_{i,j} + \frac{1}{4} \Delta x_i \Delta x_{i-1} \frac{\partial^2 u}{\partial x^2} \Big|_{i,j} + \frac{1}{4} \Delta x_{i-1}^2 \frac{\partial^2 u}{\partial x^2} \Big|_{i,j} + HOT \right) \right] \\
& + \frac{\rho_{i,j}}{\left( \frac{\Delta y_j + \Delta y_{j-1}}{2} \right)} \left[ \left( u_{i,j} + \frac{1}{8} \Delta x_i^2 \frac{\partial^2 u}{\partial x^2} \Big|_{i,j} + HOT \right) - \left( u_{i,j} - \frac{1}{2} \Delta y_j \frac{\partial u}{\partial y} \Big|_{i,j} - \frac{1}{2} \Delta y_{j-1} \frac{\partial u}{\partial y} \Big|_{i,j} \right. \right. \\
& \left. \left. + \frac{1}{8} \Delta y_j^2 \frac{\partial^2 u}{\partial y^2} \Big|_{i,j} + \frac{1}{4} \Delta y_j \Delta y_{j-1} \frac{\partial^2 u}{\partial y^2} \Big|_{i,j} + \frac{1}{4} \Delta y_{j-1}^2 \frac{\partial^2 u}{\partial y^2} \Big|_{i,j} + \frac{1}{8} \Delta x_i^2 \frac{\partial^2 u}{\partial x^2} \Big|_{i,j} + HOT \right) \right] = 0
\end{aligned} \tag{3.14}$$

Eq. 3.15 simplifies and reorganizes Eq. 3.14 into the original PDE terms and the error terms. Eq. 3.16 shows the relationship between  $\Delta x_i$  and  $\Delta x_{i-1}$  (and  $\Delta y_j$  and  $\Delta y_{j-1}$ ) in terms of  $r$ , which is the same  $r$  used in Eq. 2.16 and Eq. 2.17. Eq. 3.17 removes the PDEs from 3.15 and substitutes Eq. 3.16 to simplify the equation. The terms in square brackets are assumed to be constant as a function of the grid spacing and outside the square brackets are the error terms.

$$\begin{aligned}
& u_{i,j} \left( \frac{\partial \rho}{\partial y} \Big|_{i,j} + \frac{\partial \rho}{\partial x} \Big|_{i,j} \right) + u_{i,j} \left( \frac{1}{2} \left( \frac{\Delta y_{j-1}}{y_j} - 1 \right) \frac{\partial u}{\partial y} \Big|_{i,j} \right. \\
& + \frac{1}{2} \left( \frac{\Delta x_{i-1}}{x_i} - 1 \right) \frac{\partial u}{\partial x} \Big|_{i,j} - \left( \frac{1}{8} \Delta y_j + \frac{1}{4} \Delta y_{j-1} + \frac{1}{8} \Delta y_{j-1} \frac{\Delta y_{j-1}}{\Delta y_j} \right) \frac{\partial^2 u}{\partial y^2} \Big|_{i,j} \\
& \left. + \left( \frac{1}{8} \Delta x_i + \frac{1}{4} \Delta x_{i-1} + \frac{1}{8} \Delta x_{i-1} \frac{\Delta x_{i-1}}{\Delta x_i} \right) \frac{\partial^2 u}{\partial x^2} \Big|_{i,j} \right) \\
& + \rho_{i,j} \left( \frac{\partial u}{\partial y} \Big|_{i,j} + \frac{\partial u}{\partial x} \Big|_{i,j} \right) \\
& - \rho_{i,j} \left( \frac{1}{4} (\Delta y_j + \Delta y_{j-1}) \frac{\partial^2 u}{\partial y^2} \Big|_{i,j} + \frac{1}{2} (\Delta x_{i-1}) \frac{\partial^2 u}{\partial x^2} \Big|_{i,j} \right) = 0
\end{aligned} \tag{3.15}$$

$$r = \frac{\Delta x_i}{\Delta x_{i-1}} \text{ or } \Delta x_i = r \Delta x_{i-1} \tag{3.16}$$

$$\begin{aligned}
& \left( \frac{1}{r} - 1 \right) \left[ \frac{u_{i,j}}{2} \frac{\partial u}{\partial y} \Big|_{i,j} + \frac{u_{i,j}}{2} \frac{\partial u}{\partial x} \Big|_{i,j} \right] - (\Delta y_{j-1}) \left( \frac{(r+1)^2}{r} \right) \left[ \frac{u_{i,j}}{8} \frac{\partial^2 u}{\partial y^2} \Big|_{i,j} \right] \\
& - (\Delta x_{i-1}) \left( \frac{(r+1)^2}{r} \right) \left[ \frac{u_{i,j}}{8} \frac{\partial^2 u}{\partial x^2} \Big|_{i,j} \right] \\
& - (\Delta y_{j-1}) (r+1) \left[ \frac{\rho_{i,j}}{4} \frac{\partial^2 u}{\partial y^2} \Big|_{i,j} \right] - (\Delta x_{i-1}) \left[ \frac{\rho_{i,j}}{2} \frac{\partial^2 u}{\partial x^2} \Big|_{i,j} \right]
\end{aligned} \tag{3.17}$$

There are five unique terms in Eq. 3.17: one cell size ratio term, one  $\Delta x_{i-1}$  term, and three combination terms that are made up between a  $\Delta y_{j-1}$  or  $\Delta x_{i-1}$  term and a ratio term.

### 3.2.2 Conservation of Momentum MEA

An MEA of the two dimensional, steady, constant density conservation of momentum equation in the x direction with non-constant cell sizing was used. An MEA using a non-uniform grid was necessary to complete because the grid generated in STAR-CCM+ was not guaranteed to be uniform. The conservative inviscid 2D momentum equation in the x



direction is shown in Eq. 3.18. Assuming a constant pressure drop and density, Eq. 3.18 turns into Eq. 3.19 and the general discretized form of Eq. 3.19 is shown in Eq. 3.20.

$$\frac{\partial \rho u^2}{\partial x} + \frac{\partial \rho uv}{\partial y} = -\frac{\partial P}{\partial x}. \quad (3.18)$$

$$\frac{\partial u^2}{\partial x} + \frac{\partial uv}{\partial y} = -\frac{1}{\rho} \frac{\partial P}{\partial x} = \text{const.} \quad (3.19)$$

$$\frac{(u_{i+1,j} - u_{i,j})^2}{\Delta x_i} + \frac{\left(u_{i+\frac{1}{2},j+\frac{1}{2}} - u_{i+\frac{1}{2},j-\frac{1}{2}}\right) \left(v_{i+\frac{1}{2},j+\frac{1}{2}} - v_{i+\frac{1}{2},j-\frac{1}{2}}\right)}{\frac{1}{2}(\Delta y_j + \Delta y_{j+1})} = \text{const.} \quad (3.20)$$

Eq. 3.21 is the upwinded discretization of Eq. 3.18. Note: the first velocity in the x direction upwinds the second velocity in the x direction term and the velocity in the y direction upwinds the velocity in the x direction term.

$$\frac{\left(u_{i+\frac{1}{2},j} - u_{i-\frac{1}{2},j}\right) (u_{i+1,j} - u_{i,j})}{\Delta x_i} + \frac{\left(u_{i+\frac{1}{2},j} - u_{i+\frac{1}{2},j-1}\right) \left(v_{i+\frac{1}{2},j+\frac{1}{2}} - v_{i+\frac{1}{2},j-\frac{1}{2}}\right)}{\frac{1}{2}(\Delta y_j + \Delta y_{j+1})} = \text{const.} \quad (3.21)$$

where:

$$\begin{aligned} u_{i+1,j} &\approx \frac{1}{2} \left( u_{i+\frac{3}{2},j} + u_{i+\frac{1}{2},j} \right) \\ u_{i,j} &\approx \frac{1}{2} \left( u_{i+\frac{1}{2},j} + u_{i-\frac{1}{2},j} \right) \\ v_{i+\frac{1}{2},j+\frac{1}{2}} &\approx \frac{1}{2} \left( v_{i,j+\frac{1}{2}} + v_{i+1,j+\frac{1}{2}} \right) \\ v_{i+\frac{1}{2},j-\frac{1}{2}} &\approx \frac{1}{2} \left( v_{i,j-\frac{1}{2}} + v_{i+1,j-\frac{1}{2}} \right) \\ \Delta x_i &= x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}} \\ \Delta x_{i+1} &= x_{i+\frac{3}{2}} - x_{i+\frac{1}{2}} \\ \Delta y_j &= y_{j-\frac{1}{2}} - y_{j-\frac{3}{2}} \\ \Delta y_{j+1} &= y_{j+\frac{1}{2}} - y_{j-\frac{1}{2}} \end{aligned}$$

Next, the Taylor series is computed around  $i + \frac{1}{2}, j$  for both the x and y direction. Eq. 3.22 through Eq. 3.25 are the Taylor series for velocity in the x direction and Eq. 3.26 through Eq. 3.29 are the Taylor series for velocity in the y direction.

$$u_{i+\frac{1}{2},j} = u_{i+\frac{1}{2},j} \quad (3.22)$$

$$u_{i-\frac{1}{2},j} = u_{i+\frac{1}{2},j} + \frac{(-\Delta x_i)}{1!} \frac{\partial u}{\partial x} \Big|_{i+\frac{1}{2},j} + \frac{(-\Delta x_i)^2}{2!} \frac{\partial^2 u}{\partial x^2} \Big|_{i+\frac{1}{2},j} + HOT \quad (3.23)$$

$$u_{i+\frac{3}{2},j} = u_{i+\frac{1}{2},j} + \frac{(\Delta x_{i+1})}{1!} \frac{\partial u}{\partial x} \Big|_{i+\frac{1}{2},j} + \frac{(\Delta x_{i+1})^2}{2!} \frac{\partial^2 u}{\partial x^2} \Big|_{i+\frac{1}{2},j} + HOT \quad (3.24)$$

$$u_{i+\frac{1}{2},j-1} = u_{i+\frac{1}{2},j} + \frac{(-\frac{1}{2}(\Delta y_j + \Delta y_{j+1}))}{1!} \frac{\partial u}{\partial x} \Big|_{i+\frac{1}{2},j} + \frac{(-\frac{1}{2}(\Delta y_j + \Delta y_{j+1}))^2}{2!} \frac{\partial^2 u}{\partial x^2} \Big|_{i+\frac{1}{2},j} + HOT \quad (3.25)$$

$$v_{i,j+\frac{1}{2}} = v_{i+\frac{1}{2},j} + \frac{(-\frac{\Delta x_i}{2})}{1!} \frac{\partial v}{\partial x} \Big|_{i+\frac{1}{2},j} + \frac{(-\frac{\Delta x_i}{2})^2}{2!} \frac{\partial^2 v}{\partial x^2} \Big|_{i+\frac{1}{2},j} + \frac{(\frac{\Delta y_{j+1}}{2})}{1!} \frac{\partial v}{\partial y} \Big|_{i+\frac{1}{2},j} + \frac{(\frac{\Delta y_{j+1}}{2})^2}{2!} \frac{\partial^2 v}{\partial y^2} \Big|_{i+\frac{1}{2},j} + \frac{(\frac{\Delta y_{j+1}}{2})}{1!} \frac{(-\frac{\Delta x_i}{2})}{1!} \frac{\partial}{\partial y} \left( \frac{\partial v}{\partial x} \right) \Big|_{i+\frac{1}{2},j} + HOT \quad (3.26)$$

$$v_{i,j-\frac{1}{2}} = v_{i+\frac{1}{2},j} + \frac{(-\frac{\Delta x_i}{2})}{1!} \frac{\partial v}{\partial x} \Big|_{i+\frac{1}{2},j} + \frac{(-\frac{\Delta x_i}{2})^2}{2!} \frac{\partial^2 v}{\partial x^2} \Big|_{i+\frac{1}{2},j} + \frac{(-\frac{\Delta y_{j+1}}{2})}{1!} \frac{\partial v}{\partial y} \Big|_{i+\frac{1}{2},j} + \frac{(-\frac{\Delta y_{j+1}}{2})^2}{2!} \frac{\partial^2 v}{\partial y^2} \Big|_{i+\frac{1}{2},j} + \frac{(-\frac{\Delta y_{j+1}}{2})}{1!} \frac{(-\frac{\Delta x_i}{2})}{1!} \frac{\partial}{\partial y} \left( \frac{\partial v}{\partial x} \right) \Big|_{i+\frac{1}{2},j} + HOT \quad (3.27)$$

$$v_{i+1,j+\frac{1}{2}} = v_{i+\frac{1}{2},j} + \frac{(\frac{\Delta x_i}{2})}{1!} \frac{\partial v}{\partial x} \Big|_{i+\frac{1}{2},j} + \frac{(\frac{\Delta x_i}{2})^2}{2!} \frac{\partial^2 v}{\partial x^2} \Big|_{i+\frac{1}{2},j} + \frac{(\frac{\Delta y_{j+1}}{2})}{1!} \frac{\partial v}{\partial y} \Big|_{i+\frac{1}{2},j} + \frac{(\frac{\Delta y_{j+1}}{2})^2}{2!} \frac{\partial^2 v}{\partial y^2} \Big|_{i+\frac{1}{2},j} + \frac{(\frac{\Delta y_{j+1}}{2})}{1!} \frac{(\frac{\Delta x_i}{2})}{1!} \frac{\partial}{\partial y} \left( \frac{\partial v}{\partial x} \right) \Big|_{i+\frac{1}{2},j} + HOT \quad (3.28)$$

$$\begin{aligned}
v_{i+1,j-\frac{1}{2}} = & v_{i+\frac{1}{2},j} + \frac{\left(\frac{\Delta x_i}{2}\right)}{1!} \frac{\partial v}{\partial x} \Big|_{i+\frac{1}{2},j} + \frac{\left(\frac{\Delta x_i}{2}\right)^2}{2!} \frac{\partial^2 v}{\partial x^2} \Big|_{i+\frac{1}{2},j} + \frac{\left(\frac{-\Delta y_{j+1}}{2}\right)}{1!} \frac{\partial v}{\partial y} \Big|_{i+\frac{1}{2},j} \\
& + \frac{\left(\frac{-\Delta y_{j+1}}{2}\right)^2}{2!} \frac{\partial^2 v}{\partial y^2} \Big|_{i+\frac{1}{2},j} + \frac{\left(\frac{-\Delta y_{j+1}}{2}\right)}{1!} \frac{\left(\frac{\Delta x_i}{2}\right)}{1!} \frac{\partial}{\partial y} \left( \frac{\partial v}{\partial x} \right) \Big|_{i+\frac{1}{2},j} + HOT
\end{aligned} \quad (3.29)$$

Next, velocity averages in the x direction from Eq. 3.21 are calculated in Eq. 3.30 and Eq. 3.31.

$$\frac{1}{2} \left( u_{i+\frac{3}{2},j} + u_{i+\frac{1}{2},j} \right) = u_{i+\frac{1}{2},j} + \frac{1}{2} \frac{(\Delta x_{i+1})}{1!} \frac{\partial u}{\partial x} \Big|_{i+\frac{1}{2},j} + \frac{1}{2} \frac{(\Delta x_{i+1})^2}{2!} \frac{\partial^2 u}{\partial x^2} \Big|_{i+\frac{1}{2},j} + HOT \quad (3.30)$$

$$\frac{1}{2} \left( u_{i+\frac{1}{2},j} + u_{i-\frac{1}{2},j} \right) = u_{i+\frac{1}{2},j} + \frac{1}{2} \frac{(-\Delta x_i)}{1!} \frac{\partial u}{\partial x} \Big|_{i+\frac{1}{2},j} + \frac{1}{2} \frac{(-\Delta x_i)^2}{2!} \frac{\partial^2 u}{\partial x^2} \Big|_{i+\frac{1}{2},j} + HOT \quad (3.31)$$

Next, velocity averages in the y direction from Eq. 3.21 are calculated in Eq. 3.32 and Eq. 3.33.

$$\begin{aligned}
\frac{1}{2} \left( v_{i,j+\frac{1}{2}} + v_{i+1,j+\frac{1}{2}} \right) = & v_{i+\frac{1}{2},j} + \frac{\left(\frac{-\Delta x_i}{2}\right)^2}{2!} \frac{\partial^2 v}{\partial x^2} \Big|_{i+\frac{1}{2},j} + \frac{\left(\frac{\Delta y_{j+1}}{2}\right)}{1!} \frac{\partial v}{\partial y} \Big|_{i+\frac{1}{2},j} \\
& + \frac{\left(\frac{\Delta y_{j+1}}{2}\right)^2}{2!} \frac{\partial^2 v}{\partial y^2} \Big|_{i+\frac{1}{2},j} + HOT
\end{aligned} \quad (3.32)$$

$$\begin{aligned}
\frac{1}{2} \left( v_{i,j-\frac{1}{2}} + v_{i+1,j-\frac{1}{2}} \right) = & v_{i+\frac{1}{2},j} + \frac{\left(\frac{-\Delta x_i}{2}\right)^2}{2!} \frac{\partial^2 v}{\partial x^2} \Big|_{i+\frac{1}{2},j} + \frac{\left(\frac{-\Delta y_{j+1}}{2}\right)}{1!} \frac{\partial v}{\partial y} \Big|_{i+\frac{1}{2},j} \\
& + \frac{\left(\frac{-\Delta y_{j+1}}{2}\right)^2}{2!} \frac{\partial^2 v}{\partial y^2} \Big|_{i+\frac{1}{2},j} + HOT
\end{aligned} \quad (3.33)$$

The error term from the  $\frac{\partial u^2}{\partial x}$  term is computed next. The Taylor series from Eq. 3.22 and Eq. 3.23 and the averaged terms in Eq. 3.30 and Eq. 3.31 are substituted into the

upwind discretized  $\frac{\partial u^2}{\partial x}$  term (Eq. 3.34). Eq. 3.35 simplifies and rearranges Eq. 3.34 in its original PDE form plus the error term.

$$\begin{aligned}
& \frac{\left( u_{i+\frac{1}{2},j} u_{i+1,j} - u_{i-\frac{1}{2},j} u_{i,j} \right)}{\Delta x_i} \\
&= \frac{\left( u_{i+\frac{1}{2},j}^2 + \frac{1}{2} (\Delta x_{i+1}) \frac{\partial u}{\partial x} \Big|_{i+\frac{1}{2},j} u_{i+\frac{1}{2},j} + \frac{1}{2} \frac{(\Delta x_{i+1})^2}{2!} \frac{\partial^2 u}{\partial x^2} \Big|_{i+\frac{1}{2},j} u_{i+\frac{1}{2},j} + HOT \right)}{\Delta x_i} \\
& \quad - \frac{1}{\Delta x_i} \left( u_{i+\frac{1}{2},j}^2 + \frac{3}{2} (-\Delta x_i) \frac{\partial u}{\partial x} \Big|_{i+\frac{1}{2},j} u_{i+\frac{1}{2},j} + \frac{3}{2} \frac{(-\Delta x_i)^2}{2!} \frac{\partial^2 u}{\partial x^2} \Big|_{i+\frac{1}{2},j} u_{i+\frac{1}{2},j} \right. \\
& \quad \quad \quad \left. + \frac{1}{2} (-\Delta x_i)^2 \frac{\partial u}{\partial x} \Big|_{i+\frac{1}{2},j}^2 + HOT \right)
\end{aligned} \tag{3.34}$$

$$\begin{aligned}
& 2u_{i+\frac{1}{2},j} \frac{\partial u}{\partial x} \Big|_{i+\frac{1}{2},j} + \frac{1}{2} \left( \frac{\Delta x_{i+1}}{\Delta x_i} - 1 \right) u_{i+\frac{1}{2},j} \frac{\partial u}{\partial x} \Big|_{i+\frac{1}{2},j} \\
& + \frac{1}{4} \left( \frac{\Delta x_{i+1}^2}{\Delta x_i} - 3\Delta x_i \right) \frac{\partial^2 u}{\partial x^2} \Big|_{i+\frac{1}{2},j} u_{i+\frac{1}{2},j} - \frac{\Delta x_i}{2} \frac{\partial u}{\partial x} \Big|_{i+\frac{1}{2},j}^2 + HOT
\end{aligned} \tag{3.35}$$

Eq. 3.35 removes the PDE terms, Eq. 3.16 is substituted in, and is rewritten as a function of  $\Delta x$  terms. This helped inform which grid metrics were used in the LASSO analysis. The terms in square brackets of Eq. 3.36 were assumed to be constant as a function of the grid spacing and outside the square brackets are the error terms.

$$\left( \frac{1}{r} - 1 \right) \left[ \frac{1}{2} u_{i+\frac{1}{2},j} \frac{\partial u}{\partial x} \Big|_{i+\frac{1}{2},j} \right] - (\Delta x_{i+1}) \left( \frac{1}{r} \right) \left[ \frac{1}{2} \frac{\partial^2 u}{\partial x^2} \Big|_{i+\frac{1}{2},j} u_{i+\frac{1}{2},j} + \frac{1}{2} \frac{\partial u}{\partial x} \Big|_{i+\frac{1}{2},j}^2 \right] + HOT \tag{3.36}$$

There are 2 unique error terms in Eq. 3.36: one is a  $\Delta x$  ratio term and a  $\Delta x$  term divided by a  $\Delta x$  ratio term.

The error term from the  $\frac{\partial uv}{\partial y}$  term is computed next. The Taylor series from Eq. 3.22 and Eq. 3.25 and the averaged terms from Eq. 3.32 and Eq. 3.33 are substituted into the upwind discretized  $\frac{\partial uv}{\partial y}$  term (Eq. 3.37). Eq. 3.38 simplified and Eq. 3.39 rearranged Eq. 3.37 in its original PDE form plus the error term.

$$\begin{aligned}
& \frac{\left(u_{i+\frac{1}{2},j}v_{i+\frac{1}{2},j+\frac{1}{2}} - u_{i+\frac{1}{2},j-1}v_{i+\frac{1}{2},j-\frac{1}{2}}\right)}{\frac{1}{2}(\Delta y_j + \Delta y_{j+1})} \\
&= \frac{1}{\frac{1}{2}(\Delta y_j + \Delta y_{j+1})} \left[ v_{i+\frac{1}{2},j}u_{i+\frac{1}{2},j} + \frac{\left(\frac{\Delta y_{j+1}}{2}\right)}{1!} \frac{\partial v}{\partial y} \Big|_{i+\frac{1}{2},j} u_{i+\frac{1}{2},j} \right. \\
&\quad \left. + \frac{\left(\frac{-\Delta x_i}{2}\right)^2}{2!} \frac{\partial^2 v}{\partial x^2} \Big|_{i+\frac{1}{2},j} u_{i+\frac{1}{2},j} + \frac{\left(\frac{\Delta y_{j+1}}{2}\right)^2}{2!} \frac{\partial^2 v}{\partial y^2} \Big|_{i+\frac{1}{2},j} u_{i+\frac{1}{2},j} + HOT \right] \\
&- \frac{1}{\frac{1}{2}(\Delta y_j + \Delta y_{j+1})} \left[ u_{i+\frac{1}{2},j} + \frac{\left(-\frac{1}{2}(\Delta y_j + \Delta y_{j+1})\right)}{1!} \frac{\partial u}{\partial y} \Big|_{i+\frac{1}{2},j} \right. \\
&\quad \left. + \frac{\left(-\frac{1}{2}(\Delta y_j + \Delta y_{j+1})\right)^2}{2!} \frac{\partial^2 u}{\partial y^2} \Big|_{i+\frac{1}{2},j} + HOT \right] \left[ v_{i+\frac{1}{2},j} + \frac{\left(\frac{-\Delta x_i}{2}\right)^2}{2!} \frac{\partial^2 v}{\partial x^2} \Big|_{i+\frac{1}{2},j} \right. \\
&\quad \left. + \frac{\left(\frac{-\Delta y_{j+1}}{2}\right)}{1!} \frac{\partial v}{\partial y} \Big|_{i+\frac{1}{2},j} + \frac{\left(\frac{-\Delta y_{j+1}}{2}\right)^2}{2!} \frac{\partial^2 v}{\partial y^2} \Big|_{i+\frac{1}{2},j} + HOT \right] \tag{3.37}
\end{aligned}$$

$$\begin{aligned}
\frac{u_{i+\frac{1}{2},j-1}v_{i+\frac{1}{2},j-\frac{1}{2}}}{\frac{1}{2}(\Delta y_j + \Delta y_{j+1})} &= \frac{1}{\frac{1}{2}(\Delta y_j + \Delta y_{j+1})} \left[ u_{i+\frac{1}{2},j}v_{i+\frac{1}{2},j} + \frac{\left(\frac{\Delta x_i}{2}\right)^2}{2!} \frac{\partial^2 v}{\partial x^2} \Big|_{i+\frac{1}{2},j} u_{i+\frac{1}{2},j} \right. \\
&\quad + \frac{\left(\frac{-\Delta y_{j+1}}{2}\right)}{1!} \frac{\partial v}{\partial y} \Big|_{i+\frac{1}{2},j} u_{i+\frac{1}{2},j} + \frac{\left(\frac{-\Delta y_{j+1}}{2}\right)^2}{2!} \frac{\partial^2 v}{\partial y^2} \Big|_{i+\frac{1}{2},j} u_{i+\frac{1}{2},j} \\
&\quad + \frac{\left(-\frac{1}{2}(\Delta y_j + \Delta y_{j+1})\right)}{1!} \frac{\partial u}{\partial y} \Big|_{i+\frac{1}{2},j} \frac{\left(\frac{-\Delta y_{j+1}}{2}\right)}{1!} \frac{\partial v}{\partial y} \Big|_{i+\frac{1}{2},j} \\
&\quad + \frac{\left(-\frac{1}{2}(\Delta y_j + \Delta y_{j+1})\right)}{1!} \frac{\partial u}{\partial y} \Big|_{i+\frac{1}{2},j} v_{i+\frac{1}{2},j} \\
&\quad \left. + \frac{\left(-\frac{1}{2}(\Delta y_j + \Delta y_{j+1})\right)^2}{2!} \frac{\partial^2 u}{\partial y^2} \Big|_{i+\frac{1}{2},j} v_{i+\frac{1}{2},j} + HOT \right] \tag{3.38}
\end{aligned}$$

$$\begin{aligned}
& \frac{\partial v}{\partial y} \Big|_{i+\frac{1}{2},j} u_{i+\frac{1}{2},j} + \frac{\partial u}{\partial y} \Big|_{i+\frac{1}{2},j} v_{i+\frac{1}{2},j} + \frac{\left(\frac{\Delta y_{j+1}}{\Delta y_j} - 1\right)}{\left(\frac{\Delta y_{j+1}}{\Delta y_j} + 1\right)} \frac{\partial v}{\partial y} \Big|_{i+\frac{1}{2},j} u_{i+\frac{1}{2},j} \\
& - \frac{1}{2} (\Delta y_{j+1}) \frac{\partial u}{\partial y} \Big|_{i+\frac{1}{2},j} \frac{\partial v}{\partial y} \Big|_{i+\frac{1}{2},j} - \frac{1}{4} (\Delta y_j + \Delta y_{j+1}) \frac{\partial^2 u}{\partial y^2} \Big|_{i+\frac{1}{2},j} v_{i+\frac{1}{2},j} + H O T
\end{aligned} \tag{3.39}$$

Eq. 3.40 removes the PDE terms, Eq. 3.16 is substituted in, and is rewritten as a function of  $\Delta x$  and  $\Delta y$  terms. This helped inform which grid metrics were used in the LASSO analysis. The terms in square brackets of Eq. 3.40 were assumed to be constant as a function of the grid spacing and outside the square brackets are the error terms.

$$\begin{aligned}
& \left(\frac{1}{r} - 1\right) \left[ \frac{\partial v}{\partial y} \Big|_{i+\frac{1}{2},j} u_{i+\frac{1}{2},j} \right] - (\Delta y_{j+1}) \left[ \frac{1}{2} \frac{\partial u}{\partial y} \Big|_{i+\frac{1}{2},j} \frac{\partial v}{\partial y} \Big|_{i+\frac{1}{2},j} \right] \\
& - (\Delta y_{j+1}) \left(\frac{1}{r} + 1\right) \left[ \frac{1}{4} \frac{\partial^2 u}{\partial y^2} \Big|_{i+\frac{1}{2},j} v_{i+\frac{1}{2},j} \right] + H O T
\end{aligned} \tag{3.40}$$

There are three unique error terms in Eq. 3.40: one  $\Delta y$  ratio term, one  $\Delta y_{j-1}$  term, and one combination term between a  $\Delta y_{j-1}$  term and a  $\Delta y$  ratio term.

### 3.2.3 Cell Size Estimation Terms

The terms from the above MEA were all functions of the cell size. There were two main ways to estimate the cell size: the cell size specification used during grid generation or the number of cells. Both were implemented in the LASSO analysis to determine which was a better cell size estimator. The next two sections describe calculation of the two cell size estimators.

#### Cell Size Estimation Term - Cell Size Specification

The user defined cell size method is a method to estimate the average cell size by using the user defined cell size "b" used by the grid generator to build the grid. For fine grids without local refinements, this method typically estimates the average cell size well. For

coarse grids, a small change to the cell size input does not guarantee a change in the grid. Therefore, other terms were necessary to characterize the average cell size.

### **Cell Size Estimation Term - Number of Cells**

The number of cells method is a method to estimate the average cell size based on the number of cells and the volume of the domain. Eq. 3.41 shows the calculation of the average cell size. Using this metric improved the calculation of the average cell size for coarse grids

$$h = \left( \frac{V}{N_c} \right)^{\frac{1}{3}} \quad (3.41)$$

where  $V$  is the volume of the domain and  $N_c$  is the total number of cells.

### **Summary of Error Terms from the Numerical Scheme**

After reviewing the terms that are in Eq. 3.17, Eq. 3.36, and Eq. 3.40, the unique terms were classified into three categories: grid spacing, grid spacing ratio, and a combination of grid spacing terms. The grid spacing was a function of the cell size ( $h$ ), the grid spacing ratio was a function of how the cell size changes, and the combination term were a function of both. The grid spacing ratio was described using grid quality metrics, which are described in the following section.

#### **3.2.4 Grid Quality Terms**

In addition to error terms due to the numerical scheme used, grid quality terms were used to describe error due to poor grid quality. The grid quality metrics that were calculated by STAR-CCM+ were used to describe errors due to a low quality grid. Figures 3.5 through 3.12 and paraphrased grid metric definitions are from the STAR-CCM+ user manual [14] to provide completeness.

### Cell Aspect Ratio

Cell aspect ratio is a metric that describes the length of the cell in one direction is to another direction. Values can be between 0.0 and 1.0. Perfect hexahedral cells are cube and have an aspect ratio of 1.0. As one direction becomes larger, the aspect ratio becomes less than 1.0. This parameter was not added to the error model because it is more useful to be used as a screening method by accepting or rejecting individual grids. A less than ideal cell aspect ratio is shown in Figure 3.5.

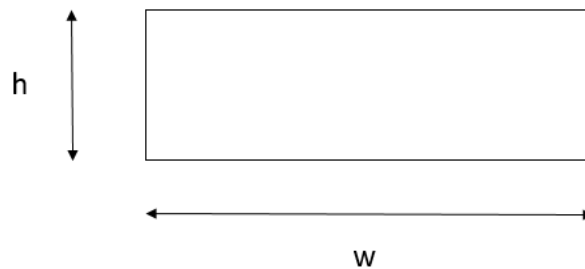


Figure 3.5: Cell Aspect Ratio Example. Reprinted from [14].

### Face Validity

Face validity is an area-weighted measure of the correctness of the face normal. The face validity is good when the face normal is pointing outward away from the cell and is bad when the face normal is pointing inwards towards the cell. Values between 0.5 and 1.0 are considered good, while values below 0.5 are bad. This parameter was not added to the error model because it is more useful to be used as a screening method by accepting or rejecting individual grids. An example of a good cell and a bad cell are shown in Figure 3.6.



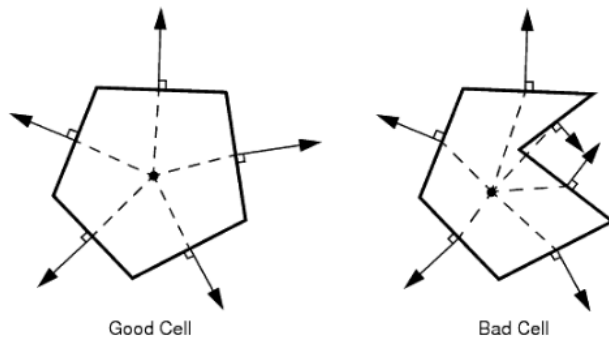


Figure 3.6: Face Validity Example. Reprinted from [14].

### Cell Quality

The cell quality is a hybrid Gauss and least-squares method for cell gradients. When the cell have large gradients between the cell centroids of nearby cell centroids, the cell is considered to have a poor cell quality. Values between  $1.0e^{-5}$  and 1.0 are considered good, while values below  $1.0e^{-5}$  are bad. This parameter was not added to the error model because it is more useful to be used as a screening method for accepting or rejecting individual grids. An example of a good cell and a bad cell are shown in Figure 3.7

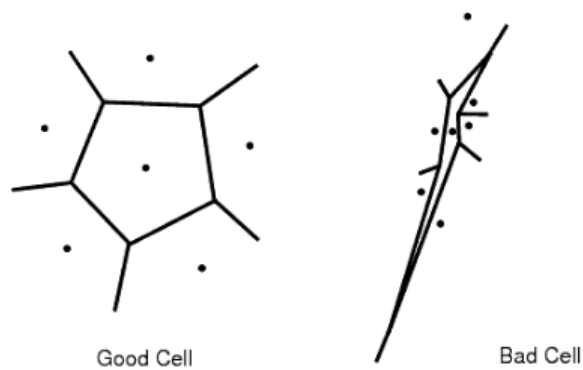


Figure 3.7: Cell Quality Example. Reprinted from [14].

## Volume Change

Volume change is a metric that quantifies the volume change between the cell of interest and the largest neighbor. Large changes in the volume between cells can cause inaccuracies. Volume changes between 0.01 and 1.0 are acceptable, but volume changes below 0.01 are considered bad cells. An example of a good cell and a bad cell are shown in Figure 3.8.

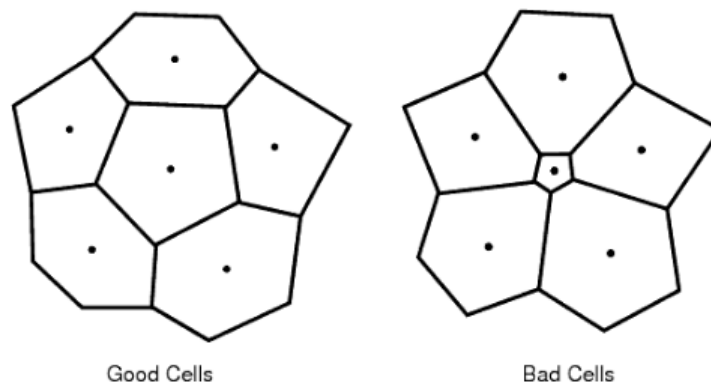


Figure 3.8: Volume Change Example. Reprinted from [14].

## Cell Skewness Angle

The cell skewness angle is the angle between the face normal and the line connecting the two centroids. For a structured hexahedral, this is typically not a problem, but should be included for future studies. Skewness angles between  $0^\circ$  and  $85^\circ$  are considered acceptable and angles above  $85^\circ$  are considered bad. This parameter was not added to the error model because it is more useful to be used as a screening method by accepting or rejecting individual grids. Figure 3.9 shows how the cell skewness angle is calculated.

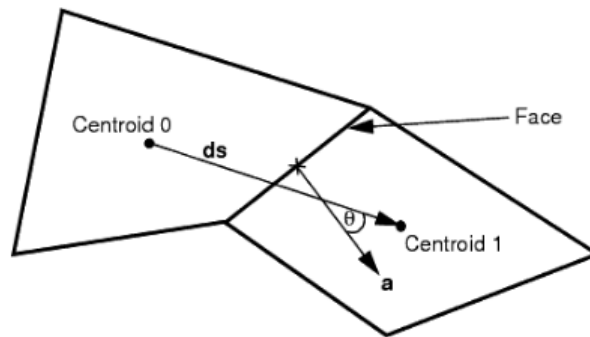


Figure 3.9: Cell Skewness Example. Reprinted from [14].

### Chevron Quality Indicator

Chevron quality indicators identify slender cells that the line connecting the cell centroid does not pass through the connecting face. Values are either one or zero, where Chevron cells are marked as one and all other cells are zero. This metric was not added to the error model because it is more useful to be used as a screening method by accepting or rejecting individual grids. An example of a good cell and a bad cell are shown in Figure 3.10.

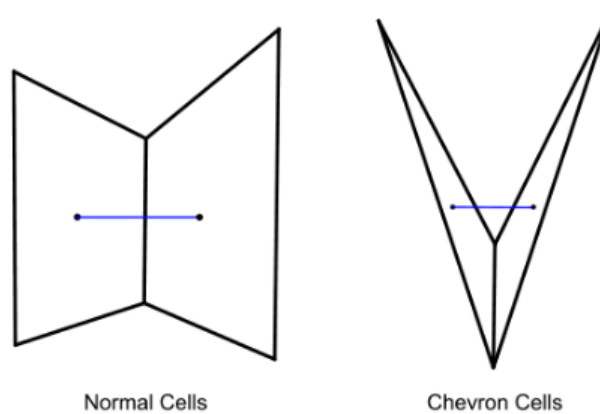


Figure 3.10: Chevron Cell Example. Reprinted from [14].

### Least Squares Quality

The least squares quality is an indication of the quality of the cell by using the least squares of the location of a cell centroid to the cell centroid of its face-neighbors. This metric was not added to the error model because it is more useful to be used as a screening method by accepting or rejecting individual grids. An example of the least squares quality calculation is shown in Figure 3.11

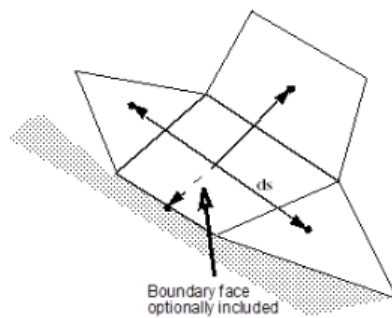


Figure 3.11: Least Squares Quality Example. Reprinted from [14].

### Cell Warpage Quality

Cell warpage quality is to identify thin or warped cells in the grid, which is shown in Figure 3.12. Cell warpage is between 0.0 and 1.0 where values between 0.15 and 1.0 are considered acceptable and values below 0.15 are considered bad. This metric was not added to the error model because it is more useful to be used as a screening method by accepting or rejecting individual grids.

### Summary of Error Terms from Grid Quality

Based on the grid metric definitions above, some grid metrics are similar to the MEA terms, while others are not. The grid metric that was similar to the MEA terms (volume change) was used to describe the grid spacing ratio term. The rest of the grid metrics were used to eliminate poor grids. The volume change grid metric term was included in a

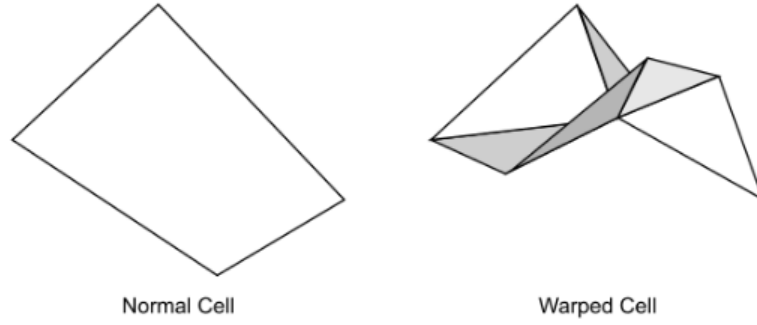


Figure 3.12: Cell Warpage Example. Reprinted from [14].

LASSO analysis and shown in Eq. 3.42

$$Error = c_1 \left( \frac{1}{G_1} - 1 \right) + c_2 h^p + c_3 h^p \left( \frac{\frac{1}{G_1} - 1}{\frac{1}{G_1} + 1} \right) + c_4 h^p \left( \frac{(G_1 + 1)^2}{G_1} \right) + c_5 h^p G_1 + c_6 h^p \frac{1}{G_1} \quad (3.42)$$

where  $G_1$  is the normalized volume change.

### 3.3 LASSO Analysis

A LASSO analysis was performed with some of the parameters listed in the previous section in addition to two different ways to calculate cell sizes. Four different powers of the terms that estimate the cell size were implemented, which is shown in Eq. 3.43. Since non-linear terms cannot be implemented into LASSO, having different powers of the terms aided in determining a rough estimate of the order of accuracy, which helped inform the range of the order of accuracy for later studies.

$$\begin{aligned}
Error = & \beta_1 \left( \frac{1}{G_1} - 1 \right) + \beta_2 h_{CSSM} + \beta_3 h_{CSSM}^{1.5} + \beta_4 h_{CSSM}^2 + \beta_5 h_{CSSM}^3 \\
& + \beta_6 h_{NoCM} + \beta_7 h_{NoCM}^{1.5} + \beta_8 h_{NoCM}^2 + \beta_9 h_{NoCM}^3 + \beta_{10} h_{NoCM} \left( \frac{\frac{1}{G_1} - 1}{\frac{1}{G_1} + 1} \right) \\
& + \beta_{11} h_{NoCM}^{1.5} \left( \frac{\frac{1}{G_1} - 1}{\frac{1}{G_1} + 1} \right) + \beta_{12} h_{NoCM}^2 \left( \frac{\frac{1}{G_1} - 1}{\frac{1}{G_1} + 1} \right) + \beta_{13} h_{NoCM}^3 \left( \frac{\frac{1}{G_1} - 1}{\frac{1}{G_1} + 1} \right) \\
& + \beta_{14} h_{NoCM} \left( \frac{(G_1 + 1)^2}{G_1} \right) + \beta_{15} h_{NoCM}^{1.5} \left( \frac{(G_1 + 1)^2}{G_1} \right) \\
& + \beta_{16} h_{NoCM}^2 \left( \frac{(G_1 + 1)^2}{G_1} \right) + \beta_{17} h_{NoCM}^3 \left( \frac{(G_1 + 1)^2}{G_1} \right) \\
& + \beta_{18} h_{NoCM} G_1 + \beta_{19} h_{NoCM}^{1.5} G_1 + \beta_{20} h_{NoCM}^2 G_1 + \beta_{21} h_{NoCM}^3 G_1 \\
& + \beta_{22} h_{NoCM} \frac{1}{G_1} + \beta_{23} h_{NoCM}^{1.5} \frac{1}{G_1} + \beta_{24} h_{NoCM}^2 \frac{1}{G_1} + \beta_{25} h_{NoCM}^3 \frac{1}{G_1}
\end{aligned} \tag{3.43}$$

where  $h_{CSSM}$  is the cell size estimated by the Cell Size Specification Method,  $h_{NoCM}$  is the cell size estimated by the Number of Cell Method,  $G_1$  is the volume change value, and the  $\beta_i$  coefficients are represented in a matrix as beta. The beta coefficients were used as importance metrics that reduced the number of terms that were implemented in the error model. This reduces the computation power required for coefficient optimization. The LASSO function in MATLAB was used to calculate the beta coefficients including removing redundant terms by using cross-validated fits. The coefficients were post-processed and presented as a weighted percentage of the total coefficient. This calculation is shown in Eq. 3.44

$$\beta_{weight} = \frac{\beta_{m,n}}{\sum_1^N \beta_{m,n}} \tag{3.44}$$

### 3.3.1 Data Normalization

Normalization of the variables before the LASSO analysis started simplified the problem. Since the cell size used was so small, fitting the data was difficult. To avoid this

problem, the variables were normalized between zero and one. This allows LASSO to fit the data in a range that reduced the chance of an ill-conditioned problem.

### 3.3.2 Expert Opinion Check

Both the data and the results of LASSO were reviewed for outlying data or results that did not make sense for the problem at hand. For example, 166 coarse grids were removed because they had an abnormally large number of iterations to reach steady state, which is shown in Fig. 3.13. The number of iterations required to run simulations with a normalized base size of 0.6 or larger were considered to be badly built grids that wouldn't produce useful results.

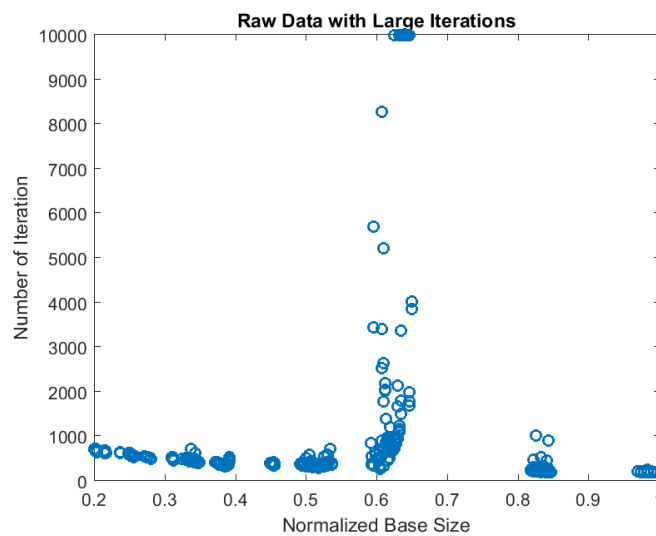


Figure 3.13: Raw Data with Large Iterations

These grids did not produce quality QoI or computationally efficacy data, so they were removed from the data set. Allowing an expert to review the data before the LASSO analysis or the results after the LASSO analysis reduces the chances of abnormal data affecting the results.

### 3.3.3 Homogeneity Check

The homogeneity check was performed using the cross-validation feature of LASSO in MATLAB [12]. Out of 200 remaining data points, 190 random data points were as LASSO model training while the other 10 were used as model testing. This makes sure that the data was fairly similar to each other. Heterogeneous data would have produced an error model that was over constrained and not useful for predictive capabilities. The results in the next section include the cross-validation feature.

### 3.3.4 LASSO Results

Table 3.3 shows the beta coefficients that were calculated as well as terms that pass the cross-validated fit reduction. It is important to note that switching the order of the terms influences the results, but the magnitudes of the coefficients were approximately the same. Therefore, the positions of the terms were switched randomly for 100 iterations and the median coefficients were used.

Table 3.3: LASSO Coefficients

Terms	Beta Coefficient	Percent Weight	Implemented in the Error Model?
$\left[\left(\frac{V}{N_c}\right)^{\frac{1}{3}}\right]^1$	0.533	52.76%	Yes
$\left[\left(\frac{V}{N_c}\right)^{\frac{1}{3}}\right]^{1.5}$	0.422	41.76%	Yes
$\left[\left(\frac{V}{N_c}\right)^{\frac{1}{3}}\right]^1 \frac{\left(\frac{1}{G_1}-1\right)}{\left(\frac{1}{G_1}+1\right)}$	-0.0554	5.48%	Yes

Based on the weight of error values in Table 3.3, the most influential term was the average cell size to the power of 1 followed by the average cell size to the power of 1.5. In addition, the average cell size multiplied by a grid ratio term had a non-zero beta value.



The resulting error model is shown in Eq. 3.45. Using the calculated  $\beta$  coefficients, the error data had an L2 norm of 0.0041 (normalized by the number of data points) and the average error was 0.038. The actual error and predicted error can be seen in Fig. 3.14. Eq. 3.45 was the error model used for further coefficient calculations, which will be described in the next section.

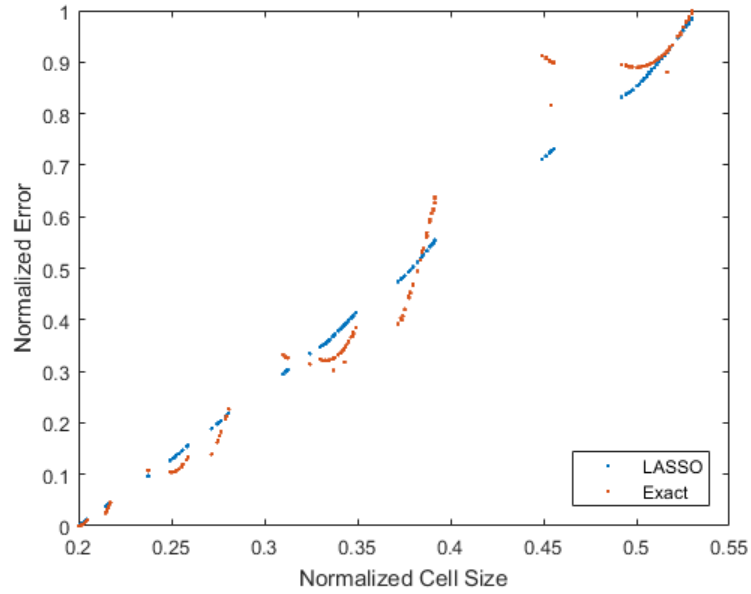


Figure 3.14: Error Prediction Using LASSO Coefficients

$$Error Model = c_1 \left[ \left( \frac{V}{N_c} \right)^{\frac{1}{3}} \right]^{p_{obs}} + c_2 \left[ \left( \frac{V}{N_c} \right)^{\frac{1}{3}} \right]^{p_{obs}} \left( \frac{\frac{1}{G_1} - 1}{\frac{1}{G_1} + 1} \right) \quad (3.45)$$

### 3.4 Coefficient Optimization

The coefficients in Eq. 3.46 were minimized using a hybrid optimization method in Dakota. The hybrid method allows for multiple optimization routines. The first routine was a genetic algorithm, which is a global sampling optimization method. The second routine was a mesh adaptive search, which is a local sampling method, and the third routine was a non-linear Fletcher-Reeves conjugate gradient optimizer, which is a local gradient

method [16]. The combination of these optimization methods were able to search the parameter space adequately in a short amount of time.

$$\begin{aligned}
 \arg \min_k \left| \Delta Error Model_{k,k+1} = c_1 \left[ \left( \frac{V}{N_{c_k}} \right)^{\frac{1}{3}} \right]^{p_{obs}} - c_1 \left[ \left( \frac{V}{N_{c_{k+1}}} \right)^{\frac{1}{3}} \right]^{p_{obs}} \right. \\
 \left. + c_2 \left[ \left( \frac{V}{N_{c_k}} \right)^{\frac{1}{3}} \right]^{p_{obs}} \left( \frac{\frac{1}{G_{1_k}} - 1}{\frac{1}{G_{1_k}} + 1} \right) - c_2 \left[ \left( \frac{V}{N_{c_{k+1}}} \right)^{\frac{1}{3}} \right]^{p_{obs}} \left( \frac{\frac{1}{G_{1_{k+1}}} - 1}{\frac{1}{G_{1_{k+1}}} + 1} \right) \right|_{\alpha}
 \end{aligned} \quad (3.46)$$

### 3.4.1 Coefficient Optimization Results

Using the hybrid optimization method in Dakota, multiple optimizations were performed on Eq. 3.46 with different vector differencing methods, which selected optimal  $c_1$ ,  $c_2$ , and  $p_{obs}$  values. Table 3.4 shows the optimal  $c_1$ ,  $c_2$ , and  $p_{obs}$  values for the data set produced using STAR-CCM+ as well as the difference between the estimated error and the actual error on a refined grid. The median case was calculated in addition to median absolute difference (MAD) in order to propagate the uncertainty due to the vector differencing fit.

Table 3.4: Optimization and Residual Data

Norms	$c_1$	$c_2$	$p_{obs}$	Error Difference
L1	0.527	$3.5 \cdot 10^{-2}$	1.60	$4.4 \cdot 10^{-3}$
L2	0.501	0.118	1.48	$6 \cdot 10^{-3}$
L4	0.490	0.194	1.37	$8 \cdot 10^{-3}$
L8	0.487	0.195	1.18	$1.4 \cdot 10^{-2}$
L inf	0.490	0.173	1.07	$2.0 \cdot 10^{-2}$
Median	$0.490 \pm 9.3 \cdot 10^{-3}$	$0.173 \pm 6.67 \cdot 10^{-2}$	$1.37 \pm 0.563$	$8 \cdot 10^{-3}$

To determine if the model was suitable for predicting the exact discretization error, the error model was compared against the exact error which is shown in Figure 3.15. The L1 and L2 norms were close to the exact discretization data, but with a spread of predictions,

it is probable that the exact discretization data will be within the uncertainty bound of the error model.

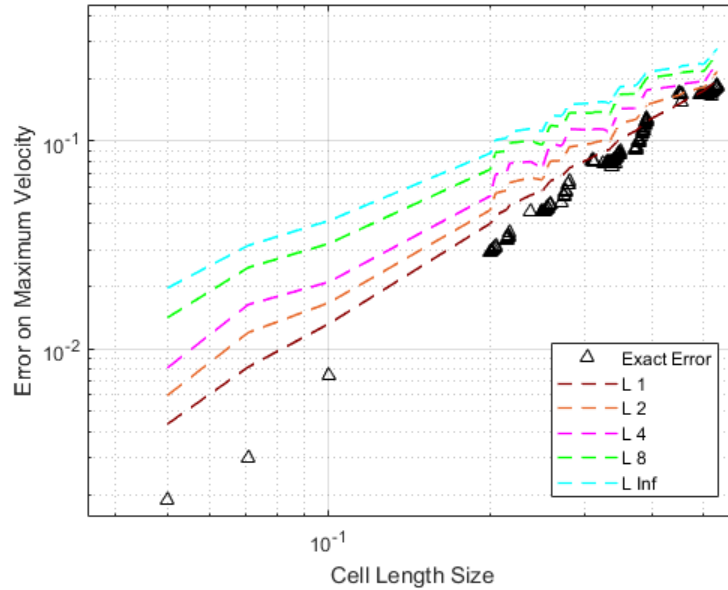


Figure 3.15: Error Model Results

### 3.5 Uncertainty Quantification

In addition to determining the best estimate error from the median coefficient value the vector differencing methods produced, 95% and 5% error bounds were calculated. The uncertainty was quantified by the spread of optimized coefficient values from different vector differencing methods. From the spread of values, the MAD was calculated, which produces a robust metric to describe the uncertainty bounds. RMR results using the optimized error model was compared against the exact discretization error and shown in Fig. 3.16.

The exact discretization error of a simulation with the grid one order of magnitude smaller than the training data was between the error model uncertainty bounds. This proves that improving the error model for RMR can predict discretization error.

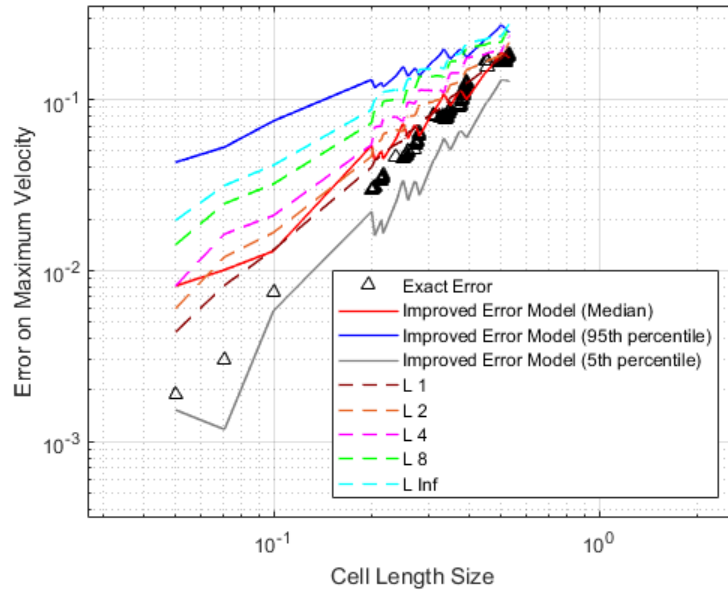


Figure 3.16: Uncertainty Estimate

### 3.6 Computational Effectiveness

The computational effectiveness was a way of quantifying the uncertainty due to a small data set. With a smaller data set, an outlier data point would have had a larger influence on the median predicted error and an even larger impact on the uncertainty bounds of the error prediction. This was shown by taking 200 data points, removing a randomly selected, but constant number of data points, and then using the remaining data points to make an error prediction. This was repeated 200 times to get a spread of predicted error values. As the number of samples increased, the less uncertainty there was in the predicted error value. This trend is shown in Fig. 3.17.

### 3.7 Comparison Between Solution Verification Methods

To determine if the improved error model provided better results than the standard  $ch^p$  error model, the two methods were compared to each other using the STAR-CCM+ data, which is shown in Figure 3.18.

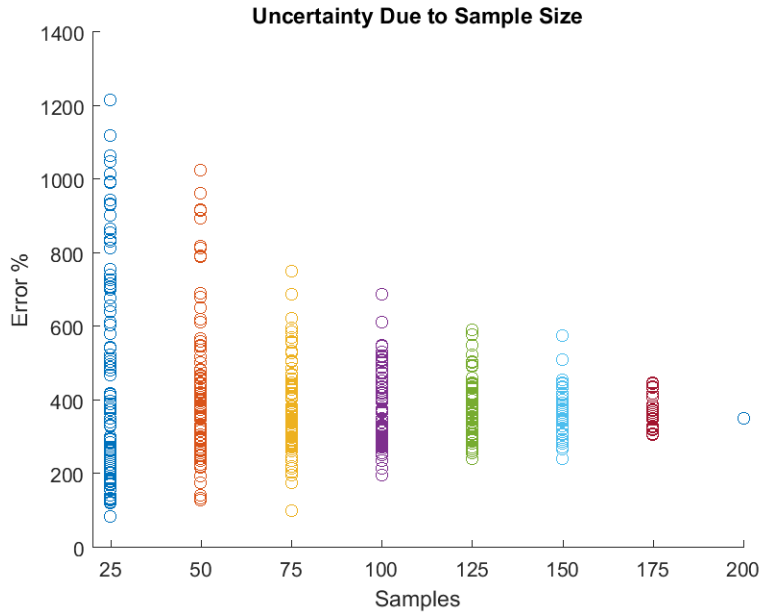


Figure 3.17: Uncertainty Due to Sample Size

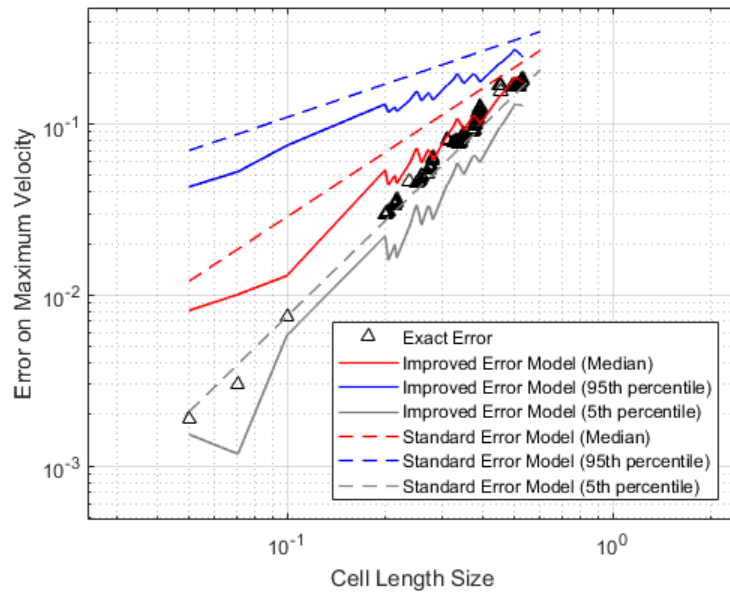


Figure 3.18: Uncertainty Estimate Comparison

Using the terms identified by LASSO produced an uncertainty bound that captured the error on a refined grid and was roughly twice as better than using the standard  $ch^{p_{obs}}$  error model. In addition, the standard error model was not able to capture the error on a refined grid within its uncertainty bound. This is because the error was both a function of average cell size and rate of refinement. When the error is a function of something other than average cell size, it degrades the order of accuracy, which was the case for this data set.

### **3.8 Teacher-Student Analysis**

A teacher-student analysis was performed as a more rigorous test of the framework used with STAR-CCM+ data. A major downfall of using a full CFD code to test out the methodology was that full control of the solution shape and the exact form of the error was not possible to obtain. The strength of using the teacher-student analysis was that the solution shape and the exact error equation were known and able to be manipulated through the use of MMMEAS. The next few sections use the MMMEAS to generate artificial data points, which were used to test LASSO's error predictive capabilities.

#### **3.8.1 One Dimensional MEA for Steady Continuity Equation**

A one-dimensional steady state continuity equation with constant velocity and non-constant cell sizing was used for the teacher-student study. The equation used to represent the solution was complex enough to provide a realistic number of terms in the exact error equation, but was simple enough to test and complete sensitivity analyses. In addition, the teacher equation included one more higher-order term than the student equation, so the start of the asymptotic region could be determined. Eq. 3.47 is the one-dimensional steady state continuity equation and Eq. 3.48 is the upwind discretization with non-constant cell size of Eq. 3.47.

$$u_0 \nabla \rho = u_0 \left( \frac{\partial \rho}{\partial x} \right) = 0 \quad (3.47)$$

$$u_0 \left( \frac{\rho_i - \rho_{i-1}}{\Delta x_i} \right) = 0 \quad (3.48)$$

where:

$$\Delta x_i = x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}$$

$$\Delta x_{i-1} = x_{i-\frac{1}{2}} - x_{i-\frac{3}{2}}$$

The density terms were expanded around  $\rho_i$  using a Taylor series, which is shown in Eq. 3.49.

$$\rho_{i-1} = \rho_i + \frac{\left( \frac{-\Delta x_i - \Delta x_{i-1}}{2} \right)}{1!} \frac{\partial \rho}{\partial x} \Big|_i + \frac{\left( \frac{-\Delta x_i - \Delta x_{i-1}}{2} \right)^2}{2!} \frac{\partial^2 \rho}{\partial x^2} \Big|_i + \frac{\left( \frac{-\Delta x_i - \Delta x_{i-1}}{2} \right)^3}{3!} \frac{\partial^3 \rho}{\partial x^3} \Big|_i + HOT \quad (3.49)$$

Eq. 3.49 was then substituted back into Eq. 3.48, which is shown in Eq. 3.50. Next, Eq. 3.50 was rearranged in terms of the original PDE plus error terms, which is shown in Eq. 3.51. Eq. 3.52 shows the exact local error equation, Eq. 3.53 simplifies and substitutes Eq. 3.16 into Eq. 3.52, and Eq. 3.54 shows the exact global error equation using an L2 norm of the local error.

$$u_0 \frac{1}{\Delta x_i} \left( \rho_i - \left( \rho_i + \frac{\left( \frac{-\Delta x_i - \Delta x_{i-1}}{2} \right)}{1!} \frac{\partial \rho}{\partial x} \Big|_i + \frac{\left( \frac{-\Delta x_i - \Delta x_{i-1}}{2} \right)^2}{2!} \frac{\partial^2 \rho}{\partial x^2} \Big|_i + \frac{\left( \frac{-\Delta x_i - \Delta x_{i-1}}{2} \right)^3}{3!} \frac{\partial^3 \rho}{\partial x^3} \Big|_i + HOT \right) \right) = 0 \quad (3.50)$$

$$\begin{aligned}
& u_0 \frac{\partial \rho}{\partial x} \Big|_i \\
& + u_0 \left( \frac{1}{2} \left( \frac{\Delta x_{i-1}}{\Delta x_i} - 1 \right) \frac{\partial \rho}{\partial x} \Big|_i - \frac{1}{8} \left( \Delta x_i + 2\Delta x_{i-1} + \Delta x_{i-1} \left( \frac{\Delta x_{i-1}}{\Delta x_i} \right) \right) \frac{\partial^2 \rho}{\partial x^2} \Big|_i \right. \\
& + \frac{1}{48} \left( \Delta x_i^2 + 3\Delta x_i \Delta x_{i-1} + 3\Delta x_{i-1}^2 + \Delta x_{i-1}^2 \left( \frac{\Delta x_{i-1}}{\Delta x_i} \right) \right) \frac{\partial^3 \rho}{\partial x^3} \Big|_i \\
& \left. + HOT \right) = 0
\end{aligned} \tag{3.51}$$

$$\begin{aligned}
Local\ Error_i &= \left( \frac{\Delta x_{i-1}}{\Delta x_i} - 1 \right) \left[ \frac{u_0}{2} \frac{\partial \rho}{\partial x} \Big|_i \right] - (\Delta x_i) \left[ \frac{u_0}{8} \frac{\partial^2 \rho}{\partial x^2} \Big|_i \right] \\
& - (\Delta x_{i-1}) \left[ \frac{u_0}{4} \frac{\partial^2 \rho}{\partial x^2} \Big|_i \right] - (\Delta x_{i-1}) \left( \frac{\Delta x_{i-1}}{\Delta x_i} \right) \left[ \frac{u_0}{8} \frac{\partial^2 \rho}{\partial x^2} \Big|_i \right] \\
& + (\Delta x_i^2) \left[ \frac{u_0}{48} \frac{\partial^3 \rho}{\partial x^3} \Big|_i \right] + (\Delta x_i \Delta x_{i-1}) \left[ \frac{u_0}{16} \frac{\partial^3 \rho}{\partial x^3} \Big|_i \right] \\
& + (\Delta x_{i-1}^2) \left[ \frac{u_0}{16} \frac{\partial^3 \rho}{\partial x^3} \Big|_i \right] + (\Delta x_{i-1}^2) \left( \frac{\Delta x_{i-1}}{\Delta x_i} \right) \left[ \frac{u_0}{48} \frac{\partial^3 \rho}{\partial x^3} \Big|_i \right] + HOT
\end{aligned} \tag{3.52}$$

$$\begin{aligned}
Local\ Error_i &= \underbrace{\left( \frac{1}{r} - 1 \right) \left[ \frac{u_0}{2} \frac{\partial \rho}{\partial x} \Big|_i \right]}_{Term\ 1} - \underbrace{(\Delta x_{i-1}) \left( \frac{(r+1)^2}{r} \right) \left[ \frac{u_0}{8} \frac{\partial^2 \rho}{\partial x^2} \Big|_i \right]}_{Term\ 2} \\
& + \underbrace{(\Delta x_{i-1}^2) \left( \frac{(r+1)^3}{r} \right) \left[ \frac{u_0}{48} \frac{\partial^3 \rho}{\partial x^3} \Big|_i \right]}_{Term\ 3} + HOT
\end{aligned} \tag{3.53}$$

$$Global\ Error = \frac{1}{\sqrt{N}} \sqrt{\sum_{i=1}^N (Local\ Error_i)^2} \tag{3.54}$$

### 3.8.2 Solution Equation and Local Error Equation (Teacher Equation)

Now that the error equation is known up to the third order term, the function of density was chosen. To identify how well LASSO works in the asymptotic range, a function with



a known start to the asymptotic region was chosen. A polynomial function was a viable function because of the ability to analytically calculate the derivatives. Two polynomials were used to evaluate the teacher equation. A linear equation (Eq. 3.55) was used to verify LASSO would zero out the first-order error term because the second derivative was zero (and therefore the MEA coefficient for the first-order error term would be zero). A second-order polynomial (Eq. 3.56) was used to verify LASSO would be able to identify the MEA coefficients for both the zeroth and first-order term. The second-order polynomial study was completed three times with three different coefficient sizes. The first study used a zeroth-order coefficient that was relatively large compared to the first-order coefficient. The second study used a zeroth-order coefficient that was relatively similar to the first-order coefficient. The third study used a zeroth-order coefficient that was relatively small compared to the first-order coefficient. Table 3.5 lists the coefficients used for all four studies.

$$\rho_{first}(x) = \rho_0 + \rho_1 x \quad (3.55)$$

$$\rho_{second}(x) = \rho_0 + \rho_1 x + \rho_2 x^2 \quad (3.56)$$

Now that two functional forms of density are known, the derivatives were computed for each polynomial to be substituted back into the Eq. 3.52. For the first-order polynomial, the first derivative density function is shown in Eq. 3.57, the second derivative density function is shown in Eq. 3.58, and the third derivative density function is shown in Eq. 3.59. For the second-order polynomial, the first derivative density function is shown in Eq. 3.60, the second derivative density function is shown in Eq. 3.61, and the third derivative density function is shown in Eq. 3.62.

$$\frac{\partial \rho_{first}(x)}{\partial x} = \rho_1 \quad (3.57)$$

$$\frac{\partial^2 \rho_{first}(x)}{\partial x^2} = 0 \quad (3.58)$$

Table 3.5: Teacher Equation Coefficients

Study Number	Functional Form of $\rho$	Relative Coefficient Size	$\rho_0$	$\rho_1$	$\rho_2$	Length of Domain
1	$\rho_{first}$	First Term Coefficient is non-zero	1000	20	0	$\pi$
2	$\rho_{second}$	First Term Coefficient » Second Term Coefficient	1000	20	0.1	$\pi$
3	$\rho_{second}$	First Term Coefficient $\approx$ Second Term Coefficient	1000	10	20	$\pi$
4	$\rho_{second}$	First Term Coefficient « Second Term Coefficient	1000	2.0	20	$\pi$

$$\frac{\partial^3 \rho_{first}(x)}{\partial x^3} = 0 \quad (3.59)$$

$$\frac{\partial \rho_{second}(x)}{\partial x} = \rho_1 + 2\rho_2 x \quad (3.60)$$

$$\frac{\partial^2 \rho_{second}(x)}{\partial x^2} = 2\rho_2 \quad (3.61)$$

$$\frac{\partial^3 \rho_{second}(x)}{\partial x^3} = 0 \quad (3.62)$$

The exact local and global error were used as a benchmark for LASSO to be able to calculate the error estimate based on the LASSO coefficients when the form of Eq. 3.52 was known.

### 3.8.3 Grid Refinement Methods

Using Eq. 2.17, the four different families of grids were built analytically. Each family was refined in a unique way, as described in the methodology section. This allowed for

studying how the error decreased as the grids were refined. Each family of grids decreased in error at different rates, which was expected. The student equation needs to have enough terms in the equation to describe each grid family's refinement trend to be able to predict the error.

This procedure represents possible refinement strategies within the STAR-CCM+'s grid generator. When a CFD user tries to complete a uniform refinement of a grid using STAR-CCM+'s grid generator, it is possible that the refinement is not a uniform refinement of the cell size, but a mix of refining the cell size and the grid refinement ratio. Identifying different refinement possibilities allowed for studying of LASSO's ability to determine which terms in the student equation are dominant and which terms are non-dominant.

#### **3.8.4 Grid Comparison and LASSO Analysis - Local Error**

Once the four family of grids were generated, the exact local error was quantified for all cells on all grid of a particular grid family. The cell errors were then compared to determine the  $\Delta_{local\ error}$  of two cells on a given grid. This was completed for the four different density equations. This represents the same process as the STAR-CCM+ analysis, which used STAR-CCM+ grids and simulations to calculate the  $\Delta QoI$  for two different simulations, but avoids the assumption that the local error model is a good global error model. Since each grid had a few hundred cells, the number of  $\Delta_{local\ error}$  data points was on the order of 1,000,000+. To reduce the computational demand for the LASSO algorithm, cells were randomly removed so that the number of  $\Delta_{local\ error}$  data points was on the order of 100,000+. This drastically reduced the computational cost, while still providing the LASSO algorithm a large amount of data. The  $\Delta_{local\ error}$  was then used with the input grid variables to complete a LASSO analysis using the student equation (Eq. 3.63). Eq. 3.63 is derived from Eq. 3.53, but does not include the second-order error term and replaces the derivatives and fractions with unknown coefficients. The terms between

brackets in Eq. 3.53 were approximated as constants because the value of the derivative are approximately constant as the grid is refined.

$$Error_{student} = c_0 \left( \frac{1}{r} - 1 \right) - c_1 (\Delta x_{i-1}) \left( \frac{(r+1)^2}{r} \right) \quad (3.63)$$

LASSO was able to determine the exact coefficient within 30%, which is sufficient to determine the importance of the term. Table 3.6 shows the LASSO coefficients and percent weight of each term for the four different grid families for Study 1, Table 3.7 for Study 2, Table 3.8 for Study 3, and Table 3.9 for Study 4. For Study 3 and 4, it was challenging to achieve the correct magnitude of the coefficients without introducing large variability in the coefficients due to the first derivative being a function of space. Therefore, the derivative was changed to a constant to avoid the variability causing problems with the LASSO predicted coefficients. In addition, the error from the last cell in the grid was removed because it effectively was used a ghost cell to properly calculate the previous cell's error.

Table 3.6: Teacher-Student Local LASSO Results (Study 1)

Grid Family Number	Terms	$\beta$ Coefficient	Percent Weight	Mean Analytic Coefficient
1	First Term Coefficient	9.71	100.00%	10.00
1	Second Term Coefficient	0.00	0.00%	0.00
2	First Term Coefficient	9.72	100.00%	10.00
2	Second Term Coefficient	0.00	0.00%	0.00
3	First Term Coefficient	9.71	100.00%	10.00
3	Second Term Coefficient	0.00	0.00%	0.00
4	First Term Coefficient	9.71	100.00%	10.00
4	Second Term Coefficient	0.00	0.00%	0.00

Table 3.7: Teacher-Student Local LASSO Results (Study 2)

Grid Family Number	Terms	$\beta$ Coefficient	Percent Weight	Mean Analytic Coefficient
1	First Term Coefficient	10.01	99.67%	10.04
1	Second Term Coefficient	0.033	0.33%	0.025
2	First Term Coefficient	9.79	99.73%	10.04
2	Second Term Coefficient	0.027	0.27%	0.025
3	First Term Coefficient	9.93	99.76%	10.04
3	Second Term Coefficient	0.024	0.24%	0.025
4	First Term Coefficient	9.99	99.69%	10.04
4	Second Term Coefficient	0.031	0.31%	0.025

Table 3.8: Teacher-Student Local LASSO Results (Study 3)

Grid Family Number	Terms	$\beta$ Coefficient	Percent Weight	Mean Analytic Coefficient
1	First Term Coefficient	5.85	49.99%	5.00
1	Second Term Coefficient	5.85	50.01%	5.00
2	First Term Coefficient	5.07	49.31%	5.00
2	Second Term Coefficient	5.22	50.69%	5.00
3	First Term Coefficient	5.17	49.23%	5.00
3	Second Term Coefficient	5.33	50.77%	5.00
4	First Term Coefficient	4.50	46.18%	5.00
4	Second Term Coefficient	5.24	53.82%	5.00

Table 3.9: Teacher-Student Local LASSO Results (Study 4)

Grid Family Number	Terms	$\beta$ Coefficient	Percent Weight	Mean Analytic Coefficient
1	First Term Coefficient	1.04	16.54%	1.00
1	Second Term Coefficient	5.27	83.46%	5.00
2	First Term Coefficient	0.75	12.72%	1.00
2	Second Term Coefficient	5.12	87.28%	5.00
3	First Term Coefficient	0.71	11.80%	1.00
3	Second Term Coefficient	5.30	88.20%	5.00
4	First Term Coefficient	1.22	17.52%	1.00
4	Second Term Coefficient	5.73	82.48%	5.00



### 3.8.5 Grid Comparison and LASSO Analysis - Global Error

Since the four families of grids have already been generated, the global error was quantified on all grids. Then the grids were compared to determine the  $\Delta_{global\ error}$  of the two grids. This represents the same process as the local error analysis, but uses the global error on two grids to calculate the  $\Delta_{global\ error}$  instead of the local error on two cells to calculate the  $\Delta_{local\ error}$ . The  $\Delta_{global\ error}$  was then used with the input grid variables to complete a LASSO analysis using the student equation (Eq. 3.63). Eq. 3.63 is derived from Eq. 3.53, but does not include the second-order error term. The terms between brackets in Eq. 3.53 were approximated as constants because the value of the derivative are approximately constant as the grid is refined. This was able to determine which terms in the student equation were dominant in each of the families of grids. Table 3.10 shows the LASSO coefficients and percent weight of each term for the four different grid families.

The global error coefficients were within 3% of the local coefficients. This shows that relationship between local and global error can be determined. It also shows that a good local error model makes a good global error model.

### 3.8.6 Teacher-Student Data Comparison

Using the  $\beta$  coefficients from Table 3.6 through Table 3.9, local error models were built for each specific grid for a given study. Using the  $\beta$  coefficients from Table 3.10, global error models for each grid family were built. To determine how well LASSO performed, both types of error models were compared to the actual error, which are shown in Fig. 3.19 through Fig. 3.34 for local error and Fig. 3.35 through Fig. 3.38 for global error.

Table 3.10: Teacher-Student Global LASSO Results

Grid Family Number	Terms	$\beta$ Coefficient	Percent Weight	Mean Analytic Coefficient
1	First Term Coefficient	9.21	100.00%	10.00
1	Second Term Coefficient	0.00	0.00%	0.00
2	First Term Coefficient	9.84	100.00%	10.00
2	Second Term Coefficient	0.00	0.00%	0.00
3	First Term Coefficient	9.73	100.00%	10.00
3	Second Term Coefficient	0.00	0.00%	0.00
4	First Term Coefficient	9.73	100.00%	10.00
4	Second Term Coefficient	0.00	0.00%	0.00

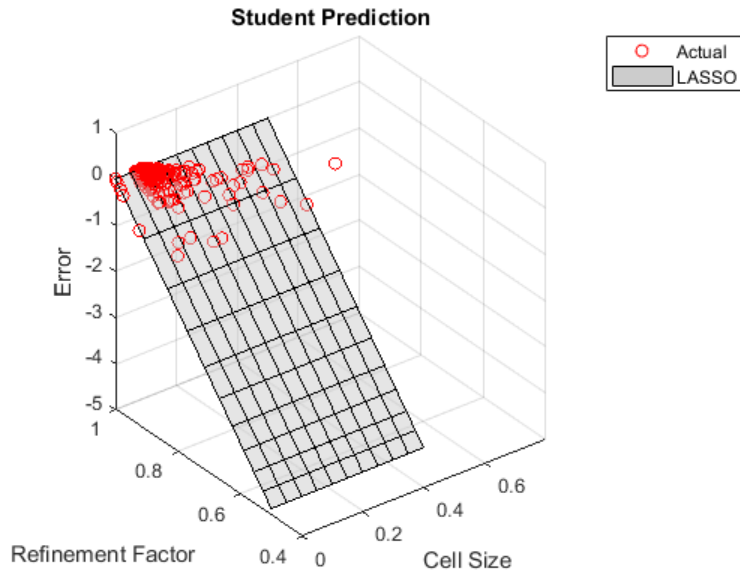


Figure 3.19: Teacher-Student Study 1 Grid 1 Local Error Comparison Plot

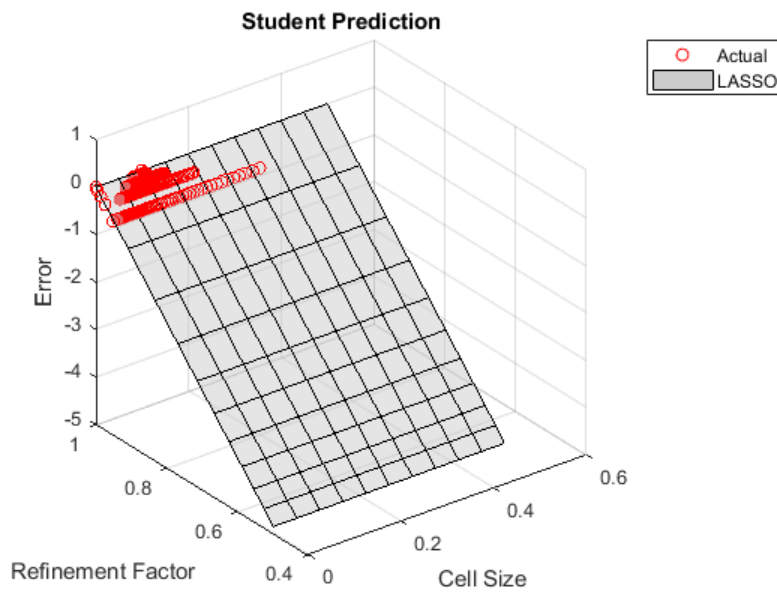


Figure 3.20: Teacher-Student Study 1 Grid 2 Local Error Comparison Plot

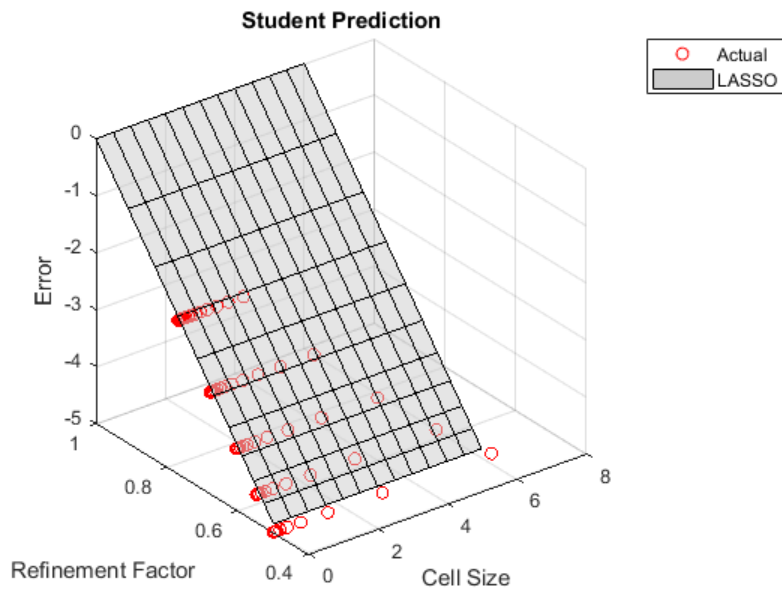


Figure 3.21: Teacher-Student Study 1 Grid 3 Local Error Comparison Plot

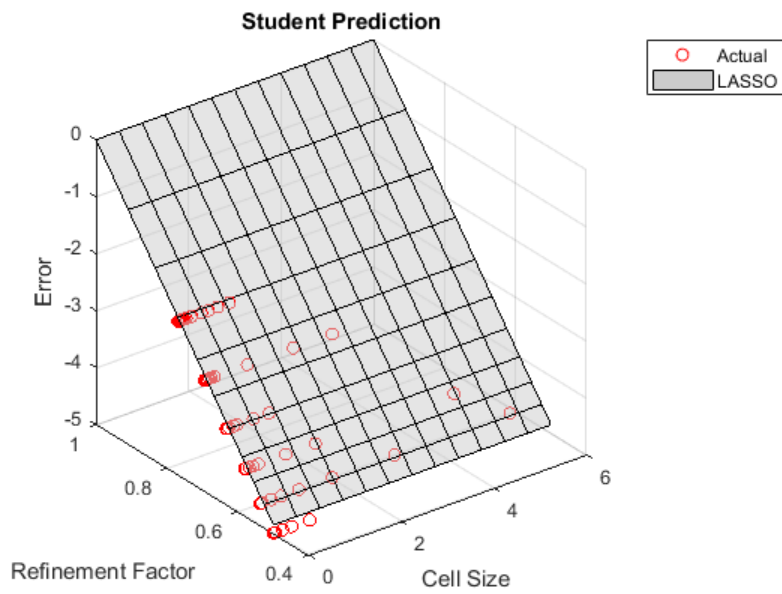


Figure 3.22: Teacher-Student Study 1 Grid 4 Local Error Comparison Plot

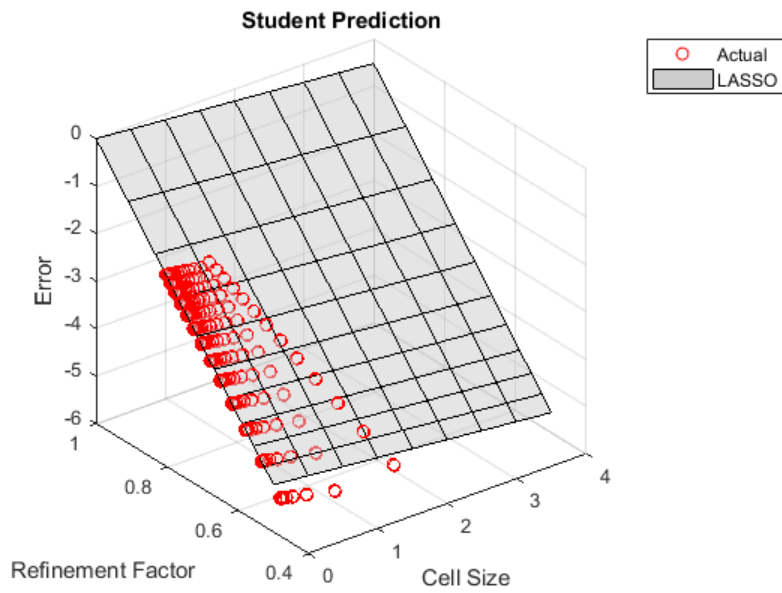


Figure 3.23: Teacher-Student Study 2 Grid 1 Local Error Comparison Plot

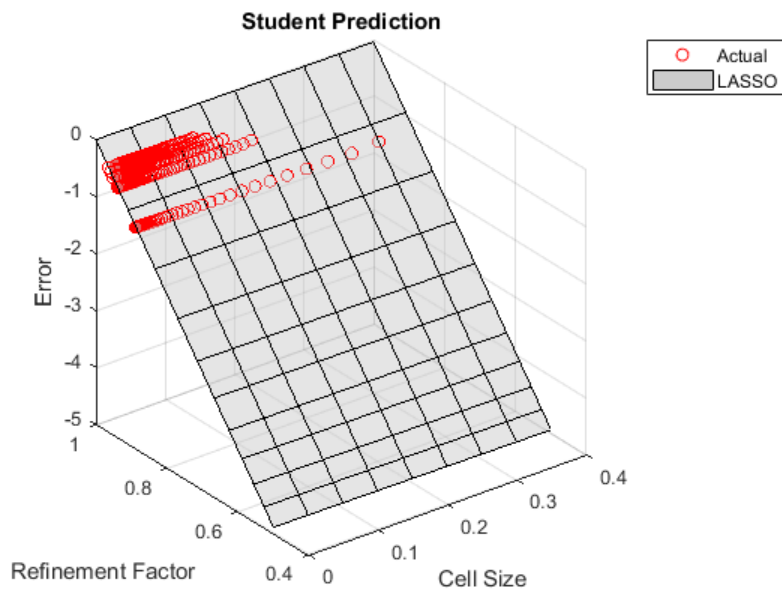


Figure 3.24: Teacher-Student Study 2 Grid 2 Local Error Comparison Plot

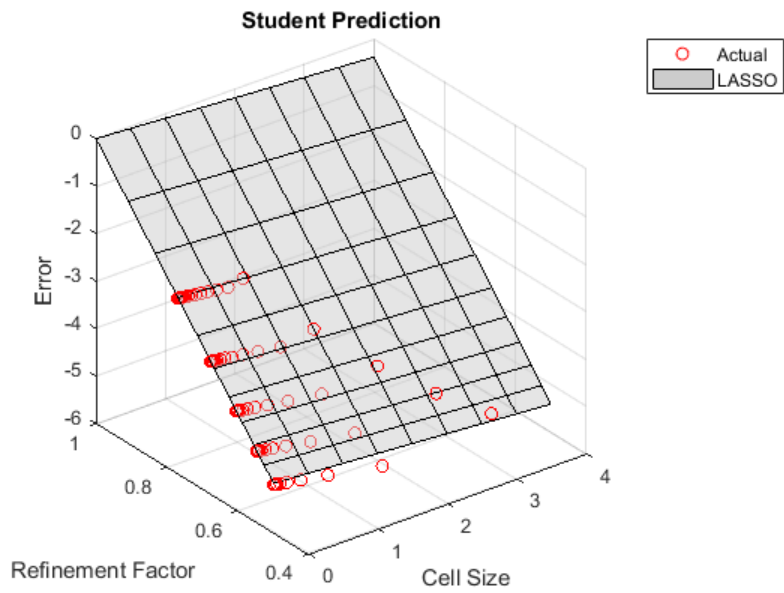


Figure 3.25: Teacher-Student Study 2 Grid 3 Local Error Comparison Plot

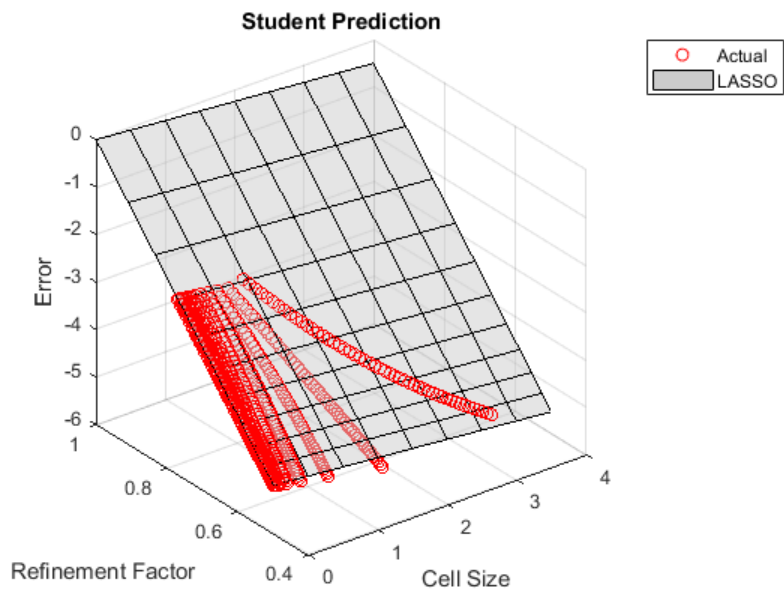


Figure 3.26: Teacher-Student Study 2 Grid 4 Local Error Comparison Plot

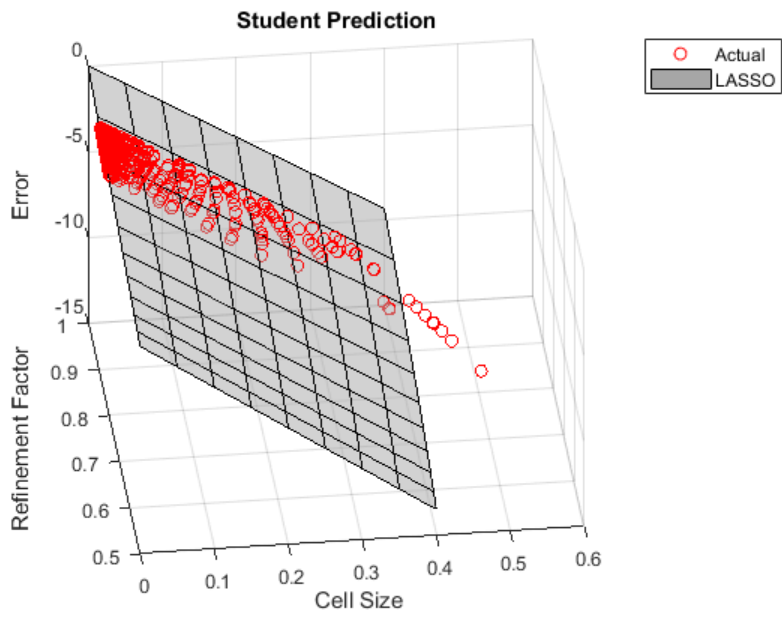


Figure 3.27: Teacher-Student Study 3 Grid 1 Local Error Comparison Plot

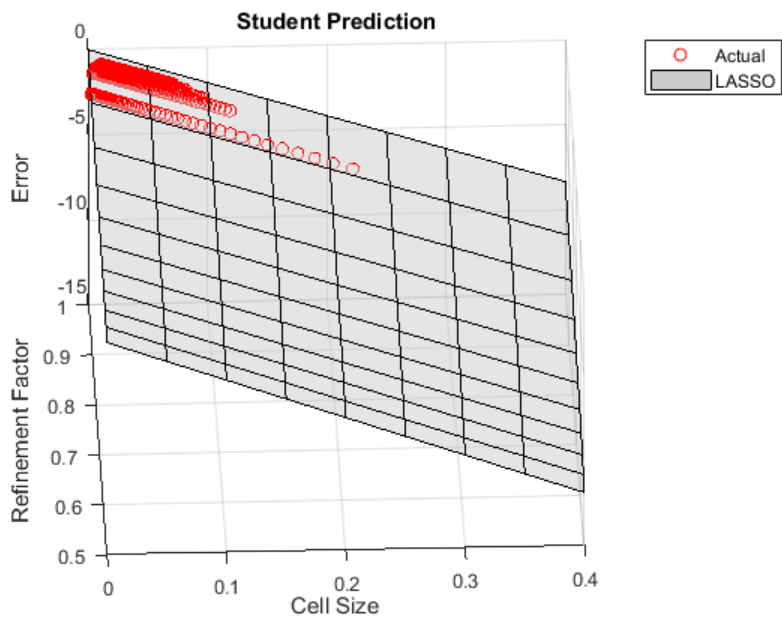


Figure 3.28: Teacher-Student Study 3 Grid 2 Local Error Comparison Plot

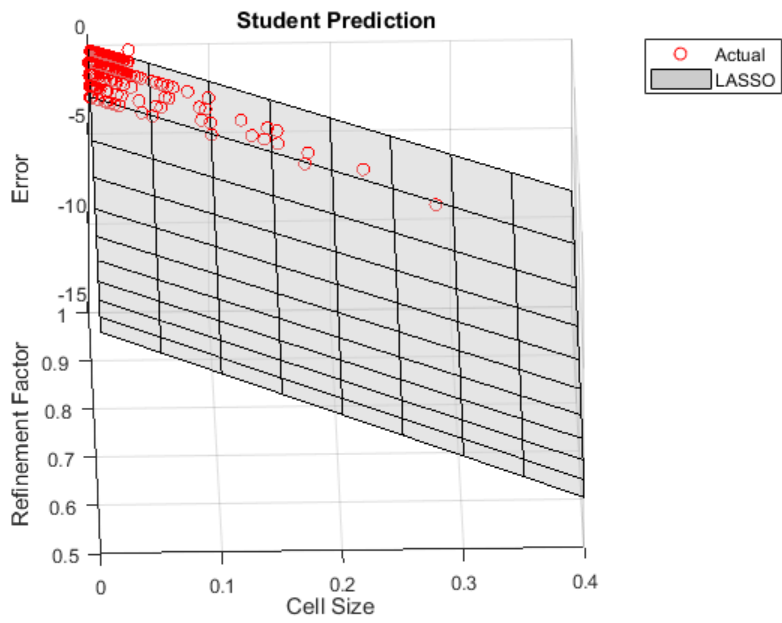


Figure 3.29: Teacher-Student Study 3 Grid 3 Local Error Comparison Plot

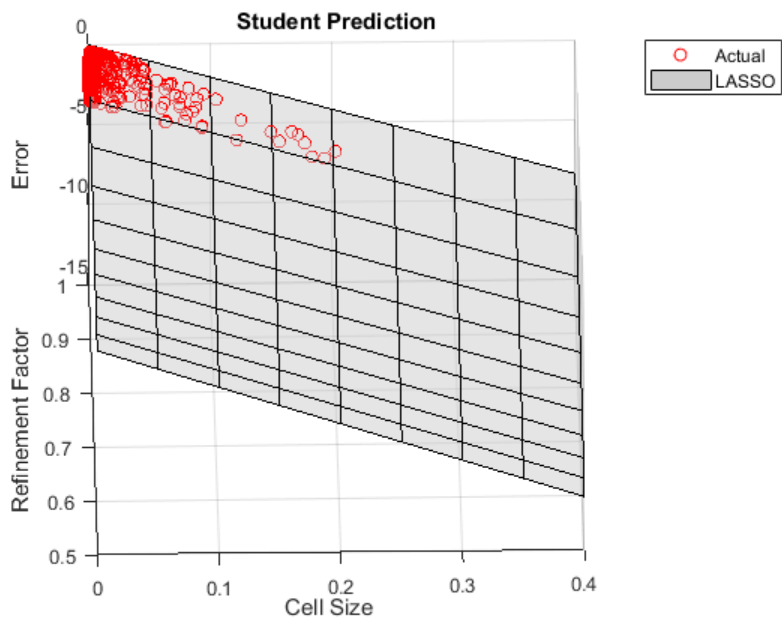


Figure 3.30: Teacher-Student Study 3 Grid 4 Local Error Comparison Plot



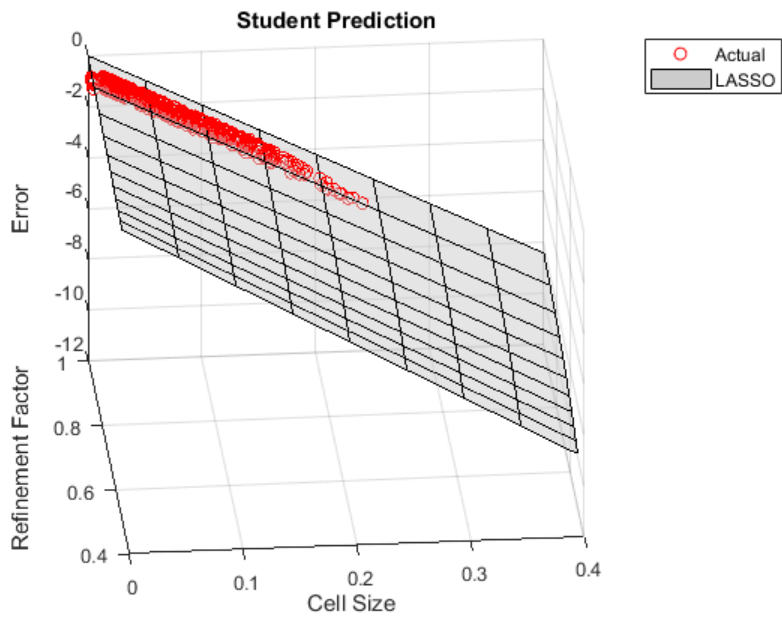


Figure 3.31: Teacher-Student Study 4 Grid 1 Local Error Comparison Plot

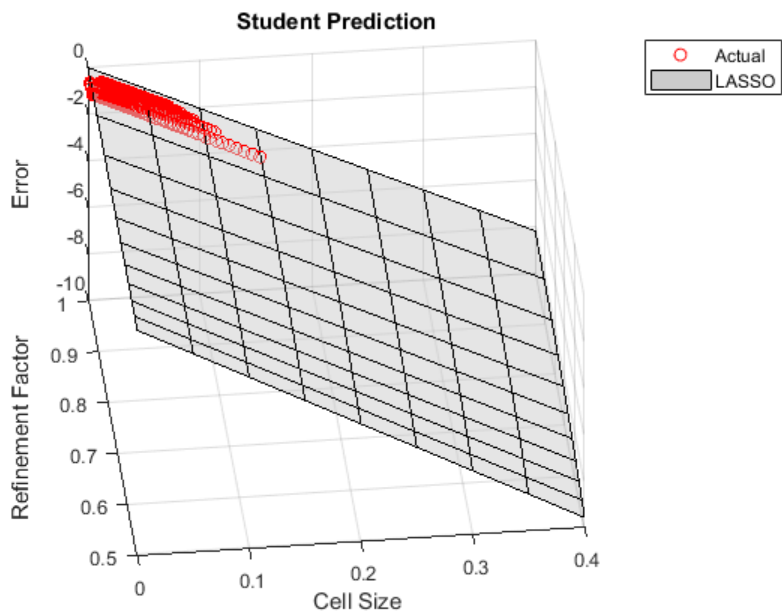


Figure 3.32: Teacher-Student Study 4 Grid 2 Local Error Comparison Plot

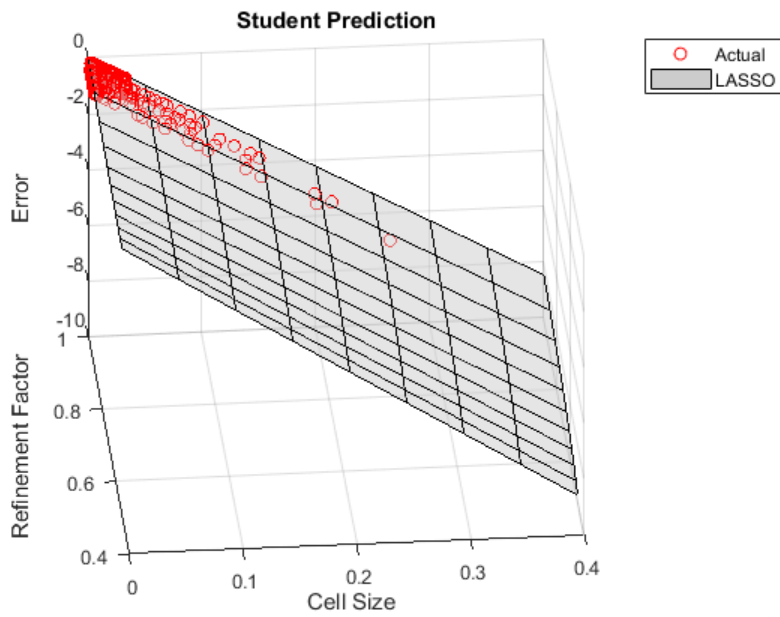


Figure 3.33: Teacher-Student Study 4 Grid 3 Local Error Comparison Plot

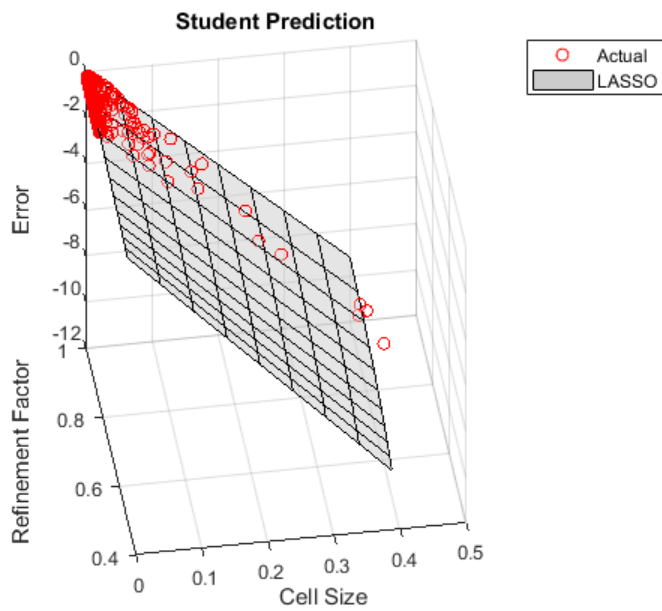


Figure 3.34: Teacher-Student Study 4 Grid 4 Local Error Comparison Plot

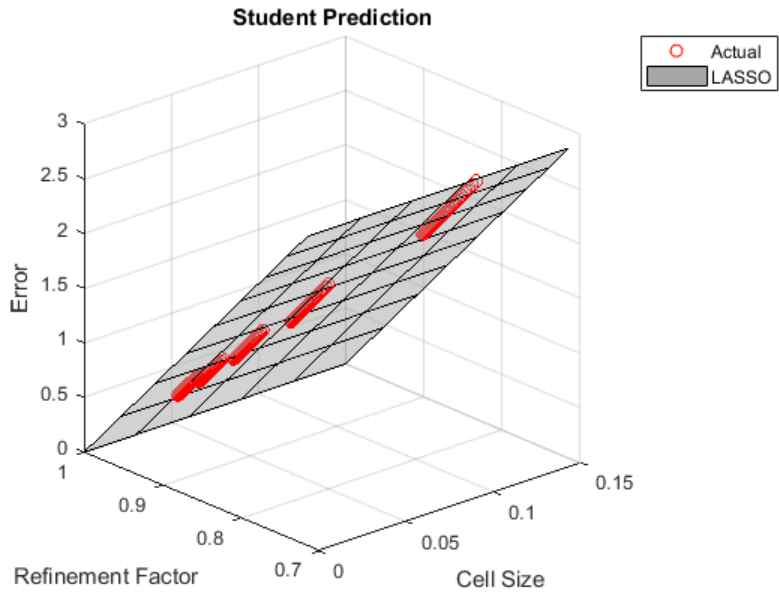


Figure 3.35: Teacher-Student Study 1 Grid 1 Global Error Comparison Plot

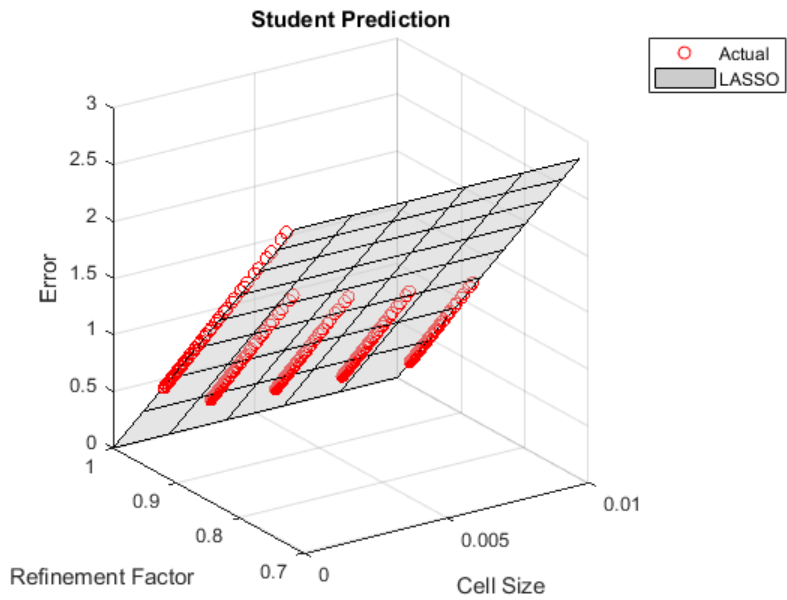


Figure 3.36: Teacher-Student Study 1 Grid 2 Global Error Comparison Plot

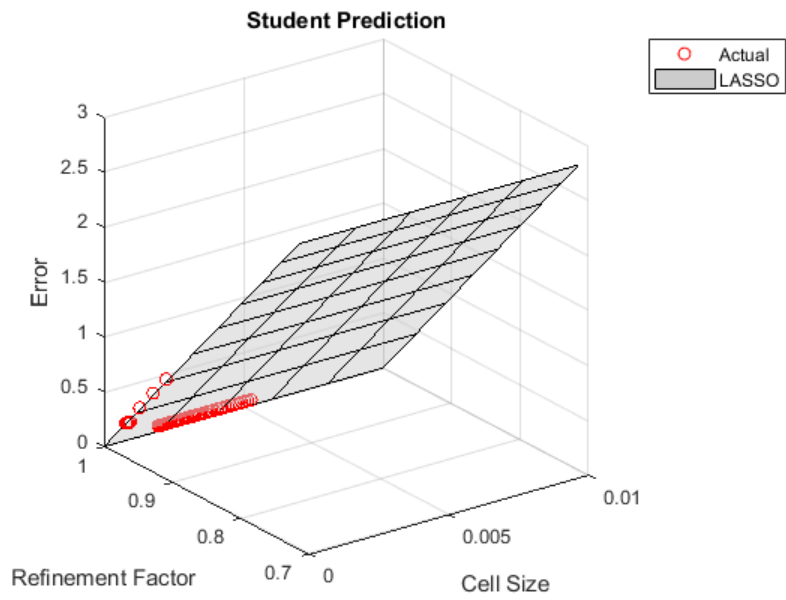


Figure 3.37: Teacher-Student Study 1 Grid 3 Global Error Comparison Plot

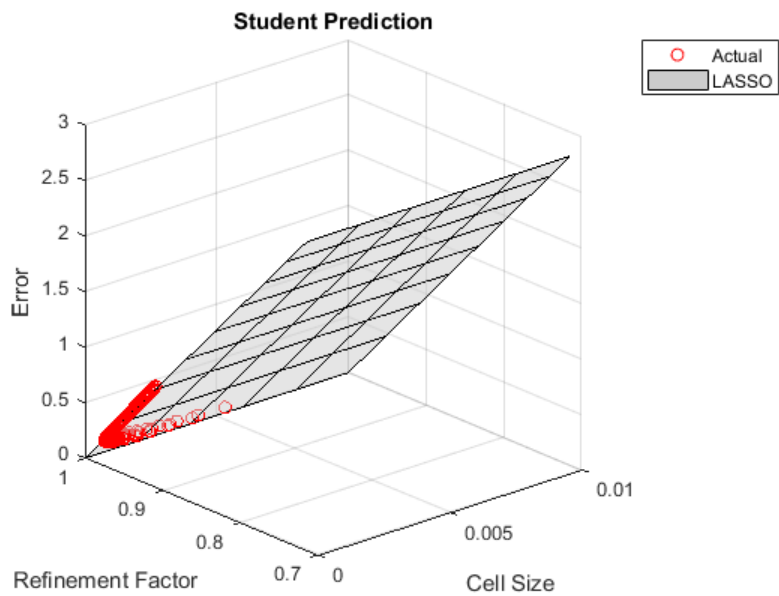


Figure 3.38: Teacher-Student Study 1 Grid 4 Global Error Comparison Plot

The local and global error models reproduced the exact error fairly well, which sufficiently proves that LASSO has the ability to determine the dominant terms in the Teacher equation for the four types of grid refinement.

## 4. CONCLUSION

Even after a few decades of work, solution verification methods are still evolving. Currently, RMR solution verification method shows promise to be able to predict discretization error with simulations close to and outside the asymptotic range. With the use of more generic error models than the historical power law model, the error estimate is closer to the exact error with smaller uncertainty bounds and approximately the same or less computational power.

### 4.1 STAR-CCM+ Analysis

While using STAR-CCM+ data is more difficult and time consuming to generate, the framework was able to automatically generate a grid, run the simulation, and extract QoI data from the simulation. The MEA was difficult to generate, so symbolic manipulation software was used to confirm the results. The LASSO analysis was able to identify the most dominant terms and reconstruct the error fairly well. The coefficient optimization and uncertainty quantification analyses were able to predict the discretization error on a grid that was generated approximately one order of magnitude more refined than the calibration data. RMR with the improved error model was able to correctly estimate error on a refined grid, while RMR with the standard error model was not. In addition, a qualitative relationship between computational power and uncertainty was developed.

### 4.2 Teacher-Student Analysis

The teacher-student analysis was able to test the same procedure as the STAR-CCM+ analysis, but the teacher-student analysis had more control variables. Since the solution shape, error function, and grid generation were all known, a more precise analysis was possible. The result was a more reliable justification test which shows that LASSO was able

to identify the dominant terms to one significant figure in an error model. The MMMEAS tests helped justify applying this problem in complex solution verification problems when the exact solution is not known.

### **4.3 Future Work**

While this analysis tested the framework, many improvements could be made. Some uncertainties that were not quantified were the uncertainty due to sample size and uncertainty due to extrapolation. In addition to quantifying the new sources of uncertainty, quantifying the computational power required to reduce the uncertainty would also aid in solution verification efforts. Another useful exercise would be to compare other widely used solution verification methods, such as GCI, least-square GCI, and the factor of safety method, to RMR methods on classic solution verification problems. This would aid in RMR's reputation as a valid solution verification technique for grids with a large amount of cells. This would involve completing a more complex MEA to represent second-order terms. Additional ways to aid in robust verification's credibility would be to complete a teacher-student analysis on fully coupled equations and apply the analysis on multi-physics and multi-mesh problems. These types of analyses would help show strengths and weaknesses of the robust verification method.

## REFERENCES

- [1] P. J. Roache, “Perspective: a method for uniform reporting of grid refinement studies,” *Transactions-American Society of Mechanical Engineers Journal of Fluids Engineering*, vol. 116, pp. 405–405, 1994.
- [2] W. L. Oberkampf and C. J. Roy, *Verification and validation in scientific computing*. Cambridge University Press, 2010.
- [3] P. Knupp and K. Salari, *Verification of computer codes in computational science and engineering*. CRC Press, 2002.
- [4] P. J. Roache, “Code verification by the method of manufactured solutions,” *Transactions-American Society of Mechanical Engineers Journal of Fluids Engineering*, vol. 124, no. 1, pp. 4–10, 2002.
- [5] H. F. Stripling, M. L. Adams, R. G. McClarren, and B. K. Mallick, “The method of manufactured universes for validating uncertainty quantification methods,” *Reliability Engineering & System Safety*, vol. 96, no. 9, pp. 1242–1256, 2011.
- [6] L. F. Richardson, “The approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stresses in a masonry dam,” *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, vol. 210, pp. 307–357, 1911.
- [7] W. Rider, W. Witkowski, J. R. Kamm, and T. Wildey, “Robust verification analysis,” *Journal of Computational Physics*, vol. 307, pp. 146–163, 2016.
- [8] C. W. Hirt, “Heuristic stability theory for finite-difference equations,” *Journal of Computational Physics*, vol. 2, no. 4, pp. 339–355, 1968.



- [9] R. Warming and B. Hyett, “The modified equation approach to the stability and accuracy analysis of finite-difference methods,” *Journal of computational physics*, vol. 14, no. 2, pp. 159–179, 1974.
- [10] R. B. Bird, W. E. Stewart, and E. N. Lightfoot, *Transport phenomena*. John Wiley & Sons, 2007.
- [11] R. Tibshirani, “Regression shrinkage and selection via the lasso,” *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.
- [12] MATLAB, *version R2017a*. Natick, Massachusetts: The MathWorks Inc., 2017.
- [13] G. B. Thomas and R. L. Finney, *Calculus and analytic geometry*. Addison Wesley Publishing Company, 1979.
- [14] Siemens, *STAR-CCM+ Users’ Guide*. 12.02.010-r8 ed., 2017.
- [15] S. V. Patankar and D. B. Spalding, “A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows,” *International journal of heat and mass transfer*, vol. 15, no. 10, pp. 1787–1806, 1972.
- [16] B. Adams, W. Bohnhoff, K. Dalbey, J. Eddy, M. Ebeida, J. Frye, G. Gerarci, P. Hough, K. Hu, J. Jakeman, M. Khalil, K. Manupin, J. Monschke, E. Ridgway, A. Rushdi, L. Swiler, J. Stephens, D. Vigil, and T. Wildey, “Dakota, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis: version 6.6 user’s manual,” *Sandia National Laboratories, Tech. Rep. SAND2014-4253*, 2017.