Spring 1-1-2012

Error Analysis on Multi-Fidelity Design Spaces through Surrogate Modeling with an Emphasis on Scramjet Combustion

Kevin Daniel Basore
University of Colorado at Boulder, kevin.basore@gmail.com

Error Analysis on Multi-Fidelity Design Spaces through Surrogate Modeling with an Emphasis on Scramjet Combustion

by

Kevin Daniel Basore

B.S., University of Colorado, 2010

A thesis submitted to the

Faculty of the Graduate School of the

University of Colorado in partial fulfillment

of the requirements for the degree of

Master of Science

Department of Aerospace Engineering Sciences

2012

This thesis entitled:
Error Analysis on Multi-Fidelity Design Spaces through Surrogate Modeling with an Emphasis on Scramjet Combustion
written by Kevin Daniel Basore
has been approved for the Department of Aerospace Engineering Sciences

Dr. Ryan Starkey

Professor Kenneth Jansen

Professor Brian Argrow

Date

The final copy of this thesis has been examined by the signatories, and we find that both the content and the form meet acceptable presentation standards of scholarly work in the above mentioned discipline.

Basore, Kevin Daniel (M.S, Aerospace Engineering)

Error Analysis on Multi-Fidelity Design Spaces through Surrogate Modeling with an Emphasis on Scramjet Combustion

Thesis directed by Dr. Ryan Starkey

Hypersonic vehicle design is a challenging problem. The main obstacle that makes this work challenging is the non-linear interdependency between each of the vehicle's systems. Thus, when generating a vehicle design model for a hypersonic system, a large number of parameters must be considered. The addition of a single parameter into the design space leads to an exponential increase in the number of necessary single point solutions.

This thesis will examine hypersonic vehicle design in a unique way that decreases the computational power required to achieve a more accurate solution. This was done through the examination of error vectors between multi-fidelity design spaces using Kriging surrogate models and superimposing the computed difference onto a low fidelity solver. By adding this empirical correction onto the low fidelity data, the accuracy of the computed metrics was improved by an order of magnitude in some cases.

In proving this theory a simplified airfoil test case and a generic cavity scramjet, that included combustion chemistry, were investigated. Examination of both of these geometries required a sizable amount of computing resources. To meet this requirement a custom cloud controller that could dynamically load processor cores depending on individual computer workloads was developed. This computational cloud, as well as an automated case generation algorithm, were developed in order to process the hundreds of computational runs necessary.

Once processed, these design spaces were visually and numerically inspected to verify convergence within each of the solutions. Having a verified discrete sampling in each of the design space fidelity levels, Kriging surrogate models were constructed to produce a continuous spectrum on the error between them. This continuous approximation was then used as the empirical correction to the low fidelity solver in order to achieve the order of magnitude increase in accuracy.

Dedication

I would like to dedicate this work to my father, Dr. Paul Alan Basore, for inspiring me to finish my studies.

Acknowledgements

I would like to thank my adviser Dr. Ryan Starkey as well as the rest of my defense committee Dr. Ken Jansen and Dr. Brian Argrow. In addition I would like to thank the people who gave me technical advising on the content of my thesis including Tony Castrogiovanni, David Putnam and Andrew Kearl. Finally, I would like to thank my fellow employees within the Busemann Advanced Concepts Laboratory.

Computational resources were provided by NSF-MRI Grant CNS-0821794, MRI-Consortium: Acquisition of a Supercomputer by the Front Range Computing Consortium (FRCC), with additional support from the University of Colorado and NSF sponsorship of the National Center for Atmospheric Research.

Contents

Chapter

Tables

Table

Figures

Figure

Nomenclature

α	Angle of Attack
$\frac{F}{\dot{m}_0}$	Specific Thrust, ms
$\dot{m}_f$	Mass Flow Rate Fuel
γ	Ratio of Specific Heats
γ_{air}	Specific Heat Ratio Specifically for Standard Air
μ	Mach Wave Angle
ν	Prandtl-Meyer Function
ρ_∞	Density Infinity Condition
ρ_{air}	Inflow Air Density
θ	Wedge Half Angle
C_p	Coefficient of Pressure
f_{st}	Stoichiometric Fuel Ratio
g_0	Sea Level Gravity
h_{comb}	Height of the Combustor
I_{sp}	Specific Impulse

M	Mach Number
p	Static Pressure
P_0	Total Pressure
p_∞	Pressure Infinity Condition
R	Specific Gas Constant
T	Static Temperature
T_0	Total Temperature
T_∞	Temperature Infinity Condition
V_{air}	Inflow Velocity
AHWP	Advanced Hypersonic Weapon Program
ANN	Artificial Neural Networks
AoA	Angle of Attack
AR	Aspect Ratio
BACL	Busemann Advanced Concepts Laboratory
CAD	Computer Aided Design
CAD	Computer Aided Design
CFD	Computational Fluid Dynamics
CGT	Chimera Grid Tools
DA	Diamond Analytical
DARPA	Defense Advanced Research Projects Agency

ELV	Expendable Launch Vehicle
F	Thrust, $\frac{N}{m}$
FFT	Fast Fourier Transform
GB	Gigabyte
GNU	GNU's Not Unix
GRIDED	Grid Editor for Structured Entities
GUI	Graphical User Interface
HYPGEN	Hyperbolic Field Grid Generator
ICBM	Intercontinental Ballistic Missile
IDEA	Integrated Design and Engineering Analysis
ITAR	International Traffic in Arms Regulations
LAN	Local Area Network
LEO	Low Earth Orbit
LES	Large Eddy Simulation
MARS	Multivariate Adaptive Regression Splines
MLE	Maximum Likelihood Estimates
N-S	Navier Stokes
NaN	Not a Number
NASA	National Aeronautics and Space Administration
NASP	National Aerospace Plane

NRMSE	Normalized Root Mean Square Error
OVERFLOW	OVERset grid FLOW solver
PID	Process Identification
POD	Proper Orthogonal Decomposition
PRESS	Predicted Error Sum of Squares
RAID	Redundant Array of Independent Disks
RANS	Reynold Averaged Navier-Stokes
RHS	Right Hand Side
RMS	Root Mean Square
ROM	Reduced Order Modeling
scp	secure copy
SLP	Stochastic Layered Perceptron
SR	Stretching Ratio
SRAP	Spline Redistribute And Project
ssh	Secure Shell
SURGRD	Overset Surface Grid Generator
TSTO	Two Stage To Orbit
U.S.	United States
USSR	Union of Soviet Socialist Republics
V	Velocity, $\frac{m}{s}$

VULCAN Viscous Upwind aLgorithm for Complex flow ANalysis

Chapter 1

Introduction

Hypersonic vehicle design is inherently non-linear and thus very difficult when looking for an optimal design point.[1] However, being close to the optimal design point is necessary to make the vehicle successful within its flight envelope.[2] This conundrum makes inventive solutions to achieving the optimal closed design necessary.

The methodology proposed in this work will examine the use of surrogate modeling, in particular Kriging surrogate models, to improve the accuracy of low fidelity design spaces. By increasing the accuracy that can be achieved for the same computational cost, the probability of being able bracket the optimal design solution in the initial parameter sweep grows.

To prove the method described within this work, two geometrical design configurations were examined. The first of these was a diamond airfoil or a double wedge as it is known within some communities and a generic cavity scramjet combustor. Although, this method is able to be applied to the entire vehicle, this level of analysis was not performed as the computational resources allocated would not allow it.

The first geometry, the diamond airfoil, was chosen mainly to verify the methodology of the thesis and also for verification purposes of the third party software used within the calculations. This configuration was chosen over others simplified geometries for two reasons. Firstly, there is an exact analytical solution to the diamond airfoil and secondly, there were available published works that were able to be used for comparison purposes.

The second geometry chosen was a simplified antisymmetric backwards facing cavity step

scramjet combustor. This geometry was chosen over other hypersonic waverider configurations because, in general, the combustor has more influence over the final design performance metrics. Also, like the diamond airfoil, there are multiple published works on this geometry which allowed for verification and validation of the initial results produced.

1.1 Methodology

The methodology implemented began with the diamond airfoil configuration due to its simplicity. The external flow around the airflow was meshed and then compared at three different parameter levels as will be explained within Chapter 3. From initial points taken in each of the design spaces, a discrete error space was created between the model levels. This discrete space was then turned into a continuous spectrum through the use of a Kriging surrogate model. From this continuous spectrum approximation across a multidimensional domain, an empirical approximation could be superimposed upon the low fidelity model to increase the accuracy of the metric results. Chapter 5 shows how this initial idea could be improved further in the future using orthonormal bases to remove the need to compute high fidelity data.

From the diamond airfoil, this methodology was extrapolated to the scramjet combustor. Following the same idea, discrete points were sampled in order to populate the continuous error spectrum between the fidelity levels. In order to achieve this extrapolation though, three main differences had to be accounted for. Firstly, the difference in the geometry and the associated problems in being able to produce a quality mesh from it. Secondly, out of the two chemical fidelity levels tested, there was no exact analytical expression to compute an absolute error from. Therefore, in order to verify the solution, results from outside sources were reproduced and used for comparison. Finally, the computational requirement per solution dictated that fewer solutions per design space could be produced.

Accounting for all these differences though, this methodology was successfully able to be extrapolated onto the scramjet geometry as is shown in Chapter 4.

Chapter 2

Background

2.1 Importance of Hypersonics

Hypersonic flight first began in February 1949 with a WAC-Corporal rocket attached to an acquired German V2 in the New Mexican White Sands Missile Range. From these humble beginnings in which the the first two stage rocket was built, the field of hypersonics slowly began to take root.[3] One of the more important steps in this development was the creation of hypersonic experimental facilities within the country.[3] Two of the more well known, and also, the first two facilities, was the 11 in tunnel at NASA Langley and the 10 in x 14 in tunnel at NASA Ames.[4] Both constructed in the late 1940's each of these tunnels was initially used for basic research and then for conducting experiments for later follow on projects.[3, 5] To display how much the 11 in tunnel, shown in Figure 2.1, was utilized, over 230 technical publications with an average of one every 5 weeks were produced from data acquired within the tunnel.[5]

This initial basic research not only allowed the reentry problem on ballistic missiles to be solved in the 1950's, but also allowed for necessary research to be conducted on many of the hypersonics programs to follow.[5]. The first of these programs was the X-15 experimental aircraft which flew from 1959 through 1968. The cancellation of this program was sealed after the crash and death of pilot Major Micheal J. Adams in that same year. This crash was by far not the first of the program with one of the many incidents shown within Figure 2.2, however, being the first fatality of the program, this was all that was necessary to terminate it. Over the entire 9 year history, only 1 hr 25 min 33 sec of hypersonic data was able to be collected.[5] This point highlights just

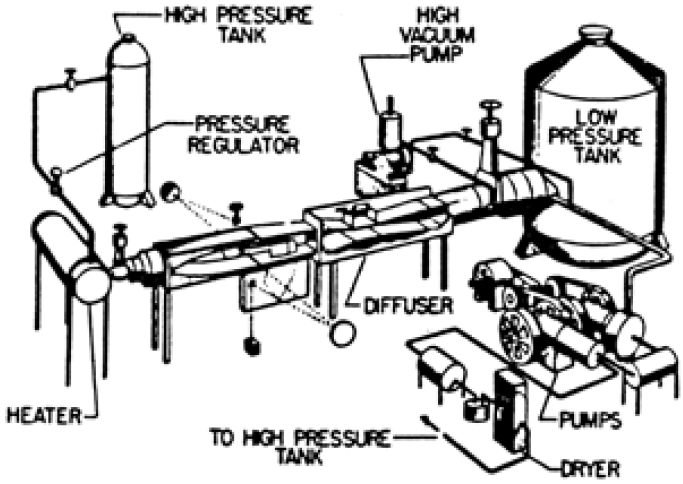


Figure 2.1: NASA Langley 11 inch Hypersonic Tunnel.[6]

how hard it is to collect hypersonic flight data when compared to something like civilian transport aircraft which can incur upwards of a few thousand flight hours a year.[7]

Despite this lack of data and fundamental knowledge in some key areas, large scale programs aimed at the promise of producing reusable hypersonic vehicles were initiated in the coming years. None were successful, with the most notable failure being the National Aerospace Plane(NASP) which was finally terminated in 1994 after 8 years because it failed to produce a working design.[3, 8] Even with these costly failures, the United States(U.S.) government as well as governments from around the world are still interested in the technology on the potential that it holds. This point can be proven through the recent commitment to new hypersonic projects including the X-43, X-51, DARPA Falcon and the U.S. Army's Advanced Hypersonic Weapon Program(AHWP).[9, 10, 11, 12, 13, 14]. For general reference, the flight envelope of some of these mentioned programs can be seen within Figure 2.3 which shows the speed and altitude that these programs were aiming for.

Figure 2.2: One of the X-15 Failures.[5]

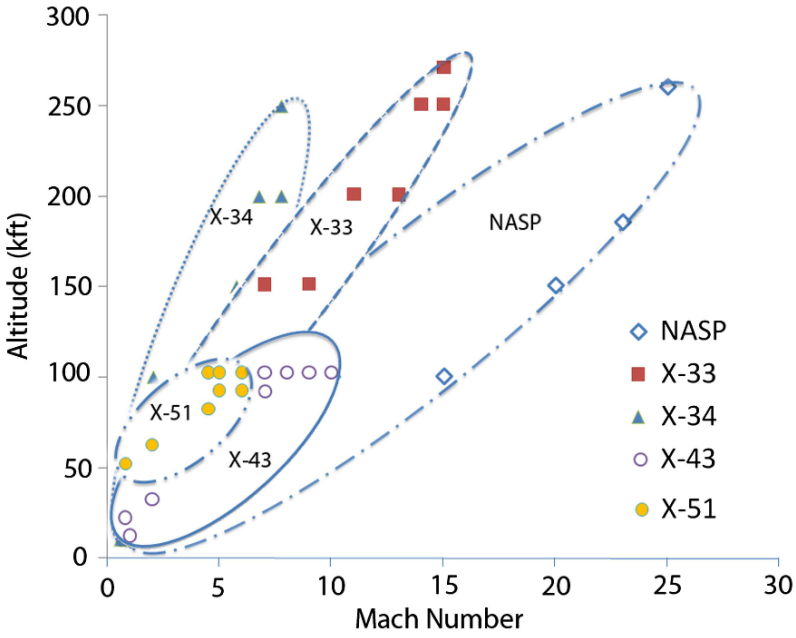


Figure 2.3: Flight Envelope for Recent Vehicles.[15]

Part of this potential that hypersonics holds, is the speed at which the vehicles can travel and the ability to traverse a non-ballistic trajectory; Figure 2.4. Using the case of the AHWP this means a reduced time to target and also the ability to fly a non-threatening ICBM trajectory. A recent example of the need for this global strike capability came from the RQ-170 stealth drone that crash landed in Iran. The use of hypersonic global strike would have been ideal within this situation as it would have been able to prevent a large tech transfer to the nation of Iran through the nearly immediate destruction of the downed aircraft.[16]

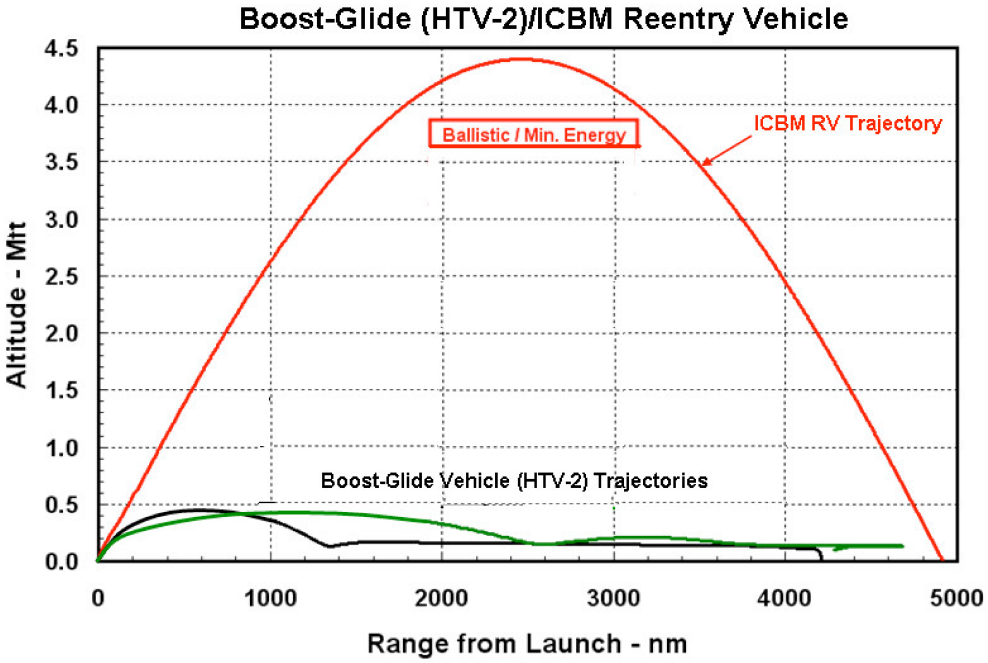


Figure 2.4: Comparison of ICBM to Hypersonic Missiles.[13]

The remaining potential for hypersonics can be seen in the self-propelled airbreathing hypersonic vehicles. These vehicles offer three more main advantages when compared to rockets: the vehicles do not have to carry their own oxidizer, the I_{sp} offered by the engine far exceeds the theoretical limit of a rocket and finally the estimated crash ratio is substantially less. The first and second advantages in not having to carry an oxidizer and the higher I_{sp} results in less weight per kilogram payload and thus substantially less cost per vehicle.

Figure 2.5 following shows the comparison between scramjet and rocket I_{sp}'s as well as some other standard propulsion systems to quantify the efficiency advantage of scramjets at high speeds. Also, Table 2.1 is shown to give a representative sample of the difference in cost per lb payload to Low Earth Orbit(LEO) between varying rocket and scramjet based vehicles. These systems include a rocket based Expendable Launch Vehicle(ELV), the reusable space shuttle, Two Stage To Orbit(TSTO) scramjet based systems staged at different Mach numbers, and a Single Stage To Orbit(SSTO) scramjet system. Although none of these systems are terrestrial based, the trend still holds the same between rockets and hypersonics for these systems.

The final advantage listed, an increase in the estimated safety of the vehicle, is also shown in Table 2.1. Again these ratings are for access to space systems but the trend holds the same for terrestrial based systems. The main two reasons for the large difference is firstly, hypersonic systems do not carry an oxidizer which substantially decreases the risk of catastrophic failure from propellant related errors. Secondly, there is no section of the flight envelope that a full mission abort cannot be performed from. This problem is currently a major issue within manned rocket-based access to space.[17]

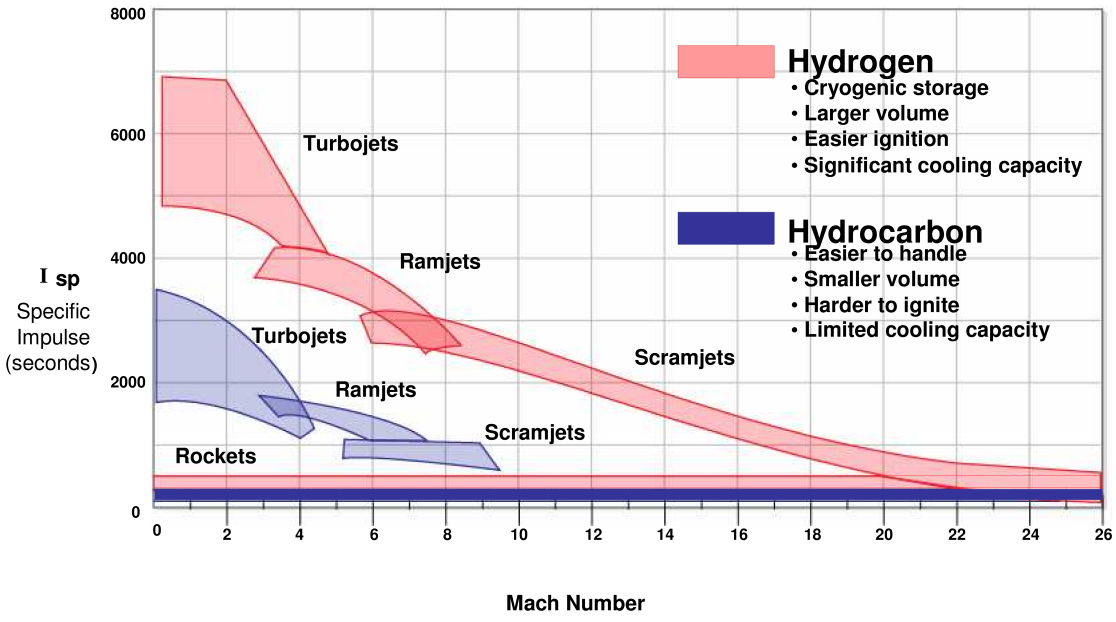


Figure 2.5: Propulsion I_{sp} Curve.[9]

Table 2.1: Estimated Cost of Hypersonic Vehicles[18]

			TSTO			
	ELV	Space Shuttle	Mach 7	Mach 10	Mach 15	SSTO
Payload Fraction	3%	1%	1-2%	3%	4%	5%
Loss of Vehicle/Payload	1:50	1:100	1:4,000	1:60,000	1:110,000	1:160,000
$/lb to LEO	$2,500	$10,000	$1,700	$2,000	$1,400	$1,000

2.2 Optimization of Hypersonic Vehicles

The number of parameters that go into designing a hypersonic vehicle vary but in general, as for any of other design space, the more parameters you include in the initial design sweep, the closer you will be able to reach the final design position.[1] As previously mentioned, the main block from modeling all possible variables is the computation time required to create a single point, let alone a multidimensional sweep of a design space. To give an example of the kinds of variables considered, the variables used within the design of the X-43A are shown following in Figure 2.6.

From the initial design sweeps in which an individual point was chosen, design iterations have to be performed to increment the design closer to the desired optimal solution. Currently, this can take upwards of months to complete a single iteration when on the order of $10e0$runs is desired. One of the current directives to combat this problem is the Integrated Design and Engineering Analysis(IDEA) code being produced within NASA. This code performs complete iterations on the vehicle from an initial design point. As mentioned, the closer you can get to final design point the less computational time that is required to iterate. Currently, the end goal for IDEA in 2014, is to reduce a single iterations down to 2 days.[1] Figure 2.7 shows all of the parameters that currently planned to be considered within IDEA. To give a representation of the inter-coupling of these parameters a flow diagram of the current standard design process is shown within Figure 2.8.

. Although not represented within the figure, each of these parameters is non-linearly coupled and thus even though the parameter list is impressive parts of the design space still have to be approximated.

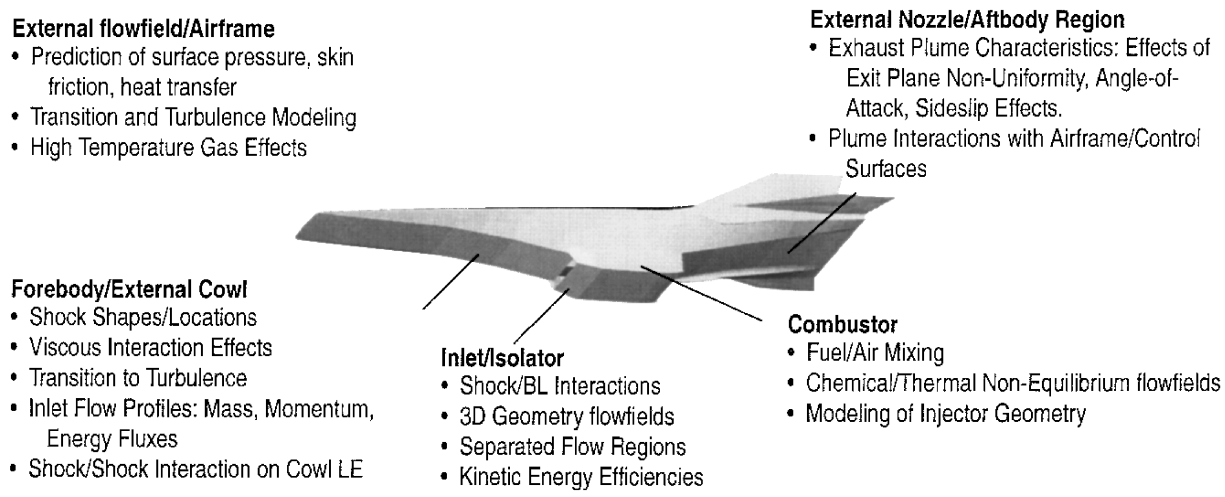


Figure 2.6: Modeling Requirements for the X-43A.[19]

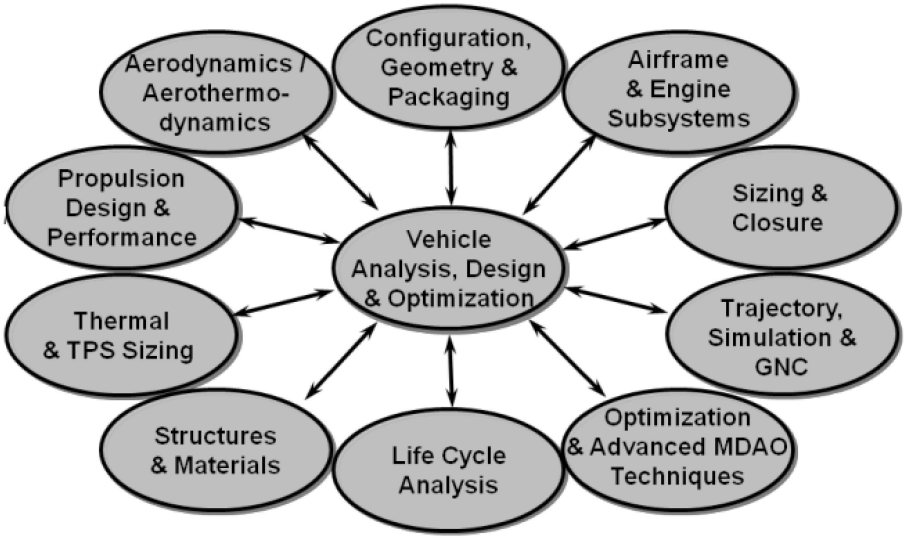


Figure 2.7: Parameters Considered in NASA IDEA.[19]

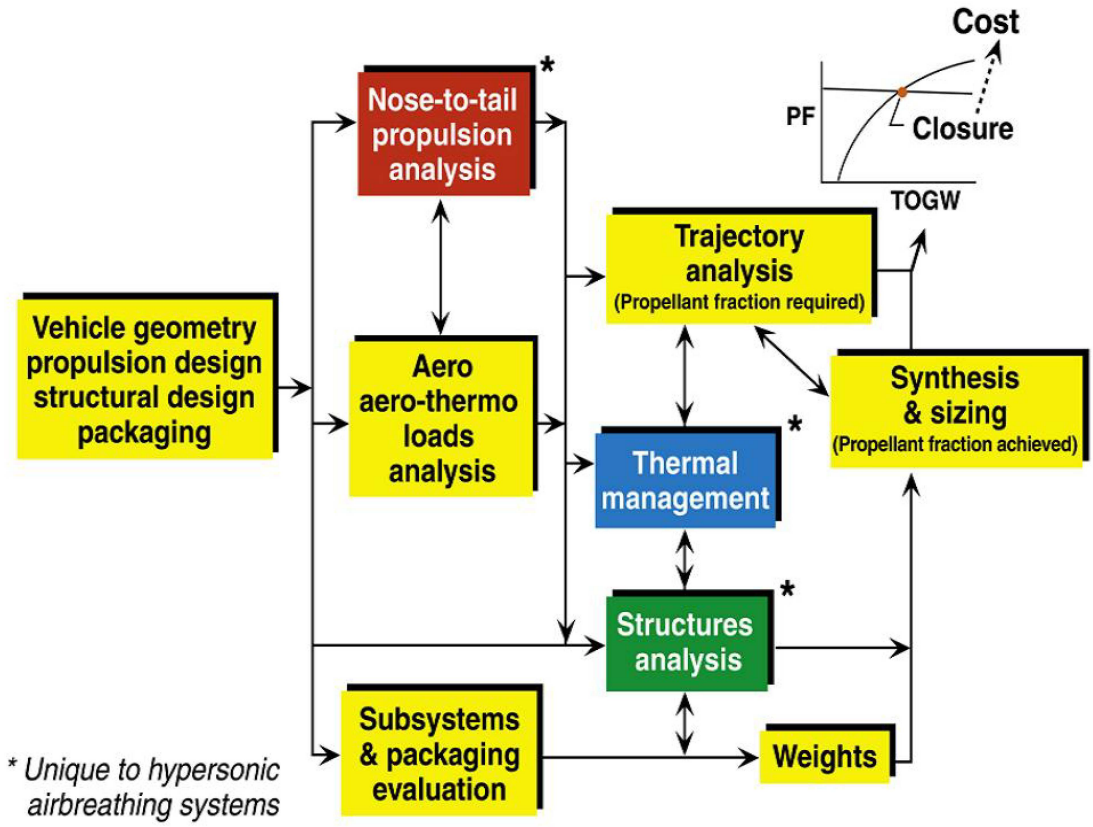


Figure 2.8: Non-Linear Interaction of Hypersonic Vehicle Design.[18]

2.3 Importance of Low Fidelity Computation Models

To prove the computational expense in modeling large parameter spaces, Equation 2.1 shows the relationship between the number of sample points and the parameter space dimensionality.[20]

$$n^k \tag{2.1}$$

Where, n is the number of sample points and k is the number of dimensions. Thus, if 10 points are sampled per dimension within a 5 dimensional space, something not unreasonable nor very big individually, this design space would calculate out to 10^5 points. To understand this better, assume each individual point solution is optimized to on the order of a min; populating the design space would still require 16,000 cpu hours. Calculated at the average cpu per hour cost of 10 cents it would thus cost $1,600 to compute all of the design point solutions.

Finally, to really show the need for low fidelity computation models, remove the assumption that the model only takes a minute to solve. Running at a low estimate of 10-20% saving in computational time, these savings can quickly become a large amount of money. For example, a 20% percent saving in computation on an hour long high fidelity model per solution using the same sampling as before would save $20,000.

2.3.1 Reduced Order Modeling

Reduced Order Modeling(ROM) has recently become more prominent as the cost of computing has continued to drop. This is because large simulations and large design space sampling has now become feasible. To show this, reference [21] explains how the cost of computing dropped 5 orders of magnitude from the 1955 to 1998 and how this initiated the science of large scale modeling simulations.

Currently, there are three main techniques within the aerodynamics community considered for ROM: Proper Orthogonal Decomposition(POD), Volterra series and surrogate modeling.[15] The method that was chosen from these options for this thesis was surrogate modeling, more specifically

Kriging surrogate models. The decision to use this ROM will be explained further within Section 2.4.

The common denominator between all of these methods is the use of linear basis functions as prediction methods to approximate off parameter design space points not specifically computed. These basis functions that each the ROM are based upon are calculated in varying ways and a brief overview for each model is shown below. Secondly, it should be noted that each of these linear basis functions are being considered spatial non-linearities and not temporal.

Proper Orthogonal Decomposition

POD is based on the Singular Value Decomposition and extracts out an optimal orthonormal basis to be used in constructing a continuous spectrum. For a two dimensional sampling matrix shown in Equation 2.2 individual points are assumed to have an infinite amount of modes as shown with Equation 2.3. These modes are constructed through the eigenvalues and eigenvectors of the system (Equation 2.4) and then truncated to provide a finite amount of modes to be evaluated. This method is similar to the Volterra series as the method becomes more accurate as more modes are included within the finite basis. In implementation of POD the basis weights within the system are found through a second modeling approximation. Within reference [22] the second model that was used was the Kriging surrogate model. This additional computational complexity makes POD less tractable for parameter spaces that do not require an orthonormal basis.[23]

$$\mathbf{Y} = \begin{bmatrix} Y\left(x_1^{(1)}\right) & \cdots & Y\left(x_1^{(n)}\right) \\ \vdots & \ddots & \vdots \\ Y\left(x_p^{(1)}\right) & \cdots & Y\left(x_p^{(n)}\right) \end{bmatrix} \tag{2.2}$$

$$x = \xi_1\psi_1 + \xi_2\psi_2 + \cdots + \xi_r\psi_r \tag{2.3}$$

$$\begin{aligned} \mathbf{Y}^T\mathbf{Y}V_i &= \Lambda_i V_i \\ \psi_i &= \frac{1}{\sqrt{n\Lambda_i}}\mathbf{Y}V_i \end{aligned} \tag{2.4}$$

Volterra Series

Volterra is based upon constructing kernels within an infinite series that is constructed using the response of the full system to a set of parameter inputs. As with POD this infinite series needs to be truncated in order to provide a finite amount of calculations. For weakly non-linear systems this truncation can be as small as a second order approximation. Unfortunately, this requires a priori knowledge of the system being solved to be able to justify truncation down to this level or else an estimate has to be made on the level of non-linearity within the system. One of the main disadvantages of this method is that the series is not guaranteed to converge and thus is not recommended for systems in which convergence may be an issue. Equation 2.5 following shows the kernel that is used as a basis for the ROM models constructed for the decomposition.[23]

For an initial state response of $\mathbf{w}(0)$ an arbitrary real parameter input $u(t)$ is applied for $t \geq 0$. Where $\mathbf{h}$ is the order kernel being solved within the series (eg. $\mathbf{h}_1$ is the linear unit impulse response).

$$\begin{aligned} \omega(t) = \quad & \mathbf{h}_0 + \int_0^t \mathbf{h}_1(t-\tau)u(\tau)d\tau \\ & + \int_0^t \int_0^t \mathbf{h}_2(t-\tau_1, t-\tau_2)u(\tau_1)u(\tau_2)d\tau_1 d\tau_2 \\ & + \sum_{n=3}^{N} \int_0^t \cdots \int_0^t \mathbf{h}_n(t-\tau_1, \ldots, t-\tau_n)u(\tau_1)\ldots u(\tau_n)d\tau_1 \ldots d\tau_n \end{aligned} \tag{2.5}$$

Surrogate Models

Surrogate models are like Volterra and POD in that these methods use basis functions to approximate continuous spaces. Surrogate modeling is not a single method like the previous two though and encompasses multiple, different methods for approximating the design space. Examples of some of these methods and the basis functions that each uses is shown below minus the Kriging model which will be explained in full in Section 2.4. These methods were chosen to be showcased here as they are mentioned later within Section 2.4.3 and are used as comparison tools to justify the logic behind choosing the Kriging surrogate.

- Polynomials

$$\hat{f}(\mathbf{x}) = c_0 + \sum_{i=1}^{n} c_i x_i + \sum_{i=1}^{n}\sum_{j\geq i}^{n} c_{ij} x_i x_j + \sum_{i=1}^{n}\sum_{j\geq i}^{n}\sum_{k\geq j}^{n} c_{ijk} x_i x_j x_k + \dots \tag{2.6}$$

  This surrogate uses polynomials as the basis functions, does not include compact support and is highly dependent on the level of polynomial used within the approximation. [24]

- Kriging

- Artificial Neural Networks(ANN)

$$\hat{f}(\mathbf{x}) = tanh\left(tanh\left(xA_0 + \theta_0\right)A_1 + \theta_1\right) \tag{2.7}$$

  Where, $A_0, \theta_0, A_1, \theta_1$ are the neuron weights in the two layered algorithm and the offset values respectively.

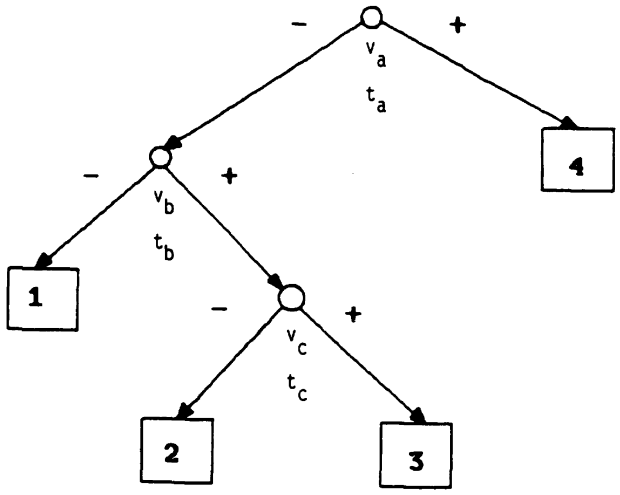
  The ANN used in Section 2.4.3 is the stochastic layered perceptron(SLP) feed-forward learning algorithm. This feed forward type of ANN reduces the training computation cost by fixing some of the neuron weights. Although, this reduces the cost of the model it also reduces some of the flexibility that a standard perceptron learning based ANN would experience. A short overview of the standard perceptron feed-forward ANN is included in Appendix A.2.[24]

- Multivariate Adaptive Regression Splines(MARS)

$$\hat{f}(\mathbf{x}) = a_0 + \sum_{m=1}^{M_1} a_m B_m(x_i) + \sum_{m=1}^{M_2} a_m B_m(x_i, x_j) + \dots \tag{2.8}$$

This surrogate uses truncated power spline as the basis functions, B_m. These functions are determined through a regressive algorithm that uses an indicator function to determine which form the basis function should take. Figure 2.9 shows the forms that each of the individual basis functions are able to take where x_{v*} is the predictor variable and t^* is the split point for the step function shown in Equation 2.9.[24, 25]

$$H[\eta] = \begin{cases} 1 & if\, \eta \geq 0, \\ 0 & otherwise, \end{cases} \tag{2.9}$$



$$B_1 = H[-(x_{v_a} - t_a)]H[-(x_{v_b} - t_b)]$$
$$B_2 = H[-(x_{v_a} - t_a)]H[+(x_{v_b} - t_b)]H[-(x_{v_c} - t_c)]$$
$$B_3 = H[-(x_{v_a} - t_a)]H[+(x_{v_b} - t_b)]H[+(x_{v_c} - t_c)]$$
$$B_4 = H[+(x_{v_a} - t_a)]$$

Figure 2.9: Basis Function Options for MARS.[25]

From above, the Volterra series or the POD method do not hold any tractable advantages over surrogate modeling for the work proposed within this thesis. For Volterra, the need of a priori knowledge and the fact that the method is not guaranteed to converge both make the method not tangible. The reason that it is was unknown if the Volterra series would have converged or not was that at the time of writing no other work of this kind had been able to be found. This meant that a lack of a prior knowledge of the type of non-linearities within the chosen systems left the question of how many basis functions would need to be used let alone the convergence conditions for each set. For the POD method, reference [22] has shown that Kriging is more accurate than POD-Kriging and the added computation for less accuracy meant that this choice was not tangible either. Figure 2.10 following shows the computed error metrics within reference [22] that show the difference between the methods as the number of sample points is increased within the continuous space being modeled.

Even though Figure 2.10 shows Kriging has more accuracy than POD there are however, situations in which surrogates are not the best choice. An example being the case mentioned previously of needing an orthonormal basis. For this example the POD method would obviously be the clear winner between POD and surrogates as none of the surrogate models were able to guarantee an orthonormal basis.

Finally, the author recommends that it would be beneficial to contrast the efficiency of all these methods on the methodology being proposed within this thesis. This work though, would be beyond the current scope and thus will be left to future researchers.

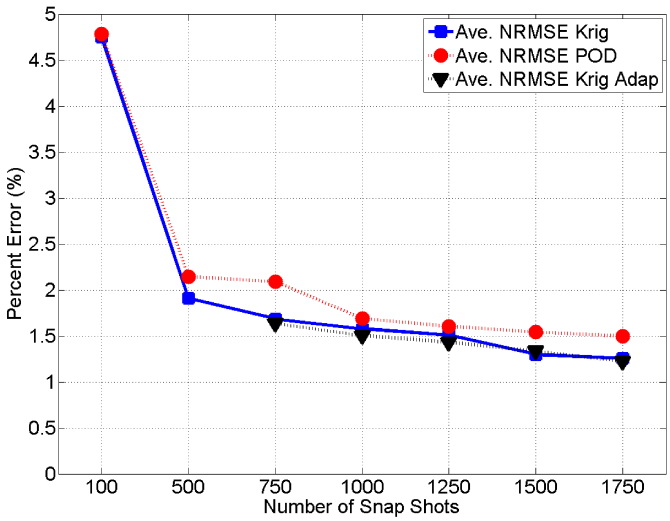


Figure 2.10: Normalized Root Mean Square Error(NRMSE) Between Kriging and POD.[22]

2.4 Kriging Surrogate Model

The surrogate model that was chosen for this work was the unaltered Kriging surrogate model. This model was chosen because it is well documented in the engineering world and secondly the model provides multi-spectral wave space support.[20] The advantage of multi-spectral support becomes important within this thesis as the multi-dimensional error space that will be experienced has a large spectrum sweep and is not unimodal dependent. Thirdly, Kriging is good at predicting deterministic data that has similar errors in closely correlated sample points.[26] Currently, the CFD algorithms and modeling techniques used within the hypersonic community produce this exact type of data.[22, 26] Finally, the previous section outlined why a surrogate model was favorable to the other choices in ROM.

Following is the full derivation of the Kriging surrogate model, including the basis function that were previously omitted in Section 2.3.1. This derivation will become important following in Section 2.4.2 in which a multi-fidelity correlation is presented as an alternative to the equations shown below.

The true unknown function can be modeled as a known function with the superposition of a Gaussian random function with a variance of σ^2:

$$y(x) = f(x) + Y(x) \tag{2.10}$$

Equation 2.11 shows that for Kriging this known function is approximated as a constant β. A representation of the Gaussian random vectors is shown below in Equation 2.12.

$$y(x) = \beta + Y(x) \tag{2.11}$$

$$\mathbf{Y} = \begin{pmatrix} Y\left(\mathbf{x}^{(1)}\right) \\ \vdots \\ Y\left(\mathbf{x}^{(n)}\right) \end{pmatrix} \tag{2.12}$$

Equation 2.13 shows the basis functions that were used for this type of surrogate model. For reference to the other surrogate model basis functions available please refer to Section 2.3.1. These basis function are based on the correlation of two sample points within the space. This is important as it allows for the method to interpolate between the sample points in n dimensional space.

$$cor\left[Y\left(\mathbf{x}^{(i)}\right), Y\left(\mathbf{x}^{(l)}\right)\right] = exp\left(-\sum_{j=1}^{k} \theta_j \left|x_j^{(i)} - x_j^{(l)}\right|^{p_j}\right) \tag{2.13}$$

For coherence to the notation reference [26] the unity vector is defined following in Equation 2.14.

$$\mathbf{f} = [1, \ldots, 1] \tag{2.14}$$

In Equation 2.15 for n dimensions an $n \times n$ correlation matrix is produced with an evaluation of each of the basis functions.

$$\mathbf{\Psi} = \begin{pmatrix} cor\left[Y\left(\mathbf{x}^{(1)}\right), Y\left(\mathbf{x}^{(1)}\right)\right] & \cdots & cor\left[Y\left(\mathbf{x}^{(1)}\right), Y\left(\mathbf{x}^{(n)}\right)\right] \\ \vdots & \cdots & \vdots \\ cor\left[Y\left(\mathbf{x}^{(n)}\right), Y\left(\mathbf{x}^{(1)}\right)\right] & \cdots & cor\left[Y\left(\mathbf{x}^{(n)}\right), Y\left(\mathbf{x}^{(n)}\right)\right] \end{pmatrix} \tag{2.15}$$

The arbitrary Gaussian vector field is related to the correlation basis functions through the covariance:

$$Cov\left(\mathbf{Y}, \mathbf{Y}\right) = \sigma^2 \mathbf{\Psi} \tag{2.16}$$

Following is the derivation that represents the ability to solve for θ_j as a vector instead of a constant as is presented within reference [26]. Equation 2.17 following shows the likelihood function for the probability distribution for this application. This function can be obtained through the deterministic assumption, as previously mentioned, about the data being analyzed.

$$\begin{aligned} L\left(\mathbf{Y}^{(1)}, \ldots, \mathbf{Y}^{(n)} | \beta, \sigma\right) &= \frac{1}{\left(2\pi\sigma^2\right)^{\frac{n}{2}}} exp\left[-\frac{\sum\left(\mathbf{Y}^{(i)} - \beta\right)^2}{2\sigma^2}\right] \\ L &= \frac{1}{\left(2\pi\sigma^2\right)^{\frac{n}{2}} |\mathbf{\Psi}|^{\frac{1}{2}}} exp\left[-\frac{(\mathbf{y} - \beta\mathbf{f})^T \mathbf{\Psi}^{-1} (\mathbf{y} - \beta\mathbf{f})}{2\sigma^2}\right] \end{aligned} \tag{2.17}$$

To solve for the maximum likelihood estimate(MLE) for β and σ the natural log of Equation 2.17 is taken, resulting in Equation 2.18. In doing this a derivative is able to be taken and set to zero to solve for the MLE's.

$$\ln\left(L\right) = \frac{n}{2}\ln\left(2\pi\right) - \frac{n}{2} ln\left(\sigma^2\right) = \frac{1}{2}\ln\left|\mathbf{\Psi}^{-1}\right| - \frac{(\mathbf{y} - \beta\mathbf{f})^T \mathbf{\Psi}^{-1} (\mathbf{y} - \beta\mathbf{f})}{2\sigma^2} \tag{2.18}$$

Equation 2.19 below is the final representations of the MLE estimates for β and σ.

$$\begin{aligned} \hat{\beta} &= \frac{\mathbf{f}^T \mathbf{\Psi}^{-1} \mathbf{y}}{\mathbf{f}^T \mathbf{\Psi}^{-1} \mathbf{f}} \\ \hat{\sigma^2} &= \frac{\left(\mathbf{y} - \beta\mathbf{f}\right)^T \mathbf{\Psi}^{-1} \left(\mathbf{y} - \beta\mathbf{f}\right)}{n} \end{aligned} \tag{2.19}$$

Substituting Equation 2.19 back into Equation 2.18 results in Equation 2.20. This equation is known as the concentrated ln-likelihood function. Numerical techniques are used to maximize this function and will not be discussed here but can be found in reference [20].

$$\ln(L) \approx -\frac{n}{2}\ln\left(\hat{\sigma}^2\right) - \frac{1}{2}\ln|\mathbf{\Psi}| \tag{2.20}$$

2.4.1 Sampling Methods

There are multiple different ways of being able to sample a design space. The two main methods that are going to be discussed here are the latin hypercube sampling and also full factorial sampling.

Full factorial sampling is the method of using linear spaced vectors within the dimensional space. Figure 2.11 shows a good representation of this sampling. Each of the vectors does not necessarily have to have the same length within the space but the definition requires that each of the vectors remain uniform within their respective dimensions. This was the sampling plan that was chosen for this work even though it has some serious disadvantages when compared to the Latin hypercube.

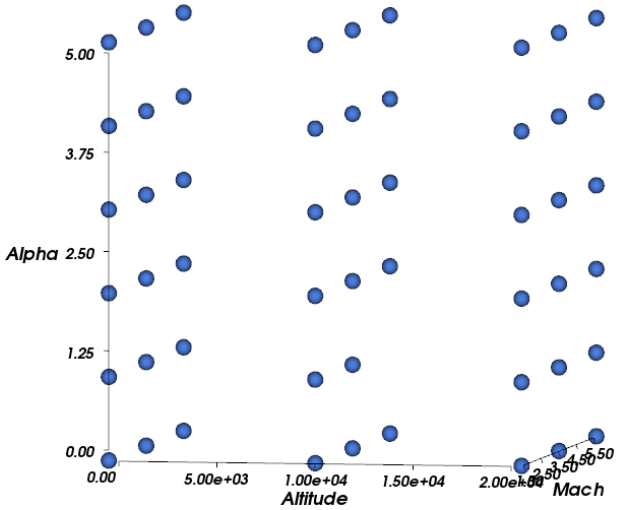


Figure 2.11: Example of Full Factorial Sampling Plan.

Latin hypercube sampling is an algorithm that makes sure that a single point cannot be projected onto another within a single dimension. One of the more well know examples of this type sampling is a sudoku puzzle which follows these rules and some additional ones. For simplicity a Latin square is shown following in Table 2.2 to help the reader visualize a multidimensional representation. The main problem with using this type of sampling plan though, is choosing an optimal sampling solution. For example, the 4 dimensional Latin square space in Table 2.2 has 576 valid solutions to the problem but can be optimized depending on the variables being calculated. Thus, when extrapolated into something such as the diamond airfoil parameter space, there is almost an infinite amount of valid solutions and thus a sizable amount of optimized points depending on the optimization parameters set.

Besides the added complexity of solving for an optimal solution which can be found in reference [20], there are two main advantages that this type of sampling algorithm have. Firstly, that points are not redundantly chosen in the sample dimension as a second dimension is discretized and secondly, that there are in general less sample points needed to span a design space of the same size.

From the cost benefit analysis performed on these metrics, it was determined that the work being performed within this thesis was not complex enough to justify implementing a Latin hypercube sampling algorithm. This work is viewed as an initial proof of concept and although there is some optimization within the calculations, implementing a hypercube was considered beyond the scope of the problem.

Table 2.2: Example Latin Square

B	A	C	D
C	B	D	A
A	D	B	C
D	C	A	B

2.4.2 Co-Kriging

Co-Kriging is a version of the Kriging interpolation method that accounts for the difference in multi-fidelity data. This method has been shown to be very effective at being able to predict the high fidelity unknown function.[27, 28, 29] The derivation differences between this method and the derivation shown within Section 2.4 are going to be presented. The reason for this is that it is currently considered the main alternative to the work presented within this thesis.

Equation 2.21 shows the difference in the samples taken within the design space and the change in the Gaussian random variable because of it. Also, it should be noted that following equations presented are going to use the notation presented within reference [28] for simplicity and for comparison with Figure 2.12.

Thus, c is equal to the "cheap" or the computationally inexpensive data, e is equal to the "expensive" data and d is the difference between the two explained further in Equation 2.22.

$$\mathbf{X} = \begin{pmatrix} \mathbf{X}_c \\ \mathbf{X}_e \end{pmatrix} \qquad \mathbf{Y} = \begin{pmatrix} \mathbf{Y}_c\left(\mathbf{X}_c\right) \\ \mathbf{Y}_e\left(\mathbf{X}_e\right) \end{pmatrix} \tag{2.21}$$

Equation 2.22 shows how the expensive unknown function is estimated from a scaled, ρ, cheap function and a difference Gaussian distribution.

$$Z_e(\mathbf{x}) = \rho Z_c(\mathbf{x}) + Z_d(\mathbf{x}) \tag{2.22}$$

The different definition within Equation 2.22 for the unknown function leads to different covariance associations shown in Equation 2.23 through 2.25. From these different covariance associations, new MLE's are needed for the cheap Gaussian distribution as well the difference Gaussian distribution for an expensive Gaussian estimate.

$$\begin{aligned} cov\left[\mathbf{Y}_c\left(\mathbf{X}_c\right), \mathbf{Y}_c\left(\mathbf{X}_c\right)\right] &= cov\left[Z_c\left(\mathbf{X}_c\right), Z_c\left(\mathbf{X}_c\right)\right] \\ &= \sigma_c^2 \mathbf{\Psi}_c\left(\mathbf{X}_c, \mathbf{X}_c\right) \end{aligned} \tag{2.23}$$

$$\begin{aligned} cov\left[\mathbf{Y}_e\left(\mathbf{X}_e\right), \mathbf{Y}_c\left(\mathbf{X}_c\right)\right] &= cov\left[\rho Z_c\left(\mathbf{X}_c\right) + Z_d\left(\mathbf{X}_c\right), Z_c\left(\mathbf{X}_e\right)\right] \\ &= \rho\sigma_c^2 \mathbf{\Psi}_c\left(\mathbf{X}_c, \mathbf{X}_e\right) \end{aligned} \tag{2.24}$$

$$\begin{aligned} cov\left[\mathbf{Y}_e\left(\mathbf{X}_e\right), \mathbf{Y}_e\left(\mathbf{X}_e\right)\right] &= cov\left[\rho Z_c\left(\mathbf{X}_e\right) + Z_d\left(\mathbf{X}_e\right), \rho Z_c\left(\mathbf{X}_e\right) + Z_d\left(\mathbf{X}_e\right)\right] \\ &= \rho^2\sigma_c^2 \mathbf{\Psi}_c\left(\mathbf{X}_e, \mathbf{X}_e\right) + \sigma_d^2 \mathbf{\Psi}_d\left(\mathbf{X}_e, \mathbf{X}_e\right) \end{aligned} \tag{2.25}$$

Where, $\mathbf{\Psi}$ is the same correlations used as basis functions as presented in Equation 2.15. Combining all of the above covariance equations, the final covariance for the multi-fidelity derivation is shown following in Equation 2.26.

$$\mathbf{C} = \begin{pmatrix} \sigma_c^2 \mathbf{\Psi}_c\left(\mathbf{X}_c, \mathbf{X}_c\right) & \rho\sigma_c^2 \mathbf{\Psi}_c\left(\mathbf{X}_c, \mathbf{X}_e\right) \\ \rho\sigma_c^2 \mathbf{\Psi}_c\left(\mathbf{X}_c, \mathbf{X}_e\right) & \rho^2\sigma_c^2 \mathbf{\Psi}_c\left(\mathbf{X}_e, \mathbf{X}_e\right) + \sigma_d^2 \mathbf{\Psi}_d\left(\mathbf{X}_e, \mathbf{X}_e\right) \end{pmatrix} \tag{2.26}$$

Equation 2.27 following is the natural log likelihood for the cheap variables and is found by taking the derivative of Equation 2.26 w.r.t. β_c and σ^2. This likelihood function is equivalent to Equation 2.20.

$$-\frac{n_c}{2}\ln\left(\hat{\sigma}_c^2\right) - \frac{1}{2}\ln|\det\left(\mathbf{\Psi}_c\left(\mathbf{X}_c, \mathbf{X}_c\right)\right)| \tag{2.27}$$

Equation 2.28 is the same likelihood parameter estimate as the equation above but accounts for the multi-dimensional data. Solving Equation 2.27 and 2.28 allows for the expensive Gaussian estimate to be computed and thus the unknown high fidelity function is able to be estimated.

$$-\frac{n_e}{2}\ln\left(\hat{\sigma}_d^2\right) - \frac{1}{2}\ln|\det\left(\mathbf{\Psi}_d\left(\mathbf{X}_e, \mathbf{X}_e\right)\right)| \tag{2.28}$$

To display how well this multi-dimensional estimate works the one dimensional data presented within reference [28] is shown following. For this example the cheap and expensive function estimates are shown in Equation 2.29.

$$\begin{aligned} f_e(x) &= (6x-2)^2 \, sin(12x-4), x \in \{0,1\} \\ f_c(x) &= Af_e + B(x-\frac{1}{2}) - C \end{aligned} \tag{2.29}$$

Where, $A = 0.5$, $B = 10$ and $C = -5$

From these estimates for a cheap and expensive function the Kriging and Co-Kriging functions are calculated and shown in Figure 2.12. To quantify how well the Co-Kriging results match the exact analytical unknown expensive function Figure 2.13 of the mean square error is included. Equation 2.30 following is the definition of the mean squared error estimate as used within reference [28]. Where, $\mathbf{c}$ is defined following in Equation 2.31.

$$s^2(\mathbf{x}) = \hat{\rho}^2\hat{\sigma}_c^2 + \hat{\sigma}_d^2 - \mathbf{c}^T\mathbf{C}^{-1}\mathbf{c} \tag{2.30}$$

$$\mathbf{c} = \begin{pmatrix} \hat{\rho}\hat{\sigma}_c^2\psi_c\left(\mathbf{X}_c, \mathbf{x}^{(n+1)}\right) \\ \hat{\rho}^2\hat{\sigma}_c^2\psi_c\left(\mathbf{X}_e, \mathbf{x}^{(n+1)}\right) + \hat{\sigma}_d^2\psi_d\left(\mathbf{X}_e, \mathbf{x}^{(n+1)}\right) \end{pmatrix} \tag{2.31}$$

As previously mentioned, Co-Kriging was shown in full here as it is currently considered the main alternative to the proposed method within this thesis. The justification for not using the method shown within this section or the derivative enhanced version as shown within references [27, 29] is that all of these methods still require high fidelity data. One of the secondary goals of this work was to examine the error space between the fidelity levels in order to find extendable mapped approximations in parameter space, such that high fidelity data would not need to be computed. Because of this goal and limitation of Co-Kriging, the methodology explained in Section 1.1 was used instead of a variation of the Co-Kriging method.

A comparison between these two methods would be considered greatly advantageous but was also considered outside the current scope and thus was not be performed within this work.

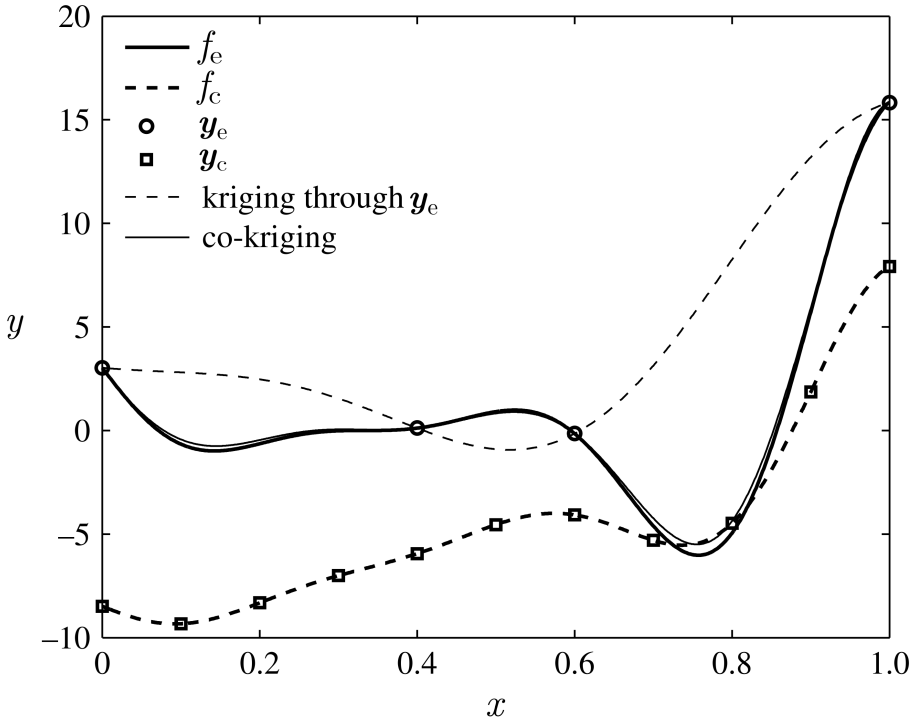


Figure 2.12: Interpolation Results for One Dimensional Co-Kriging.[28]

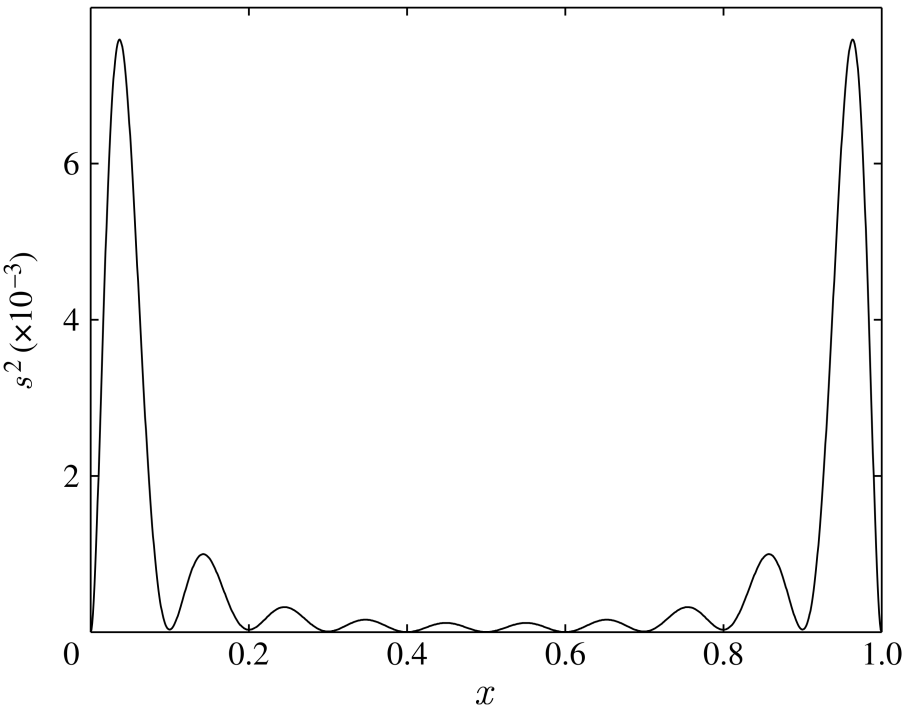


Figure 2.13: Mean Square Error Prediction for Co-Kriging Results.[28]

2.4.3 Dakota Surfpack Surrogate Models

The Sandia National Laboratories Dakota surfpack package was chosen to produce the surrogate models for this work. Firstly, this package was chosen because it is supported under the GNU general public license and is thus free. Secondly, surfpack was able to support n dimensional models which was an important criteria for the high dimensionality required within the scramjet chemistry section. Finally, this package supported other surrogate models which were used to verify that the Kriging model was the appropriate model choice for this work.[24, 30] The models able to be used within surfpack package are:

(1) Low Order Polynomials

(2) Kriging Interpolation

(3) Multivariate Adaptive Regression Splines(MARS)

(4) Artificial Neural Networks(ANN)

All of the available models are plotted against some representative deterministic CFD data provided within the package in Figure 2.14. This figure is provided to give a general overview on how each of the models performed relative to Kriging. The error metrics for these models are also shown in Table 2.3 to support the decision to use the Kriging interpolation model as analytically justified within Section 2.3.1. The definitions for each of the error metrics used below are defined within Appendix A.1.

From the example error metrics shown in Table 2.3 it was determined that the PRESS error metric would be used to verify the error associated with the Kriging models built in Chapter 3 and Chapter 4.

Table 2.3: Error Metrics for Surrogate Models in Dakota

	Models				
Error Metric	1st Poly	2nd Poly	MARS	Kriging	ANN
PRESS	1.61E−6	8.216E−7	NaN	8.21E−7	1.30E−5
RMS	9.76E−4	6.36E−4	2.52E−3	3.18E−19	1.81E−13
R^2	8.50E−1	9.36E−1	N/A	N/A	N/A

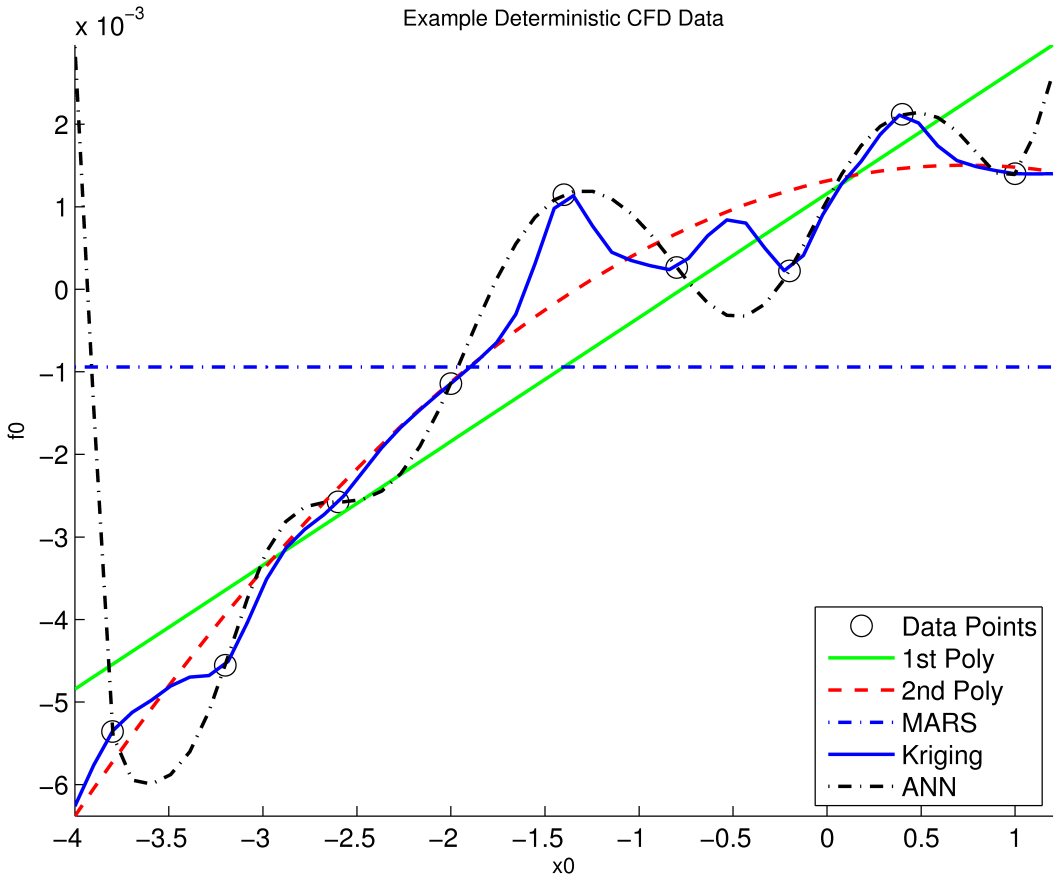


Figure 2.14: Comparison of Dakota Models.

2.5 Importance and Impact of Finite Rate Chemistry

Limiting the amount of computations that have to be conducted to perform a chemical analysis is key in being able to feasibly analyze different scramjet configurations.[31] One of the biggest features in being able to reduce the amount of computations, is choosing a simplistic fuel to analyze. For this work though, it was determined that a hydrocarbon, over a hydrogen, based fuel was necessary.

The discussion of which fuel to use within a scramjet engine is still prevalent issue within the hypersonic community.[32] The reasoning behind using a hydrocarbon based fuel within this thesis is that these fuels having a higher energy density and also because of their chemical stability. This stability gives hydrocarbon based fuels a more promising short term outlook than liquid hydrogen based systems, as the ease of handling the fuel will reduce problems associated with refueling and storage of the propellants. The instability of mass producing liquid hydrogen was discovered in the early 1950's when early rocket and hydrogen powered turbomachinery was being investigated by both the United States and the USSR.[33] This statement on the preference for hydrocarbons can also be supported by the choice of the USAF X-51 project which migrated to JP-7 from the H_2 unlike the previous X-43A program.[9, 34]

Thus, needing a hydrocarbon based fuel with a simplistic chemical composition, the propellant Ethene(Ethylene) was chosen. The chemical composition for this hydrocarbon is C_2H_4 with the chemical structure able to be seen below in Figure 2.15. The alternative choice to Ethene was Jet A-1 which has an estimated chemical composition of $C_{12}H_{23}$ and reduced order mechanisms on the order of 17 species with 14 reactions.[35] Jet A-1 is considered more real world applicable as it is already in large scale production for the civil aeronautical industry but the increased computational time that would have been required was considered infeasible with the available resources.

Having determined the fuel to be used within this thesis, proven reduced-order mechanisms were investigated. The models eventually chosen can be seen in reference [36] and had on the order of 7 species with 3 reactions and 10 species with 10 reactions. These simplistic chemical sets

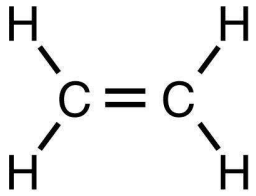


Figure 2.15: Chemical Structure for Ethene.

reduced the computational time down to a feasible level and also provided a large fidelity difference to be investigated within the scramjet configuration.

2.6 CFD Solver and Meshing Suite

The meshing programs and solver that were chosen for this dissertation are Chimera, ANSYS ICEM and VULCAN. A brief background on each will be provided following to give a general introduction to each software package and why it was chosen over other packages for this work.

2.6.1 VULCAN v.6.0.2

VULCAN is a three dimensional ITAR restricted solver produced by the Hypersonic Air-breathing Propulsion Branch at Langley Research Center, NASA. This structured solver is able to compute calculations for both turbulent reacting and non-reacting flows, from hypersonic speeds all the way to subsonic flows. Some of the features included within the solver include the ability to run V and W multi-grid algorithms, elliptical and space marching schemes for reacting and non reacting flow and a variety of turbulence models. Some of these turbulence models include Reynold Averaged Navier-Stokes (RANS), Large Eddy Simulation (LES) and also a hybrid RANS/LES model. Finally, a single gas or a mixture of gases can be modeled as either calorically perfect or thermally perfect for reacting and non-reacting flows. [37]

The main reasons that this CFD solver was chosen for this study was firstly, being written and distributed by NASA, no fees would have to be paid for the use of the solver. Unfortunately, this also meant that little to no technical support was received within the first section of running simple

generic sample cases through the solver. Secondly, the ability to solve turbulent reacting flows at supersonic to hypersonic speeds was necessary to being able to complete the work. Although the reacting flow capability was only used within the second part of this dissertation, it was considered simpler to stay within a single solver package to reduce the necessary lead time of learning a new package.

The second main fault that came with choosing this solver package was, as previously mentioned, the solver depended upon structured grids. For the test case run of the diamond airfoil, this lead to meshing issues that were derived from not achieving the correct stretching ratio. These issues would have been avoided using an unstructured mesh instead of the produced structured mesh, but the need for a reacting flow solver outweighed these problems. The meshing issues will be described further within Section 3.2.2.[37]

2.6.1.1 Turbulence Model

The turbulence model that was chosen within all the viscous runs used within this work was the k-ω RANS model. This model was chosen due to its history within the field and also the option of using wall matching functions instead of explicitly solving to the wall. No LES was performed or compared to the results produced as this was considered outside the scope of the work being produced.

2.6.2 Chimera v.2.1

Chimera Grid Tools is an ITAR restricted NASA based meshing tool suite. This tool suite was originally designed to be used with the NASA in-house produced CFD solver OVERFLOW but as the mesh output is in a plot3d format, this suite was still able to be used with VULCAN. The other primary reason that this meshing suite was chosen is that, being an in-house NASA code, it was free to acquire. From this design suite the main programs that were used were GRIDED, SRAP, HYPGEN and SURGRD, with a detailed description of each displayed in Appendix B. It is recommend that the reader become familiar with the context of each of these functions as they

will be referenced extensively in Section 3.2.[38]

2.6.3 Batch Processing

To complete the large scale data sets necessary for the surrogate models within this thesis, batch automation had to be completed for both mesh creation and the flow solver. The two main algorithms that were produced was an input automation code for VULCAN and a custom cloud controller that allowed for multiple simultaneous parallel solver instances.

Appendix E.4 contains a detailed description of the inputs that were used to batch automate the Chimera meshing suite. Currently, no automation was done in ICEM due to time restrictions, even though the program does contain the ability to be batch automated.

Appendix E.1 outlines the input algorithm that was necessary to dynamically create formatted text input files for VULCAN. This automation work was done so that design spaces could be dynamically allocated as necessary without any user input.

Finally, Appendix E.2 shows the algorithm that was used to speed up the computation of the design spaces through the use of a custom cloud cluster. This controller performed two functions including controlling the MPD ring that was set up across each of the machines incorporated into the cluster and a load balanced algorithm to minimize the amount of idle cores within the cloud. All of this was achieved through the Linux bash scripting environment, with port 22 ssh tunneling used as the networking tool to process data transfer around the entire ring.

Chapter 3

Diamond Airfoil Test Case

To simplify the coding and help facilitate understanding of the proposed goal within this thesis, a simple test case with a defined analytical solution was chosen as a verification tool. By solving the empirical error between solution spaces on a diamond airfoil, the necessary lessons could be applied to the scramjet combustion case for a relatively small computational expense. To begin this analysis an analytical solution had to be obtained that could be mass produced across large design spaces for comparison to the CFD results from VULCAN. As no program was found at the time of writing that was able to produce these results, an in-house code, Diamond Analytical(DA), was created. The following section explains the algorithm and also the work that went into producing this program.

Following the production of DA, Chimera was used to produce grids for both the Euler and N-S cases to be run through VULCAN. The accuracy of the solutions between DA, Euler and the N-S was used as the basis for the different multi-fidelity levels within the surrogate analysis. To achieve accurate CFD results at a reasonable computational expense, both a grid convergence study as well a solver optimization study were run for both the Euler and N-S equations.

After achieving accurate CFD results, all three levels of fidelity were input into the surrogate model for analysis to be performed on the error space. Both a four dimensional as well as an example one dimensional space are presented to help the reader understand the multi-dimensional aspect of the design spaces.

3.1 Diamond Analytical

Diamond Analytical is an analytical solver for the lift and drag on a diamond airfoil that is able to produce parameter sweeps within a four dimensional parameter space. The four dimensions that are able to iterated across are shown within Table 3.1. The code was written in FORTRAN 95 and was completed as a part of this thesis, as no other viable option was found to produce the necessary results. To verify the algorithm produced, the results were verified against a hand worked solution within reference [39] and shown to match exactly for the mixture of double and single precision numbers used within the code.

A high level overview of the algorithm can be seen within Appendix C, as well as a more in depth description for each of the functions that are referenced within the logic block diagrams. For a full disclosure of the program documentation or a copy of the source code please contact the author directly.

Table 3.1: Variable Dimensions able to Iterated in DA

Variable	Symbol	Min Value	Max Value	Units
Altitude		0	84.85	km
Mach	M	1.05	$\sim$10	
Wedge Half Angle	θ	0	θ_{max}	deg
Angle of Attack	α	0	90	deg

The derivation for θ_{max} as defined within Table 3.1 is shown within Equation 3.1. Also, the definition of the variable ν is shown within Equation 3.2. The geometry assumption made in Equation 3.1 is shown within Figure 3.1 following. Finally, the reason for the approximate M_{max} is because of the way that the program is currently defined. That is, for the θ's that were examined at this condition the maximum defined $\nu(M_2)$ of 130.5° is violated past this approximate M_{max} value.

$$\begin{aligned} \nu\left(M_2\right) &= \nu\left(M_1\right) + \Delta\alpha \\ \nu\left(M_2\right) &= \nu\left(M_1\right) + 2\theta \\ \nu\left(M_2\right)_{max} &= 124.7^\circ \\ \theta_{max} &= \frac{124.7 - \nu(M_1)}{2} \end{aligned} \tag{3.1}$$

$$\nu(M) = \sqrt{\frac{\gamma+1}{\gamma-1}}\tan^{-1}\sqrt{\frac{\gamma+1}{\gamma-1}\left(M^2-1\right)} - \tan^{-1}\sqrt{M^2-1} \tag{3.2}$$

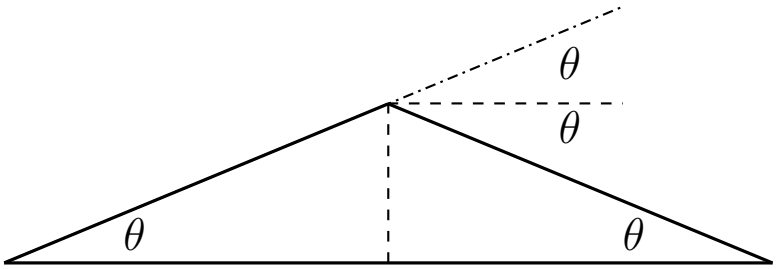


Figure 3.1: Geometry Assumption for θ_{max} Derivation.

Finally, it should also be noted that besides the standard normal Mach relations, the closed form solution to the $\theta-\beta-M$ equations was used to give a faster computation time when compared with the root finding secant method.[40] The formulation that was used can be seen below in Section 3.1.1.

3.1.1 Closed Form $\theta - \beta - M$

The closed solution $\theta-\beta-M$ that was used is of the form proposed by Wellman in reference [41] with a correction factor as displayed within reference [42]. Equations 3.4 and 3.5 following display the closed form formulation in full. Also, the original $\theta - \beta - M$ equation that the closed form solution is derived from is shown below in Equation 3.3 for completeness.

$$\frac{\tan\left(\beta-\theta\right)}{\tan\beta} = \frac{\left(\gamma-1\right)M_1^2\sin^2\beta + 2}{\left(\gamma+1\right)M_1^2\sin^2\beta} \tag{3.3}$$

$$\tan\beta = \frac{b+9ac}{2\left(1-3ab\right)} - \frac{d\left(27a^2\sqrt{c}+9ab-2\right)}{6a\left(1-3ab\right)}\tan\left[\frac{n\pi}{3} + 1/3\arctan\frac{1}{d}\right] \tag{3.4}$$

Where,

$$
\begin{aligned}
n &= \begin{cases} n = 0 & \text{Weak Solution} \\ n = 1 & \text{Strong Solution} \end{cases} \\
a &= \left[\frac{\gamma - 1}{2} + \frac{(\gamma + 1)\, c}{2}\right] \tan\theta \\
b &= \left[\frac{\gamma + 1}{2} + \frac{(\gamma + 3)\, c}{2}\right] \tan\theta \\
c &= \tan^2 \mu \\
d &= \sqrt{\frac{4\,(1 - 3ab)^3}{(27a^2c + 9ab - 2)^2} - 1}
\end{aligned}
\tag{3.5}
$$

3.1.2 Initial Results from Diamond Analytical

A representative sample is shown below for a two dimensional design space for the drag on the airfoil. The parameters used within the study can be seen below in Table 3.2 with the plotted drag metrics able to be seen within Figure 3.2. A two dimensional design space was chosen for this work because higher order design spaces are hard to visualize and it was not deemed necessary to demonstrate this capability of the program. Finally, it should be noted that the full factorial sampling plan is being used as was explained within Section 2.4.1.

Table 3.2: Design Space Parameters for 2D DA Example Space

Variable	Min Value	Max Value	Delta
Mach	1.5	9.0	0.1
AoA	0.0	4.5	0.1
Wedge Angle	5.0		
Altitude	0.0		

From Table 3.2 it can be seen that the sampled area was a very densely sampled design space. This was not possible for the higher CFD fidelity model but due to the relatively inexpensive cost for these calculations it is interesting to examine such a dense sampling. Also, it can be considered that the design space within Figure 3.2 is very benign but as the nonlinearities associated with the geometry and parameters variables chosen are fairly benign this is not unexpected.

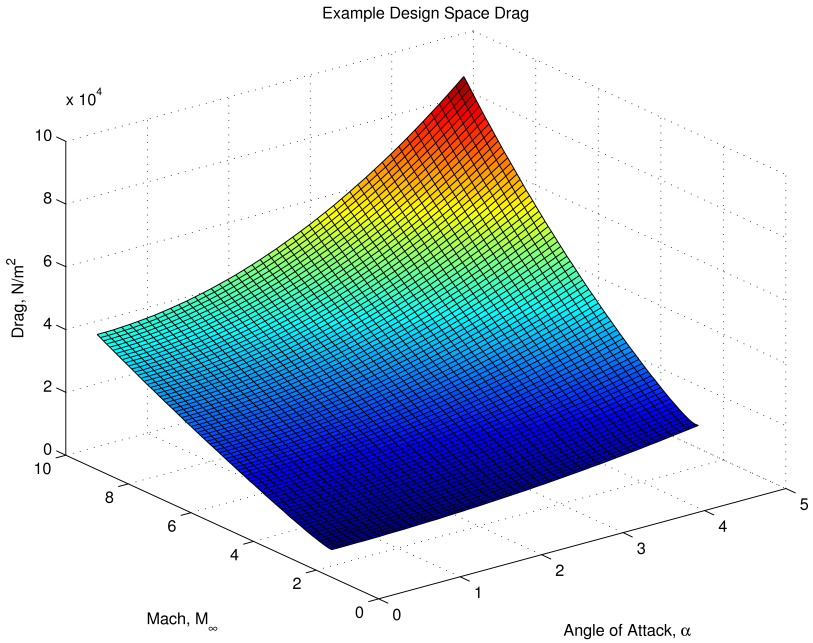


Figure 3.2: Example Two Dimensional Design Space for DA: Drag.

3.2 Diamond Airfoil Meshing

Three main iterations were performed on the grid generation process for the diamond airfoil. All of these iterations will be discussed briefly but the final process will be discussed within detail. The first two of these process was the generation of structured grids in first a C format and then secondly an H format using a hyperbolic algorithm. As mentioned within Section 2.6.2 all of the meshing performed was done with the Chimera grid tools suite and was output in the required plot3d format for input into VULCAN. For all three grid generation iterations the input format into Chimera was the same and thus will be explained following.

Figure 3.3 shows the standard geometry that was chosen for all of the meshing within Chimera. This wedge angle was chosen because it offered a reasonable turning angle for zero angle of attack conditions that allowed for debugging of the initial CFD results. Secondly, the chord length of the airfoil was chosen to standard units across the entire domain and also to reduce the Reynolds Number as seen by the airfoil which will be explained further within Section 3.6.

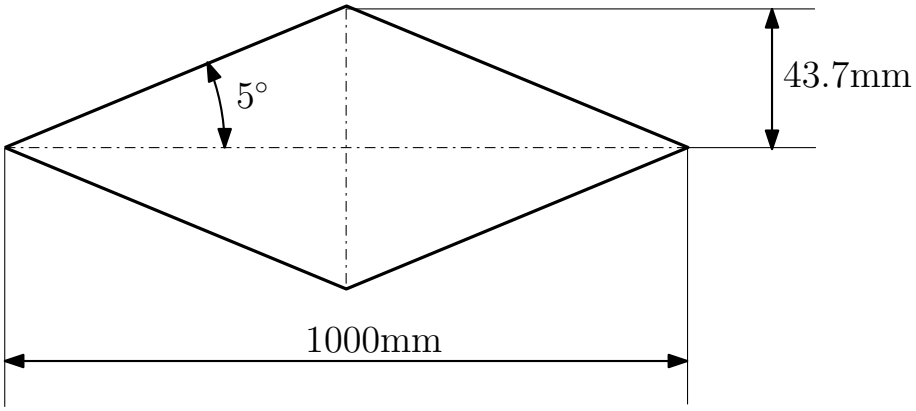


Figure 3.3: Sample Geometry Used for Meshing Trials and Grid Convergence.

3.2.1 Plot3D Input

Due to the financial cost of the proprietary CAD import function associated with Chimera, this function was not used to import geometry into the program. Instead, MATLAB algorithms were written that allowed for plot3d formatted files to be generated that matched the required multi-block format of multiple entities. For the diamond airfoil this required a minimum of four entity points to be generated that could then be processed using GRIDED and SRAP to generate the necessary points within Chimera.

To allow for batch processing of the grid generation within Chimera, the MATLAB function had to be directly called from the overlord automation generation script. This process call within the automation algorithm is outlined within Appendix E. Finally, it should be noted that the batch generation of grids was only necessary for changes in the wedge angle design parameter vector.

3.2.2 Meshing Design Iterations

HYPGEN C Grid

The first design iteration that was used was a hyperbolic algorithm to generate a C grid. This algorithm was used from the start as this allowed for a reduction in the number of generated points and also allowed for easy batch processing of AoA. Remeshing for varying AoA was later found to not be necessary but this process was still considered superior for remeshing in the θ dimension. The problem that was encountered with this mesh was the leading edge of the mesh was found to be creating instabilities from the mesh that was able to be produced from a finite point. Secondly, the ambiguity of the boundary conditions able to be placed on the mesh within VULCAN lead to further grid instabilities. From both of these reasons but more the numerical instabilities from the leading edge a different meshing approach was sought out. Figure 3.4 following shows the stretching ratio of C grid around the leading edge.

Short H Grid

From the C gird a short based H grid was produced. This grid being grown in the positive and negative y directions allowed for the finite point on the front of the airfoil to be accounted for. This type of meshing does include a larger C0 condition that had to be imposed on important part of the flow field but this was considered better than the previously induced numerical instabilities. The problem that was encountered with this type of grid was again numerical instabilities. The outflow boundary that as able to be set on the top and bottom boundaries created non-physical strong shock reflections that propagated back into the design space. Figure 3.5 shows a representative sample of the solved grid showing the shock refections within the space.

Tall H Grid

To remove the reflection condition from the short H grid the hyperbolic end condition was extended past such that the shocks exited on the back boundary condition. This removed the numerical reflection problem but also dramatically increased the number of points that had to be computed within the space as well be discussed in detail within Section 3.6. This grid was the final grid that was chosen with a representative sample shown in Figure 3.7. It has been noted that for the supersonic the lead into the airfoil could be drastically reduced but as this configuration was also sampled within a subsonic condition for debugging the inflow lead into the airfoil was left symmetric with the outflow. The pros and cons of this mesh decision will be discussed further within Section 3.5.

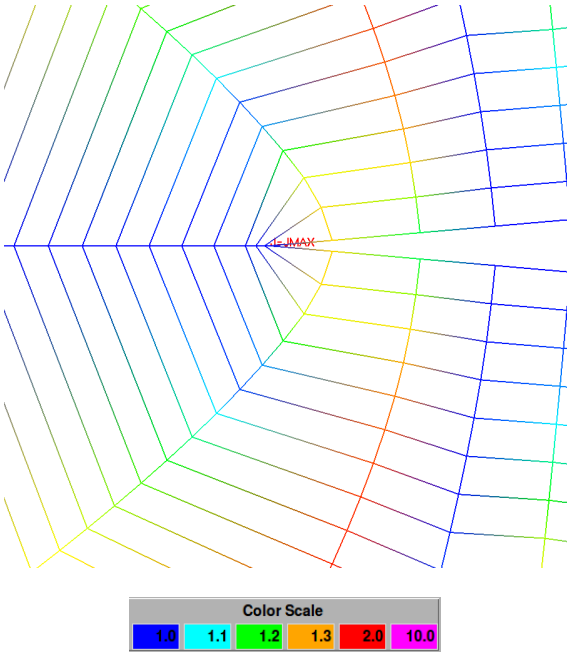


Figure 3.4: C Grid Leading Edge K Stretching Ratio.

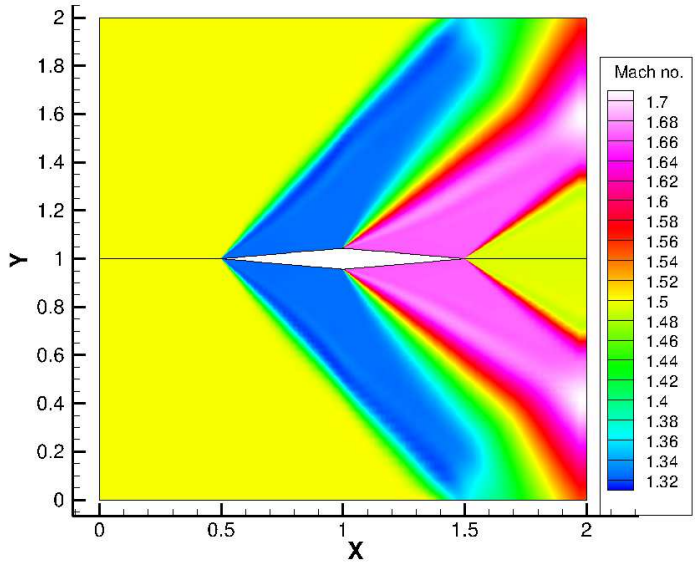


Figure 3.5: Non-Physical Shock Reflections in Grid Iteration.

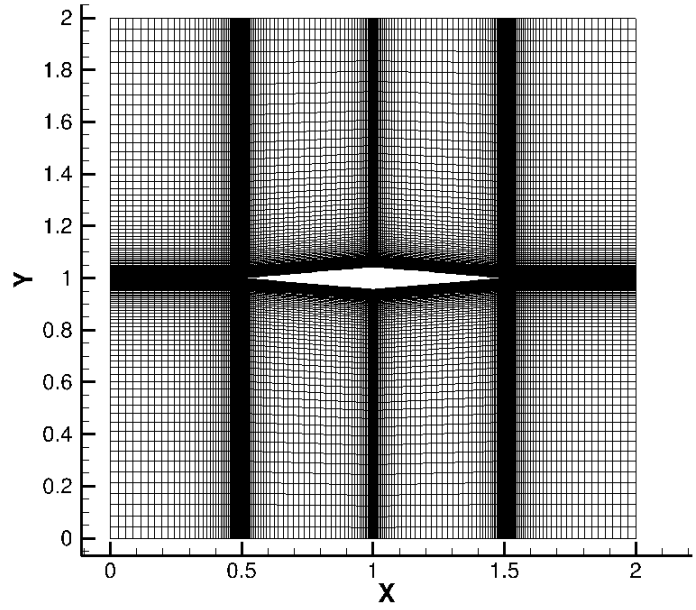


Figure 3.6: Mesh of the Short Grid Iteration.

3.2.3 Multi-Block Division

The use of multi-block division in the meshing of airfoil was chosen as the exact 50% split in the number of points per block gave a near predicted linear acceleration when using multiple cores for each instance of the solver. Also, the geometry represented by the non-meshed area within the center, Figure 3.3, made concatenation of the multiple blocks difficult. As mentioned in Section 3.2.2 this meant that multiple C0 conditions had to be placed along the block boundaries that did not correspond to the walls of the airfoil. This introduced more error into system as important flow features were generally concentrated around these block divisions. Patch conditions were considered but due to the need to have a sweep through the AoA parameter this was determined unfeasible. Also, the binary in the current form that are not self destructing, as set by the compiler, does not support patch conditions within the flow solver. Finally, smaller block divisions were considered but the divisions

3.2.4 Chimera Output Format

The output file format that was chosen from Chimera was the little endian multiple block unformatted double binary with no iblanking. This designation became important as the mesh grids used within the final grid designation as explained within Section 3.4 became very large. Also, as explained within Section E.5.2 the files had to be converted in binary form to be able to be read by the VULCAN environment. Even with the format designation the memory allocations had to be swept through for most of the file in order to extract the correct data for implementation into a new binary file.

3.3 Standard Conditions

Across all the optimization and convergence studies shown following a standard set of condition were adopted to be able to compare the output results. Firstly, this started with the geometry shown within Figure 3.3. Secondly, the following conditions displayed in Table 3.3 following show the remainder of the fluid conditions that were assumed including the parameters for Sutherland's Law that was used for the viscosity.

Table 3.3: Standard Fluid Conditions: Airfoil

Variable	Value	Units
Mach	1.5	
AoA	0.0	deg
Wedge Angle	5.0	deg
γ	1.4	
Static Temp	300	K
Gas Constant	286.9	$\frac{m^2}{s^2 K}$
Static Pressure	101325	Pa
S_0	1.1056E+2	K
T_0	2.7311E+2	K
μ_0	1.76160E−5	$\frac{kg}{ms}$
Laminar Prandtl	7.2E−1	
Turbulent Prandtl	0.9E−1	
Calorically Perfect Gas		

3.4 Airfoil Euler Grid Convergence

To make sure that the solutions produced were grid independent, a grid convergence study was performed. Due to the previously mentioned issue of grid quality, Section 3.2.2, the grids chosen did not include collated points around the shocks and were instead uniformly distributed across each of the four entities making up the airfoil. A representative sample of one of these grids, 65 x 129 points, used within the convergence study is shown below in Figure 3.7. Following this Figure is Table 3.4 which shows each of the grid sizes that were used within this study and also some of the meshing metrics associated with each of the girds.

Please refer to Appendix D for the figures depicting the aspect ratio (AR) and also the stretching ratio (SR) of the 65 x 129 grid. The SR figures within this appendix are considered typical for the entire study and will be referred to later for the final grid that was chosen for the Euler parameter sweeps.

Table 3.4: Grid Dimensions Used within the Euler Convergence Study

Grid Dimensions					
J Points	K Points	Min K Dimension (m)	Max J SR	Max K SR	Max AR
33	65	1E−1	1.05	1.03	1.73
65	129	1E−2	1.08	1.04	3.73
193	385	1E−2	1.08	1.01	2.48
321	641	1E−3	1.11	1.02	8.00
449	897	1E−3	1.11	1.01	8.49
577	1153	1E−3	1.11	1.00	8.95

Each of the grids shown within Table 3.4 were converged to 5 orders of magnitude using the space marching scheme for a 0 deg AoA and a 5 deg θ. The space marching scheme was used for this study as it had been determined through preliminary trials that this method was faster than the elliptical solvers within VULCAN. This is discussed and proved within Section 3.4.1 for the final grid chosen.

Table 3.5 shows the CFL conditions that were used with this study and also for all the space marched solutions in Section 3.4.1.1 and 3.4.1.2. These CFL conditions were kept constant to reduce the amount of variables within each of the studies and to make cross comparison between convergence conditions easier. Finally, Figure 3.8 and Table 3.6 show the results from this study for which the analytical solution was obtained from the DA program.

Table 3.5: Euler Convergence CFL Conditions: Space Marched

Metric	Starting Plane	End Plane
Start CFL	0.5	0.5
Plane Substeps	1	1
End CFL	5.0	0.5

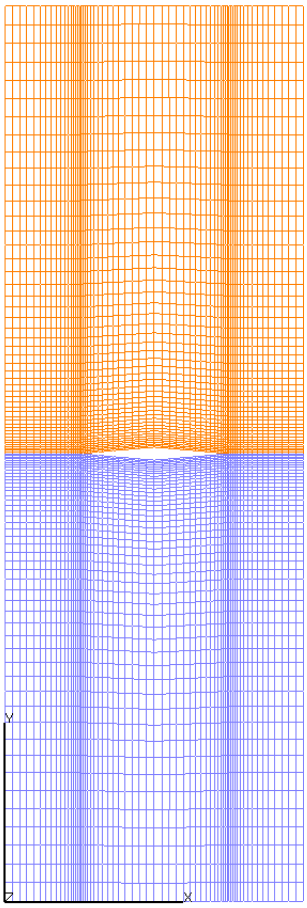


Figure 3.7: Example Grid Used within the Convergence Study.

Table 3.6: Five Orders of Magnitude Convergence Study: Euler

	Grid Dimensions						
	33_65	65_129	193_385	321_641	449_897	577_1153	Units
Drag Percent Error	1.75E+0	3.06E−1	1.68E−1	6.64E−2	4.83E−2	3.92E−2	%
Lift Abs Error	1.27E+1	2.22E−1	5.24E−4	2.85E−3	9.47E−4	1.00E−4	
Runtime	1.50E−1	3.33E−1	2.13E+0	5.92E+0	9.80E+0	1.75E+1	Mins

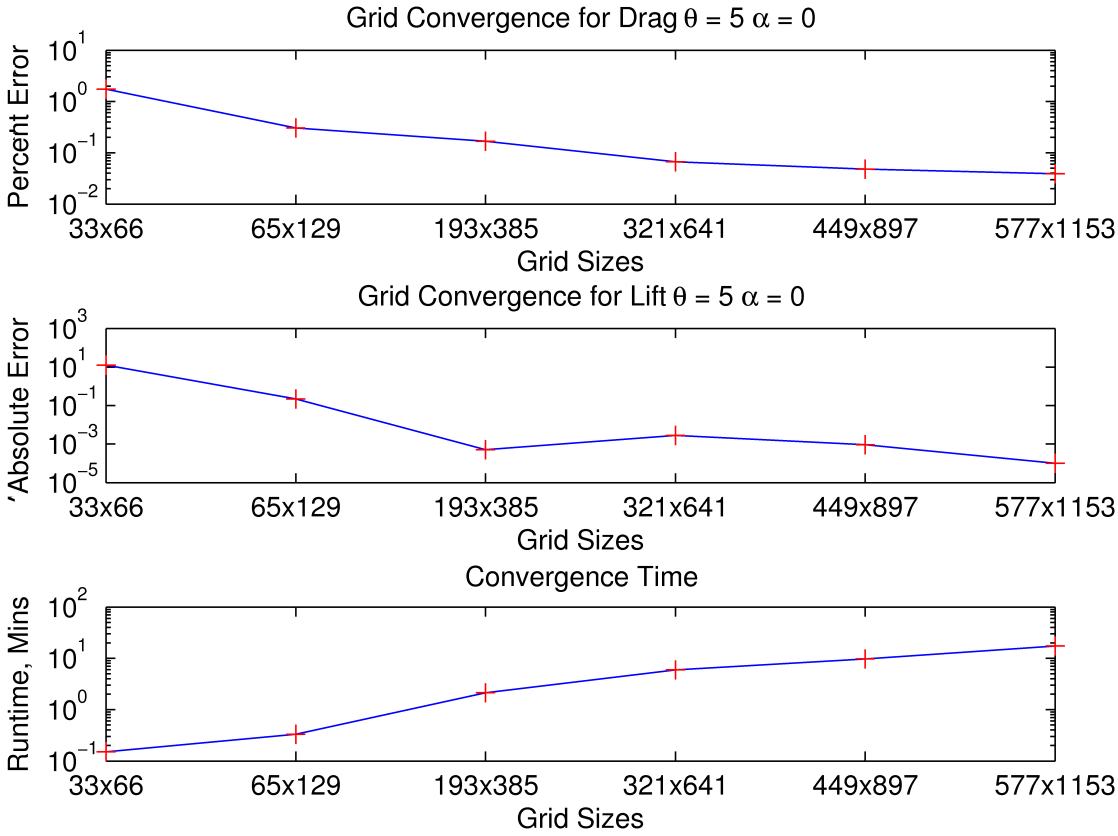


Figure 3.8: Five Orders of Magnitude Grid Convergence: Euler.

From the above convergence study, the final grid that was chosen was the 321 x 641 grid. This grid was chosen because of its relatively low runtime, drag error and also its ability to accept off parameter design from the conditions used within the study. That is, the movement of the shocks and expansion fans associated with changing the Mach and AoA of the airfoil will not affect the solution as much as they would for the 193 x 385 grid. Following in Figure 3.9 is a representation for AR of the final grid chosen. This figure is displayed to show the quality of the grid as the actual mesh is too fine to visually inspect. Figures D.1 and D.2 within Appendix D show a representative sample of the SR for the interested reader. As these values were almost constant through each of the grid sizes it was deemed not important to include them here for the final grid.

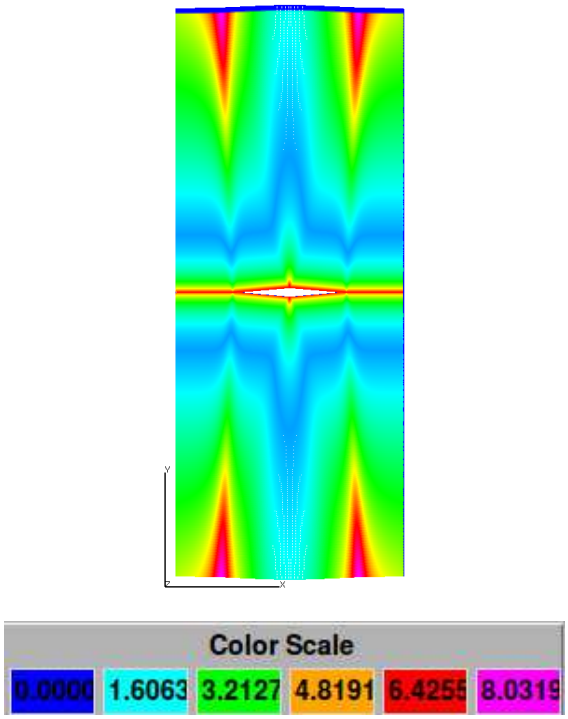


Figure 3.9: Aspect Ratio of the Final Grid Chosen: Euler.

3.4.1 Euler Point Solution Optimization

To help decrease the computational expense of populating the design spaces an order converge study as well as a solver optimization study were performed. The order convergence study was done to show that it was not necessary to have a large magnitudes of residual reduction from the maximum residual. Secondly, both the elliptical and space marching solvers were examined with different levels of multi-grid settings to find the fastest convergence time out of all the options. The solution methods that did not meet the residual convergence condition are believed to have done so because of the CFL standard conditions imposed on all of the solution methods. These solutions were rerun with an arbitrary max iteration count that was relatively low when compared to the first set of solutions. This work was done to verify that the solver was not converging to the exact solution faster than the other schemes even though the relative order convergence condition was not met.

3.4.1.1 Order Convergence Reduction

Below in Figure 3.10 and Table 3.7 are the results of the three orders of convergence grid study. It should be noted that the convergence trend was exactly the same as for the five order study as expected. The main point to note out of this study is the difference in runtimes and percent error between the two studies. Figure 3.11 following has been produced to highlight this point. From this figure it can be seen that the difference in error is on the order of 0.01% while the difference in the runtime is on the order of 10^0. Table 3.8 following shows the exact data points between the two levels of order of magnitude. This large computational saving for almost no accuracy loss meant that three orders of magnitude was chosen as the convergence condition within the parameter sweeps. Finally, it should be noted that the data point missing within the first subplot is due to an exact difference between the five order and three order solvers for this grid sizing.

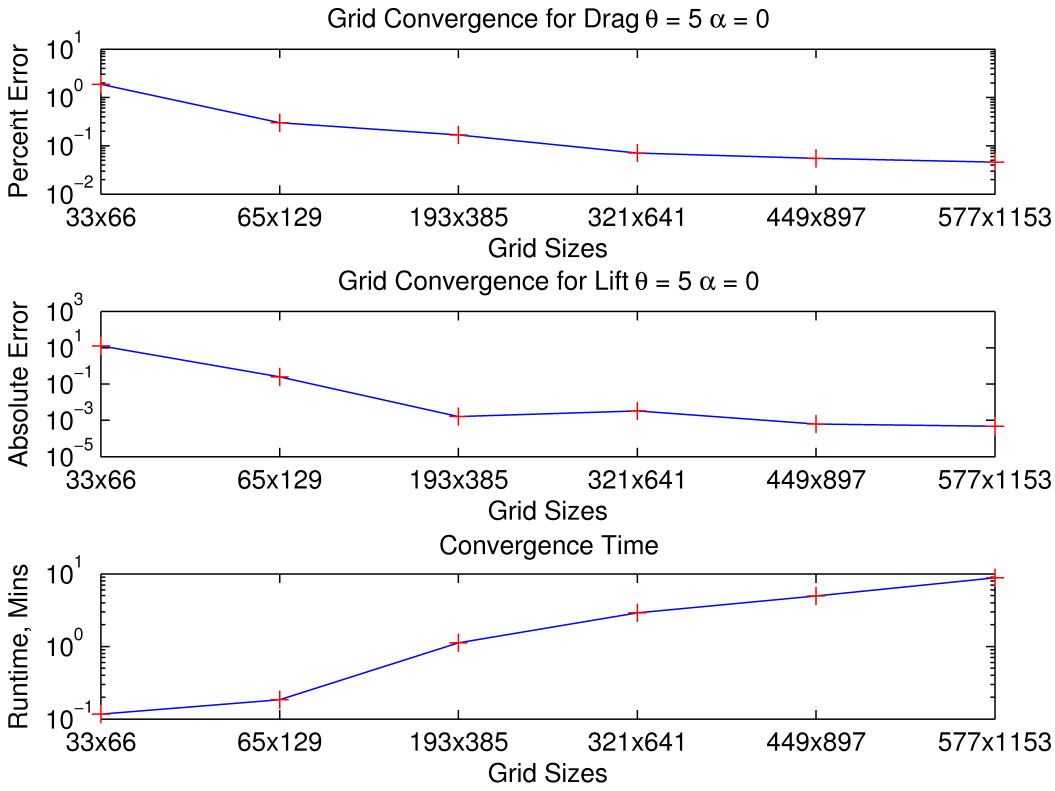


Figure 3.10: Three Orders of Magnitude Grid Convergence: Euler.

Table 3.7: Three Orders of Magnitude Convergence Study: Euler

	Grid Dimensions						
	33_65	65_129	193_385	321_641	449_897	577_1153	Units
Drag Percent Error	1.88E+0	3.00E−1	1.68E−1	7.09E−2	5.51E−2	4.60E−2	%
Lift Abs Error	1.26E+1	2.49E−1	1.60E−3	3.30E−3	6.18E−4	4.67E−4	
Runtime	1.17E−1	1.83E−1	1.12E+0	2.90E+0	4.97E+0	8.85E+0	Mins

Table 3.8: Order Reduction Comparison: Euler

	Grid Dimensions						
	33_65	65_129	193_385	321_641	449_897	577_1153	Units
Diff in Drag Error	1.31E−1	6.79E−3	0.00E+0	4.52E−3	6.79E−3	6.79E−3	%
Diff in Lift Error	1.06E−1	2.65E−2	1.07E−3	4.51E−4	3.29E−4	3.67E−4	
Diff in Runtime	3.33E−2	1.50E−2	1.02E+0	3.02E+0	4.83E+0	8.60E+0	Mins

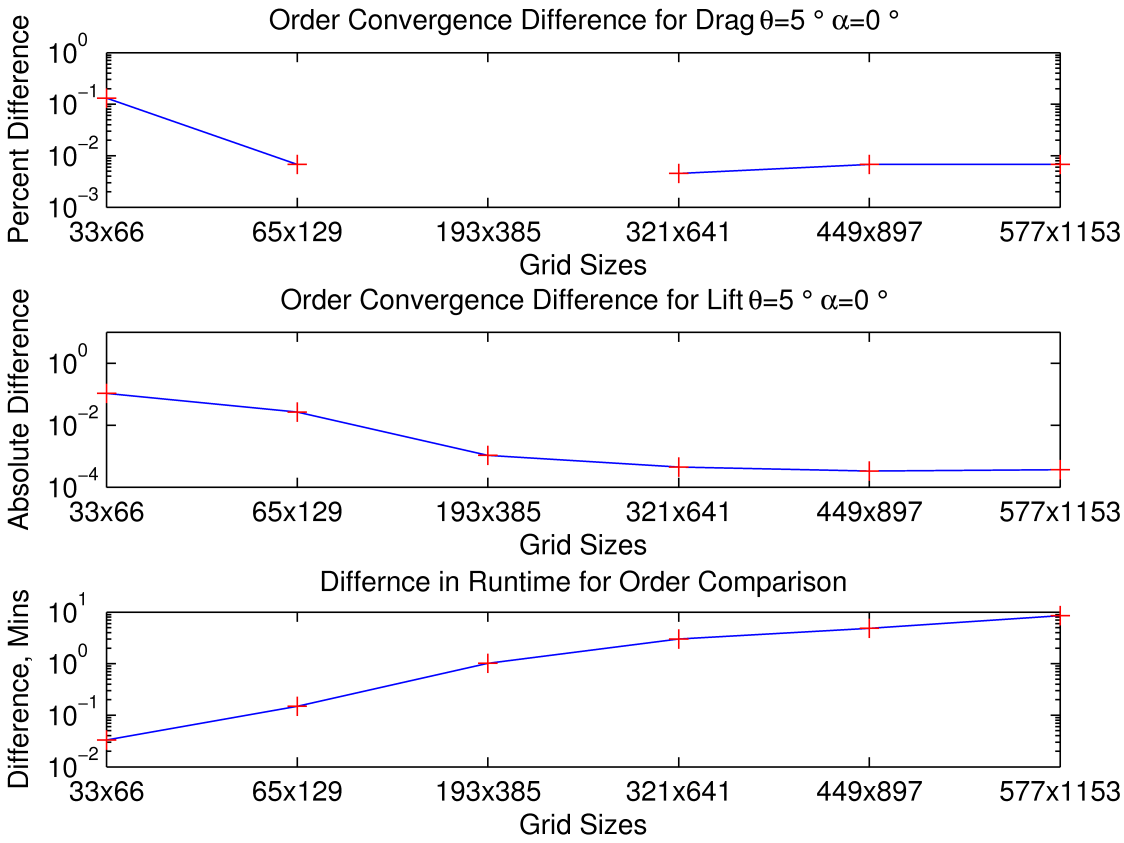


Figure 3.11: Order Reduction Comparison: Euler.

3.4.1.2 Solver Comparison

Following in Figure 3.12 and Table 3.10 are the results of the study to find the quickest solution method for the final Euler grid chosen. The names within the figure refer to the different multi-grid options that were available within VULCAN. These options were the V algorithm, the W algorithm and the non-updating layer option; I. All of these methods were compared using both an Euler solution for the spatial terms and a space marching technique to validate the work done in Section 3.4.

Table 3.5 in Section 3.4 shows the CFL conditions that were used for the space marching scheme while Table 3.11 and Table 3.9 below shows the conditions that were imposed on the serial and elliptically solved solutions. These conditions are important in that they are believed to be the culprit for the points within Figure 3.12 that were not converging properly.

Finally, a combination of the V multi-grid method with serial space marching was added into the study. This combination was given the name Space Multi Start and used multi-grids up to the 5 dimensional multi-grid plane of 17. This solver combination was added into the study as this format was used in a majority of the solutions within [37].

Table 3.9: Euler Convergence CFL Conditions: Serial

Iteration	CFL
1	0.1
10000	5.0

From Figure 3.12 it was observed that although the Euler I, Euler Serial, Euler V and Euler W solvers were not meeting residual reduction conditions they were, however, meeting the percent error condition. For this reason the study was rerun with the max iteration cap set at 200 iterations for each multi-grid level in the Euler solvers. This study was done to guarantee that the Euler solvers were in fact not converging faster than the space marching schemes. As predicted these results held true and even with the low multi-grid iteration cap the Euler solvers took on the order of 2.00E+2 times longer. The results of this comparison study can be seen following in Figure 3.13.

Table 3.10: Solver Runtime Comparison: Euler

	Convergence Order Reached	Runtime	Drag Percent Error
Euler I	3.01E+0	7.29E+1	5.96E−2
Euler Serial	2.86E+0	7.25E+1	5.96E−2
Euler V	3.01E+0	7.22E+1	5.96E−2
Euler W	3.01E+0	7.21E+1	5.51E−2
Space I	3.10E+0	6.00E−1	6.64E−2
Space Serial	3.04E+0	5.50E−1	3.33E−1
Space Multi Start	3.14E+0	1.45E+0	8.45E−2
Space V	3.14E+0	2.58E+0	7.09E−2
Space W	3.12E+0	6.17E+0	6.64E−2

Table 3.11: Euler Convergence CFL Conditions: Elliptical

Multigrid Level	Iteration	Elliptical CFL
1	1	0.1
	500	3.0
2	501	0.5
	1000	3.0
3	1001	0.5
	1500	5.0
4	1501	0.5
	2000	5.0
5	2001	0.5
	8000	5.0

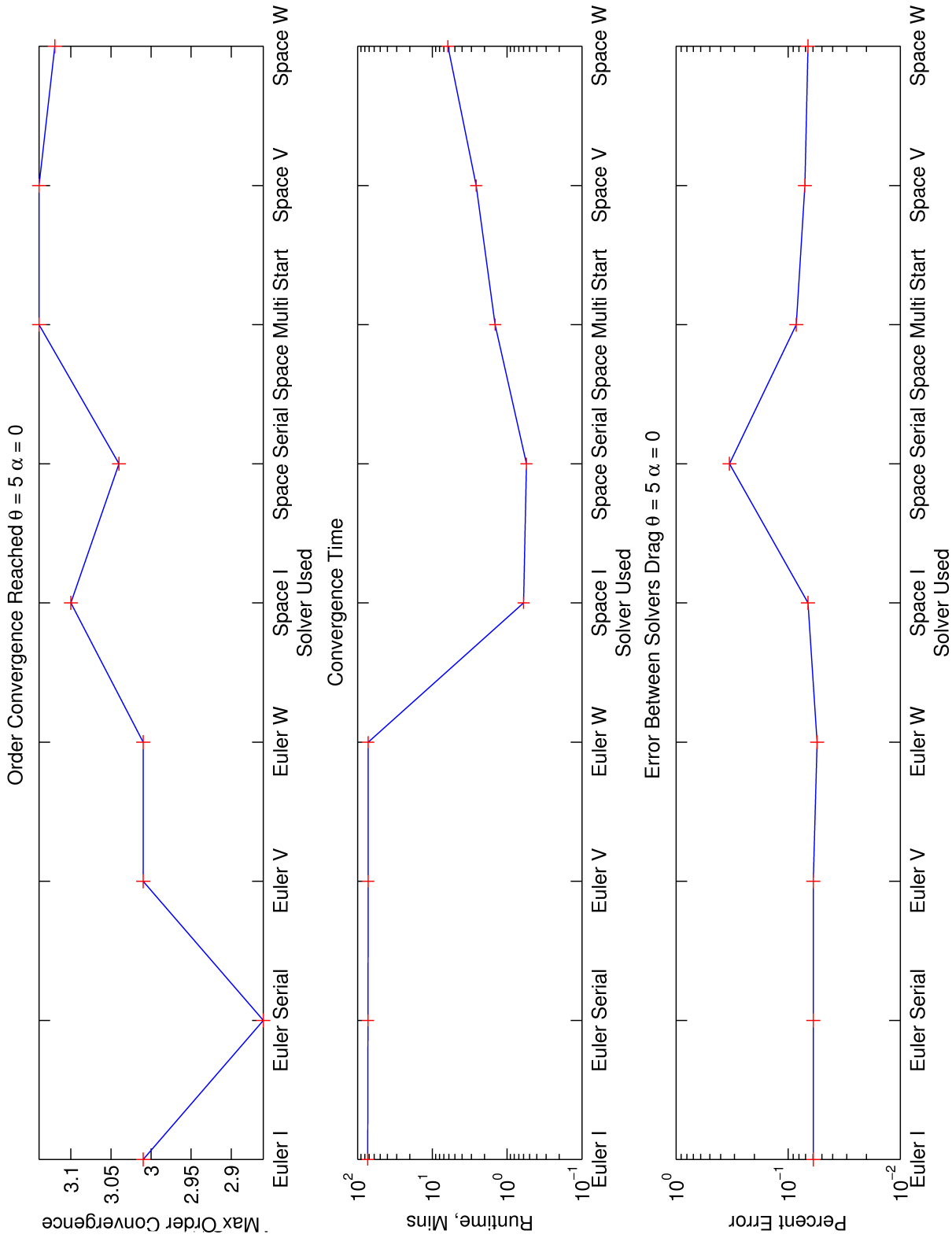


Figure 3.12: Solver Method Comparison: Euler.

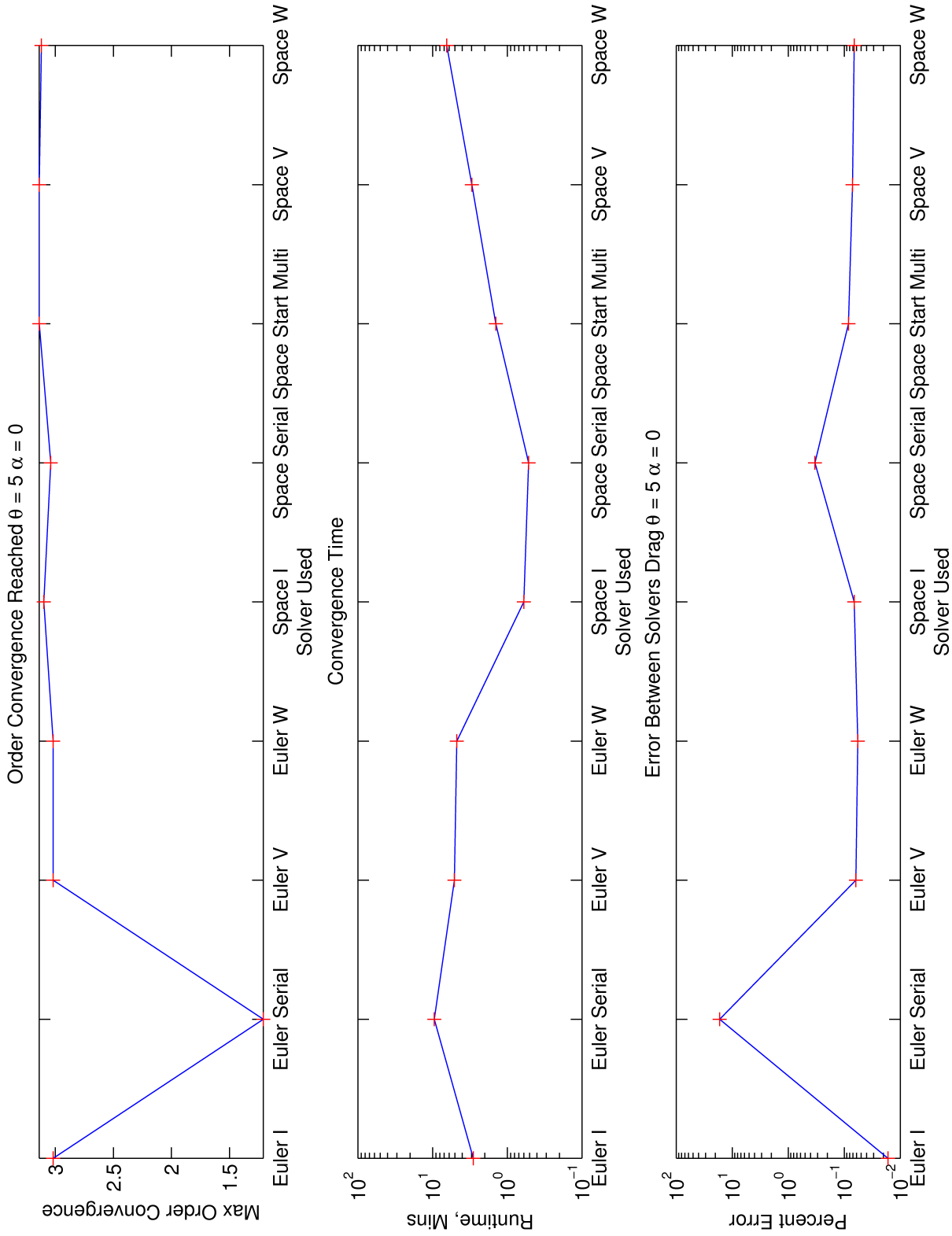


Figure 3.13: Solver Method Comparison with Iteration Cap: Euler.

3.5 Example Point Solution for Chosen Grid and Solver

A sample solution was chosen to be shown in full to give the reader an idea of what a final single design point looks like for this configuration. Following in Figure 3.14 is the contour of the Mach number as the flow propagates across this standard conditions configuration. Please refer to Table 3.3 for the definition of the standard conditions within this work.

Following the representative flow field solution is Figure 3.15, which shows the residual and order of convergence through each of the space marched planes. Within this figure it should be noted that the maximum residual seen from plane to plane never drastically increases, including across the shocks and expansion fans. Also, it should be noted that the order of convergence within the second subplot is almost zero leading up the location of the first shock within the solution. Although this appears that the solution is not converging in the first quarter of the planes within the system, this error actually occurs from the solver meeting all of the convergence criteria in a 1-2 iterations as there is no change in the flow field from the initial conditions. This lack of iterations meant that the residual could not be tracked properly within the solver and thus a false output was given for the convergence criteria. Due to this error it was propositioned that the geometry be changed to not include the lead-in meshing into the airfoil. After performing a cost benefit analysis the small decrease in computational expense when compared with having to change the standard mesh that was used within some subsonic calculations as well(debugging), was considered not worth it.

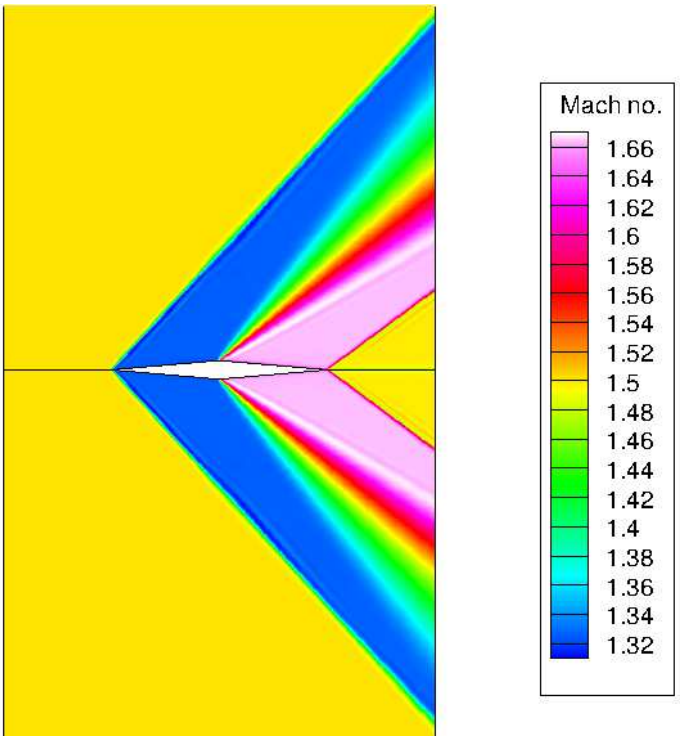


Figure 3.14: Single Solution Chosen Grid and Solver: Euler.

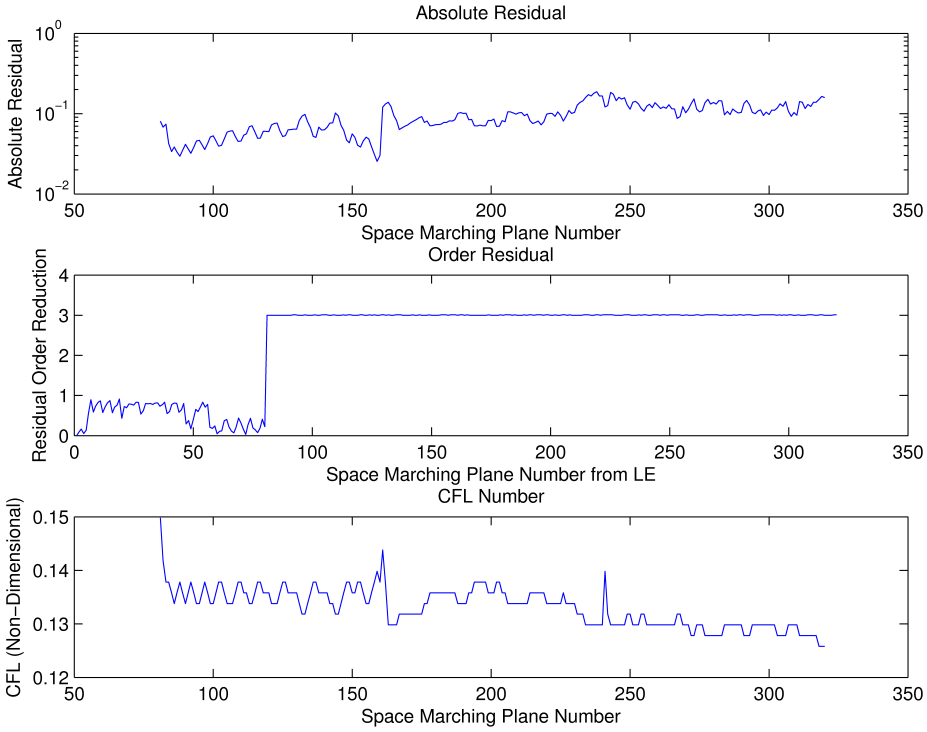


Figure 3.15: Residual Single Solution: Euler.

3.6 Airfoil Viscous Grid Convergence

As with the Euler grids within Section 3.4 a grid convergence study was run for the diamond airfoil geometry in the viscous case. This study was done again using the space marching algorithm as preliminary trials found that this solver method was much faster than the elliptical solvers. In fact this method was so many orders of magnitude faster, that a solver optimization was not conducted for the viscous solution at all. In context, producing elliptical results for the study would have taken on the order days to converge unlike the hours that it took for the space marching algorithm.

The second difference with the viscous grid convergence was that the difference in the amount of grid points used within each of the grids was a lot larger. This was done as the author was trying to decrease the y plus value within the solution down to an acceptable level and thus required large steps in the amount of grid points. The maximum y+ that was achieved as well as the other metrics associated with each of the grids within the study can be seen following within Table 3.12. The results can be seen within Figure 3.16 and Table 3.13 following this.

Table 3.12: Grid Convergence: N-S

Grid Dimensions						
J Points	K Points	Total Points	Min K Dim (m)	Max J SR	Max K SR	Max AR
577	1153	6.00E+5	1E−3	1.10	1.00	6.00
641	1281	8.00E+5	1E−4	1.10	1.01	31.5
1297	2913	3.78E+6	1E−4	1.13	1.00	16.8
3585	7809	28.0E+6	5E−5	1.14	1.00	72.7

Table 3.13: Order Reduction Comparison: Euler

	577_1153	641_1281	1297_2913	3585_7809	Units
Drag Error Euler	1.47E+0	7.30E−1	9.14E−1		%
Lift Error Euler	6.51E−1	4.43E−2	2.08E−2		
Runtime	2.64E−1	6.35E−1	1.94E+0	9.64E+0	Hours
Y Plus	7.21E+3	8.80E+2	8.59E+2		

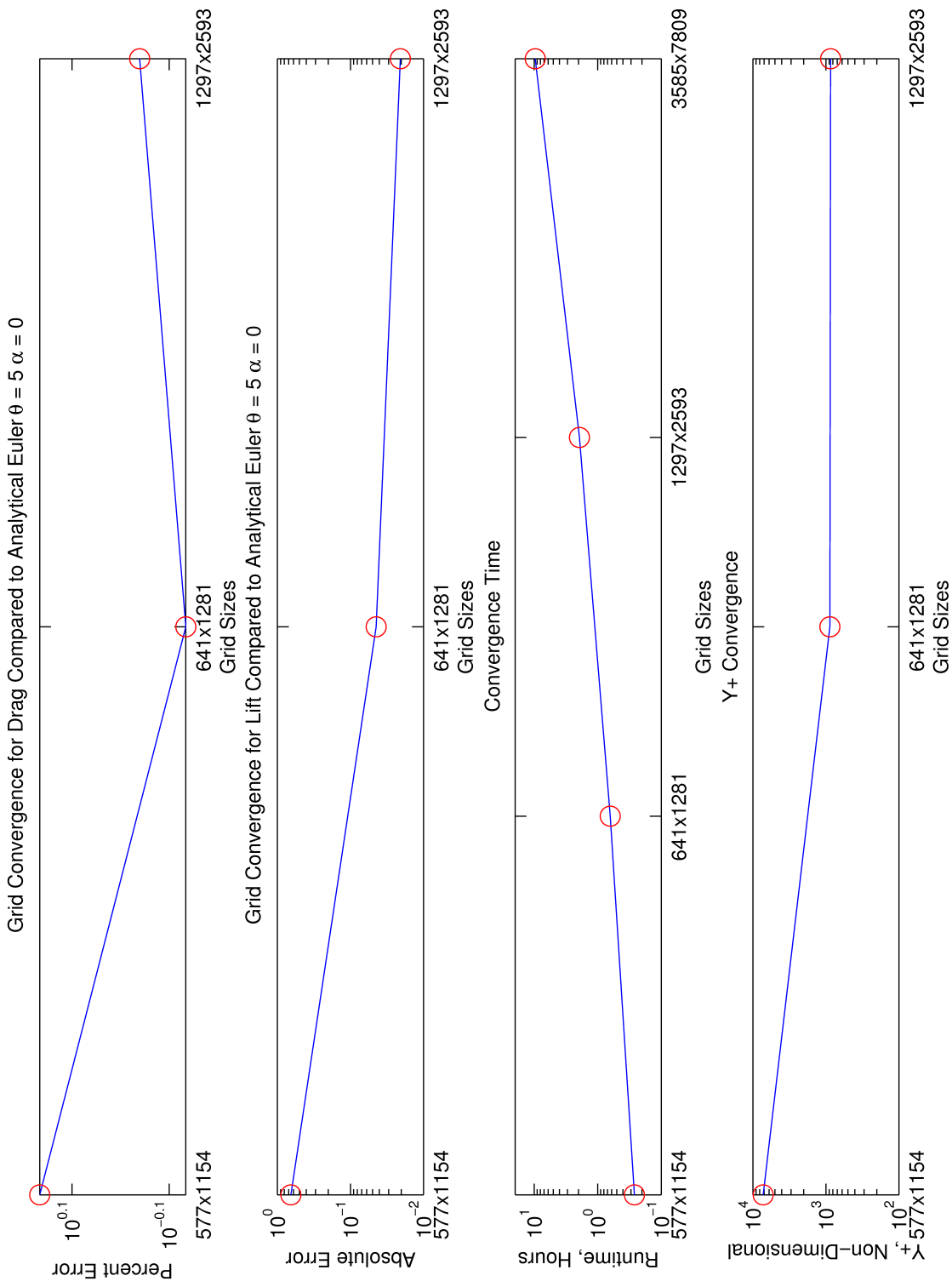


Figure 3.16: Gird Convergence: NS.

Table 3.12 has an additional data point when compared to most of the subplots within Figure 3.16. This data point was included to show the next grid size necessary to produce a half step down in the order of magnitude of meshing near the wall. This solution when ran diverged at the second shock wave present on the airfoil and thus the remaining metrics besides the runtime were not able to be included. It is believed that this grid diverged due to the large AR present at this plane. A quick examination was taken to see how many more points would be necessary to decrease the AR to a respectable level. From this examination it was determined that the order of magnitude increase on the already large grid was considered unfeasible. The 3585x7809 grid was in itself on the edge of the capabilities of the machines that the solutions were being processed on. That is the solver in the optimal space marching form took 10 hours to reach the second shock plane of 2689 with a grid file size of 1 GB. Thus, estimating a 525% increase in the non-binary output file as seen with the smaller grid sizes, each solution point would have consumed on the order of 6 GB of storage. This means that on the order of only 30 runs total, including debugging iterations, could have been performed before the available allocated hard drive space would be completely consumed. It is suggested that any person trying to repeat the results use a RAID 0 configuration to optimize the storage of such large files.

From the above study the 641x1281 grid was chosen. The reason this grid was chosen over 1297x2913 grid was that the increase in the accuracy of the solution did not increase enough to justify the extra 78 minuets that was required to converge the grid. That is the y plus did not increase substantially due to the meshing requirements that made both grids have the same minimum K dimension. Also, the change in absolute error for the lift in the airfoil only increased by a factor of 2 and not an order of magnitude. Normally, this accuracy increase would be considered a substantial increase but as the error for the 641x1281 grid was already at an order of magnitude of 10^{-2} it was not considered worth the extra runtime to reach convergence.

Secondly, it was found that the laminar sub blocks within VULCAN did not have to follow the multi-grid divisions. This mid-grid multigrid division switch to a turbulence layer within the marching scheme meant that a small amount of points could be used for the laminar sub block

instead of a full 5 layer multigrid coarsening as was originally used. In short this meant that the laminar sub block could be set to 4 grid points on the airfoil instead of 16 as the multigrid method used was a division by 2. For the test conditions used this sub block division resulted in a local Reynolds number of 4.4×10^5 which is right in the range of 3.5×10^5 to 1×10^6 as defined within reference [43].

Following in Figure 3.17 is the AR of the final grid chosen for the viscous solutions. It should be noted that the maximum occurs from the collocating of points due to the turn in the geometry. The first set of bad AR cells is assumed to not have any affect as the solver does not iterate on these cells like the rest of the lead in cells. The second set of bad AR cells impacts the trailing shock in the far field which is assumed to have a minimal impact on the wall boundary condition that is being used. These cells are the reason that the 3585x7809 grid did not converge as the maximum aspect ratio seen in these cells was 68.5, almost twice what was seen in final chosen grid for a maximum AR.

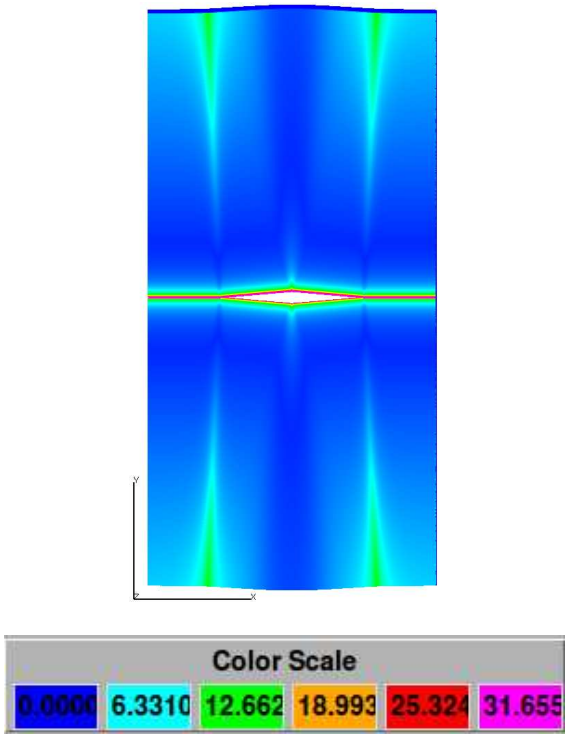


Figure 3.17: Aspect Ratio for the Chosen Viscous Diamond Grid.

3.7 Example Point Chosen for Viscous Runs

Similar to the Euler case, a single sample point solution is presented here to demonstrate what a single viscous solution looks like before any large parameter sweeps are presented. Table 3.14 shows the CFL conditions that were used for the viscous solutions. The reason there is a third point instead of the two as seen within the Euler single point solution, Section 3.5, is that VULCAN linearly increases the CFL condition from one point to the next as the solver propagates across the planes within the domain. This parameter within the solver meant that to have increasing CFL across the leading but nowhere else on the airfoil a third condition was implemented.

Table 3.14: Euler Convergence CFL Conditions: Space Marched

Metric	Starting Plane	End Leading Edge	End Plane
Start CFL	0.1	0.1	0.1
Plane Substeps	1	1	1
End CFL	0.5	3.0	3.0

Table 3.15 shows the remaining flow parameters that were not defined within the standard conditions shown in Section 3.3. The end point for the laminar subblock was chosen such that the laminar boundary layer Reynolds number assumption was imposed. For this standard condition solution point, the end condition for the laminar block calculated out to 4 grid points past the leading edge of the airfoil. The reason that all of the lead in blocks before this point were incorporated is that with no perturbation in the flow the in-flow conditions theoretically would not change and therefore remain laminar. Thus, by setting these lead in planes as laminar, VULCAN was prevented from calling the turbulence subroutines and the time to convergence for the lead in planes was dramatically decreased.

The choices made on the turbulence parameters within Table 3.15 were done so from educated guesses on what the end solution would resemble. There is currently no way of calculating out the exact number that should be used within these parameters but in general by over estimating and bracketing the optimal point these conditions will not cause instabilities within the solver. The

further from the maximum number actually experienced within the flow the longer the calculation which is why fairly low estimates were used for the turbulence percentage and also the viscosity ratio.[37]

Table 3.15: Flow Conditions Viscous Airfoil Single Point Solution

Parameter	Value
Subblock Planes	1-165
Turbulence Intensity %	1
Turbulence Viscosity Ratio	0.1

Figure 3.18 shows an overall visual representation of the solution which looks almost identical to Figure 3.14; as expected. From Figure 3.18 a zoomed in section on the leading edge of the airfoil is produced and shown in Figure 3.19 to show the grid resolution of the boundary layer on the airfoil.

Figure 3.20 is presented to show the convergence within the marching solution as it propagates through the chosen grid. As with the Euler single point solution, the solver does not appear to converge before the first shock wave. This is again due to the fact that the solver reaches an exact solution for each plane as there is no perturbations within the flow until the first shock wave is reached. Finally, Figure 3.21 is presented to show the variance of the y plus on the airfoil as a function of the plane number solved within the space marching algorithm.

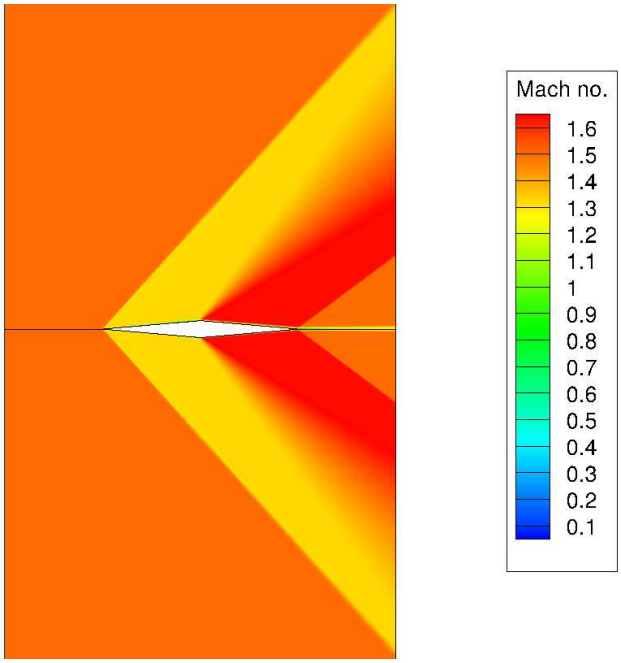


Figure 3.18: Single Point Solution Viscous Diamond Airfoil.

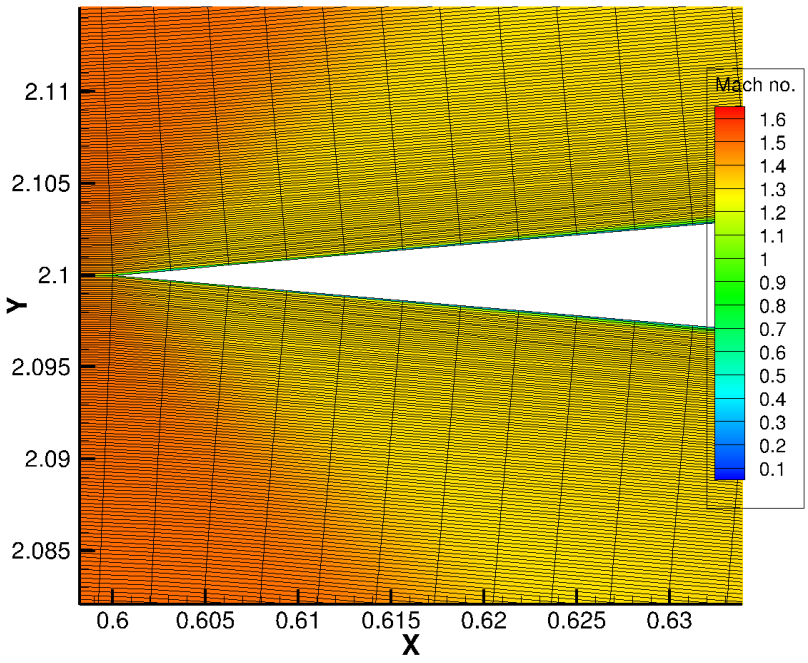


Figure 3.19: Boundary Layer in Single Point Solution for Viscous Airfoil.

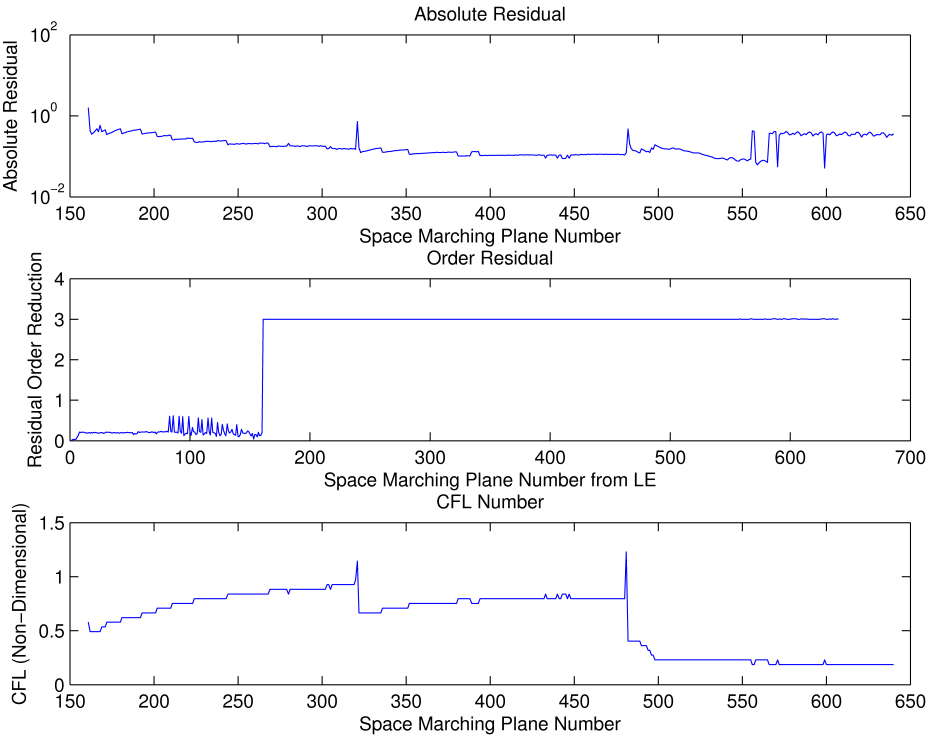


Figure 3.20: Plane Convergence for Single Point Solution Viscous Airfoil.

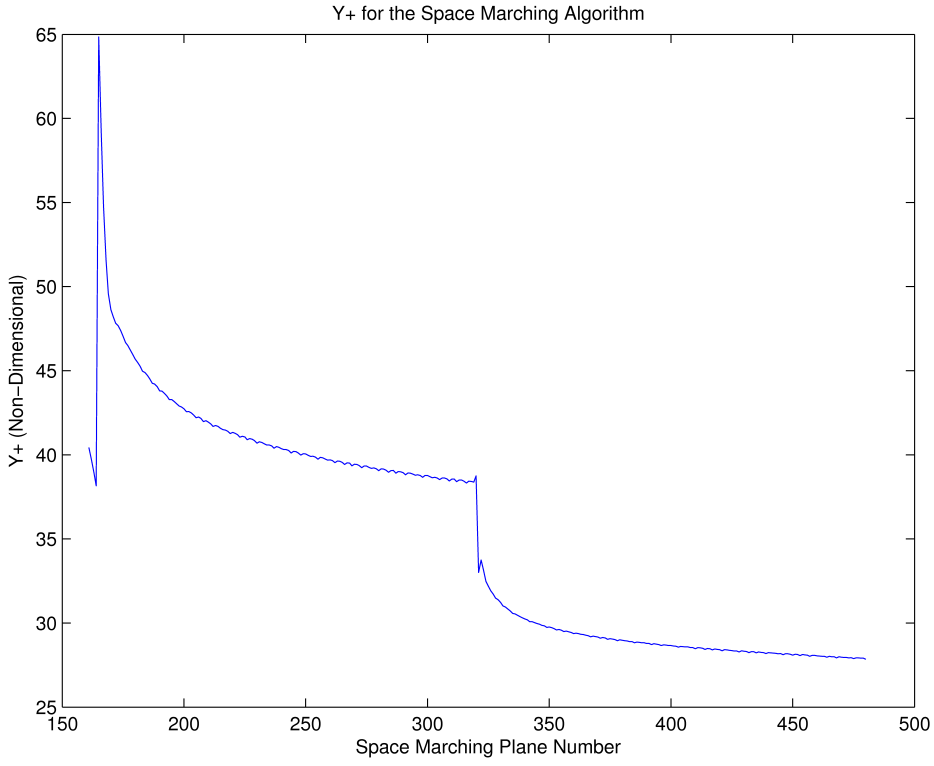


Figure 3.21: Y Plus over Space Marched Planes Single Point Solution.

3.8 One Dimensional Angle of Attack Sweep

The first multi-fidelity case that was run was a one dimensional parameter sample. The parameter that was chosen for this study was a positive AoA on the airfoil. The reason that this dimension was chosen over others was because of the pronounced difference between the visual CFD outputs for each of the parameter points. This visual difference made it easier to inspect individual solutions to verify that each of the solutions was converging as intended. A sample output from Tecplot Chorus showing the visual verification of the Euler run convergence and also the difference in the CFD solutions can be seen following in Figure 3.22.

Table 3.16 below shows the test conditions that were used within this one-dimensional study. The standard test conditions that were used are repeated here for completeness. Also, it should be noted that substepping was added into the laminar subblock on the airfoil to increase the accuracy of at this location.

Table 3.16: Test Conditions One Dimensional Multi-Fidelity Study

Solution Constants		Parameter Variables				
Constant	Value	Parameter	Start	Stop	Delta	Units
Solver Used	Space Marching	Altitude	0	0	0	m
Turbulence Model	k-ω	Mach	1.5	1.5	0	
Laminar Sub Steps	4	AoA	0	6	0.5	$^\circ$
Laminar Prandtl Number	0.72	θ	5	5	0	$^\circ$
Turbulent Prandtl Number	0.9					

3.8.1 One Dimensional Metrics Comparison

Figure 3.23 and Figure 3.24 show the variation of the lift to drag in each of the fidelity models respectively. Although the drag versus AoA has already been presented for the viscous case in Figure 3.26, this figure shows the difference between each of the fidelity levels and also the non-linearity within each model. Please note that analytical solution has not been included on these figures due to some errors within the model.

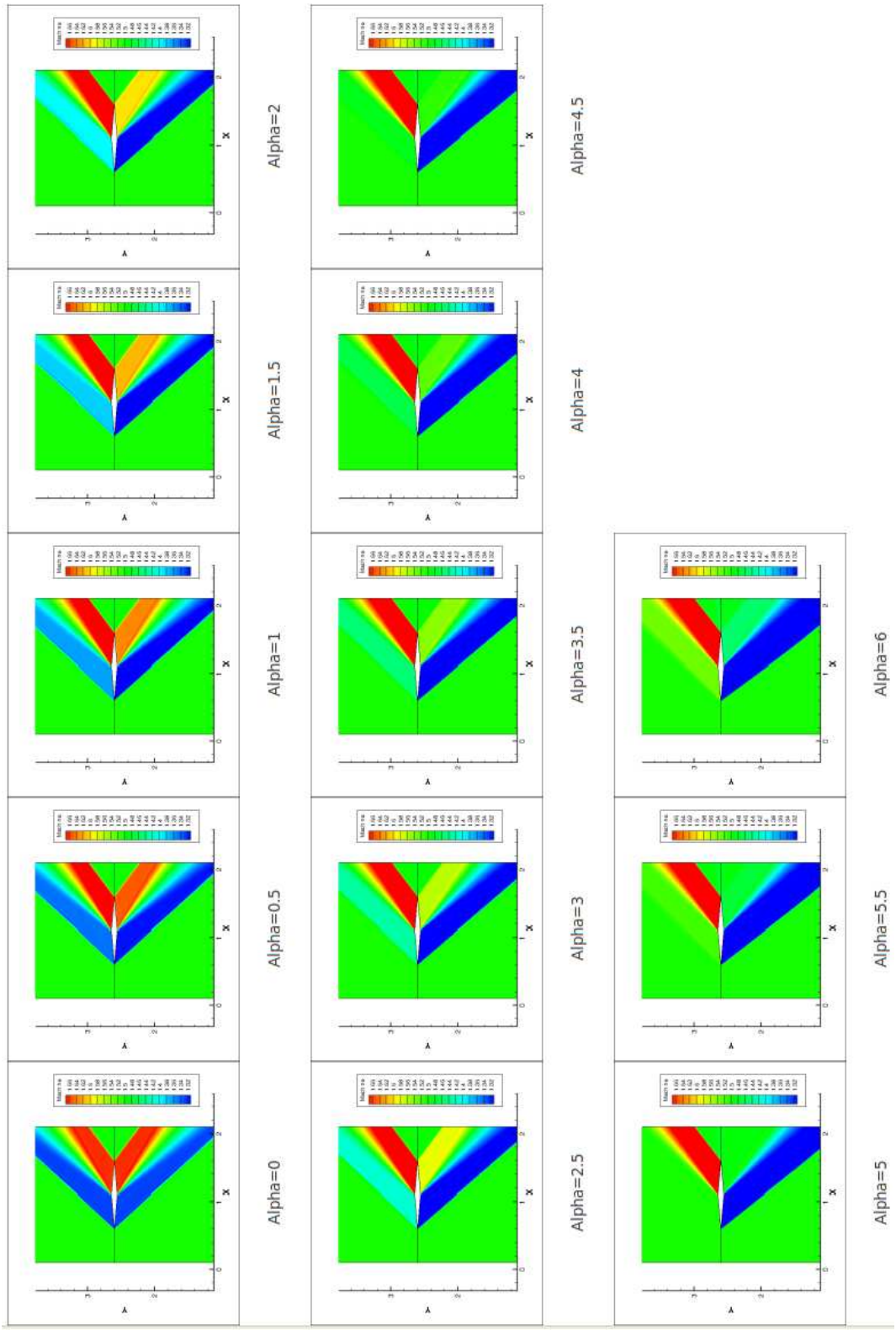


Figure 3.22: Sample Tecplot Chorus Output to Inspect Convergence.

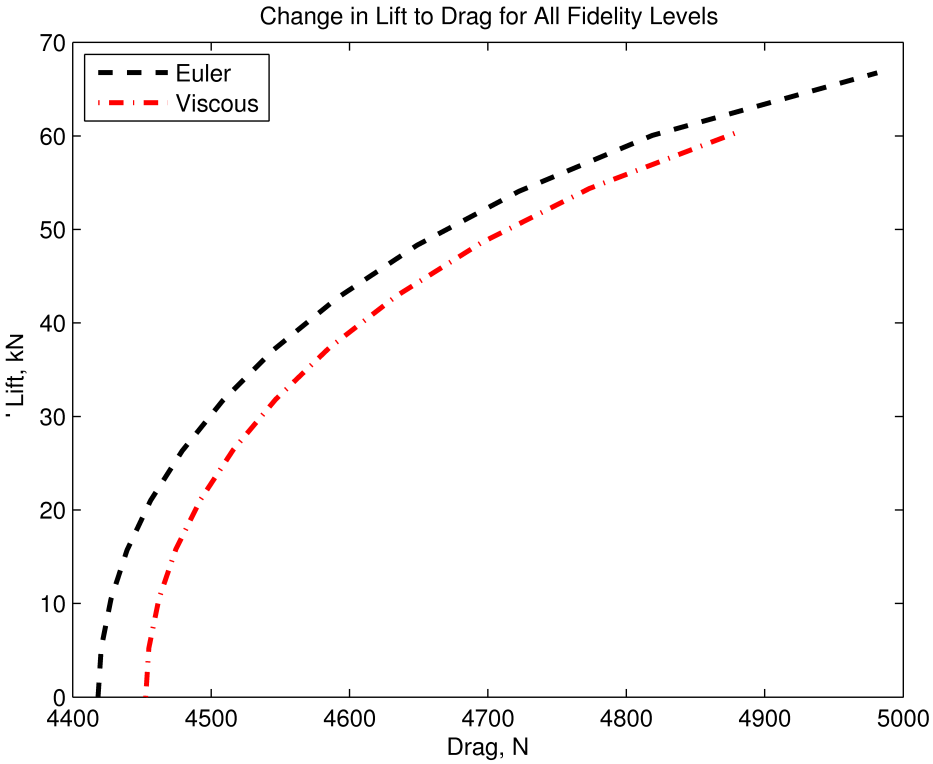


Figure 3.23: Lift as a Function of Drag All Fidelity Levels: AoA Single Dimension.

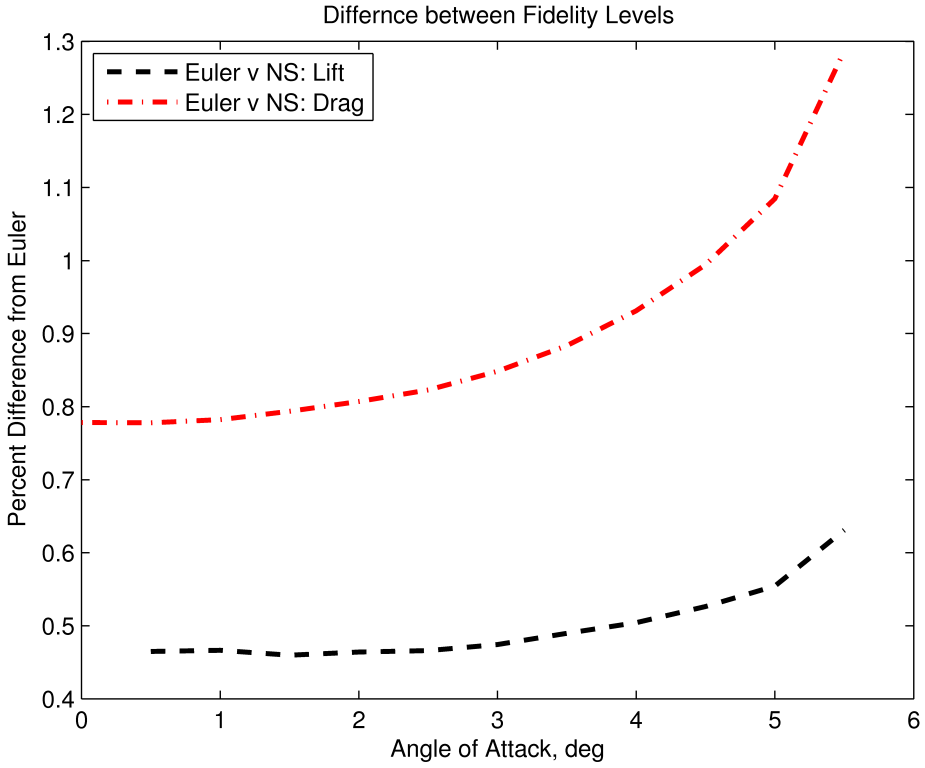


Figure 3.24: Percent Difference between Models: AoA Single Dimension.

Figure 3.25 shows a comparison between the Euler CFD results and the analytical calculations. The first subplot of the figure shows the large discrepancy experienced for the drag. As no bugs were able to found within the code and also because the lift calculations were so close, this error was considered another approximation to be accounted for between the fidelity levels. Finally, it should be noted that the divergence increases with angle of attack and that at zero AoA the two results are almost identical which was the condition used for the initial CFD validation.

Having verified the non-linear nature of the dimensional space, a Kriging approximation was conducted on the one-dimensional sample taken at the viscous N-S level. This approximation can be seen in Figure 3.26 and was produced to give a representative sample of the Kriging method when applied to in-house produced results.

Table 3.17 shows the total PRESS error for each of the Kriging models that were produced. It should be noted how the analytical value for this total error is multiple orders of magnitude higher than the other two models. This drastic difference is mainly due to the zero lift condition at zero angle of attack being enforced in the analytical solver which when removed creates a large amount of error in the PRESS calculation.

Table 3.17 also shows individual PRESS calculations for three varying off-parameter points. These points were checked because the Kriging model did not have to be coarsened to calculate the PRESS error on these points. Thus, these individual PRESS errors presented an accuracy metric for the Kriging models being produced and also verified the built-in PRESS calculations in DAKOTA.

Table 3.17: Press Error in AoA Kriging Models

	Euler		N-S		Analytical	
	Lift	Drag	Lift	Drag	Lift	Drag
Original Model	95.59	7.327	617.42	17.76	2.36E+4	182.7
0.25°	0.4489	0.0012	2.7556	7.84E−4	122.6	0.1726
3.10°	0.0225	1.96E−4	0.4900	4.90E−5	6.372	0.0233
4.75°	0.0225	0.0077	1.7689	0.0331	1.21E+3	2.6122

Table 3.18 and Table 3.19 show the percentage error for the original low-fidelity approximation

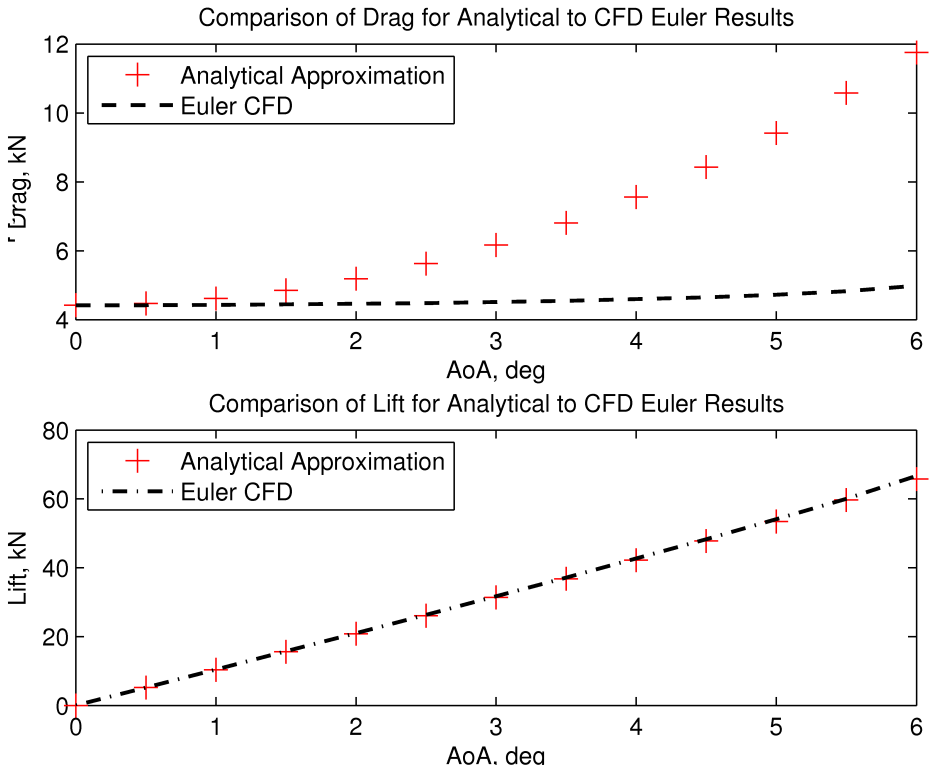


Figure 3.25: AoA Euler Analytical Comparison.

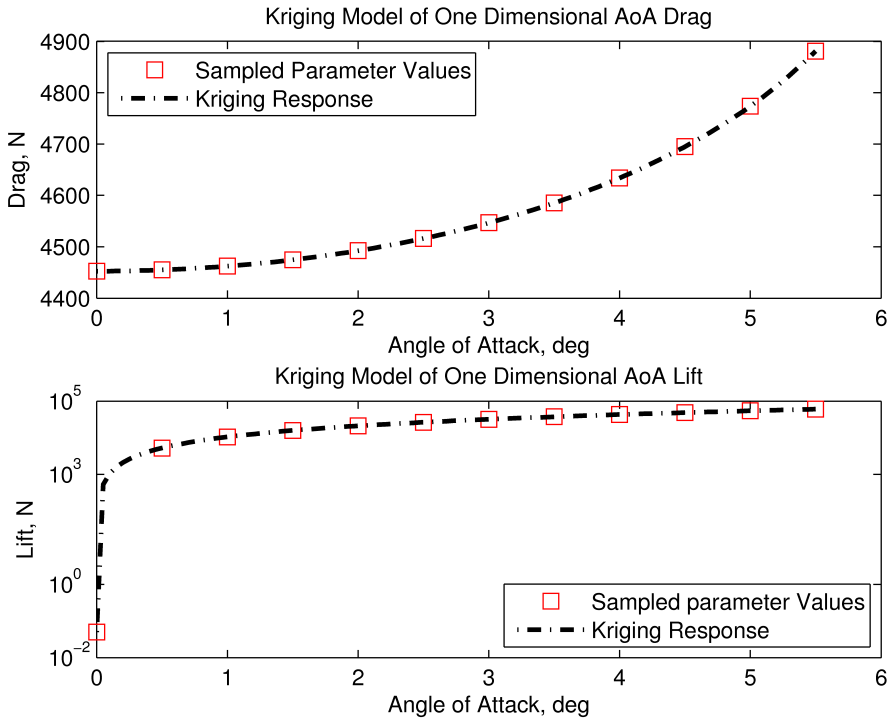


Figure 3.26: Kriging Approximation to Single AoA Dimension Viscous Airfoil.

and the error percentage after the correction is applied. As mentioned the benign nature of this design space means that the approximation is almost exact after the correction. The results are still promising that in general an order of magnitude improvement is seen from the original low fidelity approximation computed.

Finally, in both Table 3.18 and Table 3.19 all three of the off-parameter points computed in in Table 3.17 were examined instead of just a single point. These multiple calculations were done to show that low-fidelity correction is indeed dependent upon the independent variables within the design space as is expected. Secondly, the variance of the correction within this design space can be examined and seen to be a relatively stable dependent result.

Table 3.18: Lift Percentage Error For Uncorrected and Corrected Off-Parameter Points

		Euler			Analytical		
	Viscous	Value	Orig %	Corr %	Value	Orig %	Corr %
0.25°	2.626E+3	2.614E+3	0.4641	0.0377	2.594E+3	1.227	0.3584
3.10°	3.294E+4	3.278E+4	0.4758	0.0017	3.248E+4	1.4079	0.0098
4.75°	5.144E+4	5.116E+4	0.5393	0.0029	5.057E+4	1.6802	0.0651

Table 3.19: Drag Percentage Error For Uncorrected and Corrected Off-Parameter Points

		Euler			Analytical		
	Viscous	Value	Orig %	Corr %	Value	Orig %	Corr %
0.25°	4.453E+3	4.419E+3	0.7702	0.0014	4.433E+3	0.4499	0.0100
3.10°	4.554E+3	4.515E+3	0.8468	0.0002	6.283E+3	37.9771	0.0032
4.75°	4.732E+3	4.683E+3	1.0261	0.0057	8.903E+3	88.1619	0.0303

To prove the computational savings from the low-fidelity models used in the correction calculations, Table 3.20 below shows the difference in the runtimes between the fidelity levels. Please note that the analytical solution was not tracked specifically as the intrinsic time models incorporated in gfortran were not able to produce the resolution necessary. Therefore, the minimum resolution is displayed instead with a relative sense of how much the computation was under this value.

Table 3.20: Difference In Computation Time

Fidelity Level	Average Single Runtime	Total Serial Runtime
Euler	3.56	43.6
N-S	42.7	524
DA	$\ll 0.0167$	< 0.0167

3.8.2 Multi-Spectral Wavespace Example

A separate dense AoA parameter space was sampled for comparison to show the accuracy of sparse sampling data sets and also to show the non uni-modal frequency dependence as mentioned within Section 2.4. The sampling frequency that was chosen was 20 Hz because from examination of Figure 3.26 it was seen that the frequencies present in the system where not very high. Thus, sampling at 20 Hz allowed for a Nyquist frequency of 10 Hz which was assumed to be high enough to bracket a majority of the wavespace domain and not induce aliasing.

The transformation algorithm that was used in this work was the Fast Fourier Transform (FFT) with the preparation section taken from reference [44]. This algorithm does not support local adaption or compact support but for the functions being estimated on this geometry, this was not considered a large error.[44] This error does becomes more prominent in Chapter 4 but again global wavespace is seen as a good assumption and not considered a substantial error.

A representative sample of the wavespace computed from the Euler drag data in the dense sampled space is shown in the first subplot in Figure 3.27. This figure is shown to display the dominant frequencies present within the system. As expected the major contributor is close to the zero frequency. This lack of oscillation within the system is due to benign nature of the parameter space sampled. More importantly, the second subplot within Figure 3.27 shows the normalized power difference between the data wavespace and the Kriging surrogate model wavespace. This figure demonstrates that the Kriging model was able to capture the dominant frequencies within the data sets being analyzed which will become important for the data analyzed in Section 4.

To show that the sparse sampling of the domain did not greatly impact the final surrogate

results, an absolute difference was taken between the two sample sets for both the lift and drag metrics. These results are displayed in Figure 3.28 while the PRESS error calculated for the dense sampled Kriging model is shown in Table 3.21. In Figure 3.28 it should be noted that near the boundaries of the parameter space is where the largest difference was seen between the two models. Even with this divergence the absolute percentage difference was still under 0.1% and thus this was not considered an issue even though the PRESS error changed so drastically from these end divergence conditions.

Table 3.21: Press Error in AoA Dense Sampled Kriging Models

	Euler		N-S	
	PRESS	% Diff Sparse	PRESS	% Diff Sparse
Lift	24.77	74	57.58	91
Drag	7.292	0.48	0.7756	96

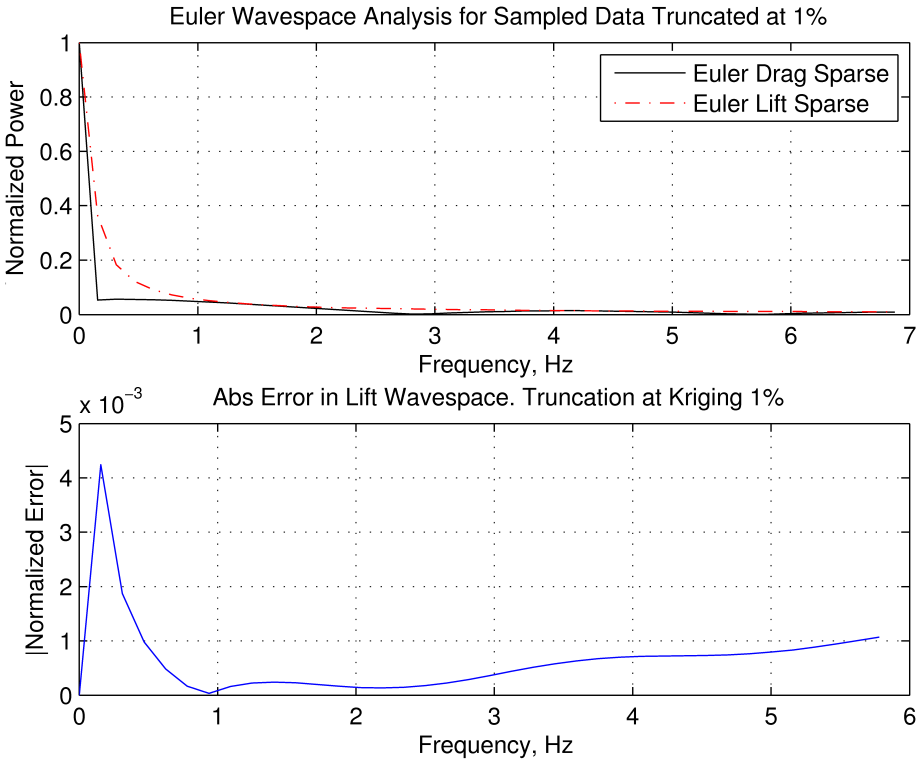


Figure 3.27: Dense Sampled AoA Wavespace Example.

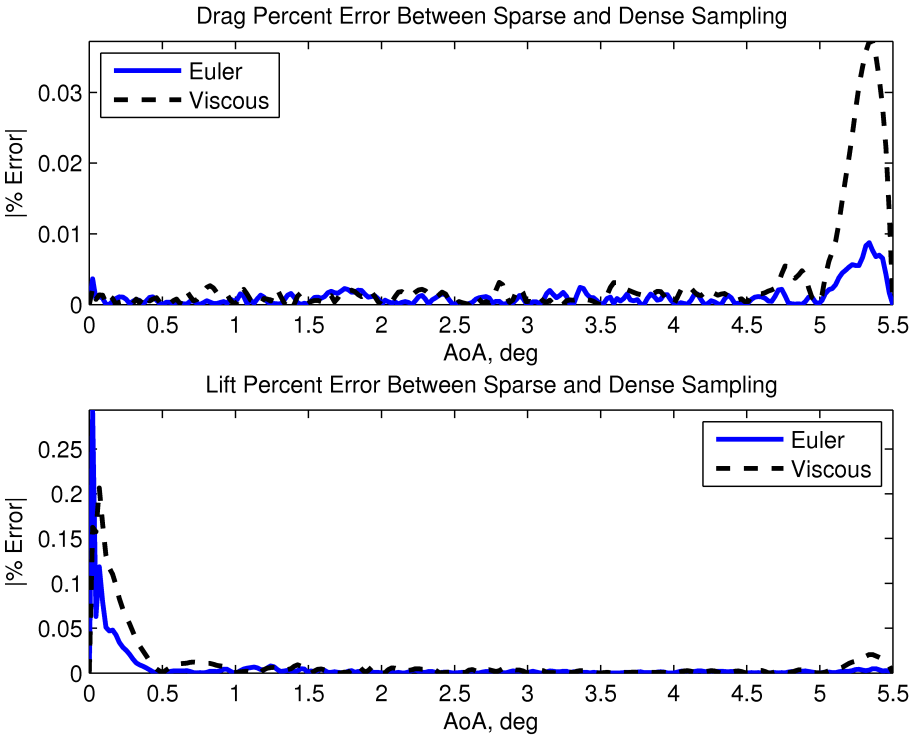


Figure 3.28: Difference in Sparse and Dense Sampled Kriging Models.

3.9 Four Parameter Diamond Airfoil Space

From the successful application of the method to a single AoA space, a four dimensional parameter space that varied the Mach number, wedge angle, altitude and AoA was created. Table 3.22 below shows the space bracketing and also the full factorial sampling that was used within this space. It should be noted that the altitude calculations used the 1976 standard atmosphere to define the pressure, density and temperature inflow conditions. This altitude primitive variable analysis is explained further in Appendix E.

Table 3.22: Four Dimension Airfoil Parameter Sampling

	Min	End	Delta	Units
Altitude	0000	20000	10000	m
Mach	1.5	5.5	2.0	
Theta	0.0	6.0	2.0	deg
AoA	0.0	6.0	2.0	deg

Table 3.23 shows the PRESS error that was associated with original sampled points. The Euler error is by far the worst out of all the values with this large difference again being caused by the divergence of the metrics at the space boundaries.

Table 3.23: Four Dimension Kriging PRESS Error

	Euler		Viscous	
	Lift	Drag	Lift	Drag
PRESS	1.24E+7	9626	7999.63	3694.15

From the calculated Kriging models new individual off-parameter points were computed and verified against the continuous spectrum produced. Initial results showed that the space was not densely sampled enough to get continuous converged solutions throughout the entire domain. This meant that the interpolated solution diverged between the sampled points and due to the small sample size taken in each dimension this meant that the interpolated results were worse than the original error in some cases. Due to computing restrictions a new space was unable to be calculated

so instead the divergence condition was approached from a sampled point in the domain until the interpolated Kriging results at the Euler fidelity level diverged by more than 1%.

Table 3.24 shows the results of this divergence study in two of the four sampled dimensions. Only two dimensions were varied as the point of this study was to show that the method could be applied to multiple dimensions and only using two dimensions allowed for a larger off-parameter distance in each individual dimension to be analyzed before the Kriging model diverged.

Table 3.24: Dimension Values for Off-Parameter Points

	Point One	Point Two	Point Three	Point Four	Point Five
Altitude	0000	0000	0000	0000	0000
Mach	3.5	3.5	3.5	3.5	3.5
Theta	2.0	2.1	2.5	2.7	3.0
AoA	2.1	2.1	2.5	2.7	3.0
% Divergence Drag	0.3579	0.2578	0.3528	1.0276	1.8543
% Divergence Lift	0.2744	0.3138	0.3765	0.0166	0.7099

From Table 3.24 the off-parameter Point Two and Three were chosen to be analyzed at the viscous fidelity level. Table 3.25 and Table 3.26 following show the results where the original error between the two fidelity levels as well as the corrected percentage from the Kriging models are shown.

Table 3.25: Corrected Drag Results: Four Dimensional Airfoil

	Viscous	Euler	Orig %	Corr %
Point Two	1.43E+3	1.41E+3	1.33	0.340
Point Three	2.03E+3	2.00E+3	1.10	0.0551

Table 3.26: Corrected Lift Results: Four Dimensional Airfoil

	Viscous	Euler	Orig %	Corr %
Point Two	3.89E+4	3.84E+4	1.29	0.3052
Point Three	4.65E+4	4.59E+4	1.29	0.3527

Chapter 4

Scramjet Combustor

The geometry that was investigated was a backward step cavity scramjet combustor. The dimensions chosen for this sample design space geometry were referenced off other researcher's work to provide a validation tool as there is no analytical solution to this problem unlike the previous geometry of the diamond airfoil. Also, the main difference in the design spaces investigated for this geometry is, that instead of physical parameters for the design sweeps, empirically defined coefficients within the chemistry models were varied. This work was done as a preliminary analysis to an uncertainty quantification on the design space designated by varying the reacting chemical coefficients with the CFD solver.

4.1 Meshing of the Scramjet Combustor

The scramjet combustor that was chosen to be referenced against for the final geometry produced came from reference [45]. This paper examined the impact that varying inflow conditions had on the combustor efficiency. One of the conditions that was investigated was uniform flow which was used as the baseline property for comparison within the paper. As no geometry conditions were given on the inlets that were used, the primitive variable associated with the uniform inflow conditions were adopted for validation purposes.

Originally, it had been planned to use the exact geometry depicted within reference [45] but due to Chimera not being able to perform the necessary functions to replicate the design and also time restrictions this became unfeasible. Instead a predetermined cavity mesh incorporated with

the VULCAN package was altered to give a geometry similar to that within reference [45]. The main differences between the two meshes was the distance and turning angle associated with the cavity ramp as well the height to the symmetry condition.

The full two dimensional projection of the cavity from reference [45] is shown in Figure 4.1. The actual geometry used within the calculations is shown within Figure 4.2 and highlights the dimensional differences mentioned. Because the exact dimensions could not be replicated the produced metrics from reference [45] could not be compared directly but instead a visual comparison was performed to give a respective high level overview of convergence to a physical solution.

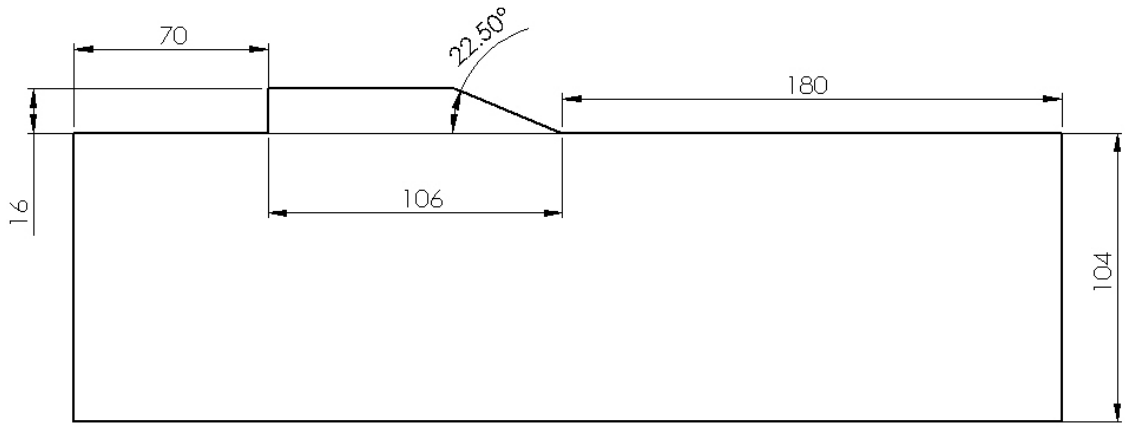


Figure 4.1: Dimensions of the Cavity Referenced, mm. [45]

To generate this mesh as mentioned a prediscretized geometry from VULCAN was altered. The original mesh that was altered can be seen following in Figure 4.3. From this mesh the geometry shown in Figure 4.2 was achieved from truncating the cavity ramp and hyperbolically growing the outlet with y standard boundary conditions enforced. The resulting and final mesh used within the calculations is shown following within Figure 4.4. Although able to provide a converged solution the truncation of the ramp lead to some instability problems from the wall matching turbulence function as the y plus was not able to be resolved down to a lower number. This will be discussed in further detail within Section 4.2.

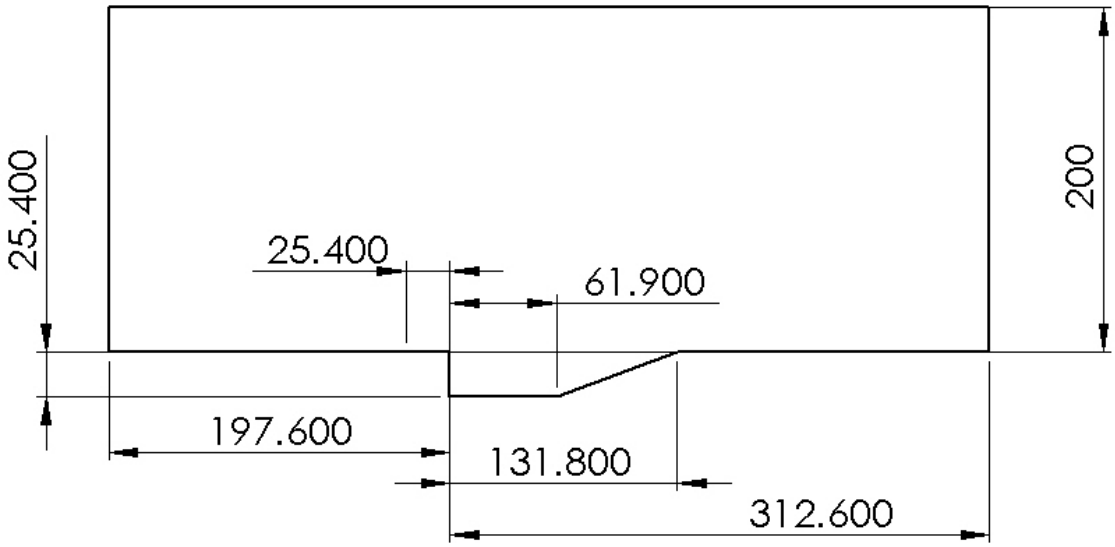


Figure 4.2: Dimensions of the Cavity Used in Calculations, mm.

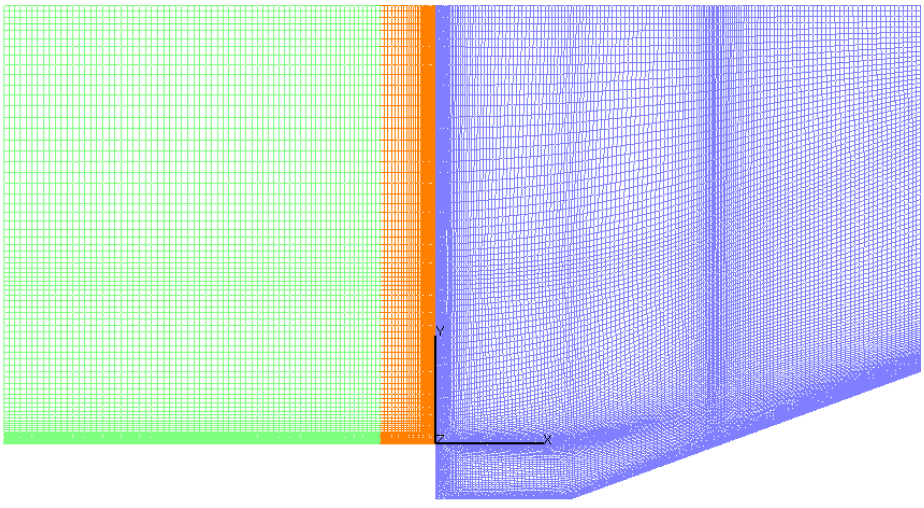


Figure 4.3: Original VULCAN Mesh That was Altered.

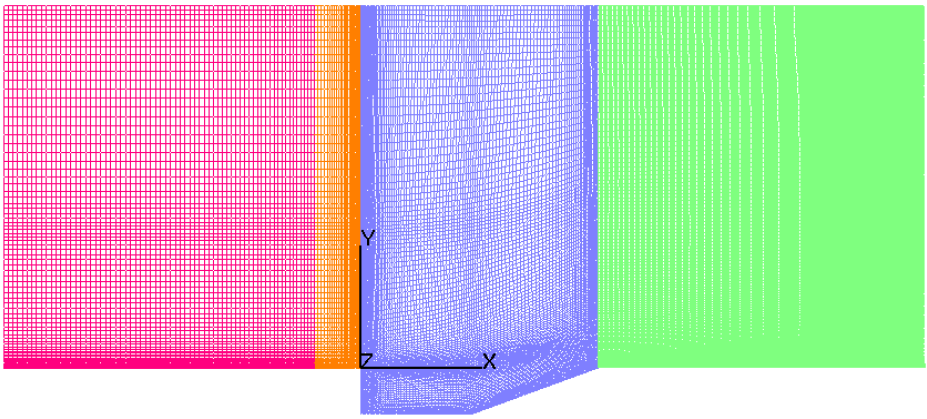


Figure 4.4: 2D Mesh Generated for Calculations.

4.2 Combustor Cold Flow Analysis

To check the converge of the grid produced a cold flow analysis was performed with a standard air mixture with no fuel. The air mixture composition used was 79% N_2 21% O_2.[46] This combination was used over higher fidelity models, such as the one originally programmed into VULCAN, as this allowed for a simplified stoichiometric combustion fuel percentage to be calculated as seen in Equation 4.1 and 4.2. Also, performing this standard air analysis allowed for a calorically perfect, instead of a thermally perfect, analysis to be run. Although using a calorically perfect analysis is not necessarily a good approximation, this assumption allowed for a 3 order magnitude relative converge to be reached faster. As this cold flow analysis was the first to be performed, this increase in convergence time was necessary in order to define the parameters that would be used within the design parameter spaces.

Figure 4.5 shows a general overview of the resolved 3 orders of magnitude relative convergence for the full geometry. Also, a set of streamtraces are placed near the shear layer for the backward step to show the amount of penetration into the cavity from the general flow field.

It should be noted in Figure 4.5 does not include the full geometry as depicted within Figure 4.4. This is because the lead into the combustor was analyzed separately and feed into the cavity geometry section as a profile import. The reason that these two sections were run separately was

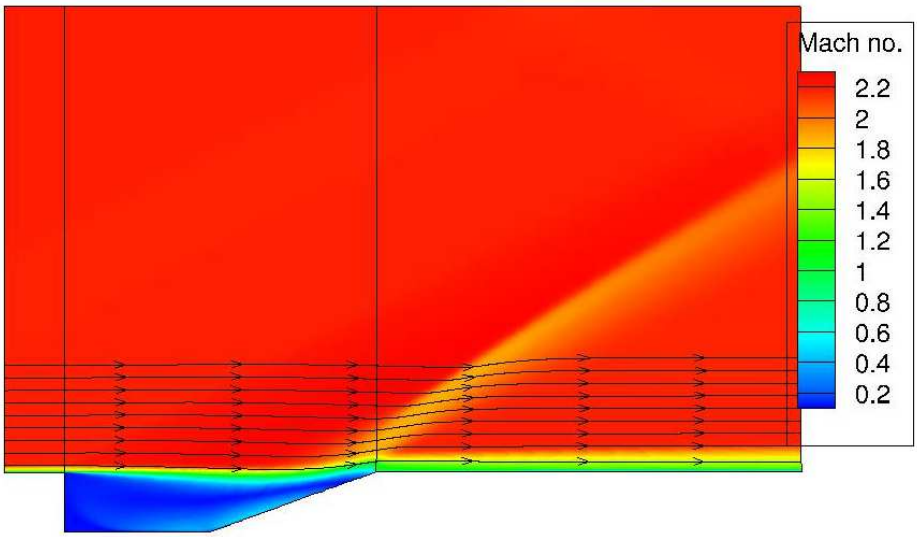


Figure 4.5: General Overview of Cold Flow Combustor Analysis.

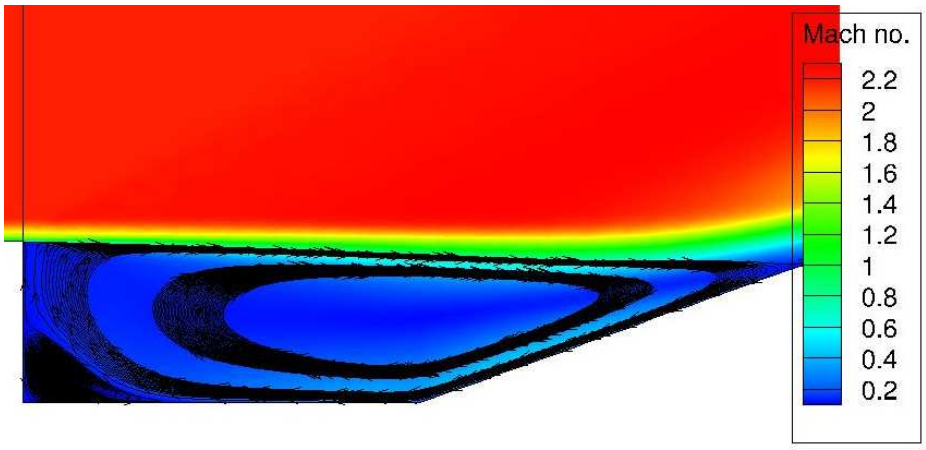


Figure 4.6: Streamtraces inside of the Cavity.

that this increased the convergence within the entire solution and also allowed the full wall-matching turbulence functions to be used within the cavity combustor. Using this methodology obviously did not allow information to be propagated upstream through the boundary layer but after examination of sensitivity to upstream perturbations within reference [45] this error incursion was considered acceptable.

One of the problems that introduced large oscillations into this analysis was the large y plus number associated with the trailing straight section of the combustor. The y plus experienced for this cold flow analysis was on the order of 50. As the cavity section within the combustor could not be remeshed this meant that convergence conditions had to be enforced on the mesh through options within the solver. This made the wall-matching turbulence equations a necessity in order to finally converge the problem. Even with the wall-matching functions the large y plus number brought the solver close to divergence for low CFL conditions. As will be explained within Section 4.5 this was actually the worst case scenario and thus was not considered a problem for the dimensional sweeps to follow.

4.3 Chemistry Model

Originally, three different chemistry models were planned to be compared as fidelity levels in a similar fashion to the diamond airfoil. Time constraints restricted this analysis down to two chemistry models with the third model able to be seen in Section 5.2. Each of these models was based upon the hydrocarbon Ethene (C_2H_4) fuel due to the simplicity of the analysis and also more importantly as this allowed for a hydrocarbon, over a hydrogen, analysis to be performed. Section 2.5 within the Background explains in detail the importance of performing this type of analysis.

4.3.1 Premixed Stoichiometric Condition

To simplify the calculations within the geometry a premixed inflow instead of injection was assumed. Thus, to define the fuel to air ratio within the system a stoichiometric condition was also assumed. Equation 4.1 and Equation 4.2 following show how the stoichiometric condition was calculated. These equations rely on a standard air definition as described in Section 4.2 to be considered an accurate approximation. Also, it should be noted that these equations are based upon an ideal combustion in which all the hydrogen oxygen and carbon are consumed within the analysis so that the only products of the reaction are carbon dioxide and water. Although idealized when compared to the chemical reactions used, this analysis gives a good baseline for designating the fuel to air ratio.

$$C_xH_y + \left(x + \frac{y}{4}\right)\left(O_2 + \frac{79}{21}N_2\right) \rightarrow xCO_2 + \frac{y}{2}H_2O + \frac{79}{21}\left(x + \frac{y}{4}\right)N_2 \tag{4.1}$$

$$f_{st} = \frac{36x + 3y}{103\,(4x + y)} \tag{4.2}$$

For Ethene (C_2H_4) this results in a stoichiometric fuel fraction of $f_{st} = 0.067961$. To keep the mass fraction at 1 for the premixed inflow conditions the fuel percentage was accounted for by removing Nitrogen from the standard air conditions so that the amount oxygen available for combustion for the stoichiometric conditions would not be affected.

4.3.2 Ethene Models

The first model shown below in Table 4.1 is the low fidelity 3 reaction model. This model incorporates 7 different species including the inert N_2, uses the Arrhenius equation for the activation energies and a 1 interval McBride curve fitting for the species coefficients.

The definition used for the Arrhenius equation can be seen below in Equation 4.3 and was sourced from reference [47].

$$k(T) = AT^{B} exp\left(\frac{-E_A}{R_u T}\right) \tag{4.3}$$

Table 4.1: Three Reaction Model Ethene

3 Reaction Model Ethene [36]	
No.	Reaction
3.1	$C_2H_4 + O_2 \rightleftharpoons 2CO + 2H_2$
3.2	$2CO + O_2 \rightleftharpoons 2CO_2$
3.3	$2H_2 + O_2 \rightleftharpoons 2H_2O$

The second model used, shown in Table 4.2, can be found in reference [36] and uses 10 species for 10 reactions. This model also uses the parameters as defined for 3 model reaction for the activation and species coefficients. It should be noted that both of these models were able to use only the single interval McBride curve fitting because the static temperature within the preliminary analysis performed did not exceed the upper limit of 5000 K. Finally, some of the reactions within this model require a catalytic third body efficiencies in order to make the reaction proceed.[48] These values can be seen following in Table 4.3 and were taken from reference [36].

Table 4.2: Ten Reaction Model Ethene

10 Reaction Model Ethene [36]

No.	Reaction
10.1	$C_2H_4 + O_2 \rightleftharpoons 2CO + 2H_2$
10.2	$CO + OH \leftrightharpoons CO_2 + H$
10.3	$CO + O + M \leftrightharpoons CO_2 + M$
10.4	$H_2 + O_2 \leftrightharpoons OH + OH$
10.5	$H + O_2 \leftrightharpoons OH + H$
10.6	$OH + H_2 \leftrightharpoons H_2O + H$
10.7	$O + H_2 \leftrightharpoons OH + H$
10.8	$OH + OH \leftrightharpoons H_2O + O$
10.9	$H + OH + M \leftrightharpoons H_2O + M$
10.10	$H + H + M \leftrightharpoons H_2 + M$

Table 4.3: Ten Reaction Model Third Body Efficiencies

Ten Reaction Model Third Body Efficiencies [37]

Reaction	Species	Efficiency
All	H_2	2.5
All	H_2O	16

4.4 Metrics Computation

In order to compare between the different models again a metrics comparison was used to simplify the analysis. In this case unlike the diamond airfoil a direct physical dimensional comparison could have been performed as the exact same grid was used for both fidelity levels but again this was considered unfeasible as this would have added 2 additional dimensions into the analysis. The three metrics that were chosen to be used for comparison were the specific thrust, I_{sp} and the total pressure loss within the combustor.

Equation 4.4 shows the mass flow formulation that was used for the inflow condition. This point was assumed before injection such that there was no premixed fuel into the system and allowed for a simplified analysis. Thus, fuel injection was assumed inside the control volume to a stoichiometric perfectly mixed fluid before entering the CFD geometry.

$$\dot{m}_{air_{in}} = \rho_{air} h_{comb} V_{air} = \dot{m}_0 \tag{4.4}$$

Please note that all of this analysis was done on a per unit length scale.

Equation 4.5 shows the equation that was used to find the specific thrust within the system. The inflow conditions were considered functions of the height in the combustor as well to standardize the calculations.

$$\frac{F}{\dot{m}_0} = \frac{1}{\dot{m}_0} \int_i^e \rho(y) V(y)^2 dy \tag{4.5}$$

To find the I_{sp} of the system the $\dot{m}_f$ had to be found to start. Equation 4.6 shows how the stoichiometric fuel percentage was used to do achieve this.

$$f = \frac{\dot{m}_0}{\dot{m}_f} \tag{4.6}$$

The above relation was used in the I_{sp} to result in Equation 4.7 below which was the second metric used in the comparison of the combustor parameter spaces.

$$I_{sp} = \frac{F}{g_0 f \dot{m}_0} \tag{4.7}$$

The final metric that was computed on the combustor was the total pressure loss within the system. The form that was used for this calculation can be seen below in Equation 4.8 where in the inlet total pressure condition is dependent upon the isentropic relation.

$$\Delta P_0 = p_i \left(1 + \frac{\gamma_{air} - 1}{2} M_i^2\right)^{\frac{\gamma_{air}}{\gamma_{air} - 1}} - \frac{1}{h_{comb}} \int_0^{h_{comb}} P_0(y) dy \tag{4.8}$$

4.5 3 Model Example Point

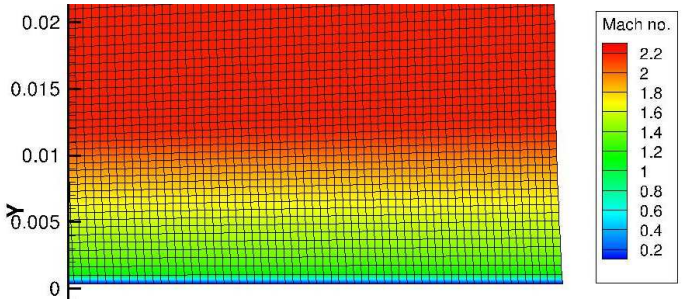
A single point solution for the 3 model chemistry set was run to define the parameters for the following design sweeps. The uniform inflow conditions for the combustor can be seen in Table 4.4 and were also applied to the 10 point single solution as seen in Section 4.6. Also, as mentioned in Section 4.3.1 a premixed assumption was made instead of injection into cavity. This assumption was made to reduce the computational time required due to time restrictions placed on the work. It is recommended by the author that any future analysis be done with injection to give a better physical representation.

Table 4.4: Uniform Inflow Conditions Used for Combustor Verification

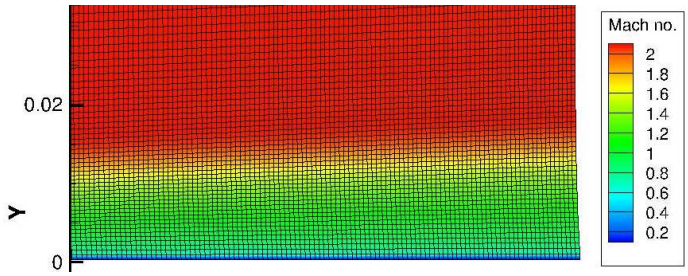
	Value	Units
Mach	1366	$\frac{m}{s}$
Static Temp	960	K
Density	0.260	$\frac{kg}{m^3}$
f_{st}	0.067961	

Using the inflow conditions defined above a preliminary optimization analysis was run. From these simplified runs it was found that using the V multigrid algorithm with 3 levels of multigrid

gave the fastest converge. Also, for this grid the maximum y plus experienced in the solution was on the order of 18. This reduction when compared to the cold flow analysis reduced the oscillations present within the solver and actually increased the convergence time. The reason that the y plus is so much smaller than that seen within the cold flow analysis is the reacting flow influencing the boundary layer propagating down the cavity wall. These reactions increase the size of boundary layer substantially which can be seen in Figure 4.7.



(a) Cold Flow

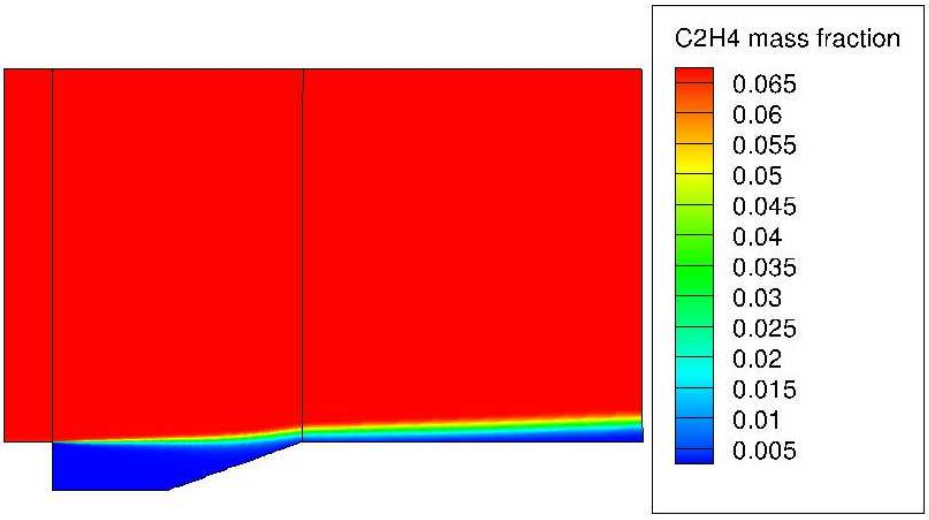


(b) 3 Model Reacting

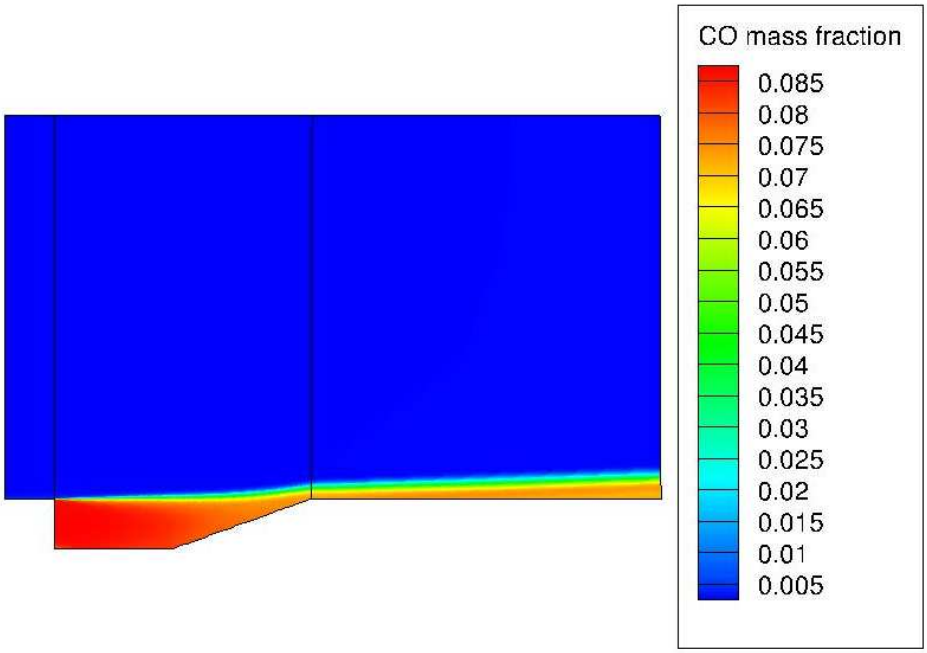
Figure 4.7: Boundary Layer Comparison.

The overview of the Mach number for this chemically reacting flow looks very similar to the cold flow analysis and thus will not be included here. Figure 4.8 below though, has been included to show the fuel fraction and one of the reactants, carbon monoxide, to show the reaction and product concentrations within the flow field for the combustor. It should be noted how little the reaction penetrates into the flow field and is mostly concentrated within the boundary layer propagating down the combustor wall.

The metrics presented within Section 4.4 where analyzed to give a baseline value for the dimensional analysis to be performed. The result of this analysis can be seen within Table 4.5 as



(a) Ethene



(b) CO Reactant

Figure 4.8: Fuel Ratio in Combustor: 3 Model.

well the resultant mass flow rates for comparison purposes.

It should be noted that all of these calculations are per unit length and also that for such a small geometry there was still a substantial total pressure loss within the system.

Table 4.5: Metrics Values for 3 Model Example Point

	Value	Units
$\frac{F}{\dot{m}_0}$	120.15	$\frac{1}{s}$
I_{sp}	180.22	ms
ΔP_0	2.926E+4	$\frac{Pa}{m}$
$\dot{m}_0$	71.05	$\frac{kg}{s}$
$\dot{m}_f$	4.83	$\frac{kg}{s}$

Finally, for this 3 model example point Figure 4.9 and Figure 4.10 are included to show the convergence within the system and also the y plus as function of the number of iterations conducted. It should be noted how in Figure 4.10 the y plus converges down to the maximum, grid dependent, end condition of 18. This convergence is due to the viscosity near the wall approaching the correct value within solution.

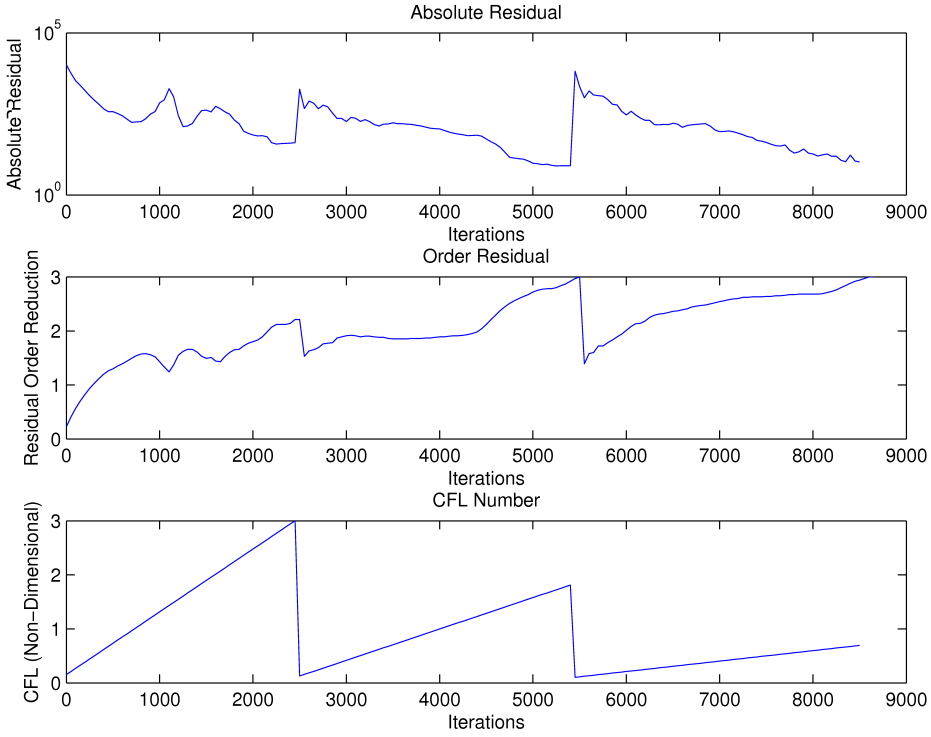


Figure 4.9: 3 Model Residual Convergence.

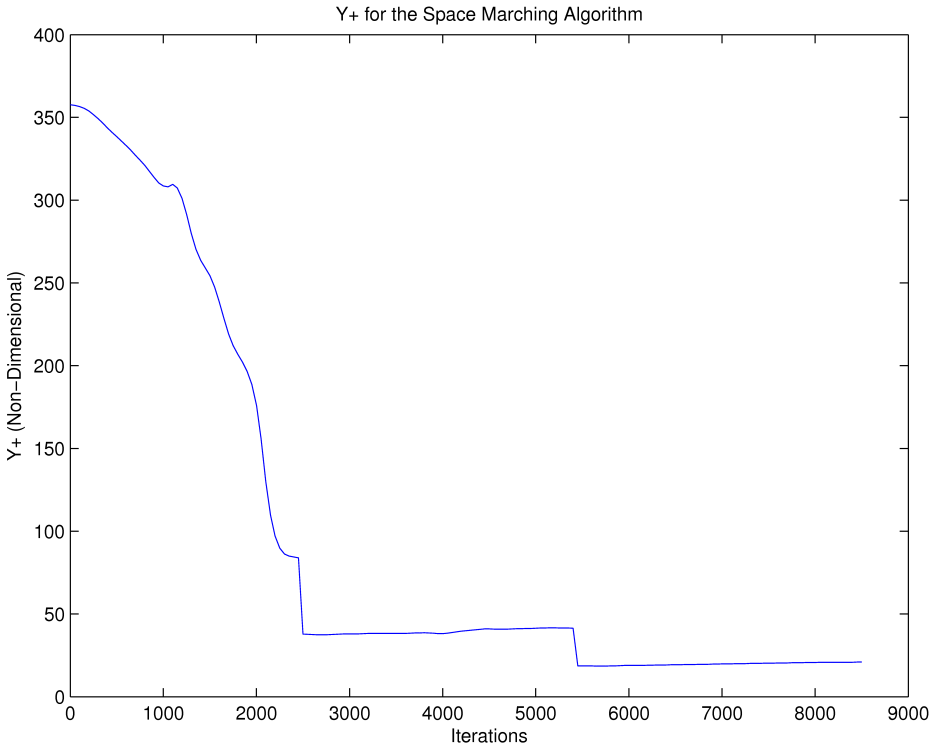


Figure 4.10: 3 Model Y Plus Convergence.

4.6 10 Model Example Point

In similar fashion to the 3 model example, a sample point was run to define the metrics used for the high fidelity chemical analysis. This model was also found to converge the fastest on a 3 level V algorithm multigrid for the defined geometry. Secondly, the y plus seen within the system was substantially less than the cold flow analysis but was higher than the maximum y plus seen in the 3 model example point. The maximum y plus value that was experienced was on the order of 28. Due to the increase in computational time from the additional chemical equations, an increase in the computation from oscillations in the solver could not be determined.

The general flow field and also the reaction penetration within the combustor are not going to be displayed here as they have the same general appearance as the cold flow Mach and also the 3 model reactants.

Following in Table 4.6 is the metrics that were computed for this fidelity level chemical analysis.

Table 4.6: Metrics Values for 10 Model Example Point

	Value	Units
$\frac{F}{\dot{m}_0}$	115.48	$\frac{1}{s}$
I_{sp}	173.05	ms
ΔP_0	1.843E+4	$\frac{Pa}{m}$
$\dot{m}_0$	71.05	$\frac{kg}{s}$
$\dot{m}_f$	4.83	$\frac{kg}{s}$

Similar to the 3 model example point the residual convergence and y plus are shown following in Figure 4.12 and Figure 4.11. The convergence seen within Figure 4.12 is very similar to the convergence in Figure 4.9 which shows that the difference in the chemical fidelity level by itself does not greatly affect the convergence of the solution. In comparing the y plus convergence in Figure 4.11 to the 3 model the only major difference was the divergence in the y plus seen at the second multigrid level. This small divergence did not affect the overall convergence of the solution, therefore, it was not considered a problem.

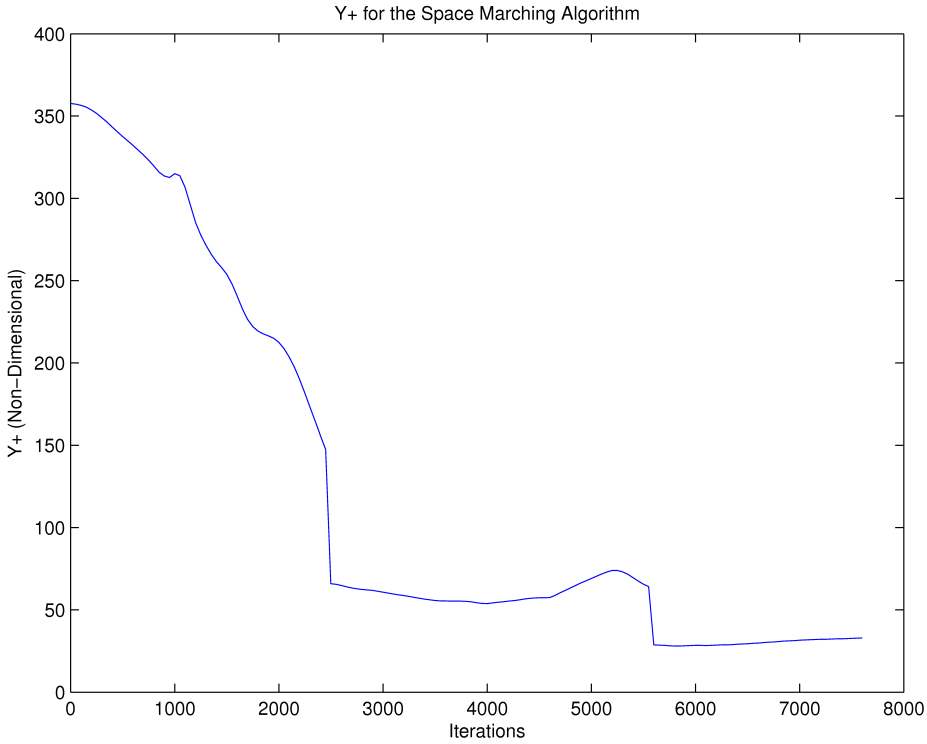


Figure 4.11: 10 Model Y Plus Convergence.

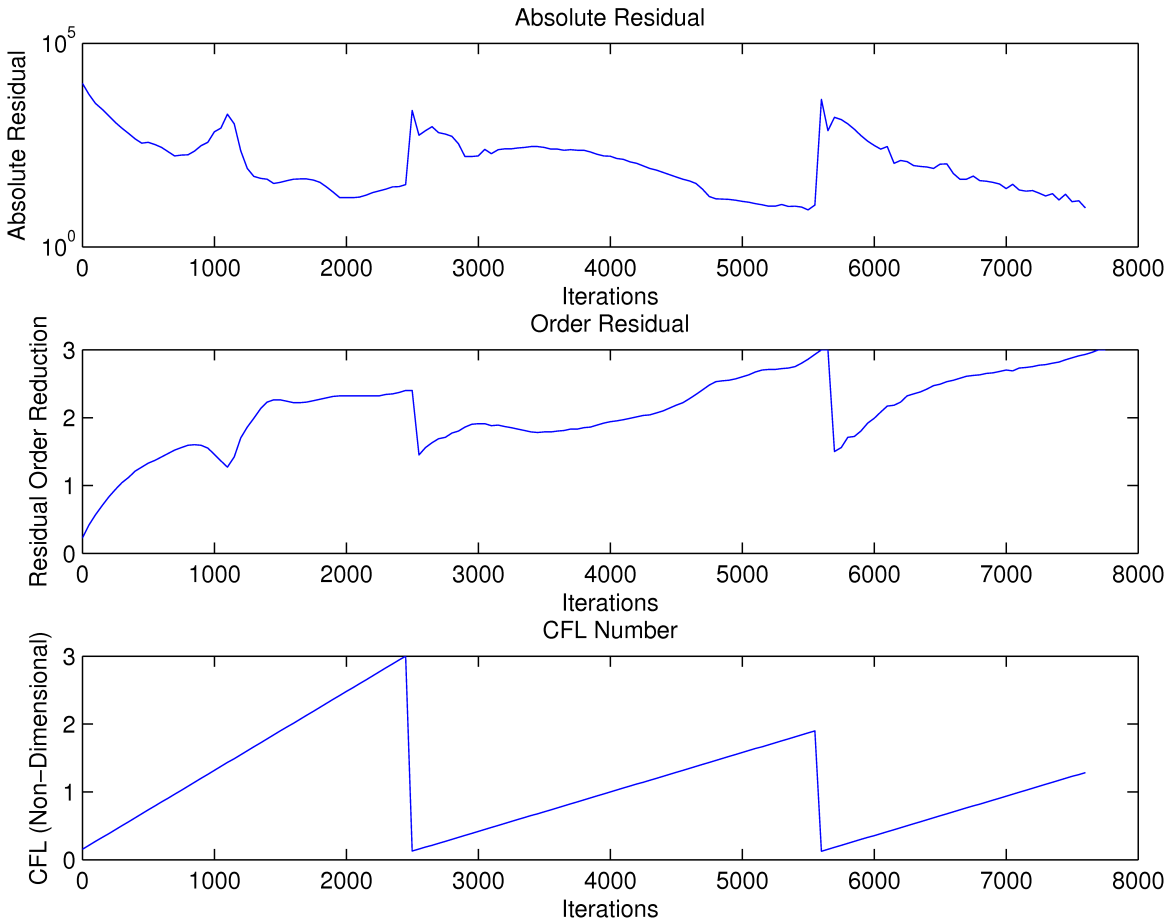


Figure 4.12: 10 Model Residual Convergence.

4.7 One Dimensional Analysis

Similar to the test case, the first parameter space examined was a single dimension. The dimension that was chosen to be examined was the exponent E_A term on the Arrhenius equation, Equation 4.3, on the first chemical reaction in both fidelity levels. This parameter was chosen as preliminary analysis showed that the Arrhenius coefficient had a very low sensitivity to perturbations and thus did not result in a design space that was representative of the complexities of scramjet chemistry analysis. Secondly, the first equation in both chemical sets was chosen as this is the chemical representation for the direct reaction with the Ethene fuel. Although, not necessarily the most perturbation sensitive, these equations give a good physical association with the metrics produced from the design sweep. Finally, it should be noted that the dimensional spaces created by varying the coefficients, exponents and catalytic effects on each of the chemical reactions become extremely large very quickly.

Table 4.7 below shows the full factorial sampling used within the design space for upper and lower bounds of ±40% and increments of 4%.

Table 4.7: Combustor Single Dimension Sweep Parameters

	Base Value	± Increment
3 Reactions	18015.3	720.612
10 Reactions	17866.1	714.644

Similar to the example points, the flow field did not change substantially so a figure of it will not be provided here. Figure 4.13, Figure 4.14, and Figure 4.15 following show the Kriging interpolation calculated for each of the metrics at the respective fidelity levels. It should be noted that for both the 3 model and the 10 model the solution began to diverge in the negative increment domain. Due to this fact the Kriging domain was set with the upper and lower bounds of the 10 model convergence conditions such that a full spectrum comparison could be made between the two fidelity levels. Please note that the accuracy of Kriging drops dramatically within unbounded points and thus the unbounded points in the negative spectrum of the 3 model should be taken as

a guide instead of accurately interpolated results. Finally, it should be noted that the nonlinearity within this design space is much more pronounced than that see within Section 3.8 as was theorized.

Figure 4.16 shows the continuous Kriging approximation to the interpolation space between the fidelity levels. The data within this figure was normalized such that all three metrics could be placed on the same plot. By doing this it can be seen that the interpolation approximation for both I_{sp} and specific thrust are the same. This was expected as these two metrics are only differentiated by space independent constants added into the term. Knowing this fact, the main reason that the specific impulse metric was included in the analysis was to provide the ability to make a baseline comparison to other propulsion systems.

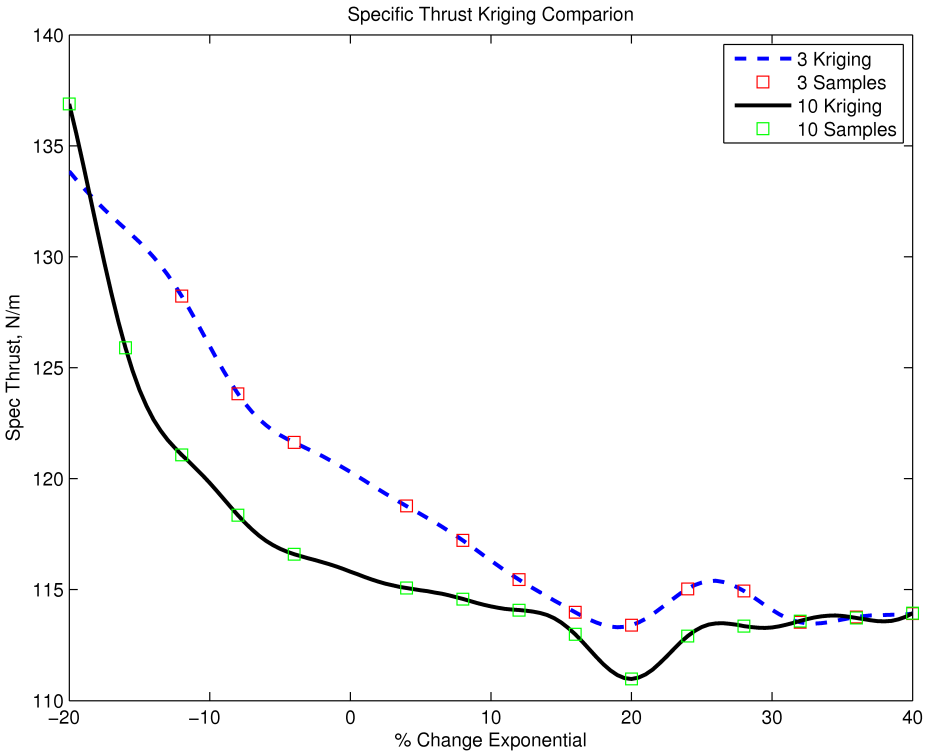


Figure 4.13: Specific Thrust Scramjet One Dimensional Sweep.

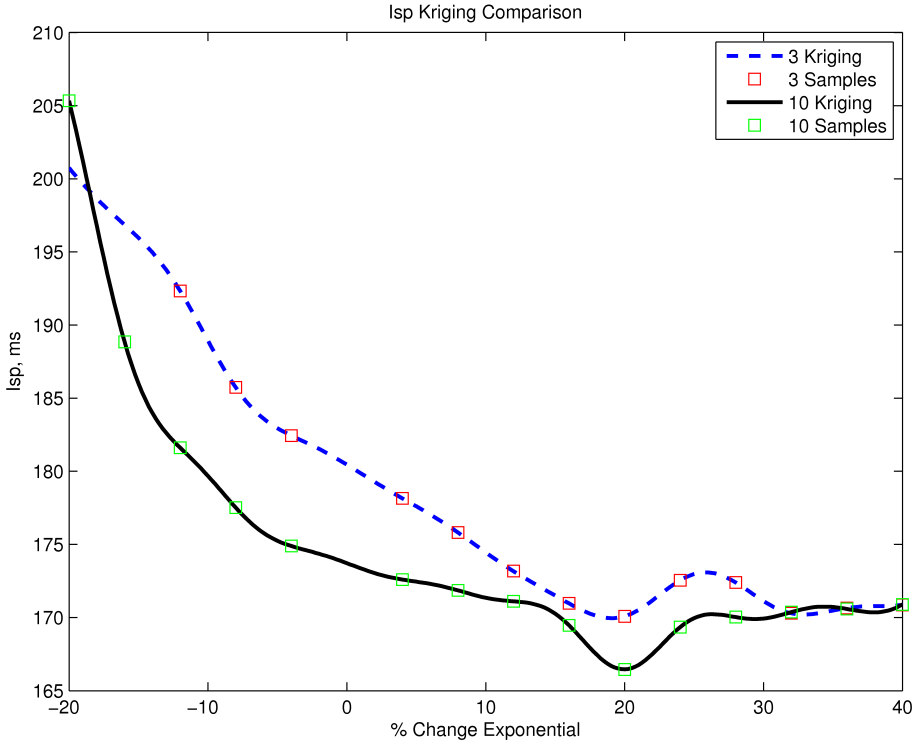


Figure 4.14: Specific Thrust Scramjet One Dimensional Sweep.

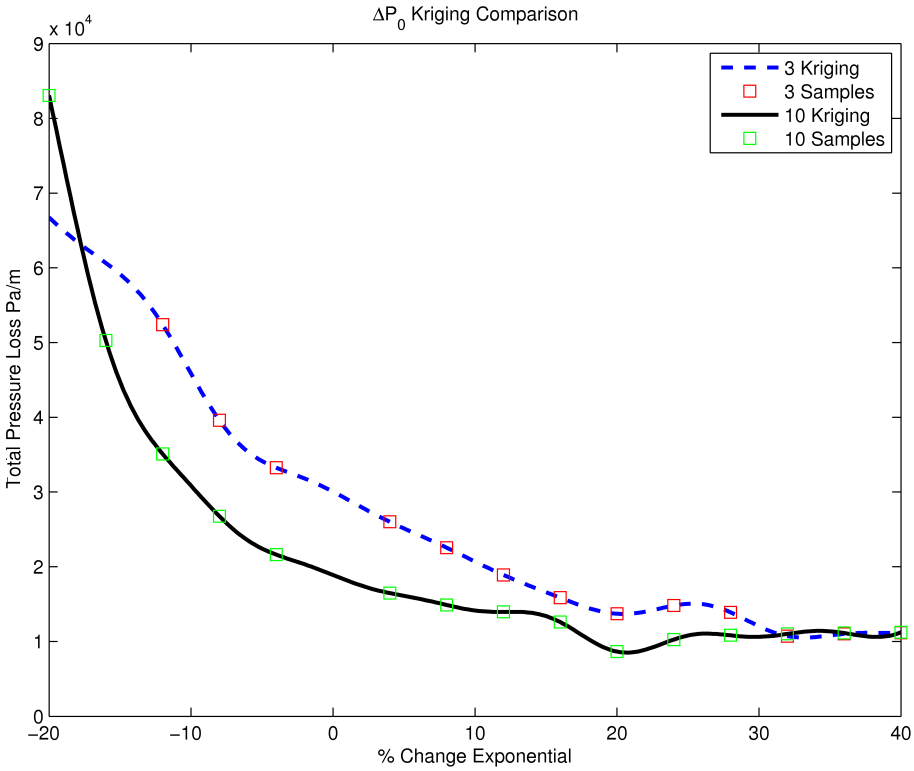


Figure 4.15: Total Pressure Loss Scramjet One Dimensional Sweep.

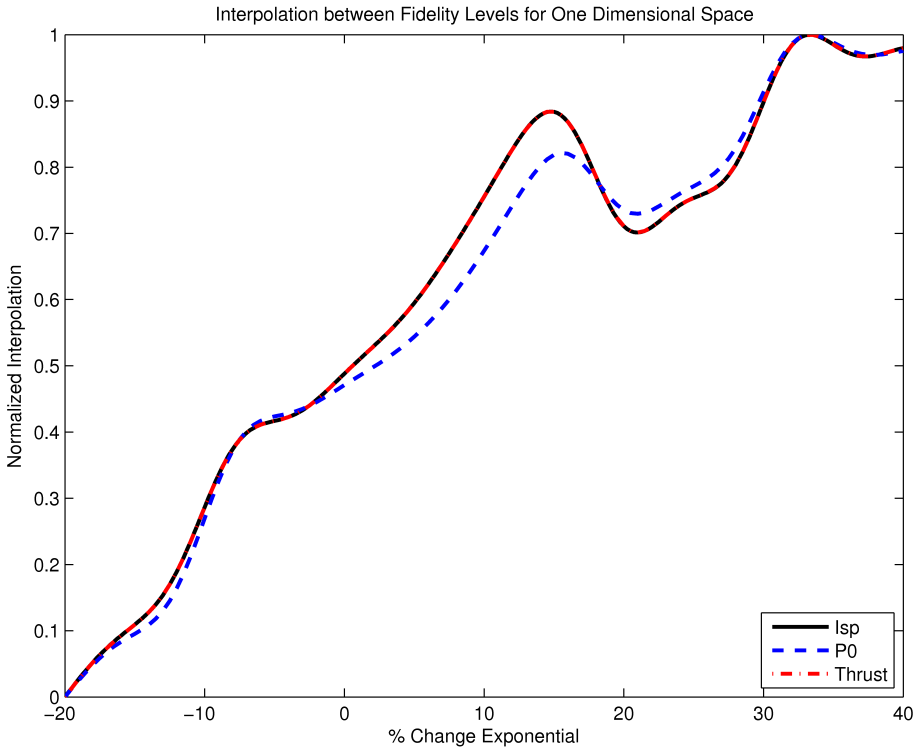


Figure 4.16: Kriging Approximations to Scramjet Metrics.

The PRESS error associated with each of the models presented above is shown following in Table 4.8 to show the accuracy of the Kriging results between the fidelity levels and also across the metrics.

Table 4.8: PRESS Error for One Dimensional Scramjet Design Space

	PRESS		
	Thrust	I_{sp}	P0
3 Model	0.9344	2.106	0.1406
10 Model	6.925	15.53	0.7900
Interpolation	0.5525	1.243	0.0344

From the interpolation spectrum the approximation for the high fidelity data at the 0 increment point was calculated for each metric. Table 4.9 shows these results as well as the original error between the fidelity levels without any added approximations. It should be noted that on average at least an order of magnitude improvement was seen. This is similar to the benign diamond airfoil case and shows that the additional nonlinearities present within the design space did not affect the order of magnitude improvement in predicting the high fidelity data.

Table 4.9: Interpolation Results One Dimensional Scramjet Combustor

	Low Fid	High Fid	Units	Original Error %	Corrected Error %
Specific Thrust	120.15	115.48	$\frac{1}{s}$	4.044	0.2002
I_{sp}	180.22	173.05	ms	4.143	0.2959
ΔP_0	2.926E+4	1.843E+4	$\frac{Pa}{m}$	58.76	0.5739

Chapter 5

Future Work

5.1 Proposed Solution to Viscous Y Plus

The y+ that was able to be resolved for the diamond airfoil was on the order of approximately 50. This value was far larger than the desired value of approximately 1. The reason as mentioned within Section 3.6 that this was not able to be reduced further was do the restrictions of the hardware that was being used to perform the calculations. To circumvent this hard ceiling on the number of points that could be run the proposed meshing solution is shown below within Figure 5.1. The reason that this meshing solution was not attempted was the lack of time available at the time of writing. This meshing solution would split the meshing into at least 4 blocks and shorten the lead into the airfoil geometry from what is currently being used.

In further detail block one could be meshed to meet the necessary y plus on the leading edge and the overall height of the block could be shortened as the shock would still be passing through the RHS I max boundary. For block 2 and 4 the minimum dimension within the K direction would able to be relaxed as the boundary layer growth would allow for larger grid spacing. This larger grid spacing would allow for less points to be used within the K dimensions to keep the AR meshing requirements and thus would reduce the computation required. For the inlet conditions on the second block, as long as the shock is contained within the boundary on RHS of block 1, as it should be, then everything above the top boundary of block 1 could be considered inflow conditions. The input conditions on block 3 could be done through profile updating or through patch conditions depending on how the user wishes to split the computational loading among the cores available.

It is expected that this meshing procedure would allow for y plus on the order of 1 and decrease the computational time required space marching the entire domain as the amount of overall points to be iterated on would be reduced drastically.

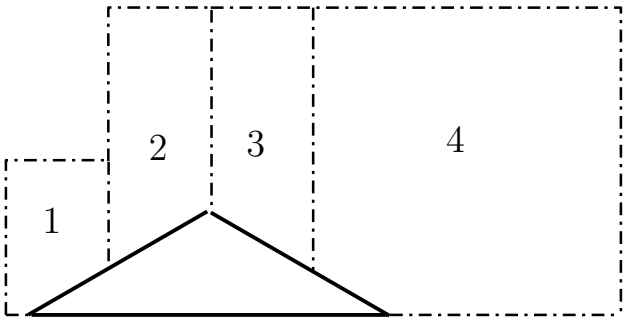


Figure 5.1: Proposed Airfoil Viscous Meshing to Improve Y+: NS.

5.2 Additional Scramjet Analysis

The following are future work extensions recommended for the scramjet combustor geometry:

(1) Time restrictions at the time of writing did not allow for a multi-dimensional analysis to be performed on the chemistry sets within the scramjet combustor. This analysis would be the first recommended future work to be performed on this geometry as the resultant parameter spaces would begin to show the uncertainty associated with the empirically derived chemical equations within each of the sets.

(2) The next recommendation would be to include injection into the model from the premixed assumption made. By including injection in the model the variance within the parameter spaces created would become larger and more unstable due to the lack of fuel within the flow stream.

(3) To show a larger difference in the variance of the chemical equations examined it would advantageous to add a third chemical model. Incorporating a third model into the scramjet analysis was originally planned to be included in this thesis but due to time restrictions

this no longer became feasible. Table 5.1 shows the chemical model that was chosen for this work and is the most logical step in expanding to larger chemical Ethene model.

(4) In order to begin examining the mapped parameter spaces into unknown dimensions it is theorized that creating an orthonormal basis on the correction space would help in finding a mappable basis coefficient vector between dimensional spaces. The current proposed idea would be to use wavelets to achieve this orthonormal basis as this transformation algorithm allows for compact local support in n dimensions and would not require as many calculations as a POD transformation. An investigation would need to be performed to prove this comparison between wavelets and POD but from this analysis variance of the basis coefficients for dimensional extrapolation could begin.

Table 5.1: Twenty Reaction Model Ethene

20 Reaction Model Ethene [45]

No.	Reaction
20.1	$C_2H_4 \rightleftharpoons C_2H_2 + H_2$
20.2	$C_2H_4 + O_2 \rightleftharpoons 2CO + 2H_2$
20.3	$CO + O + M \leftrightharpoons CO_2 + M$
20.4	$CO + O_2 \leftrightharpoons CO_2 + O$
20.5	$CO + OH \leftrightharpoons CO_2 + H$
20.6	$OH + H_2 \leftrightharpoons H_2O + H$
20.7	$O + OH \leftrightharpoons O_2 + H$
20.8	$O + H_2 \leftrightharpoons OH + H$
20.9	$2OH \leftrightharpoons H_2O$
20.10	$2H + M \leftrightharpoons H_2 + M$
20.11	$H + OH + M \leftrightharpoons H_2O + M$
20.12	$H + O + M \leftrightharpoons OH + M$
20.13	$2O + M \leftrightharpoons O_2 + M$
20.14	$N + NO \leftrightharpoons N_2 + O$
20.15	$N + O_2 \leftrightharpoons NO + O$
20.16	$N + OH \leftrightharpoons NO + H$
20.17	$H_2 + O_2 \leftrightharpoons 2OH$
20.18	$2H + H_2 \leftrightharpoons 2H_2$
20.19	$2H + H_2O \leftrightharpoons H_2 + H_2O$
20.20	$2H + CO_2 \leftrightharpoons h_2 + CO_2$

Chapter 6

Conclusion

Two geometries, of a diamond airfoil and a scramjet combustor, were analyzed in order to find the continuous spectrum between multi-fidelity design spaces. Three levels of fidelity were examined for the diamond airfoil including an analytical solution to the problem that was used for verification and validation purposes. Once a valid solution was able to be obtained for the geometry it was shown that the lower fidelity results could be corrected for an average of an order of magnitude improvement in both a one dimensional and a multi-dimensional parameter space.

For the scramjet combustor, once the meshing issues were resolved, a single dimensional analysis was run on the geometry. Unlike the test case scenario of the diamond airfoil, the dimension examined was not a physical dimension but a sensitivity between fidelity levels on the empirically derived constants within the chemical equations. After applying the correction term onto the off-parameter points computed in this non-physical space, an order of magnitude increase in the error of the low fidelity data was again able to be achieved. Although only a single dimension analysis was able to be run for this geometry, it is expected that similar results would be obtained for multiple dimensions.

The results presented, on average, showed an order of magnitude improvement in accuracy for almost no increase in computational cost. It can be seen how this work will make it easier for future vehicle designers to produce more accurate, densely sampled design spaces that will allow for the end optimal solution to be more precisely defined.

Bibliography

[1] Robinson, J. S., "An Overview of NASA's Integrated Design and Engineering Analysis(IDEA) Environment," 17th AIAA International Space Planes and Hypersonic Systems And Technologies Conference, AIAA, April 2011.

[2] Schmidt, D. K. and Lovell, T. A., "Mission Performance and Design Sensitivites of Air-Breathing Hypersonic Launch Vehicles," Journal of Spacecraft and Rockets, Vol. 34, No. 2, March-April 1997, pp. 158–164.

[3] Peebles, C., Eleven Seconds into the Unknown, AIAA, 2011.

[4] Hallion, D. R. P., editor, The Hypersonic Revolution: Case Studies in the History of Hypersonic Technology, Vol. 1, Air Force History and Museums Program, 1998.

[5] Jenkins, D. R., X-15: Extending the Frontiers of Flight, NASA, December 2009, ebook.

[6] Facing the Heat Barrier: A History of Hypersonics, NASA, 2007.

[7] Geddes, R. C., "Inspection of Demonstrator Bonded Repairs on a QANTAS 747-300," Tech. rep., Defense Science and Technology Organization, September 1999, Australia.

[8] Schweikart, D. L., The Hypersonic Revolution: Case Studies in the History of Hypersonic Technology, Vol. 3, Air Force History and Museums Program, 1998.

[9] Tang, M. and Mamplata, C., "Two Steps Instead of a Giant Leap - An Approach for Air breathing Hypersonic Flight," 17th AIAA International Hypersonic Space Planes and Hypersonic Systems and Technologies Conference, AIAA, April 2011.

[10] Walker, S., Tang, M., Morris, S., and Mamplata, C., "Falcon HTV-3X - A Reusable Hypersonic Test Bed," 15th AIAA International Space Plane and Hypersonic Systems and Technologies Conference, AIAA, April 2008.

[11] Warwick, G., "DARPA Cancels Hypersonic Blackswift," Aviation Week and Space Technology, October 2008.

[12] Norris, G., "Telemetry Lost During Hypersonic Test Flight," Aviation Week and Space Technology, August 2011.

[13] Walker, S., Sherk, J., Shell, D., Schena, R., Bergmann, J. F., and Gladbach, J., "The DARPA/AF Falcon Program: The Hypersonic Technology Vehicle #2 (HTV-2) Flight Demonstration Phase," 15th AIAA International Space Planes and Hypersonic Systems and Technologies Conference, AIAA, April 2008.

[14] Space, U. A. and Command, M. D. C. F. S., "Advanced Hypersonic Weapon Program: Environmental Assessment," Tech. rep., United States Army, June 2011.

[15] McNamara, J. J. and Friedmann, P. P., "Aeroelastic and Aerothermoelastic Analysis in Hypersonic Flow: Past, Present and Future," AIAA Journal, Vol. 49, No. 6, June 2011, pp. 1089–1122.

[16] Fulghum, D. and Sweetman, B., "Seen and Unseen: Overflights by U.S. Unmanned Surveillance Aircraft Revealed by Desert Crash," Aviation Week and Space Technology, December 2011, pp. 18–21.

[17] Mohan, S., Lalithambika, V., Rajeev, U., and Ajithkumar, B., "Ascent Trajectory and Guidance Design without Black Zone," ICTT2010, Jan 2011.

[18] McClinton, C. R., "High Speed/Hypersonic Aircraft Propulsion Technology Development," RTO/AVT/VKI Lecture Series on Advances on Propulsion Technology for High-Speed Aircraft, Von Karman Institute, March 2007.

[19] Jr., C. E. C., Engelund, W. C., Bittner, R. D., Jentink, T. N., Dilley, A. D., and Frendi, A., "Integrated Aeropropulsive Computational Fluid Dynamics Methodology for the Hyper-X Flight Experiment," Journal of Spacecraft and Rockets, Vol. 38, No. 6, November-December 2001, pp. 836–843.

[20] Forrester, A. I. J., Sóbester, A., and Keane, A. J., Engineering Design via Surrogate Modeling: A Practical Guide, Vol. 226 of Progress in Astronautics and Aeronautics, John Wiley & Sons, 2008.

[21] Mehta, U. B., Oberkampf, W. L., Sindir, M. M., and Conlisk, A. T., "Guide for the Verification and Validation of Computational Fluid Dynamics Simulations," Tech. rep., American Institute of Aeronautics and Astronautics, 1998.

[22] Crowell, A. R., McNamara, J. J., Kecskemety, K. M., and Goerig, T. W., "A Reduced Order Aerothermodynamic Modeling Framework for Hypersonic Aerothermoelasticity," 51st AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, 2010.

[23] Lucia, D. J., Beran, P. S., and Silva, W. A., "Reduced-Order Modeling: New Approaches for Computational Physics," Progress in Aerospace Sciences, Vol. 40, 2004, pp. 51–117.

[24] Giunta, A., Swiler, L., Brown, S., Eldred, M., Richards, M., and Cyr, E., "The Surfpack Software Library for Surrogate Modeling of Sparse Irregularly Spaced Multidimensional Data," 11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, AIAA, 2006, pp. 1708–1736.

[25] Friedman, J. H., "Multivariate Adaptive Regression Splines," The Annals of Statistics, , No. 1, 1991.

[26] Giunta, A., Watson, L., and Koehler, J., "A Comparison of Approximation Modeling Techniques: Polynomial versus Interpolating Models," 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Design, 1998, pp. 392–404.

[27] Yamazaki, W. and Mavriplis, D. J., "Derivative-Enhanced Variable Fidelity Surrogate Modeling for Aerodynamic Functions," 49th AIAA Aerospace Sciences Meeting including the New Horizon Forum and Aerospace Exposition, AIAA, January 2011.

[28] Forrester, A. I., Sóbester, A., and Keane, A. J., "Multi-Fidelity Optimization via Surrogate Modelling," Proceedings of the Royal Society, Vol. 463, No. 2088, December 2007, pp. 3251–3269.

[29] Han, Z.-H., Zimmerman, R., and Görtz, S., "A New CoKriging Method for Variable-Fidelity Surrogate Modeling of Aerodynamic Data," 48th AIAA Aerospace Sciences Meeting including the New Horizon Forum and Aerospace Exposition, AIAA, January 2010.

[30] Giunta, A. A., Richards, M. D., Cyr, E. C., Swiler, L. P., Brown, S. L., and Eldred, M. S., Surfpack Version 1.0 User's Manual, Sandia National Laboratories, 2006.

[31] Starkey, R. P., "Effects of Chemistry Modeling on Scramjet Engine Design," 45th AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit, AIAA, 2009.

[32] Chang, J. S. and Lewis, M. J., "Development of a Jet-A/Silane/Hydrogen Reaction Mechanism for Modeling a Scramjet Combustor," 35th AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit, June 1999.

[33] Rich, B. and Janos, L., Skunk Works: A Personal Memoir of my Years at Lockheed, Little, Brown and Company, 1994.

[34] Hank, J. M., Murphy, J. S., and Mutzman, R. C., "The X-51A Scramjet Engine Flight Demonstration Program," 15th AIAA International Space Planes and Hypersonics and Technologies Conference, AIAA, April 2008.

[35] O'Brien, T. F., Starkey, R. P., and Lewis, M. J., "Quasi-One-Dimensional High-Speed Engine Model with Finite-Rate Chemistry," Journal of Propulsion and Power, Vol. 17, No. 6, November-December 2001, pp. 1366–1374.

[36] Baurle, R. A., Mathur, T., Gruber, M. R., and Jackson, K. R., "A Numerical and Experimental Investigation of a Scramjet Combustor for Hypersonic Missile Applications," 34th AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit, 1998.

[37] Baurle, R. A., VULCAN User Input Manual, NASA, January 2008, Version 6.0.2.

[38] Chan, W. M., Rogers, S. E., Pandya, S. A., Kao, D. L., Buning, P. G., Meakin, R. L., Boger, D. A., and Nash, S. M., Chimera Grid Tools User's Manual, NASA, March 2010, Version 2.1.

[39] Anderson, J. D., Modern Compressible Flow with Historical Perspective, Vol. International Edition, McGraw Hill, 3rd ed., 2004.

[40] vonEggers Rudd, L. and Lewis, M. J., "Comparison of Shock Calculation Methods," Journal of Aircraft, Vol. 35, No. 4, July-August 1998, pp. 647–649, Engineering Note.

[41] Wellmann, J., Vereinfachung von Rechnungen am schiefen Verdichtungsstoß, Deutsche Luftund Raumfahrt, 1972.

[42] Starkey, R. P., Investigation of Air-Breathing, Hypersonic Missile Configurations Within External Box Constraints, Ph.D. thesis, University of Maryland, 2000.

[43] Schlichting, H. and Gersten, K., Boundary Layer Theory, Springer-Verlag Berlin Heidelber, 8th ed., 2000, Enlarged Edition.

[44] Weeks, M., Digital Signal Processing: Using MATLAB and Wavelets, Jones and Bartlett Publishers, 2nd ed., 2011.

[45] Malo-Molina, F. J., Gaitonde, D. V., Ebrahimi, H. B., and Ruffin, S. M., "Three-Dimensional Analysis of a Supersonic Combustor Coupled to Innovative Inward-Turning Inlets," AIAA Journal, Vol. 48, No. 3, March 2010, pp. 572–582.

[46] Heiser, W. H., Pratt, D. T., Daley, D. H., and Mehta, U. B., Hypersonic Airbreathing Propulsion, AIAA Education Series, 1994.

[47] Turns, S. R., An Introduction to Combustion: Concepts and Applications, McGraw-Hill International Editions, 2nd ed., 2006.

[48] Kee, R. J., Coltrin, M. E., and Glarborg, P., Chemically Reacting Flow: Theory and Practice, John Wiley and Sons Interscience, 2003.

[49] Jain, A. K., Mao, J., and Mohiuddin, K., "Artificial Neural Networks: A Tutorial," Computer, March 1996, pp. 31–44.

[50] Jr., J. D. A., Introduction to Flight, McGraw-Hill, 5th ed., 2005.

[51] Kuethe, A. M. and Chow, C.-Y., Foundations of Aerodynamics: Bases of Aerodynamics Design, John Wiley & Sons, 5th ed., 1998.

[52] Burden, R. L. and Faires, J. D., Numerical Analysis, Thomson Brooks/Cole, 8th ed., 2005.

Appendix A

Additional Background

A.1 Surrogate Model Error Metrics

All of the following error metrics presented within Equation A.1 through Equation A.3 are defined within reference [24] with brief description displayed below. Also, all of the metrics are displayed for a single variable dimension but are able to be extrapolated to as many dimensions as desired.

R^2 Error

$$R^2 = \frac{\sum_{i=1}^{N} \left(\hat{f}_i - \bar{f}\right)^2}{\sum_{i=1}^{N} \left(f_i - \bar{f}\right)^2} \tag{A.1}$$

Where $\bar{f}$ is the mean of the stochastic random field $f(x)$ and $\hat{f}$ is the polynomial approximation. This metric normalizes the squared difference term to the polynomial approximation with the known random field variables. This metric is only used for the polynomial approximations.

RMS Error

$$RMS = \sqrt{\frac{\sum_{i=1}^{N} \left(f_i - \hat{f}_i\right)^2}{N}} \tag{A.2}$$

The RMS error is not particularly useful for the interpolation methods as the error becomes close to zero for these models. However, this metric does give a good comparison to the R^2 metric for the polynomial approximations shown above. Please note that the error is

non-zero due to machine precision and round off errors within the algorithms used within the program. [24]

PRESS Error

$$PRESS = \sum_{i=1}^{N} \left(f_i - \hat{f}_{(i)} \right)^2 \tag{A.3}$$

PRESS is used to validate the accuracy of the model when single points are removed from the random field. This metric is very useful for the interpolation methods even though a full factorial sampling instead of a hypercube sampling was used. Accepting this error, this metric was used to compare bracketed solutions within the design spaces tested within Section 3.8 and Section 4.7.

A.2 Multi-Layered Feed-Forward Perceptron Artificial Neural Networks

Figure A.1 shows a general configuration for a feed-forward multi-layered ANN. This configuration is not self organizing but still allows for recursion between the neurons as the algorithm traverses the layers within the neuron map. Following this is the steps taken within the standard perceptron learning algorithm to find the weights within the neuron map to optimize the solution.In the case used within Section 2.4.3 this would mean finding the optimal weights in order to optimize the basis function shown within Equation 2.7.[49]

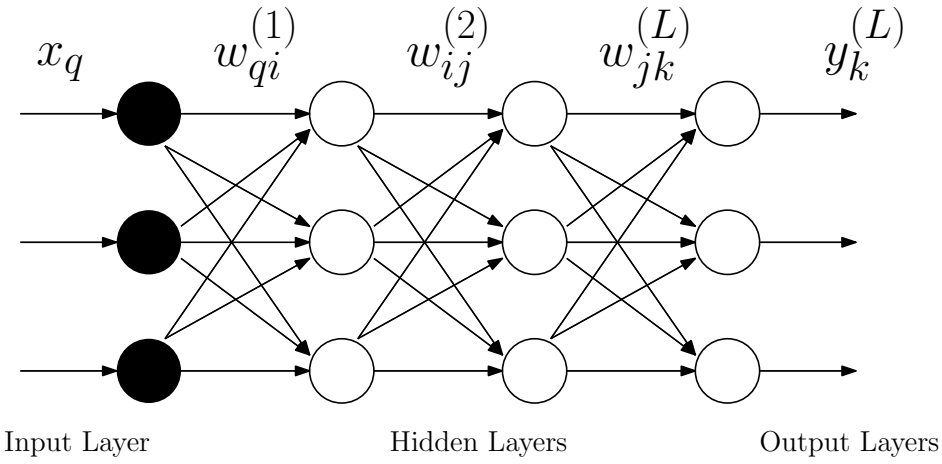


Figure A.1: Three Layered Perceptron ANN Configuration.[49]

A.2.1 Perceptron Learning Algorithm

(1) Initialize the weights to small random numbers

(2) Present a pattern vector $(x_1, x_2, \ldots, x_n)^t$ which in this case would be the initial sample points within a single dimensional space

(3) Update the weights according to Equation A.4

$$w_j(t+1) = w_j(t) + \eta(d-y)x_j \tag{A.4}$$

Where, d is the desired output, t is the iteration step and $\eta(0 < \eta < 1)$ is an input gain into each of the artificial neurons. In the case of this work the desired output for the sampled points is the same as the input. As this is not a strong forced condition within the mapping though there is no guarantee of this condition as shown with Section 2.4.3.

Appendix B

Chimera Grid Tools Descriptions

B.1 Grid Editor for Structured Entities (GRIDED)

GRIDED is a container for multiple subfunctions necessary within the Chimera Grid Tools suite. Some of these include the ability to translate, concatenate and extrapolate entities. All of these subfunctions act like a preprocessors for the other programs within Chimera and are particularly useful when importing CAD models into the meshing suite.[38]

B.2 Hyperbolic Field Grid Generator (HYPGEN)

HYPGEN is a function that generates volume grids over three dimensional surfaces using hyperbolic marching schemes from curved entities. This function was originally used within the creation of C grids for the Diamond Airfoil test case as it simplified the batch processing of rotated airfoil entities with respect to the standard reference coordinate frame. With the conversion to H grids for the airfoil as explained within Section 3.2.2, HYGEN was originally used for the generation of 2-D surfaces for use within SURGRD. After discovering grid quality issues from the method, HYPGEN was finally applied as the main grid generation tool. This implies that HYPEN was used to hyperbolically grow two dimensional grids from single entities to be reflected about the y axis.

It should also be noted that the instabilities within the hyperbolic algorithm meant that the 2-D surfaces originally generated for SURGRD had to be manually edited to prevent the error from propagating further into the meshing process. These instabilities were on the order of the solution at a magnitude of 10^{-5} which was the desired meshing point for the viscous grids. [38]

B.3 Spline Redistribute And Project (SRAP)

SRAP is program used to redistribute points across entities in three dimensions. This program was originally used to collate points around the leading and trailing edges within the diamond airfoil meshes. After resolving the grid quality issues this function was instead used to uniformly distribute points across the entity. Please note that this function only works on single dimension entities and thus had to be used before any surface meshing is attempted.[38]

B.4 Overset Surface Grid Generator (SURGRD)

SURGRD is a surface generation tool based upon reference surface geometry and in general is used on CAD surfaces that need additional meshing. Modifying the inputs into this algorithm allowed for the first generation of H grid meshes to be produced before the switch was made to HYGEN. This program is not longer included in the meshing process but is still documented here for the benefit of the reader in helping to explain the grid iteration process.

For the generation of H grids using this program opposing entities had to be generated with the same point distribution. From these entities an algebraic algorithm was used across a surface generated from HYGEN to grow a new grid that translated between the opposing entities. This algebraic algorithm allowed for exact boundary conditions to be set on the grid and also for a rudimentary control over the collating of points within the grid.[38]

Appendix C

Diamond Analytical Documentation

The following documentation starts out with block diagrams for the flow of information within the program. It takes a representative sample of $\alpha > 0$ and $\theta > \alpha$ to display the organization of the algorithm. Following these figures is a more complete documentation of each of the functions presented in the block diagrams to give further information on the algorithm developed. Each of these represented functions are organized into their containing modules as would be found within the source code. It is recommended that you examine the block diagrams for an overview of how Diamond Analytical works and look into the representative documentation if you still need further information on the functions used.

The source code for this program is not being published as a part of this dissertation but can be obtained upon request from the author. This program was written in FORTRAN 95 and is designed for the Linux OS architecture. Adaption to other OS systems is possible but manual editing of the recursive makefiles will be necessary.

C.1 Code Block Diagrams

Figure C.1 following shows the flow through the main of the program and how the looping structure within the algorithm is called.

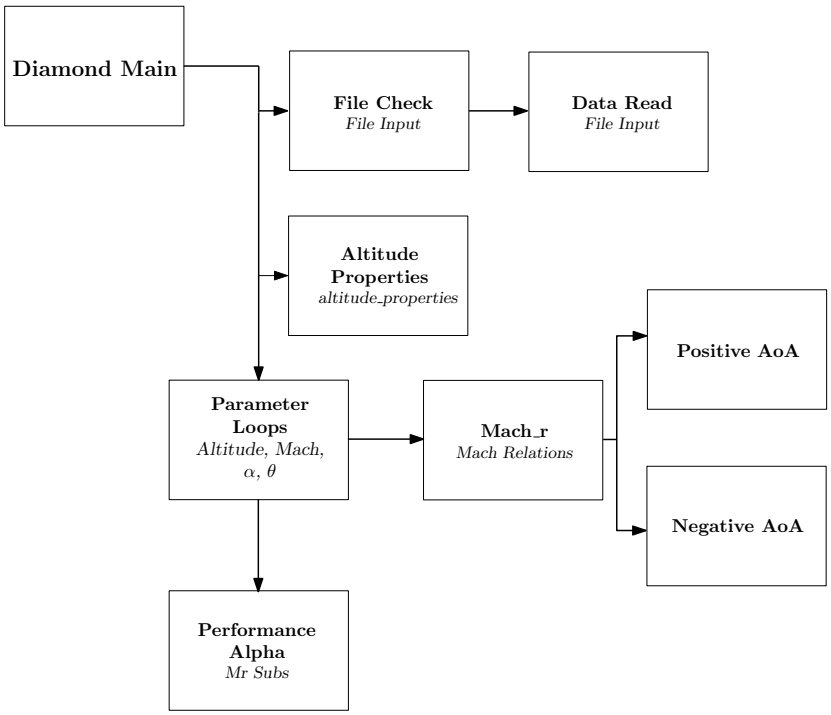


Figure C.1: Block Diagram for Code Flow In Diamond Main.

Figure C.2 shows the propagation of information through the main secondary header Mach Relations. This function is bifurcated into negative and positive AoA sections as shown in Figure C.1. From this initial bifurcation the algorithm is further split depending upon the α and θ conditions input into the algorithm. From these splits the necessary shock and expansion wave relations are called for both the upper and lower surfaces. The shock relations are based upon the $\theta - \beta -$ Mach formulation while the expansion fan relations are based upon the Prandtl-Meyer Function.

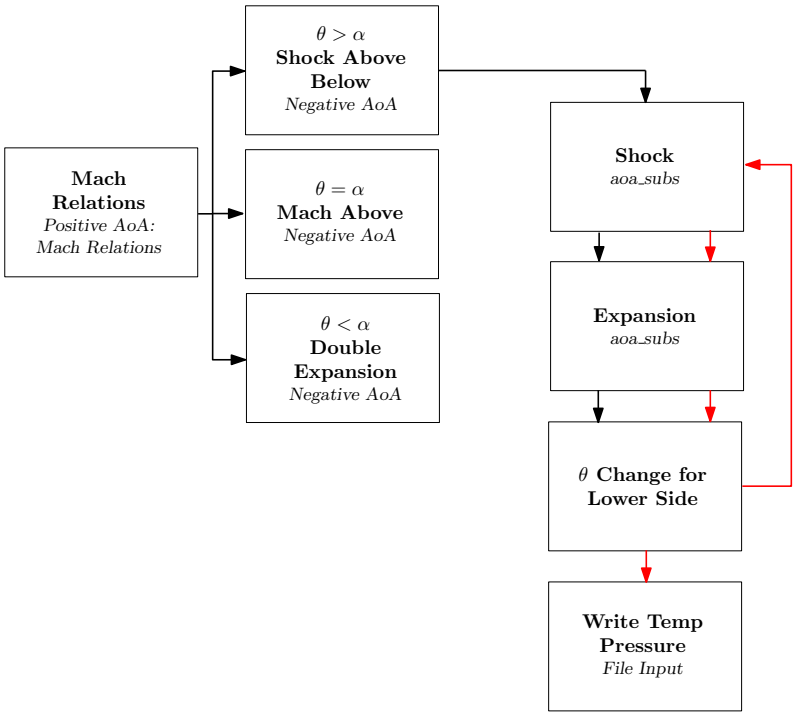


Figure C.2: Block Diagram for the Mach Relations Wrapper.

C.2 Program: Diamond Analytical

Purpose

The theoretical calculation of the shock and expansion wave properties of a diamond airfoil

Inputs

None

Outputs

None

Modules Used

Altitude Properties, Mach Relations, User Inputs, Mr Subs

Reference Code

Diamond Main

Synopses

- The overall high level function for the program that contains the loops for the different parts of each variable vector. The four variable loops currently incorporated into the program, as listed in sequential order, are altitude, Mach number, the wedge half angle and the angle of attack of the airfoil. Combing variations on each of these four parameters produces the four dimensional design sweep that was used in the comparison to VULCAN.
- The main has a decision tree to see if any input file is given to the program. If not the program asks for screen inputs of the desired solution space.
- There is a secondary decision tree based on if the solution space is zero order dimensioned in each of the design spaces.

C.3 Module: Altitude Properties

Purpose

Uses the 1972 Standard Atmosphere to obtain the temperature, pressure and density based upon the altitude requested.

Inputs

Altitude

Outputs

Temperature, Pressure, Density

Modules Used

None

Reference Code

Altitude Properties

Synopses

- This code is a single subroutine module which makes it a little redundant but it was considered necessary to separate this subroutine from the other modules within the system.
- This is only the linear model for geometric altitude and does not include the exponential model for large altitudes. Therefore, the altitudes sweeps are not going to be conducted for large altitudes because of this restriction. [50]

C.4 Module: Angle of Attack Subfunctions

Purpose - By Individual Function

Shock

This function acts as a redirection to another function for calculations but was included in this module to complete all the options that are possible to be called.

Expansion

This function uses the Prandtl-Meyer Function to compute the expansion fan angle from the incoming Mach number and thus find the primitive variables on the other side of the fan.

Mach Wave

This function is for the singular case when the angle of attack is equal to the wedge half angle and thus a Mach wave is produced instead of a shock wave or an expansion fan on the leading edge. [51]

Inputs - By Individual Function

Shock

γ, π, T_∞, p_∞, ρ_∞, M, θ, α

Expansion

The primitive variables following are for before the fan

M, θ, γ, π, T_0, P_0, R, Root Finding Tolerance, Max Iterations

Mach Wave

The primitive variables following are for before the mach wave

M, γ, π, P_0, T_0, R

Outputs - By Individual Function

Shock

The primitive variables following are for after the shock wave

M, p, P_0, T, T_0

Expansion

The primitive variables following are for after the fan

M, T, p, ρ,

Mach Wave

The primitive variables following are for before the mach wave

M, T, p, ρ,

Modules Used

M Functions, Root Finding, Mr Subs

Reference Code

aoa_subs

Synopses

Shock

- The total pressure and temperature are actually calculated within this redirection to be thrown up a single workspace to be written out to the output file.

Expansion

- The root finding technique used for this function was the standard secant method. As the convergence for this implementation was on the order of 6 iterations to reach double precision no convergence acceleration term was deemed necessary. [52]
- To guarantee convergence for the root finding method the root had to bracketed. [52] This meant that in the expansion function a bracket was determined depending on if the expansion fan was centered on the center of the airfoil or the

leading/trailing edge.

Mach Wave

- This function takes in the same inputs and outputs as the expansion wave for a reason. This was to simplify the interface within the code but the Mach number is the same before and after the Mach wave therefore the variable is redistributed before the static pressure and temperature are calculated from the isentropic relations. [51]

C.5 Module: File Input

Purpose - By Individual Function

File Check

Opens the pointers onto the hard disk and checks that they exist before continuing into Read Data. This function also contains the option to run single values if no input file is detected.

Data Read

This functions checks the formatting within the input file to make sure that it is correct and then reads in the data within the input file.

Write Temp Pressure

Writes out the pressure and temperature data to designated output files.

Inputs

File Check

Ios Read, Iterate Choice

Data Read

Altitude, Iterate Choice

Write Temp Pressure

The following two primitive variables are read in for the four sides of the diamond airfoil

p, T

Outputs

File Check

Ios Read, Iterate Choice

Data Read

M_∞, θ, α, Iterate Choice

Write Temp Pressure

None

Modules Used

None

Reference Code

file_input

Synopses

File Check

- Writes the headers for the output files and also calls Data Read if the files exist with a design sweep designated.

Data Read

- The correct formatting requires 25 spaces between the variable and the value. This is hard coded into the algorithm and thus is necessary to follow.
- The arrays are read in as min, max and delta. From these values a linearly spaced vector is created to be used within the parameter sweep.
- Each of the array parameters as well as the size of the array is written out to the output files so that post processing the size of each becomes simpler.

Write Temp Pressure

- The values are written out in formatted Ascii and not binary. It was not deemed necessary to convert the output files to binary from the debugging formatted output for the final version.

C.6 Module: Mach Relations

Purpose

This function acts as the bifurcation point within the algorithm directing the program to the necessary functions depending on the angle of attack and half wedge angle for this instance call.

Inputs

T_∞,p_∞, ρ_∞, γ, π, M, α, θ, Iteration Choice

Outputs

The following two variables are output for all four sides

p, M

Modules Used

Negative AoA Subfunctions

Reference Code

mach_relatations

Synopses

- The parameters for the root finding subfunction are defined within the header of this function.
- Although this bifurcation slows the optimization for the processor stack it is deemed necessary and not able to be removed.

C.7 Module: Mr Subs - Performance Alpha Function

Purpose

This function provides single valued metrics for each of the parameter sweep runs including the lift and drag on the airfoil.

Inputs

θ, α, M, p for all four sides, π, γ, p_∞, Airfoil Thickness, Iterate Choice

Outputs

None

Modules Used

None

Reference Code

mr_subs

Synopses

- The Newtonian flat plate approximation is calculated for comparison to make sure that the calculated force balance makes sense
- A separate section is left for when $\theta = 0$ or the airfoil is in fact a flat plate
- The non-dimensionalized C_p lift to drag is also computed for comparison
- The last calculation is a vector balance per unit area which is the Drag and Lift used in comparing to the CFD results produced from VULCAN

Appendix D

Diamond Airfoil Convergence Study Plots

D.1 Euler Convergence Study

Following in Figures D.1 through D.3 is the additional plots for the Euler convergence study shown in Section 3.4. The figures show the stretching ratio and also the aspect ratio for the 65 x 129 point grid that was used within the study. The stretching ratio figures were considered a representative sample for the entire study and the aspect ratio was included for completeness. Please refer to section 3.4 for the aspect ratio of the chosen grid that varies slightly from the one shown following.

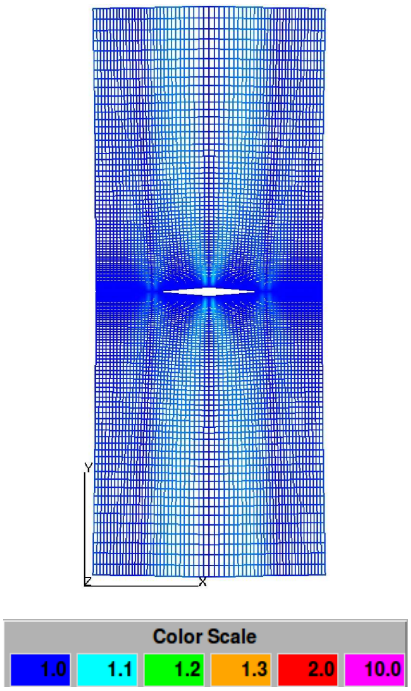


Figure D.1: Stretching Ratio in the J Direction for 65 x 129 Point Euler Grid.

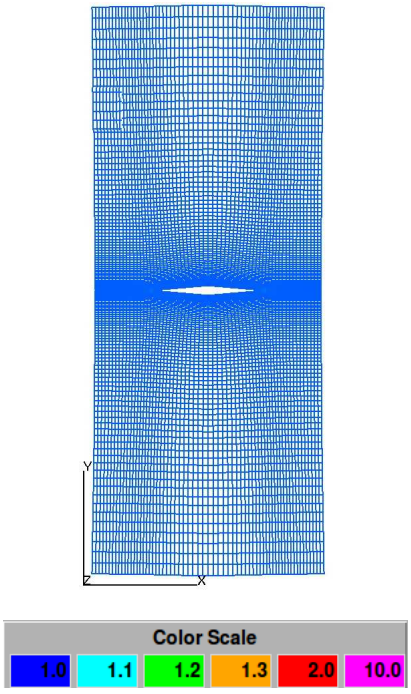


Figure D.2: Stretching Ratio in the K Direction for 65 x 129 Point Euler Grid.

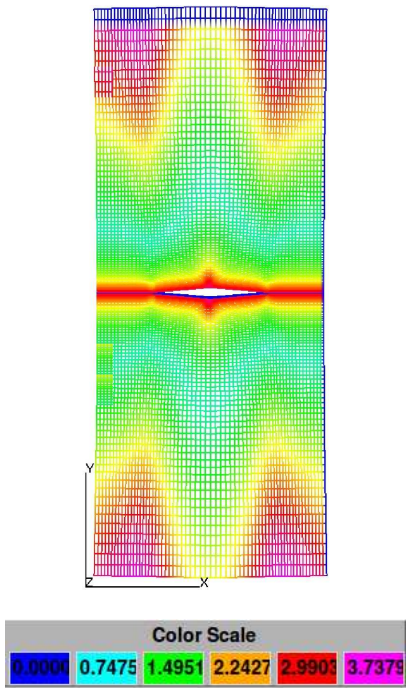


Figure D.3: Aspect Ratio for the 65 x 129 Point Euler Grid.

Appendix E

Script Automation of VULCAN

E.1 Input File Automation

In order to be able to create the large design spaces that were necessary of the work performed within this thesis automation of the input files and the meshes generated for the geometries chosen was necessary. This section will outline the input file automation algorithm that was used. Please refer to Section E.4 for the automation of the meshing software.

Also, it should also be noted that the program used for the text input into the bash environment that this algorithm and the one presented within Section E.2 was GNU awk.

Figure E.1 following shows the block diagram of the algorithm that is going to be explained. This section is going to use points to explain each of the steps within the block diagram and thus it is recommended that the reader refer to this figure for the remainder of this section.

(1) The first process carried out was the creation of the vectors to be sampled within the defined parameter space via file input into the algorithm. Section 2.4.1 within the main text has a full explication on the different sampling techniques and why the hypercube sampling technique was not used within this work.

(2) Having definitions on the bounds and the sampling to be carried out within the space, skeleton files are copied into a new directory bearing the definition of the parameter space. Defining each of the parameter spaces became important during the debugging phase and also in the creation of multiple spaces in order to optimize solution times for some of the

results being produced.

(3) The loops that are displayed within Figure E.1 are that of the diamond airfoil test case and thus are going to be referenced within this explanation. The small differences that were made for the scramjet combustor geometry will be described within Section E.3.

(4) Within the first loop is a call to a FORTRAN 95 program which is a standalone instance of the algorithm defined within Section C.3. Calling of this program allowed for the primitive variables input conditions to the mesh to be set within the skeleton input file.

(5) Within the third level of the diamond loops is a call to the meshing automation that is explained within Section E.4.

(6) Once at the bottom level of the looping structure the calculated input conditions were written into the skeleton file to be placed within the final solution point directory. This process was also done through gawk and was done for a standardized solver method. For the case of the diamond airfoil this was the space marching scheme with 5 levels of multigrid.

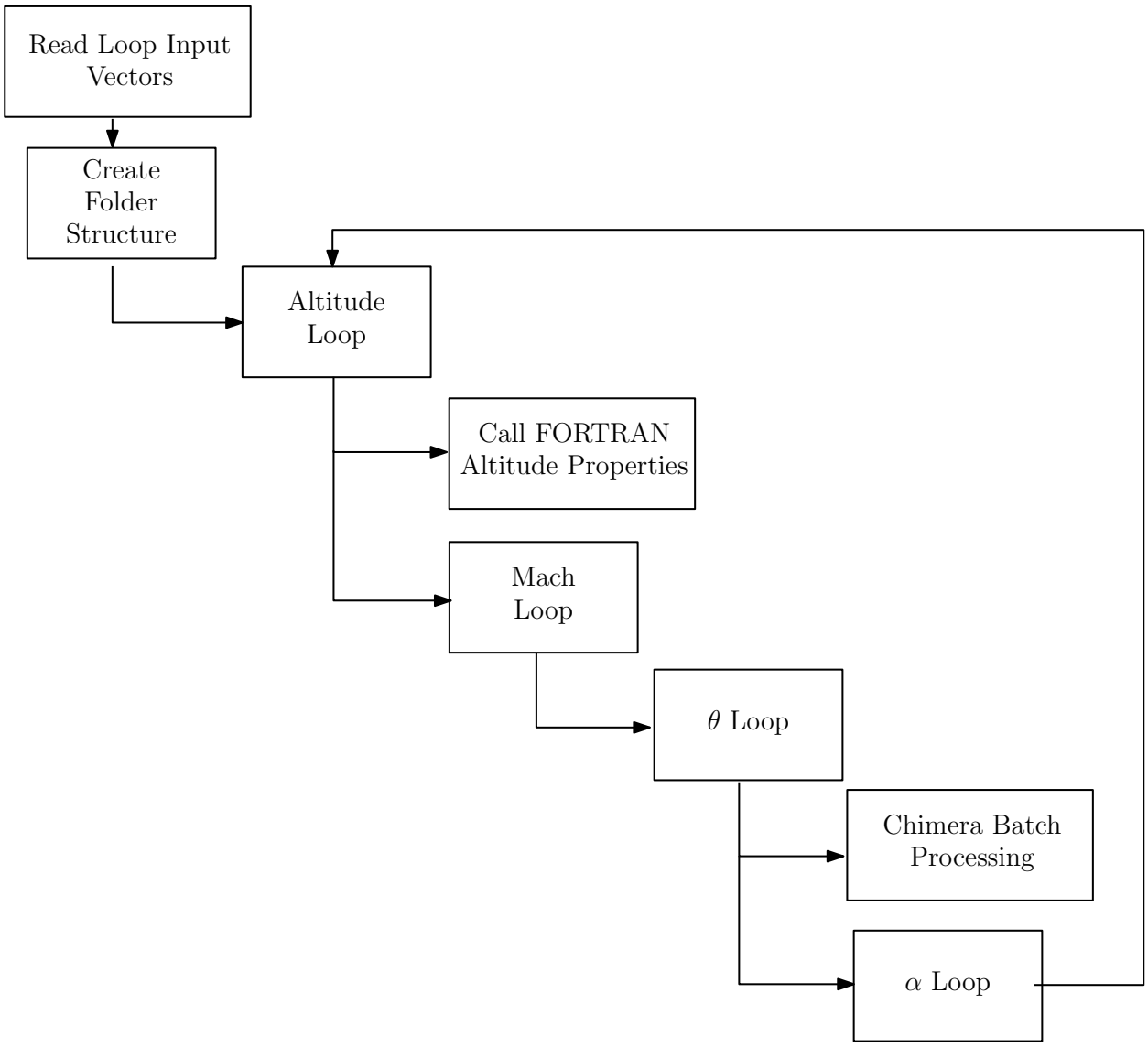


Figure E.1: VULCAN Script Automation Block Diagram.

E.2 Creation of a Custom Cluster

The computation of a single solution on for example, the viscous solution of the airfoil, took on the order of 50 minutes. With the requirement to be able to calculate large design spaces to prove the theory outlined within this thesis there became an apparent need for large amounts of computing power. Unable to widely distribute the VULCAN solver due to the ITAR restrictions(Section 2.6.1) placed on the program; a unique solution had to be adapted. This unique solution became the power to dynamically siphon computing resources off a custom built MPD cloud cluster built from the computers that were available within the Busemann Advanced Concepts Laboratory(BACL). The cloud controller was written in the Linux bash script environment, the same as the algorithm explained within the previous section. The use of this computing environment allowed for ssh tunneling through port 22 on all the computers to be used. The use of this tunneling technique removed the need to implement complex networking algorithms. Secondly, the decision to use dynamic allocating depending upon the CPU loading was done as the machines used within the custom MPD ring were not solely dedicated to this work and thus were also being used by other BACL employees concurrently.

Figure E.2 following shows the block diagram for the logic used within the algorithm. It is recommend, again, that the reader follow along with this figure as the algorithm is explained.

(1) To begin the entire cluster was pinged to verify that each of the computers was turned on and able to receive data. An exit condition was placed in the algorithm such that if any of the computers were turned off it would allow for the full ring to be booted up. From this statement it should be noted that this algorithm is very poor at off parameter conditions in general but this was not considered a problem due to the tight control that was able to placed upon the machines.

(2) The directory structure within the entire parameter space was imported into memory and calculated for the size and amount of points. This allows for a time estimate to be output for the solution of the entire design space. This estimate was based upon empirical data

taken during the debugging phase.

(3) The same as Section E.1, the looping structure for the test case is shown here. Please refer to Section E.3 for the differences in the algorithm that were made to accommodate for the scramjet combustor geometry.

(4) At the bottom of the parameter loops is where the distribution and calculations began. This began with a check of the CPU loading to verify which of the computers were over or underloaded. From here the VULCAN solver was soft linked into the folder instead of copied to allow for standardization of the solver script across all the solution points instigated.

(5) A sleep condition was initiated for when the full capacity was reached within the entire MPD ring. This sleep condition was placed here as this is right before when a solution instance is started and thus allows for the dynamic post processing of previous solutions that have completed. Within this sleep condition was a secondary sleep condition that accounted for other users present on the cloud machines as the main sleep loop is based upon the CPU loading.

(6) The resource that was used to track the backgrounded instances was the Linux Process Identification(PID) tracker used through the Linux top program. Thus, all the PID's for each of the instances initiated were tracked to verify if they were still present within the process environment on each of the cloud cluster machines being used. For completed PID's the tracker value was set to high and the processed data was secure copied(scp) back to the host hard drive. This was done as much to consolidate all the data created as much as the hard drive space on each of the cloud machines was already accounted for by other users.

(7) In order to be able to use the data output from Chimera the binary files had to be formatted for proper entry into the VULCAN environment. This is explained further in Section E.5.2.

(8) This is the initiation of a new instance on the machine with the least loading in a fall

through decision tree. It should be noted that all of the instances were backgrounded and not actively controlled through the controller due to the need for parallelization of post processing. Also, the entire data set for each instance was scped to the slave computer being allocated the instance to reduce lag in reading files through the LAN in the BACL.

(9) Completed data sets within the design space were secondary post processed. This included tracking of metrics for the design point and also tracking of the residual convergence through the solution point that is explained further within Section E.5.1.

(10) This block is outside of the main parameter loops and is a loop to sleep the controller until all the initiated instances are completed. This loop encompasses the PID check to follow and does not need to account for user loading.

(11) This PID check and scp are the same as within the parameter loops.

(12) All of the instances have completed and copied back to the host computer. Any of the instances that were not dynamically post processed are post processed here. There is a flag such that instances are not double processed within this function.

(13) The time metrics and force metrics are pulled for each of the instances. These metrics are the lower dimension projections of the results for each of the instances such that comparisons can be made between the different fidelity levels.

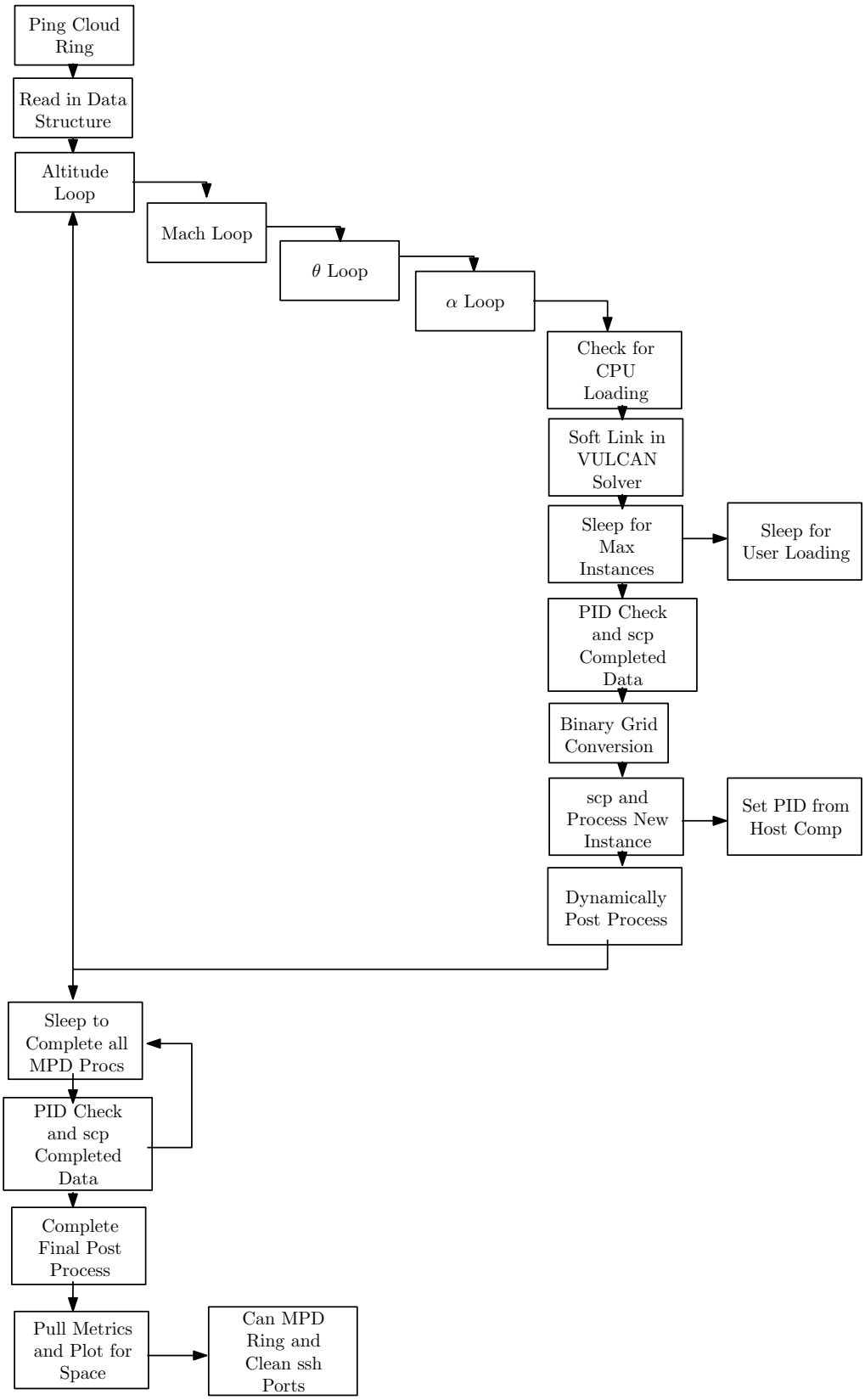


Figure E.2: Cloud Controller Block Diagram.

E.3 Scramjet Automation Differences

E.3.1 Input File Automation Differences

For the one dimensional analysis that was run for the scramjet combustor, the necessary alterations to the input files were all done on the chemical reaction file. This meant that no changes had to be made in the main VULCAN input file and thus this automation was simplified and done in MATLAB instead of bash. The same algorithm as seen in Figure E.1 was implemented but as only one dimension was iterated there was no need for the nested loops.

Secondly, the file structure that was produced had no need to be as recursive as that seen for the diamond airfoil as Tecplot Chorus was not used on this configuration. This reduction in file structure lead to a large difference in the overlord controller.

E.3.2 Cloud Controller Differences

Due to the mentioned file structure difference and also the single dimension sweep for the combustor this meant that the overlord cloud controller was greatly simplified. Single instances were designated to computers in the mpd ring but were not dynamically loaded or verified during the computation. This meant the post-processing did not occur on the fly while computing the set and also the total computation time for the sweep increased drastically. This increased time was considered acceptable due to the cost-benefit analysis performed on adding dynamic loading to the new file structure. Obviously, this analysis proved that non-dynamic loading would take less overall time when factoring in code development.

E.4 Batch Automation of Chimera

The only automated meshing that occurred was for the diamond airfoil test case. The scramjet combustor was considered to have enough parameters for a large parameter space without the need to generate new geometry. The automation algorithm will be explained following in the order that each of the functions were called within the overlord automation script.

(1) Automated Point Geometry Generation

MATLAB was used to read the desired θ angle from the overlord workspace and generate the appropriate wedge height. The airfoil chord was kept constant so that the wetted area of the airfoil would change with the respective wedge angle changes. This change in wetted area allowed for a true parameter design space instead of a design sweep of aerodynamic coefficients.

(2) Bunching of Boundary Points

SRAP was used with standard defined stretching ratio of 1.05 with 5 levels of multigrid enabled. By constraining the multigrid condition the VULCAN input files did not have to be altered for varying levels of multigrid capability.

(3) Concatenation of Entities

GRIDED was used for the concatenation of the entities and reversal of J directions to guarantee the correct surfaces were generated within HYPGEN.

(4) Hyperbolic Marching

HYPGEN was used with a hyperbolic tangent option and constant enforced end x coordinates. This enforcement meant some splaying of top symmetry condition which was not considered a major issue.

(5) Post Generation Analysis

To guarantee proper input into VULCAN the generated surfaces had to be organized into a single dimensional plane in the coordinate directions generated from HYPGEN. After this post analysis the MATLAB script was called to remove the the third dimension from the plot3d files to guarantee no errors were produced within VULCAN from the mesh input file.

E.5 MATLAB Batch Post Process

E.5.1 Batch Residual Plotting

The first of the two MATLAB batch processing routines worth noting was the automated residual plotting for each solution point. This automated plotting algorithm was completed in order to be able to check the convergence conditions of the individual solutions and also the history of the convergence within the solution. The data from the VULCAN output files was actually read through an gawk script but was controlled through MATLAB.

This algorithm was optimized to be able to handle both elliptically and space marched data so that the program could be used on both the diamond airfoil and the scramjet combustor. An example of the figures produced can be seen within Figure 3.15 and Figure 3.20 in Section 3.

E.5.2 Formatting Binary VULCAN Input Files

The second MATLAB automated algorithm that was written for this work was the automated editing of the binary mesh files that were output from Chimera for input into VULCAN. Chimera output binary files in a non-standard double configuration with a third dimension attached. To correct this an automated MATLAB algorithm was created to correct these errors and was called within the cloud controller as was described within E.2 to guarantee any auto-generated meshes were able to be properly formatted for input into VULCAN.