

A MODEL REIFICATION APPROACH TO FUSING INFORMATION FROM
MULTIFIDELITY INFORMATION SOURCES

A Thesis

by

WILLIAM DILLON THOMISON

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

Chair of Committee, Douglas Allaire
Committee Members, Daniel McAdams
Raymundo Arroyave
Head of Department, Andreas Polycarpou

May 2017

Major Subject: Mechanical Engineering

Copyright 2017 William Dillon Thomison

ABSTRACT

While the growing number of computational models available to designers can solve a lot of problems, it complicates the process of properly utilizing the information provided by each simulator. It may seem intuitive to select the model with the highest accuracy, or fidelity. Decision makers want the greatest degree of certainty to increase their efficacy. However, high fidelity models often come at a high computational expense. While comparatively lacking in veracity, low fidelity models do contain some degree of useful information that can be obtained at a low cost. We propose a method to utilize this information to generate a fused model with superior predictive capability than any of its constituent models. Our methodology estimates the correlation between each model using a model reification approach that precludes the need for observational data. The correlation is then used in an updating procedure whereby uncertain outputs from multiple models may be fused together to better estimate some quantity or quantities of interest.

DEDICATION

To my family and friends who have supported me.

ACKNOWLEDGMENTS

I would like to thank my advising professor, Dr. Douglas Allaire, for giving me the opportunity to conduct this research and providing exceptional mentorship. Special thanks to my colleagues in the Computational Design Laboratory, Fatemeh Ghoreishi, Benson Isaac, Samuel Friedman, and Brian Burrows.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supported by a thesis committee consisting of Professor Douglas Al-laire and Professor Daniel McAdams of the Department of Mechanical Engineering and Professor Raymundo Arroyave of the Department of Materials Science and Engineering.

Funding Sources

This work was supported by the AFOSR MURI on multi-information sources of multi-physics systems under Award Number FA9550-15-1-0038, program manager Dr. Jean-Luc Cambier, and by AFOSR grant FA9550-11-1-0339 under the Dynamic Data-Driven Application Systems Program, program manager Dr. Frederica Darema.

NOMENCLATURE

CFD	Computational Fluid Dynamics
FEA	Finite Element Analysis
V2V	Vehicle-to-Vehicle
CI	Covariance Intersection Algorithm
SU2	Stanford University Unstructured
MSE	Mean Squared Error
RANS	Reynolds-Averaged Navier Stokes
GPML	Gaussian Processes for Machine Learning
NASA	National Aeronautics and Space Administration
AGARD	Advisory Group for Aerospace Research and Development
C_L	Coefficient of Lift
C_D	Coefficient of Drag
M	Mach Number
α	Angle of Attack

TABLE OF CONTENTS

	Page
ABSTRACT	ii
DEDICATION	iii
ACKNOWLEDGMENTS	iv
CONTRIBUTORS AND FUNDING SOURCES	v
NOMENCLATURE	vi
TABLE OF CONTENTS	vii
LIST OF FIGURES	viii
LIST OF TABLES	x
1. INTRODUCTION AND LITERATURE REVIEW	1
2. BACKGROUND	4
2.1 Fusion Under Known Correlation	8
2.2 Fusion Under Unknown Correlation	11
3. METHODOLOGY	15
4. DEMONSTRATION	19
4.1 Analytical Example	21
4.2 1D CFD Demonstration	29
4.3 2D CFD Demonstration	34
5. SUMMARY AND CONCLUSIONS	41
REFERENCES	43

LIST OF FIGURES

FIGURE	Page
2.1 Relationship between two physics-based numerical simulations.	7
2.2 Trends in the fused estimate due to changes in variance and correlation coefficient. The fused model is the black solid line. Models 1 and 2 are blue and red, respectively.	9
2.3 Geometric interpretation of the Covariance Intersection algorithm.	12
2.4 The covariance ellipse for any choice of ω encloses the intersection of the covariance ellipses of the information sources.	14
4.1 Output from each simulator along with the truth model.	23
4.2 Gaussian process regression of uncertainty of Model 1.	24
4.3 Mean and two sigma confidence interval of Model 1 and Model 2.	25
4.4 Mean and two sigma confidence interval for the fused model using Covariance Intersection Algorithm, Zero Correlation Assumption, and Reification	27
4.5 Example outputs of NACA 0012 airfoil from XFOIL and SU2.	29
4.6 Coefficient of lift estimates from SU2, XFOIL, and wind tunnel data.	30
4.7 Gaussian process regression of uncertainty of SU2.	31
4.8 SU2 and XFOIL coefficient of lift predictions with two sigma confidence interval.	32
4.9 Results of no correlation case, Covariance Intersection algorithm, and the reification approach.	33
4.10 Gaussian process regression of SU2 coefficient of lift uncertainty with two sigma confidence interval mesh.	36
4.11 SU2 coefficient of lift predictions with two sigma confidence interval mesh.	37

4.12 Results of the reification method for C_L . XFOIL is yellow, SU2 is blue, truth is green, the reification method mean is red, and its two sigma confidence interval is the black mesh. 38

LIST OF TABLES

TABLE		Page
4.1	Mean squared error for Model 1, Model 2, fusion assuming zero correlation, the Covariance Intersection algorithm, and the reification approach. .	28
4.2	Mean squared error for XFOIL and SU2 without fusion, the no correlation case, the Covariance Intersection algorithm, and the reification approach. .	34
4.3	Mean squared error for XFOIL and SU2 without fusion, the no correlation case, the Covariance Intersection algorithm, and the reification approach. .	39

1. INTRODUCTION AND LITERATURE REVIEW*

Advancements in computing capability, mathematics, and a greater understanding of the physical world has facilitated the development of mathematical models that can simulate many natural systems. Computer modeling has become ubiquitous in the aerospace industry in response to the increasing complexity and performance requirements of today's aircraft. These simulators are incredibly useful for gaining insight into complex, dynamic systems such as those modeled using Computational Fluid Dynamics (CFD) and Finite Element Analysis (FEA). However, there is a problem with using computational models. As the statistician George Box famously stated, "All models are wrong, but some are useful." [1] This fact yields some degree of uncertainty of the output of a model. Minimizing this uncertainty is critical to achieving accurate predictions of real-world performance.

There are many sources of uncertainty that exist in the modeling process including parameter uncertainty, model inadequacy, residual variability, parametric variability, observation error, and code uncertainty [2]. Section 2.1 from Ref. [2] has defined these types of uncertainty as follows. Parameter uncertainty refers to the uncertainty associated with code inputs. Model inadequacy is attributed to the discrepancy between what is predicted by the model and the real-world results, even when the true values for all model inputs are known. Due to random variability in natural processes, the output is compared to the mean value of the real-world process. Parametric uncertainty is introduced by allowing the input parameters to vary according to some joint distribution. Observation error refers to the imperfection in measuring the real-world process. Code uncertainty is included in cases where it is impractical to run the code for every possible configuration of the input.

*Reprinted from "A Model Reification Approach to Fusing Information from Multifidelity Information Sources", 19th AIAA Non-Deterministic Approaches Conference, AIAA SciTech Forum, (AIAA 2017-1949), Copyright © 2017 by William D. Thomison and Douglas L. Allaire.

While relevant uncertainties will be accounted for, this paper will primarily be focused on reducing uncertainty due to model inadequacy.

For many applications, there exist multiple models that simulate the same system. These models typically rely on different mathematical formulae or make varying assumptions that simplify the problem. This leads to models with varying degrees of discrepancy from the true output, or fidelity. Generally speaking, a high fidelity model is more accurate than a low fidelity one at the cost of computational expense. Intuitively, using the high fidelity model leads to a reduction in uncertainty due to model inadequacy. However, a low fidelity model can still provide information that should be considered in any decision making process. The problem lies in appropriately utilizing this information in concert with the higher fidelity model. A method is needed to fuse the information from both models to create a single, fused model that can better predict some quantities of interest. Ideally, this method could fuse any number of multifidelity models to create a single model with greater predictive capabilities than any of its constituent models.

Information fusion is a broad field with diverse applications from imaging to control theory. Ref. [3] defines information fusion as “the study of efficient methods for automatically or semi-automatically transforming information from different sources and different points in time into a representation that provides effective support for human or automated decision making.” While the reification approach clearly falls into the information fusion research domain under this definition, it has a specific use case for reducing uncertainty due to model inadequacy in situations where multiple models are applicable. Techniques available for this application include the adjustment factors approach [4, 5, 6, 7], Bayesian model averaging [8, 9, 10, 11, 12, 13, 14], fusion under known correlation [15, 16, 17], and fusion under unknown correlation using Covariance Intersection [18, 19]. Our approach adapts the methods for fusion under known correlation to cases where correlation is unknown using model reification.

The term *reified analysis* was coined by Goldstein and Rougier in 2009 [20]. While our methodology shares a similar nomenclature and purpose, it is very different in its implementation. The word reify means to consider something to be real. As detailed in the following sections, our approach generates treats information generated by each model in turn as real-world data. It is in this regard that we consider the term reification an apt description of the method.

This paper outlines a methodology to create a fused distribution using data from models of varying fidelity. First, correlation between the models is estimated if it is unknown. Assuming that there is no existing data to estimate the correlation, synthetic data is generated by assuming one model to be the truth model. Given an estimate for correlation, traditional update rules can be used to create a fused distribution. There are many update rules used for combining information from multiple models. This method will make use of the method in Ref. [17]. Details of these methods are included in Section 2. Section 3 outlines our methodology. Demonstrations using three test problems are included in Section 4. Section 5 presents our conclusions drawn from this method and future work.

2. BACKGROUND*

For this paper we will specify the model as a physics-based numerical simulation of a real-world process. Models such as these cannot possibly be perfectly accurate, thus introduce varying degrees of uncertainty due to model inadequacy. We represent this model as a function $f(\mathbf{x})$, where \mathbf{x} is a set of inputs. This function f maps the inputs to some output y . To account for uncertainty, an additive term $\delta(\mathbf{x})$ is included. Model inadequacy is a primary influencer on this term, but other types of uncertainty play a role as well. In our demonstrations, observation error and code uncertainty are incorporated into $\delta(\mathbf{x})$. It should be noted that this value is dependent upon the inputs, so it can change over the input domain. The model can be represented by the equation

$$y = f(\mathbf{x}) = \bar{f}(\mathbf{x}) + \delta(\mathbf{x}), \quad (2.1)$$

where $\bar{f}(\mathbf{x})$ is the model output at a given input \mathbf{x} . It is also the mean for model $f(\mathbf{x})$ if $\delta(\mathbf{x})$ is unbiased.

While our method does not necessitate experimental data, it can be used to better quantify the model inadequacy. It then becomes important to add in observation error to the estimation of the total uncertainty of $f(\mathbf{x})$. An example of how this can be done is shown in the demonstrations in Section IV. However, if the model simulates a complex process, it may be difficult to reliably collect accurate experimental data to validate the model over the entire design space. This means that quantifying model inadequacy will often rely on expert elicitation. We assume that this leads to a normal distribution of $\delta(\mathbf{x})$ with some mean and variance. The variance reflects the fidelity of the model. High fidelity models

*Reprinted from "A Model Reification Approach to Fusing Information from Multifidelity Information Sources", 19th AIAA Non-Deterministic Approaches Conference, AIAA SciTech Forum, (AIAA 2017-1949), Copyright © 2017 by William D. Thomison and Douglas L. Allaire.

will generally have lower variance than models of lower fidelity. However, the discrepancy term's dependence upon the input can lead to areas in the design space where the lower fidelity model has a lower variance than the high fidelity model. Rigorous quantification of model inadequacy is an area of active research with many proposed solutions [21, 22].

Our method is to be used in situations where more than one model exists that is able to simulate a given quantity or quantities of interest. The information produced by these models will be fused such that the new uncertainty will be less than any of its constituent models. There are two ways to approach the combination of model outputs. The first is to believe that each model has some probability of being true. This leads to using methods such as Bayesian model averaging and adjustment factors approach to create a weighted average of the two models. The second approach is to assume that each model contains some amount of information that can be used to estimate the quantity of interest. This approach necessitates incorporating the correlation between the models.

It is our assertion that accounting for correlation is essential in information fusion. Assumed statistical independence can lead to overconfident estimates and skewed results in cases where redundant information was used by one of the sources [23]. This type of scenario happens frequently in the real-world. Imagine that a manager asks two engineers to estimate the cost for a project. Both of these engineers have the exact same education and have been on all of the same projects. It is likely that both of their estimates will be nearly identical, which leads the manager to be more confident in their decision given they have received two similar estimates. However, these estimates were created by two people with identical experience. Virtually no new information was provided to the manager by a second report, yet the managers confidence increased. This is a dangerous effect with broad applicability.

Consider this problem from a distributed data fusion perspective. Distributed data fusion refers to a system of interconnected sensor nodes without centralized data processing.

Instead, each individual node performs the estimation of the state space using the data provided by the nodes it is connected to. Sensors can be added to the network without the need for reconfiguring some centralized controller.

The major drawback to a distributed data fusion scheme is the redundancy problem [24]. Decentralized data fusion has a propensity for data to propagate back to sensors that have already used it. This means the data is double-counted and the estimate becomes corrupted. To better understand this problem, consider the following hypothetical system:

1. Two self-driving cars are equipped with a vehicle-to-vehicle (V2V) communication system that can share data from vehicle sensors, including data interpreted by the vehicles' identical cameras.
2. The camera from Vehicle 1 sees an anomaly at location x . The machine vision system reports a slight possibility that the anomaly is a cyclist. There are many different possibilities that the computer interprets from the image, but the vehicle's algorithm places a high importance on human avoidance. This triggers the V2V system to broadcast that there is a slight possibility of a cyclist at location x .
3. A similar camera on Vehicle 2 receives the message and scans location x . It sees a similar anomaly and confirms that there is something at location x , consistent with the proposal generated by Camera 1. However, Vehicle 2 is not aware that the confirmation was based on the same visual information processed by the same camera system. This leads Vehicle 2's computer to believe it has just independently confirmed the observation of a cyclist causing the V2V system to broadcast a high probability of a cyclist at location x .
4. Vehicle 1 receives information that there is a high probability of a cyclist at location x . The system believes this is a confirmation of its earlier proposal of a cyclist,

leading to immediate evasive action to avoid what was not actually a cyclist.

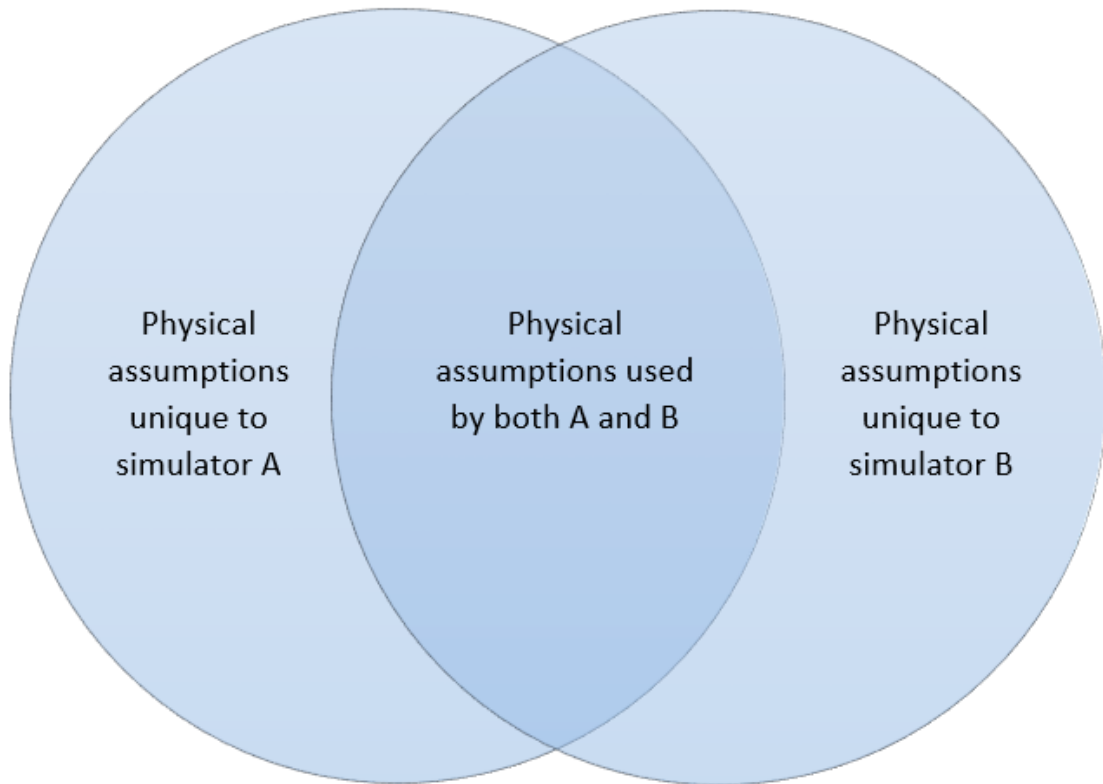


Figure 2.1: Relationship between two physics-based numerical simulations.

This hypothetical situation clearly shows how ignorance of the correlation of data can lead to overconfident assumptions and potentially dangerous actions. Due to the nature of simulating the same physical system, simulators may rely on the same formulae or physical assumptions. This redundant information is equivalent to the redundancy problem experienced by distributed data fusion and experiences the same overconfidence issue. A graphical representation of this relationship is shown in Fig. 2.1. This simplified representation is analogous to the basic statistical inclusion-exclusion principle,

$|A \cup B| = |A| + |B| - |A \cap B|$, which demonstrates the necessity of accounting for correlation. If you want to combine information from source A and B, the intersection of the two must be accounted for to avoid including redundant information.

2.1 Fusion Under Known Correlation

Although this method is capable of fusing information from any number of models, for simplicity we will assume the case where there exists two models $f_1(\mathbf{x}^*)$ and $f_2(\mathbf{x}^*)$ that both estimate the same quantity of interest. Following Eq. 2.1, $f_1(\mathbf{x}^*) = \bar{f}_1(\mathbf{x}^*) + \delta_1(\mathbf{x}^*)$ and $f_2(\mathbf{x}^*) = \bar{f}_2(\mathbf{x}^*) + \delta_2(\mathbf{x}^*)$, where \mathbf{x}^* represents a single input configuration. As shown in Refs. [18, 25], the standard deviation for each model corresponds to the following equations, $\tilde{f}_1(\mathbf{x}^*) = |f_1(\mathbf{x}^*) - \bar{f}_1(\mathbf{x}^*)|$ and $\tilde{f}_2(\mathbf{x}^*) = |f_2(\mathbf{x}^*) - \bar{f}_2(\mathbf{x}^*)|$. The fused models can be represented by the equation

$$y = k_1 f_1(\mathbf{x}^*) + k_2 f_2(\mathbf{x}^*), \quad (2.2)$$

where k_1 and k_2 are scalars such that $k_1 + k_2 = 1$. The purpose of this method is to reduce uncertainty in the output, so the scalars are chosen such that the variance of y is minimized. This is done by solving Eq. 2.3 for k_1 and k_2 assuming an unbiased, normal distribution for the uncertainty, $\delta(\mathbf{x}^*)$, such that $f_1(\mathbf{x}^*) \sim \mathcal{N}(\bar{f}_1(\mathbf{x}^*), \sigma_1^2)$ and $f_2(\mathbf{x}^*) \sim \mathcal{N}(\bar{f}_2(\mathbf{x}^*), \sigma_2^2)$.

$$\min_{\mathbf{k}} \mathbf{k}^T \Sigma \mathbf{k} \quad \text{subject to} \quad k_1 + k_2 = 1, \quad (2.3)$$

where

$$\mathbf{k} = [k_1, k_2]^T \quad \text{and}$$

$$\Sigma = \begin{bmatrix} \mathbb{E}[\tilde{f}_1(\mathbf{x}^*)^2] & \mathbb{E}[\tilde{f}_1(\mathbf{x}^*)\tilde{f}_2(\mathbf{x}^*)] \\ \mathbb{E}[\tilde{f}_2(\mathbf{x}^*)\tilde{f}_1(\mathbf{x}^*)] & \mathbb{E}[\tilde{f}_2(\mathbf{x}^*)^2] \end{bmatrix} = \begin{bmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix}.$$

Σ is the covariance matrix that correlates the two models. Recall that fusion under known correlation requires a known correlation coefficient, ρ . The solution of the minimization yields a fused model, y , that has a mean given by

$$\mathbb{E}[y] = \frac{(\sigma_2^2 - \rho\sigma_1\sigma_2)\bar{f}_1(\mathbf{x}^*) + (\sigma_1^2 - \rho\sigma_1\sigma_2)\bar{f}_2(\mathbf{x}^*)}{\sigma_1^2 + \sigma_2^2 - 2\rho\sigma_1\sigma_2}, \quad (2.4)$$

and a variance of y given by

$$\text{Var}(y) = \frac{(1 - \rho^2)\sigma_1^2\sigma_2^2}{\sigma_1^2 + \sigma_2^2 - 2\rho\sigma_1\sigma_2}. \quad (2.5)$$

The detailed proof for this set of equations is shown in Ref. [17].

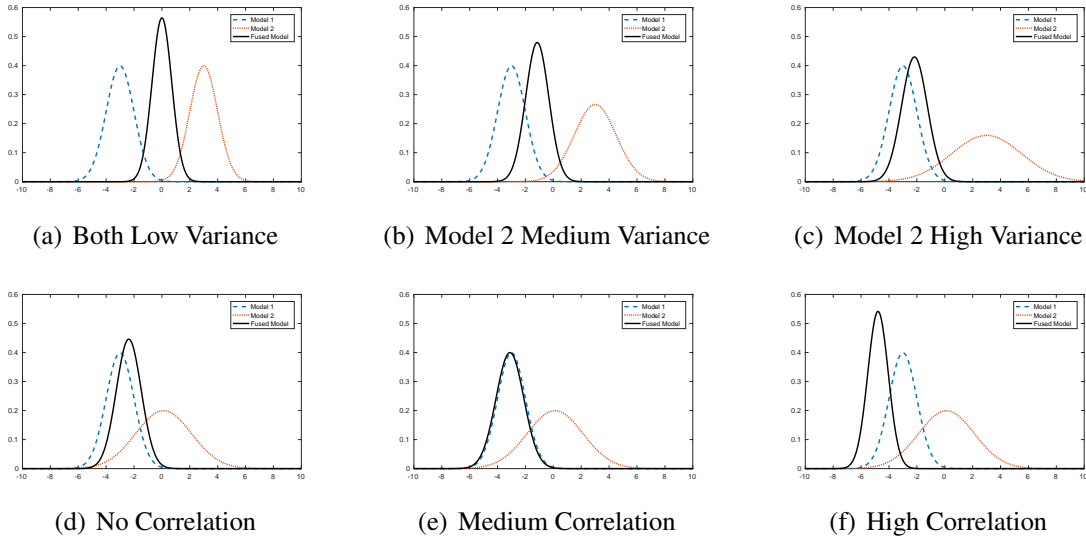


Figure 2.2: Trends in the fused estimate due to changes in variance and correlation coefficient. The fused model is the black solid line. Models 1 and 2 are blue and red, respectively.

An analysis of Eq. 2.4 and Eq. 2.5 reveals some interesting behaviors of this method

as illustrated in Fig. 2.2. The first row demonstrates what happens to the fused model as the variance of one model decreases. Note that the correlation coefficient is set to zero for this case. It is evident that the fused estimate tends toward the model of higher fidelity. Also, the confidence in the fused estimate decreases as the variance of the second model increases. The second row demonstrates the effect of the correlation coefficient when the variances are not equal. As the correlation coefficient increases, the fused estimate tends toward the model of higher fidelity. As the correlation continues to increase, the fused model moves beyond that of the higher fidelity model. This aligns with the intuition that models that are nearly identical would both produce estimates that are both either too high or too low. This is a unique trait of the fusion method that can only be leveraged when accounting for correlation. Another trend not illustrated is the effect of the correlation coefficient on models with similar variance. As the correlation coefficient increases, the fused variance increases. As the correlation approaches one, the variance is equal to that of its constituent models because no new information has been added.

It is important to note that, while this example focused on the use of two models, this method is generalized for any number of models. Ref. [17] also derives the generalized versions of Eq. 2.4 and Eq. 2.5, given by Eq. 2.6 and Eq. 2.7. The fused mean in the generalized approach is shown to be

$$\mathbb{E}[y] = \frac{\mathbf{e}^T \Sigma^{-1} \bar{\mathbf{f}}(\mathbf{x}^*)}{\mathbf{e}^T \Sigma^{-1} \mathbf{e}}, \quad (2.6)$$

where $\mathbf{e} = [1, \dots, 1]^T$, Σ^{-1} is the inverse of the covariance matrix, and $\bar{\mathbf{f}}(\mathbf{x}^*) = [\bar{f}_1(\mathbf{x}^*), \dots, \bar{f}_n(\mathbf{x}^*)]^T$, given n number of models used. The variance of the fused estimate is shown to be

$$\text{Var}(y) = \frac{1}{\mathbf{e}^T \Sigma^{-1} \mathbf{e}}. \quad (2.7)$$

2.2 Fusion Under Unknown Correlation

As shown in the previous section, the correlation coefficient plays a significant role in the distribution of the fused model. There are other methods that utilize correlation to perform information fusion. One should be selected to validate the reification approach. Traditionally, a Kalman filter would be used due to its ability to achieve the minimum mean squared estimates [26]. However, that property is contingent upon perfectly knowing the mean and covariance of the models. Covariance Intersection Algorithm (CI) was chosen for its remarkable consistency irrespective of actual correlation.

This generalized method forms its estimates using the convex combination of means and covariances in the inverse covariance space [18]. It can also account for models with multiple quantities of interest. Similar to the previous section, we will consider two models $f_1(\mathbf{x}^*)$ and $f_2(\mathbf{x}^*)$ with means $\bar{f}_1(\mathbf{x}^*)$ and $\bar{f}_2(\mathbf{x}^*)$ and deviations $\tilde{f}_1(\mathbf{x}^*)$ and $\tilde{f}_2(\mathbf{x}^*)$. The covariance matrix, $\tilde{\mathbf{P}}$, is redefined as follows

$$\tilde{\mathbf{P}} = \begin{bmatrix} \mathbb{E}[\tilde{f}_1(\mathbf{x}^*)\tilde{f}_1(\mathbf{x}^*)^T] & \mathbb{E}[\tilde{f}_1(\mathbf{x}^*)\tilde{f}_2(\mathbf{x}^*)^T] \\ \mathbb{E}[\tilde{f}_2(\mathbf{x}^*)\tilde{f}_1(\mathbf{x}^*)^T] & \mathbb{E}[\tilde{f}_2(\mathbf{x}^*)\tilde{f}_2(\mathbf{x}^*)^T] \end{bmatrix}.$$

Unlike fusion under known correlation, CI utilizes a geometric interpretation called the covariance ellipse [25]. The ellipse is created using the inequality

$$\mathcal{B}_Q(c) \triangleq \{\mathbf{x} : \mathbf{x}^T Q^{-1} \mathbf{x} < c\}, \quad (2.8)$$

where Q is a positive definite matrix, and $\mathcal{B}_Q(c)$ is the boundary of the ellipse at level c . Thus, if $Q_1 < Q_2$, then $\mathcal{B}_{Q_1}(c) \subset \mathcal{B}_{Q_2}(c)$. To use the graphical interpretation of Covariance Intersection, ellipses are first plotted for $\tilde{\mathbf{P}}_{11}$ and $\tilde{\mathbf{P}}_{22}$ for a given level c . The covariance ellipse for $\tilde{\mathbf{P}}_{yy}$ is plotted, where $\tilde{\mathbf{P}}_{yy}$ is the covariance of the fused estimate.

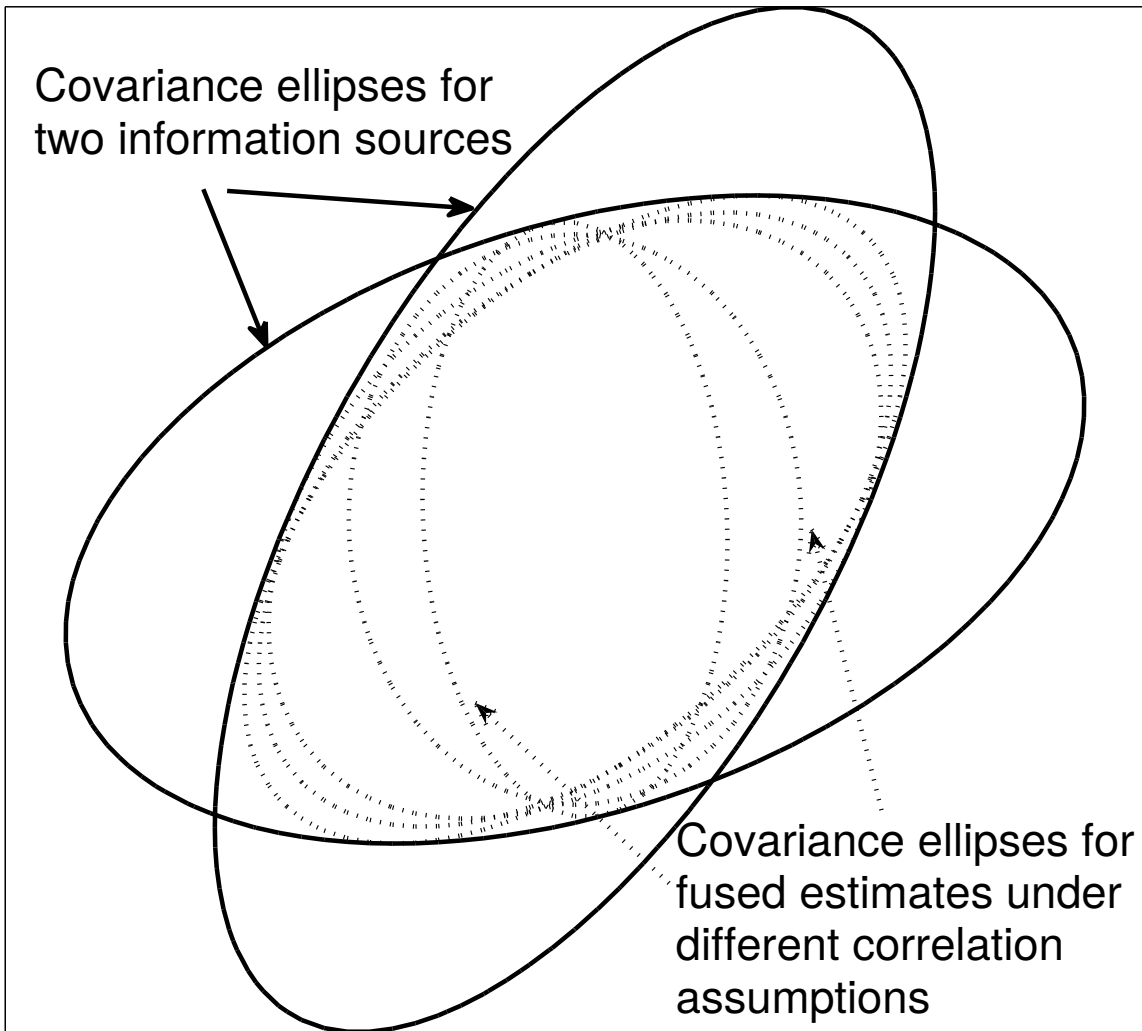


Figure 2.3: Geometric interpretation of the Covariance Intersection algorithm.

This should always be a subset of both $\tilde{\mathbf{P}}_{11}$ and $\tilde{\mathbf{P}}_{22}$, for all values of $\tilde{\mathbf{P}}_{12}$. This is shown graphically by the $\tilde{\mathbf{P}}_{yy}$ ellipse remaining inside the intersections of the $\tilde{\mathbf{P}}_{11}$ and $\tilde{\mathbf{P}}_{22}$ ellipses. An illustration of this approach is shown in Fig. 2.3.

Constructing a linear update rule is necessary to fuse information sources such that the fused estimate ellipse encloses all of the possible ellipses for the fused estimates under any correlation assumption. The closer this updated ellipse encloses the intersecting region, the

more information is incorporated into the estimate. This condition ensures the correlation coefficient is conservatively estimated, which is the primary benefit of this method over Kalman filtering. Ref. [18] defines the update equations as

$$\tilde{\mathbf{P}}_{\mathbf{y}\mathbf{y}}^{-1} = \omega\tilde{\mathbf{P}}_{11}^{-1} + (1 - \omega)\tilde{\mathbf{P}}_{22}^{-1} \quad (2.9)$$

$$\tilde{\mathbf{P}}_{\mathbf{y}\mathbf{y}}^{-1}\mathbb{E}[\mathbf{y}] = \omega\tilde{\mathbf{P}}_{11}^{-1}\bar{\mathbf{f}}_1(\mathbf{x}^*) + (1 - \omega)\tilde{\mathbf{P}}_{22}^{-1}\bar{\mathbf{f}}_2(\mathbf{x}^*), \quad (2.10)$$

where $\omega \in [0, 1]$ controls the weights applied to each model. This update ensures that the covariance ellipse of $\tilde{\mathbf{P}}_{\mathbf{q}\mathbf{q}}$ always encloses the intersection of $\tilde{\mathbf{P}}_{11}$ and $\tilde{\mathbf{P}}_{22}$, as shown in Fig. 2.4 for varying values of ω . ω is chosen such that the trace or determinant of $\mathbf{P}_{\mathbf{y}\mathbf{y}}$ is minimized. We chose to minimize the trace for our comparison to this approach.

Due to the nature of the Covariance Intersection algorithm's conservative update rule, it has a tendency to discard lower fidelity information sources to achieve the most conservative estimate possible in cases with a single quantity of interest. It requires that the information source with the highest fidelity be used for the fused estimate. While this does allow the best model to be chosen at all points in the design space, this approach leads to a loss of information that could be used to better estimate the model output if the correlation coefficients were known. The next section outlines our methodology for estimating correlation and creating a fused distribution.

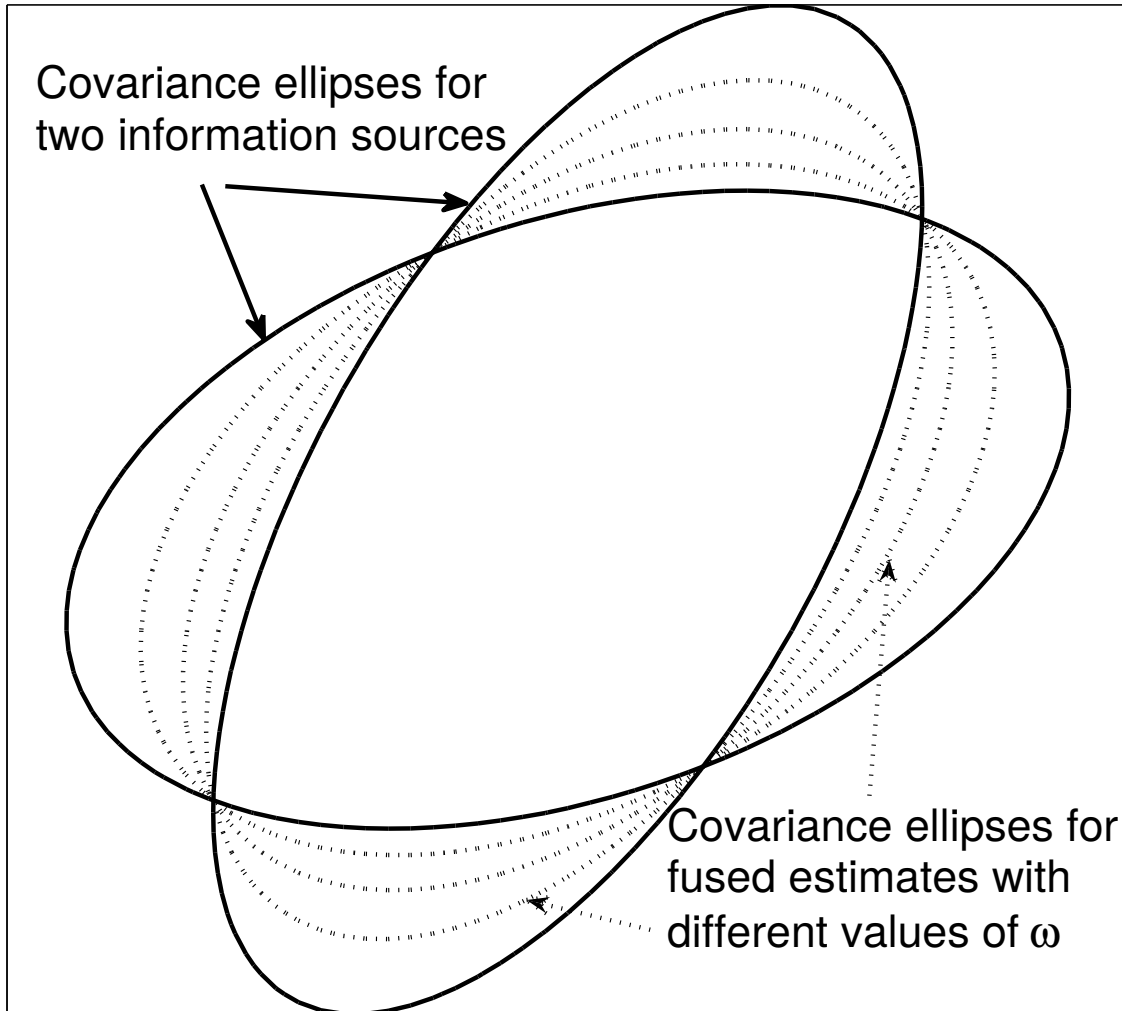


Figure 2.4: The covariance ellipse for any choice of ω encloses the intersection of the covariance ellipses of the information sources.

3. METHODOLOGY*

This section details our methodology for the complete process of fusing information from multiple models. We assume no observational data is available to estimate correlation. To mitigate this problem we use model reification to generate synthetic data. As discussed previously, model reification refers to the process of treating one of the models as the truth. This means that we assume the data generated by the model represents the true quantities of interest. We will use this data to estimate the correlation between the errors of different models. The process is then repeated for each model.

We will again assume the case where only two models, $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$, exist for estimating some quantity of interest, y . This can be estimated by using $y = \bar{f}_1(\mathbf{x}^*) + \delta_1(\mathbf{x}^*)$ or $y = \bar{f}_2(\mathbf{x}^*) + \delta_2(\mathbf{x}^*)$. In this instance, the uncertainty is assumed to be an unbiased, normal distribution given by $\delta_1(\mathbf{x}^*) \sim \mathcal{N}(0, \sigma_1^2)$ and $\delta_2(\mathbf{x}^*) \sim \mathcal{N}(0, \sigma_2^2)$. Given this information and some input configuration \mathbf{x}^* we can estimate the correlation of the errors and use Eqs. 2.4 and 2.5 to estimate the mean and variance of the new fused estimate.

We estimate the error of each model with respect to the true quantity of interest using reification. Assuming simulator $f_1(\mathbf{x})$ is chosen to reify, the deviation for the other model can be calculated using the following equations

$$\tilde{f}_1(\mathbf{x}^*) = f_1(\mathbf{x}^*) - \bar{f}_1(\mathbf{x}^*) = \delta_1(\mathbf{x}^*) \quad (3.1)$$

$$\tilde{f}_2(\mathbf{x}^*) = f_2(\mathbf{x}^*) - \bar{f}_2(\mathbf{x}^*) \quad (3.2)$$

$$= \bar{f}_1(\mathbf{x}^*) - \bar{f}_2(\mathbf{x}^*) + \delta_1(\mathbf{x}^*), \quad (3.3)$$

*Reprinted from "A Model Reification Approach to Fusing Information from Multifidelity Information Sources", 19th AIAA Non-Deterministic Approaches Conference, AIAA SciTech Forum, (AIAA 2017-1949), Copyright © 2017 by William D. Thomison and Douglas L. Allaire.

where Eq. 3.3 follows from the fact that if we reify simulator 1, the model inadequacy of simulator 2 is given as $\delta_2(\mathbf{x}^*) = \bar{f}_1(\mathbf{x}^*) - \bar{f}_2(\mathbf{x}^*) + \delta_1(\mathbf{x}^*)$. The mean squared errors are then given as

$$\mathbb{E}[\tilde{f}_1(\mathbf{x}^*)^2] = \mathbb{E}[\delta_1(\mathbf{x}^*)] = \sigma_1^2 \quad (3.4)$$

$$\mathbb{E}[\tilde{f}_2(\mathbf{x}^*)^2] = \mathbb{E}[(\bar{f}_1(\mathbf{x}^*) - \bar{f}_2(\mathbf{x}^*))^2] + \mathbb{E}[\delta_1(\mathbf{x}^*)] \quad (3.5)$$

$$= (\bar{f}_1(\mathbf{x}^*) - \bar{f}_2(\mathbf{x}^*))^2 + \sigma_1^2, \quad (3.6)$$

and the covariance is given as

$$\mathbb{E}[\tilde{f}_1(\mathbf{x}^*)\tilde{f}_2(\mathbf{x}^*)] = \sigma_1^2. \quad (3.7)$$

Thus, under the assumption that simulator 1 has been reified, the correlation coefficient is given as

$$\rho_1(\mathbf{x}^*) = \frac{\sigma_1^2}{\sigma_1\sigma_2} = \frac{\sigma_1}{\sqrt{((\bar{f}_1(\mathbf{x}^*) - \bar{f}_2(\mathbf{x}^*))^2 + \sigma_1^2)}}. \quad (3.8)$$

The subscript under the correlation coefficient denotes which model was reified. Under these assumptions, the other information available to assess model fidelity is the model inadequacy. To account for the uncertainty in which model is the highest fidelity at a particular input configuration, model 2 is also reified to estimate $\rho_2(\mathbf{x}^*)$. The average correlation between the two models can then be estimated using a variance weighted sum as follows

$$\bar{\rho}(\mathbf{x}^*) = \frac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2}\rho_1(\mathbf{x}^*) + \frac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2}\rho_2(\mathbf{x}^*). \quad (3.9)$$

This average correlation can then be used in Eqs. 2.4 and 2.5 to estimate the mean and variance of the new fused estimate.

It should be noted that all calculations were performed using a precise input configura-

tion, \mathbf{x}^* . However, such precision of input variables into a model is unrealistic. Estimates in the region surrounding the input configuration \mathbf{x}^* could provide useful information to describe the true correlation between models. There is some possibility that the mean and variance of the quantity of interest output by both simulations are identical at \mathbf{x}^* . This would produce a correlation coefficient of 1, which is highly unlikely to be true at all points in the surrounding region of the input space. It would also add no information to the fusion process. Another benefit to interrogating the region surrounding the point is to account for holes in the input space. For example, many complex models will have input configurations that cause the simulator to fail and yield no output. To mitigate these risks, the correlation will be averaged in a region surrounding \mathbf{x}^* . The optimal size of this neighborhood is a topic of future work. We assume that the size is known.

Applying this to Eq. 3.8, the new estimate for correlation after the reification of model 1 is

$$\begin{aligned}\rho_1(\mathcal{S}(\mathbf{x}^*)) &= \mathbb{E}_{\mathcal{S}(\mathbf{x}^*)}[\rho_1(\mathbf{x})] \\ &= \mathbb{E}_{\mathcal{S}(\mathbf{x}^*)} \left[\frac{\sigma_1}{\sqrt{((f_1(\mathbf{x}) - f_2(\mathbf{x}))^2 + \sigma_1^2)}} \right],\end{aligned}\tag{3.10}$$

where $\mathcal{S}(\mathbf{x}^*)$ is the region surrounding \mathbf{x}^* . For a model with few input variables, this value can be calculated using standard integration methods. However, the integral is subject to the curse of dimensionality, making it intractable as the number of inputs increases. We recommend using a Monte Carlo approach to obtain the solution. The steps for this method are shown in Algorithm 1. The average correlation coefficient can then be used with the fusion under known correlation method discussed in Section 2.1.

It should be noted that the methodology shown was assumed to be two models to simplify the explanation, but the method is generalizable to any number of models. We also simplified the model inadequacy. Variance can be added throughout the input space

Algorithm 1: Correlation Estimation with using Monte Carlo

- 1: Uniformly sample $f_1(\mathbf{x})$ m times in $\mathcal{S}(\mathbf{x}^*)$
 - 2: Add model inadequacy samples $[\delta(\mathbf{x}^1), \dots, \delta(\mathbf{x}^m)]$ to the simulator samples $[f_1^1(\mathbf{x}^1), \dots, f_1^1(\mathbf{x}^m)]$
 - 3: Calculate $\rho_1^i(\mathbf{x}^i) = \frac{\sigma_1}{\sqrt{((\bar{f}_1(\mathbf{x}^i) - \bar{f}_2(\mathbf{x}^i))^2 + \sigma_1^2)}$ for $i = 1$ to m
 - 4: Estimate $\rho_1(\mathcal{S}(\mathbf{x}^*)) \approx \frac{1}{m} \sum_{i=1}^m \rho_1^i(\mathbf{x}^i)$
 - 5: Repeat steps 1 - 4 to estimate $\rho_2(\mathcal{S}(\mathbf{x}^*))$
 - 6: Calculate $\bar{\rho}(\mathcal{S}(\mathbf{x}^*)) = \frac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2} \rho_1(\mathcal{S}(\mathbf{x}^*)) + \frac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2} \rho_2(\mathcal{S}(\mathbf{x}^*))$
-

by simply adding the dependence on \mathbf{x} in Eqs. 3.6 and 3.10. The precise specification of this variance can be done through expert elicitation or experimental data. Although experimental data is not necessary for the reification process, it can be used to validate the models and aid in the quantification of model inadequacy. The following section is a demonstration of the method.

4. DEMONSTRATION*

In this section we present the results of three demonstrations of our methodology. The first is a simple analytical problem with one-dimensional input and output. The second and third demonstrations use data from two computational fluid dynamics simulators, XFOIL and Stanford University Unstructured (SU2). One demonstration has a single input and output, while the other has two-dimensional input and output. Details regarding these simulators and their implementation are discussed in Section 4.2.

There are several assumptions and caveats to these examples that must be discussed. In order to quantify our method's predictive capability, the true value of the quantity of interest estimated by the simulators must be known. We chose to specify some "truth model" for each of these demonstrations that can be compared against to determine the method's ability to create a fused model that better predicts the true quantity of interest. The creation of this model is discussed for each example. It is important to note that such a model would never exist in real-world situations. Access to a perfect model precludes the need for other simulators.

Our methodology presumes expert elicitation will be used to quantify model inadequacy. This is not possible in the case of an analytical test problem or in situations where no expert is available. For demonstrations 2 and 3, we do not have the expertise to make accurate estimates of the fidelity level of XFOIL and SU2. Given that we specify a model that represents the true quantities of interest, this data can also be used to consistently estimate model inadequacy for each simulator. However, it would be unrealistic to have complete knowledge of the model inadequacy at all points in the design space. To keep

*Reprinted from "A Model Reification Approach to Fusing Information from Multifidelity Information Sources", 19th AIAA Non-Deterministic Approaches Conference, AIAA SciTech Forum, (AIAA 2017-1949), Copyright © 2017 by William D. Thomison and Douglas L. Allaire.

the demonstration as realistic as possible, the discrepancy between the models' output and the true values was examined at only a few points in the design space. This mimics the scenario where some experimental data is available to validate the models. A Gaussian process was then employed to perform a regression over the design space using those values as training points.

A Gaussian process is a powerful statistical modeling tool based on the principles of Gaussian distributions. A Gaussian process can be thought of as a generalized version of the Gaussian distribution that is applied over a continuous input space. Put simply, it is an infinite set of normal distributions where each set of inputs has a corresponding distribution that can be obtained using mean and covariance functions [27]. Gaussian processes were first proposed as a regression technique by O'Hagan in 1978 [28]. Gaussian process regression, much like standard regression analysis, is the process of fitting a function to represent a given dataset. The field of geostatistics coined the term *kriging* to describe this method [29]. The benefit of using a Gaussian process is that it determines the mean and variance at every point in a continuous design space while being computationally tractable. When working with sparsely populated data sets a Gaussian process can account for the changing variance in the probability distributions where data is unavailable, as is the case for most computationally expensive models.

For our demonstrations the Gaussian Processes for Machine Learning (GPML) MATLAB toolbox, version 3.6 was used to perform the regression [27]. This code requires the specification of a mean function, covariance function, likelihood function, inference method, and all corresponding hyperparameters, which are parameters that control the behavior of each function. The specific functions and hyperparameters for each demonstration will be discussed in the following subsections.

Previously we defined six types of uncertainty: parameter uncertainty, model inadequacy, residual variability, parametric variability, observation error, and code uncertainty [2].

We assume that only three of these are relevant to these demonstrations: model inadequacy, observation error, and code uncertainty. The primary focus of this research is motivated by the fact that models do not perfectly describe the true system, thus model inadequacy must be considered. To mimic the scenario where experimental data is used to validate the models, observation error is added to account for the inability to perfectly measure a real-world process. Due to the models being run only at points where experimental data was available, code uncertainty was added to account for the interpolation done between the data points. It is our assumption that the other three types of uncertainty are not as relevant to the scope of our research.

All of these assumptions were made to fairly and consistently estimate the discrepancy term, $\delta(\mathbf{x}^*)$, for each model. This term can be biased or unbiased, meaning its mean could be non-zero or zero, respectively. A biased discrepancy term would allow the model output to be adjusted up or down to better fit the truth. An unbiased discrepancy term would be of mean zero and only affect the variance of the model. While both may be used, it is our assumption unbiased estimations are much more feasible in practice. Experts can simply claim that a model loses accuracy in some area of the domain, rather than know the direction and magnitude of the model's deviation from truth. For this reason, the unbiased estimator was chosen for the demonstrations. This also makes the problem more challenging as information about the error is unused and will be a good test of the capability of the model reification approach.

4.1 Analytical Example

We consider two simulators with a one-dimensional input. A "truth model" is specified to estimate the model inadequacy and demonstrate how well the method can predict the true quantity of interest. We compare the results of our reification approach to the constituent models, the Covariance Intersection algorithm, and the assumption of zero

correlation in the fusion process.

We specify the truth model as

$$truth(\mathbf{x}) = \sin(\mathbf{x}) + noise, \quad (4.1)$$

where $noise \sim \mathcal{N}(0, 0.1^2)$. Gaussian noise is added to more realistically simulate a real-world process. This will be incorporated into the discrepancy term in the form of observation error. The two models are defined as

$$y = f_1(\mathbf{x}) = \bar{f}_1(\mathbf{x}) + \delta_1(\mathbf{x})$$

$$y = f_2(\mathbf{x}) = \bar{f}_2(\mathbf{x}) + \delta_2(\mathbf{x}),$$

where

$$\bar{f}_1(\mathbf{x}) = (\mathbf{x} - 0.1) - \frac{(\mathbf{x} - 0.1)^3}{3!} + \frac{(\mathbf{x} - 0.1)^5}{5!}$$

$$\bar{f}_2(\mathbf{x}) = (\mathbf{x} + 0.3) - \frac{(\mathbf{x} + 0.1)^3}{3!} + \frac{(\mathbf{x} - 0.1)^5}{5!}.$$

The calculation of δ_1 and δ_2 is discussed below.

These are simply modified versions of the Taylor series approximation of $\sin(\mathbf{x})$. The domain of \mathbf{x} is limited to $-\pi \leq \mathbf{x} \leq \pi$. An illustration of the truth model and two simulators is shown in Fig. 4.1. In this case, Model 1 could be considered the high fidelity model. However, Model 2 becomes much more accurate than Model 1 when $\mathbf{x} > 2$. It is evident that using only the high fidelity model would lead to significantly less accurate information than a fused model.

Accurately imitating a real-world example was an important consideration when quantifying the models' uncertainty. Given the fact that no real-world truth models exist, perfectly quantifying the model inadequacy at every point would be unrealistic. Instead, we

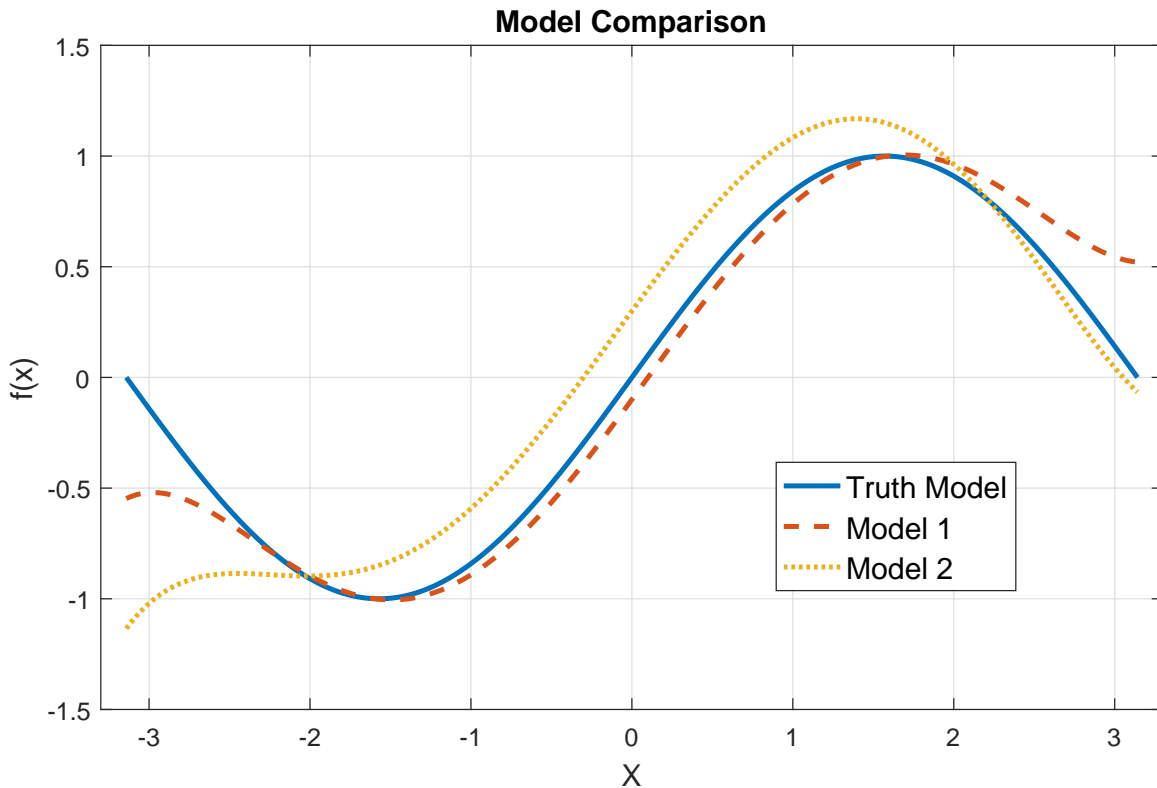


Figure 4.1: Output from each simulator along with the truth model.

chose to uniformly select 20 points in the input space to calculate the discrepancy by taking the difference between the truth model and the simulator. This mimics a situation where a limited amount of experimental data is used to validate a simulator. As discussed previously, observation error is inherent in the collection of experimental data, so that must be incorporated into the model's uncertainty. Code uncertainty was introduced in the areas between the 20 points where the models were not run.

The construction of the unbiased estimation of each model's uncertainty begins with calculating the model inadequacy at each of the 20 points selected previously, such that $model\ inadequacy = |truth(\mathbf{x}^*) - f(\mathbf{x}^*)|$. We take the absolute value to create an unbi-

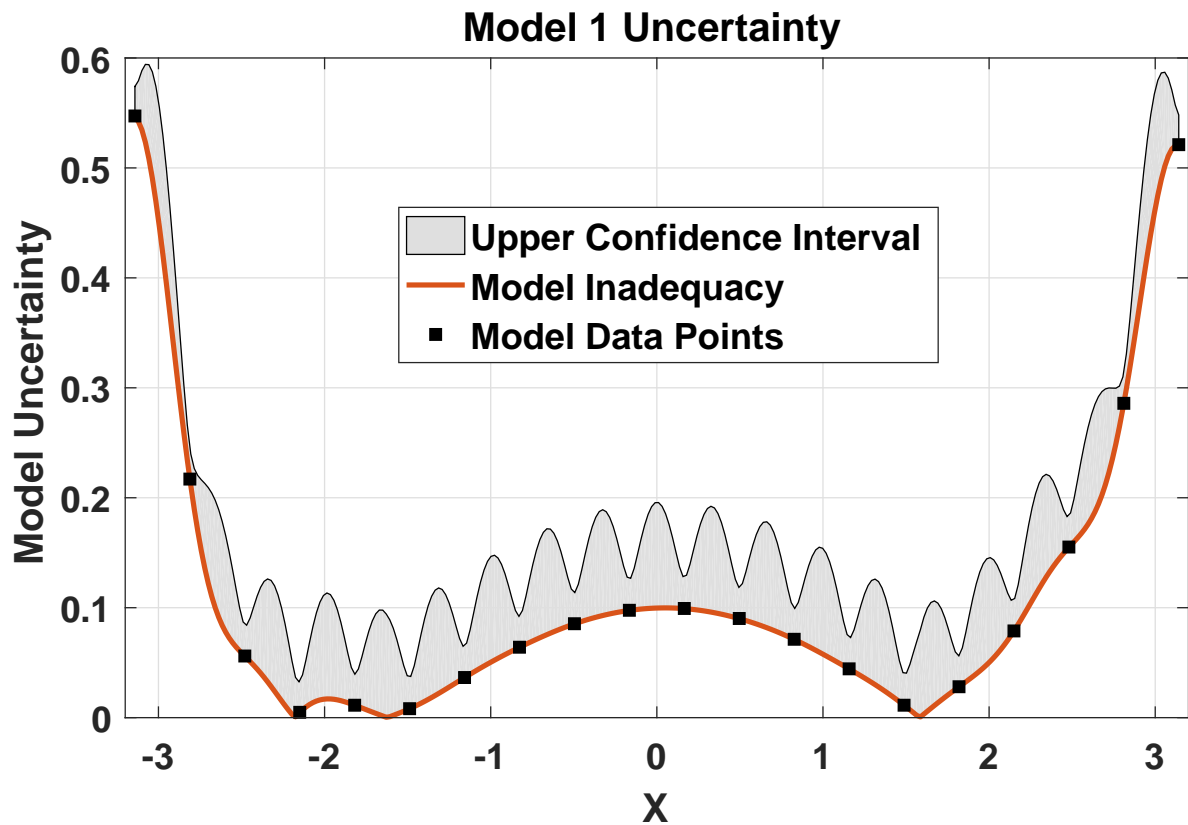


Figure 4.2: Gaussian process regression of uncertainty of Model 1.

ased estimator that ignores whether the true value is less than or greater than the model’s output. You can see each of these points represented by black squares in Fig. 4.2.

A Gaussian process regression is then performed using these model inadequacy values as training points to estimate the model’s overall uncertainty at all points in the design space. As discussed previously, a mean function, covariance function, likelihood function, and inference method must be specified. For this demonstration the mean function was simply set to zero so that it could be ignored. The covariance function was specified as squared exponential with isotropic distance measure. The hyperparameters for this function are the characteristic length scale and signal standard deviation. After some trial and error, the characteristic length scale was set to 0.25. We discovered that a good rough

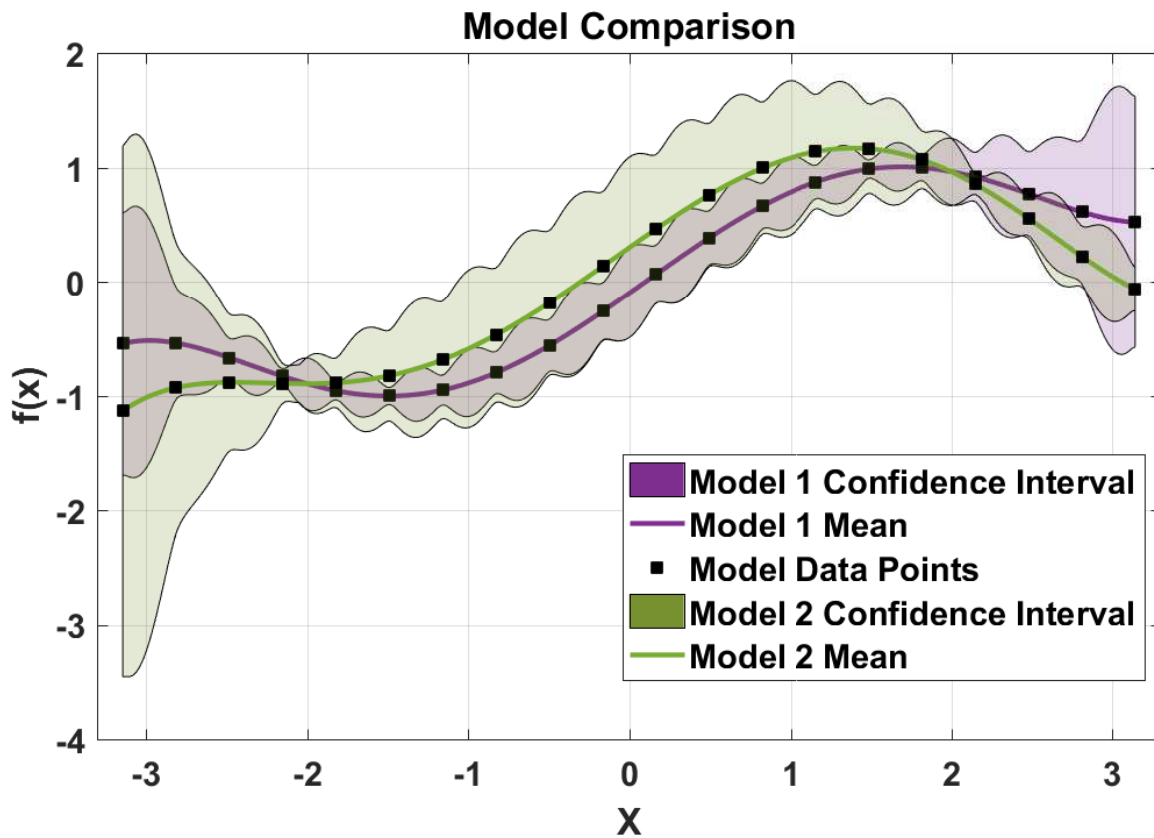


Figure 4.3: Mean and two sigma confidence interval of Model 1 and Model 2.

estimate for this number is the distance between the model data points. The signal standard deviation controls the variance added between each data point due to code uncertainty and was assumed to be 0.2. The likelihood function was chosen to be Gaussian to account for the observation error that is assumed to be Gaussian. The hyperparameter, noise standard deviation, was set to 0.1. Finally, the inference method was specified as exact.

Using these functions and hyperparameters, a plot of Model 1's uncertainty was produced as shown in Fig. 4.2. The red line is the regression through each model inadequacy data point. The gray shaded area represents the upper half of a two sigma confidence interval. This confidence interval represents the other two types of uncertainty. Note how

the confidence interval increases between each data point. This is a visual representation of the code uncertainty. Also, the confidence interval never goes to zero at the data points. This is due to observation error.

The overall uncertainty of Model 1 can be calculated from the information in this plot. The overall uncertainty will consist of a sum of two variances. The first is the variance due to model inadequacy defined as $|truth(\mathbf{x}^*) - f(\mathbf{x}^*)|^2$. The second is the variance of the Gaussian process, which is a combination of the other two uncertainties and will be denoted by σ_{GP}^2 . It is our assumption that calculating the overall uncertainty in this way is an adequate estimation that remains consistent and fair for all demonstration models. The overall model uncertainty is defined as

$$\sigma_{model}^2 = |truth(\mathbf{x}^*) - f(\mathbf{x}^*)|^2 + \sigma_{GP}^2. \quad (4.2)$$

σ_{model}^2 can now be used in the discrepancy term, such that $\delta(\mathbf{x}) \sim \mathcal{N}(0, \sigma_{model}^2)$. This process is done for both models to produce Fig. 4.3.

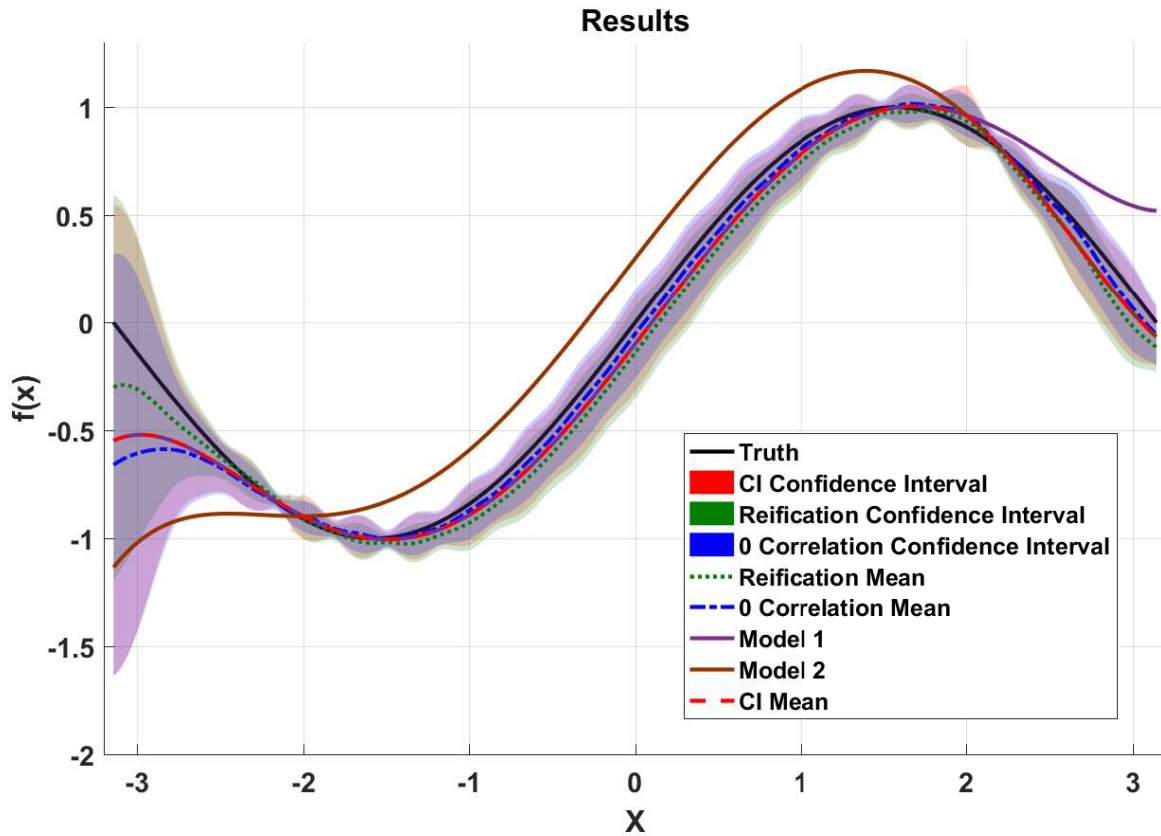


Figure 4.4: Mean and two sigma confidence interval for the fused model using Covariance Intersection Algorithm, Zero Correlation Assumption, and Reification

The last item needed in Algorithm 1 is $\mathcal{S}(x^*)$, or the neighborhood of points around x^* . This was chosen to be 10 points uniformly sampled from $-0.05 \leq x^* \leq 0.05$. This can be modified depending on the degree of non-linearity of the design space or the model's propensity for non-convergence for certain input configurations. Algorithm 1 can now be performed to estimate the correlation at all points in the design space. This correlation is used in Eqs. 2.4 and 2.5 to estimate the mean and variance of the new fused estimate. The design space was discretized such that this was repeated uniformly 300 times over the range to yield Fig. 4.4, which compares the result to Covariance Intersection algorithm and fusion assuming zero correlation.

There are several notable features illustrated in this plot. Due to this being a single dimension problem, Covariance Intersection algorithm always selects the model with the least variance at all points in the design space. While this represents a significant improvement over using either of the models individually, it is not effectively leveraging information from both models. The fused model produced by assuming zero correlation yields estimates that are always between the two models, as predicted by Fig. 2.2(a-c). From $-\pi \leq \mathbf{x}^* \lesssim -2$ the reification estimate is significantly better than Covariance Intersection and assuming zero correlation. This is similar to Fig. 2.2(f) where two correlated models allow for a prediction that is skewed beyond the higher fidelity model. This is one of the unique properties of the fusion under known correlation method that is leveraged by estimating the correlation with reification. However, this property is not always beneficial as shown from $-2 \lesssim \mathbf{x}^* \lesssim 2$ when the true value is actually between the two correlated models. We believe that the method’s ability to predict beyond both models far outweighs the slight disadvantage in these scenarios. It should be noted that the unsteady confidence interval near areas where both models converge is due to ill-conditioned matrices inherent to the Winkler fusion method. Mitigating this behavior is a topic of future work.

Comparison of Mean Squared Error (MSE)	
Fusion Method	MSE
Model 1	0.0227
Model 2	0.0865
No Correlation	0.0139
Covariance Intersection	0.0120
Reification	0.0095

Table 4.1: Mean squared error for Model 1, Model 2, fusion assuming zero correlation, the Covariance Intersection algorithm, and the reification approach.

Overall, the reification approach performs very well in this problem as shown in Ta-

ble 4.1, where a comparison of mean squared errors (MSE) between each method and truth is presented. Here, MSE is calculated by sampling a large number of uniformly spaced points in the input space, 1,000 in this case. The reification approach yields a 58% reduction in MSE when compared to using the highest fidelity model alone.

4.2 1D CFD Demonstration

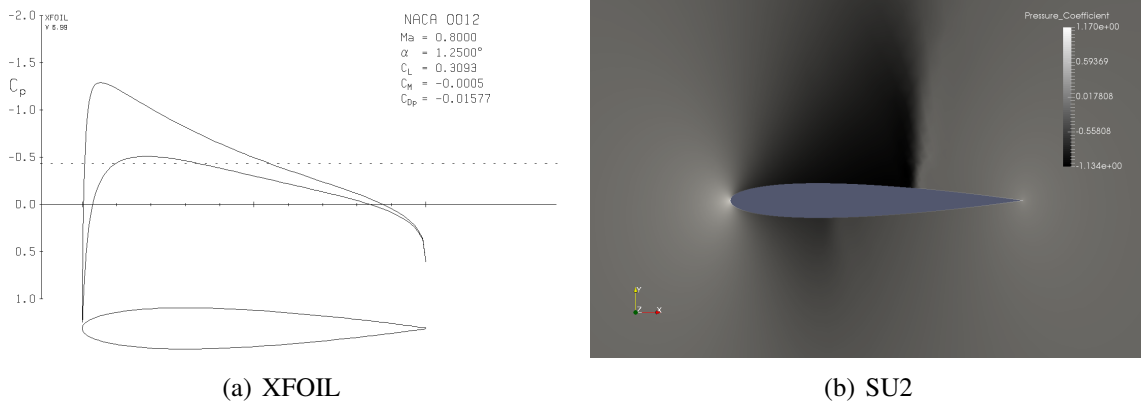


Figure 4.5: Example outputs of NACA 0012 airfoil from XFOIL and SU2.

This demonstration uses the computational fluid dynamics programs XFOIL [30] and SU2 [31] as the two simulators. The airfoil of interest is the NACA 0012, a common validation airfoil for CFD simulators. The “truth” model used to validate the method and estimate model discrepancy is constructed using real-world wind tunnel data of the NACA 0012 airfoil from the National Aeronautics and Space Administration (NASA) and the Advisory Group for Aerospace Research and Development (AGARD) [32, 33]. For this one dimensional case, the Mach number is fixed at 0.30 and the angle of attack will vary from -2.2 to 13.3 degrees. The quantity of interest is the coefficient of lift.

XFOIL and SU2 are both very powerful CFD simulators, but have different performance capabilities in various flow regimes. XFOIL is an airfoil solver for the subsonic

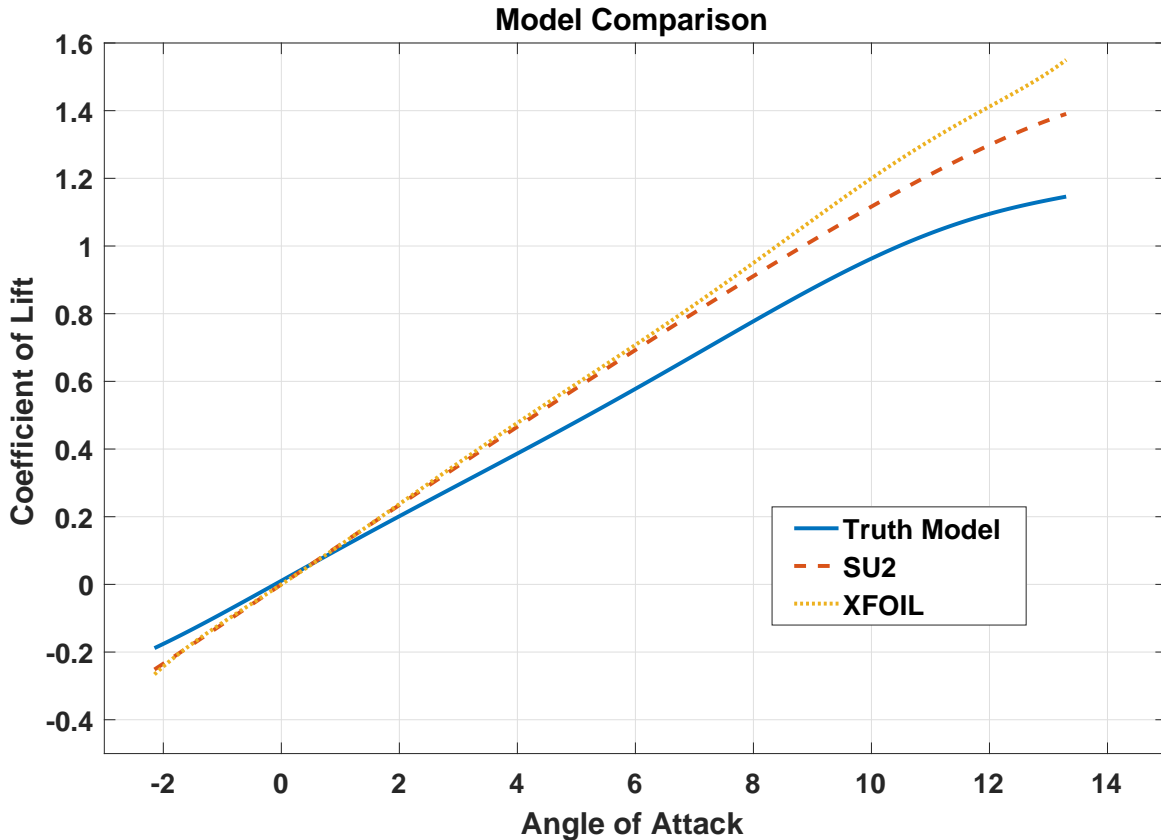


Figure 4.6: Coefficient of lift estimates from SU2, XFOIL, and wind tunnel data.

regime that combines a panel method with the Karman-Tsien compressibility correction for the potential flow with a two-equation boundary layer model. This causes XFOIL to overestimate lift and underestimate drag [34]. SU2, for the case of airfoil analysis, uses a finite volume scheme, the details of which may be found in Ref. [31]. SU2 was set to use Reynolds-averaged Navier-Stokes (RANS) method with the Spalart-Allmaras turbulence model. This allowed SU2 to be significantly more accurate than XFOIL in the more turbulent flow regimes at higher Mach and angle of attack. This accuracy comes with orders of magnitude increase in computational expense. XFOIL takes around 0.1 seconds, while SU2, in this configuration, takes about 10 minutes on average. Fig. 4.5 shows an example

output of the two simulators that illustrates the difference in fidelity levels. To validate and quantify the fidelity disparity, wind tunnel data from NASA and AGARD was used to construct a “truth model” by interpolating values between the given data points. Like the previous example, this also allows our methodology to be validated against the constituent models, the Covariance Intersection algorithm, and fusion assuming zero correlation. A comparison between SU2, XFOIL, and the truth model is shown in Fig. 4.6. As expected, SU2 performs better than XFOIL at higher angle of attack.

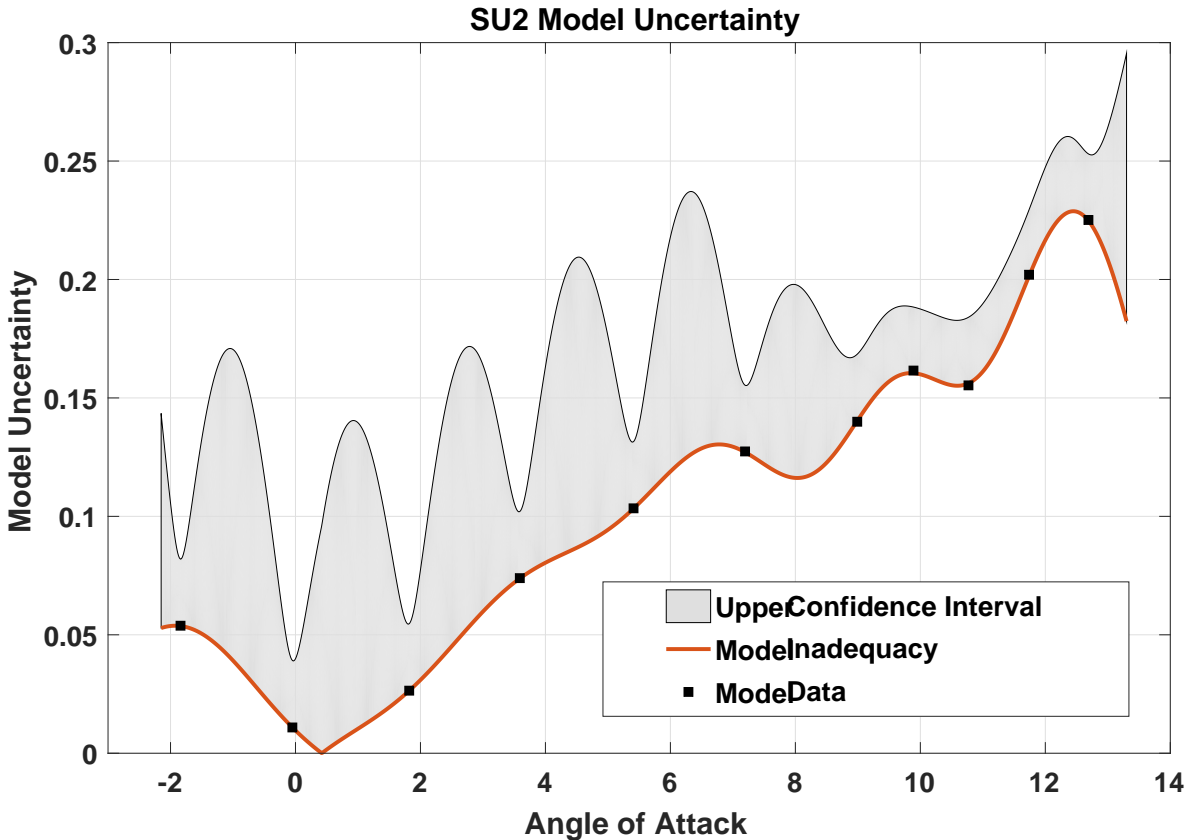


Figure 4.7: Gaussian process regression of uncertainty of SU2.

Following the same procedure as the previous demonstration, the model inadequacy is

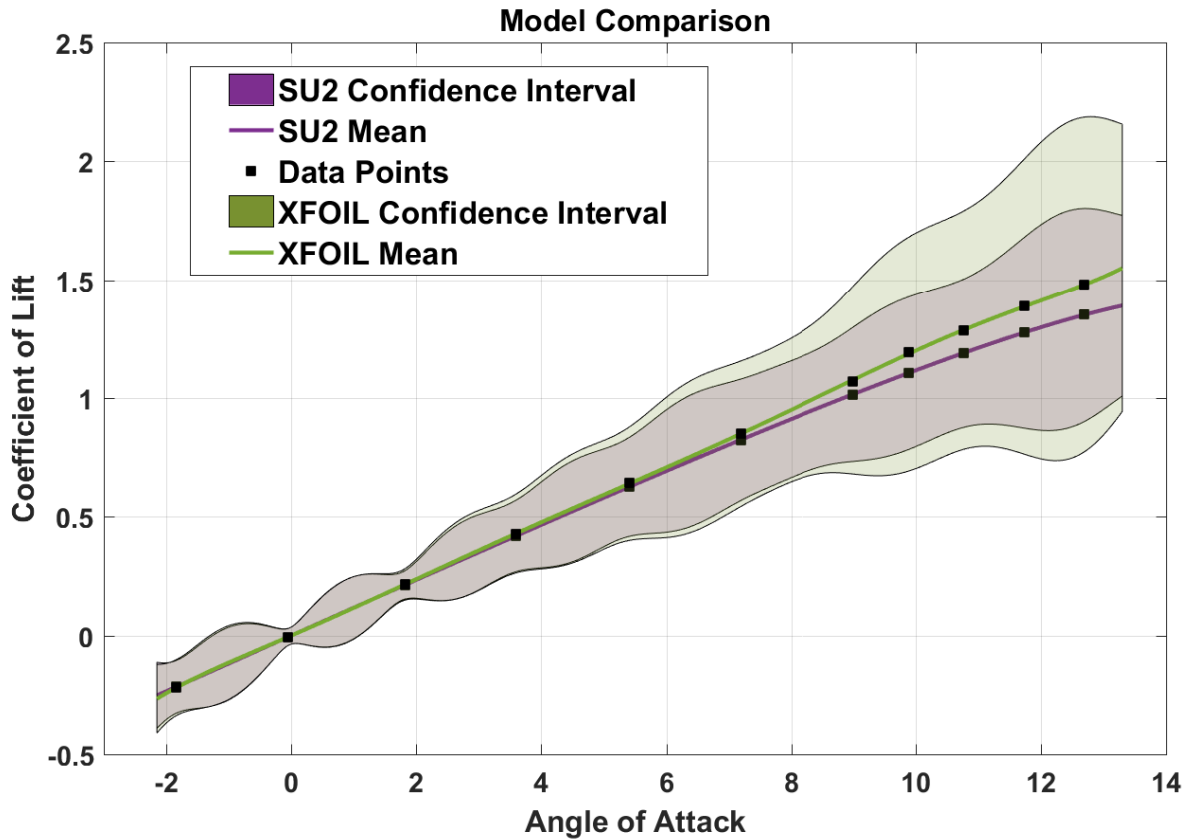


Figure 4.8: SU2 and XFOIL coefficient of lift predictions with two sigma confidence interval.

estimated by taking the difference between the wind tunnel data and our models. The wind tunnel data has 11 data points at Mach 0.30. XFOIL and SU2 are run at these points and the difference is used as training data for a Gaussian process regression. Code uncertainty and uncertainty due to observation error were added into the GP as shown in Fig. 4.7 using SU2 as an example. The same functions were used in this demonstration as the previous numerical demonstration. The only difference was the characteristic length scale was set to 1.25. The signal standard deviation and Gaussian noise standard deviation remained the same. Following Eq. 4.2, the uncertainties were incorporated to produce Fig. 4.8. Note how the confidence interval for SU2 becomes smaller than XFOIL at higher Mach, as

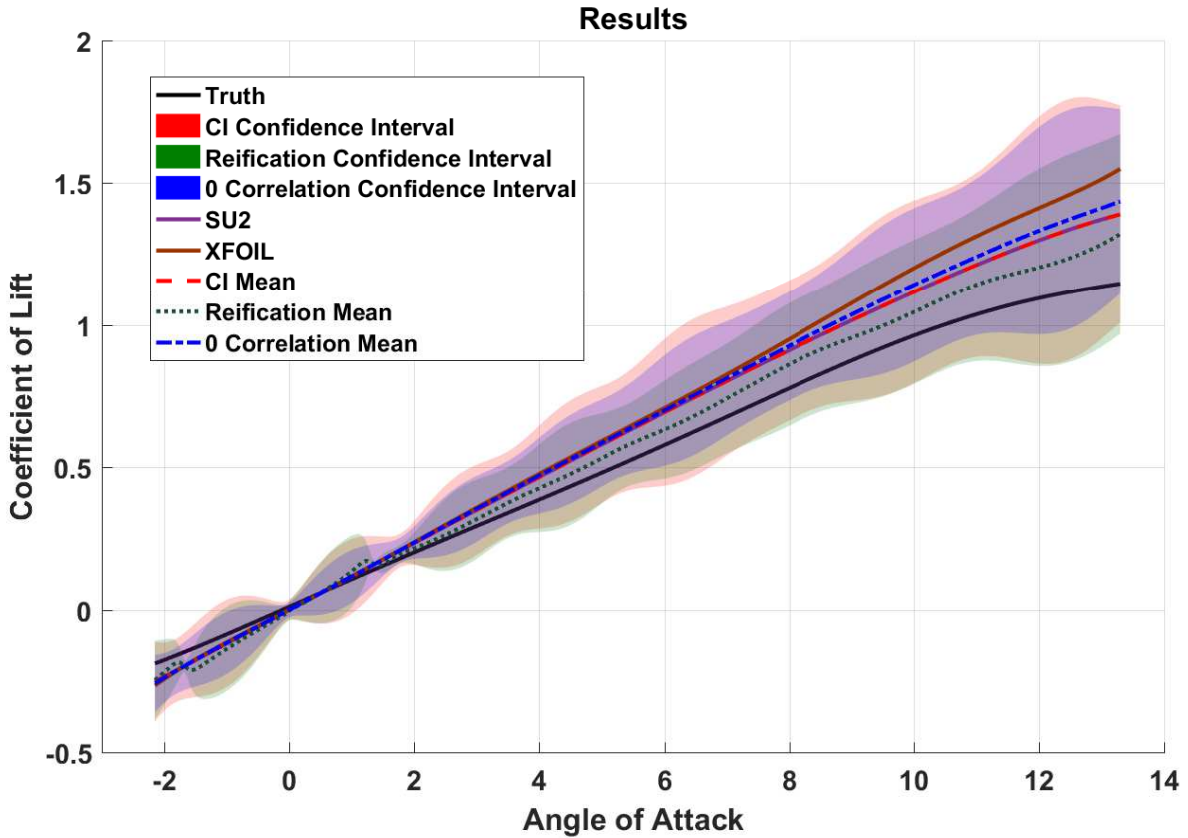


Figure 4.9: Results of no correlation case, Covariance Intersection algorithm, and the reification approach.

expected.

Similar to the previous example, 10 points were sampled uniformly from $-0.05 \leq x^* \leq 0.05$ for $\mathcal{S}(x^*)$, or the neighborhood of points around x^* . This can be modified depending on the degree of non-linearity of the design space or if the CFD solvers have trouble converging to a solution at x^* . Algorithm 1 can be performed to estimate the correlation at all points in the design space. This correlation is used in Eqs. 2.4 and 2.5 to estimate the mean and variance of the new fused estimate. Doing this over a design space that has been discretized by 1000 points yields Fig. 4.9, which compares the result to the Covariance Intersection algorithm and fusion assuming zero correlation.

There are two important things to note in this plot. First, the reification method performs better than Covariance Intersection and the zero correlation assumption in the vast majority of the design space. This is again due to the property illustrated in Fig. 2.2(f) where two correlated models allow for a prediction that is skewed beyond the higher fidelity model. Second, the areas where the two models cross exhibits interesting characteristics due to their mean and variance approaching nearly identical values which can be seen in Fig. 4.8. Mitigating the ill-conditioned matrices created by this scenario is a topic of future work.

Comparison of Mean Squared Error (MSE)	
Fusion Method	MSE
XFOIL	0.0313
SU2	0.0149
No Correlation	0.0192
Covariance Intersection	0.0149
Reification	0.0050

Table 4.2: Mean squared error for XFOIL and SU2 without fusion, the no correlation case, the Covariance Intersection algorithm, and the reification approach.

Table 4.2 shows the mean squared error for each method as well as the individual simulators. The reification method yields a 66% reduction in mean squared error compared to using only SU2. This is a huge increase in fidelity for an insignificant increase in computational expense. This result shows that the reification method has the potential to offer significant increases in accuracy for situations where multiple models are available.

4.3 2D CFD Demonstration

This demonstration again uses the computational fluid dynamics programs XFOIL and SU2 as the two simulators. The “truth” model used to validate the method and estimate

model discrepancy is the full set of real-world NACA 0012 wind tunnel data, which includes 68 data points throughout the design space. For this demonstration, the outputs of interest will be the lift coefficient, C_L , and the drag coefficient, C_D . The inputs for the analysis will be the Mach number, M , and the angle of attack, α . The Mach number varies from 0.15 to 0.75 and the angle of attack will vary from -2.2 to 13.3 degrees.

As discussed in the previous example, SU2 is considered to be the higher fidelity of the two models. We would expect to see greater accuracy relative to XFOIL as Mach number and angle of attack increases due to the introduction of more turbulent, separated flow regimes. To validate and quantify this expectation, wind tunnel data from NASA and AGARD was used to construct a "truth model" using spline interpolation to determine values between the given data points. Like the previous example, this also allows our methodology to be validated against the constituent models as well as Covariance Intersection algorithm and fusion with zero correlation.

The procedure for a two dimensional problem is very similar to the single dimension case computationally. However, it is much more challenging to illustrate the models concurrently, so there will be fewer plots shown in this demonstration. The plots included show the model mean as a colored surface, two sigma confidence interval as a black mesh, and data points as black squares. There is a high Mach, high angle of attack section that is omitted from plotting due to the simulators' inability to simulate flows in this regime.

Following the same procedure as the previous demonstration, the model inadequacy is estimated by taking the difference between the wind tunnel data and our models. The wind tunnel data has 68 data points at varying Mach and angle of attack. XFOIL and SU2 are run at these points and the difference is used as training data for a Gaussian process regression. Code uncertainty and uncertainty due to observation error were added into the GP as shown in Fig. 4.10 using SU2 coefficient of lift as an example. Due to this being a 2D case, new functions were needed for the GP to yield a smooth surface regression.

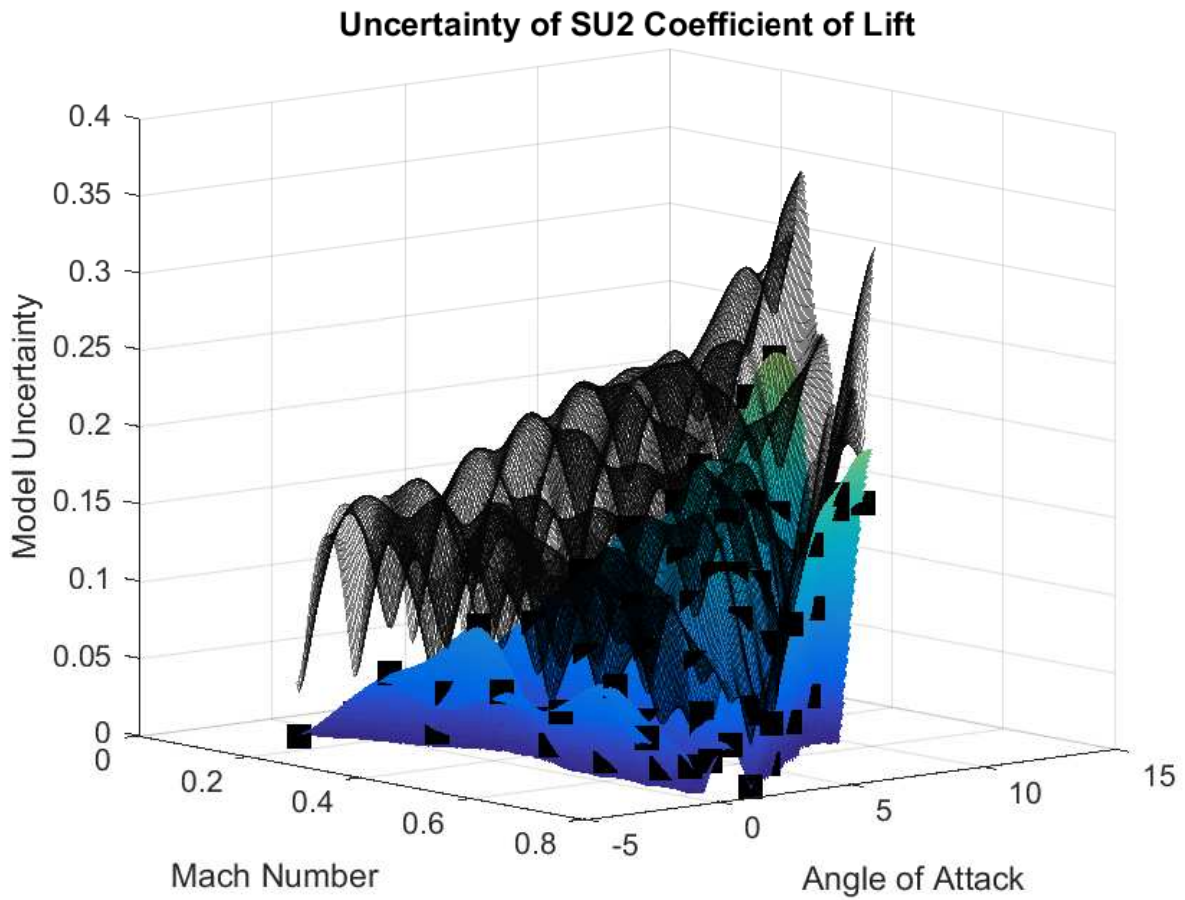


Figure 4.10: Gaussian process regression of SU2 coefficient of lift uncertainty with two sigma confidence interval mesh.

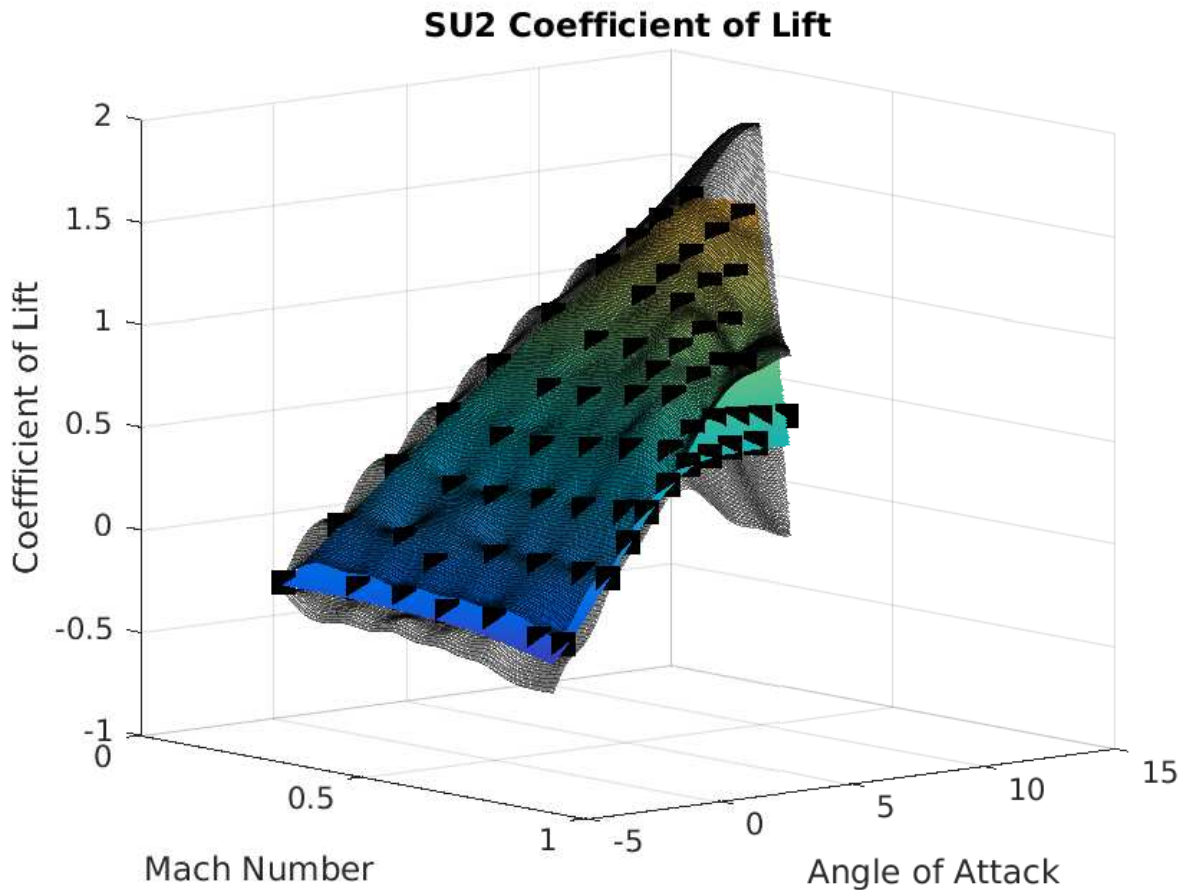


Figure 4.11: SU2 coefficient of lift predictions with two sigma confidence interval mesh.

The mean function was set to zero again. However, the covariance function was chosen to be the Matern function with automatic relevance determination. This allowed for a characteristic length scale to be specified for each dimension. This is important due to the order of magnitude difference between the values for Mach and α . The value for the Mach dimension was chosen to be 0.15, and the value for the α was 2. The other functions were unchanged. However, the hyperparameters were chosen to be 0.15 and 0.1 for the signal and Gaussian noise standard deviations for C_L , and 0.015 and 0.01 for C_D . Following Eq. 4.2, the uncertainties were incorporated to produce Fig. 4.11, which is a plot of SU2

coefficient of lift including the two sigma confidence interval. Unfortunately it is quite difficult to visualize a 3D surface plot on paper, so notable trends are obscured. This same procedure was done for XFOIL as well as C_D , although the plots are not included.

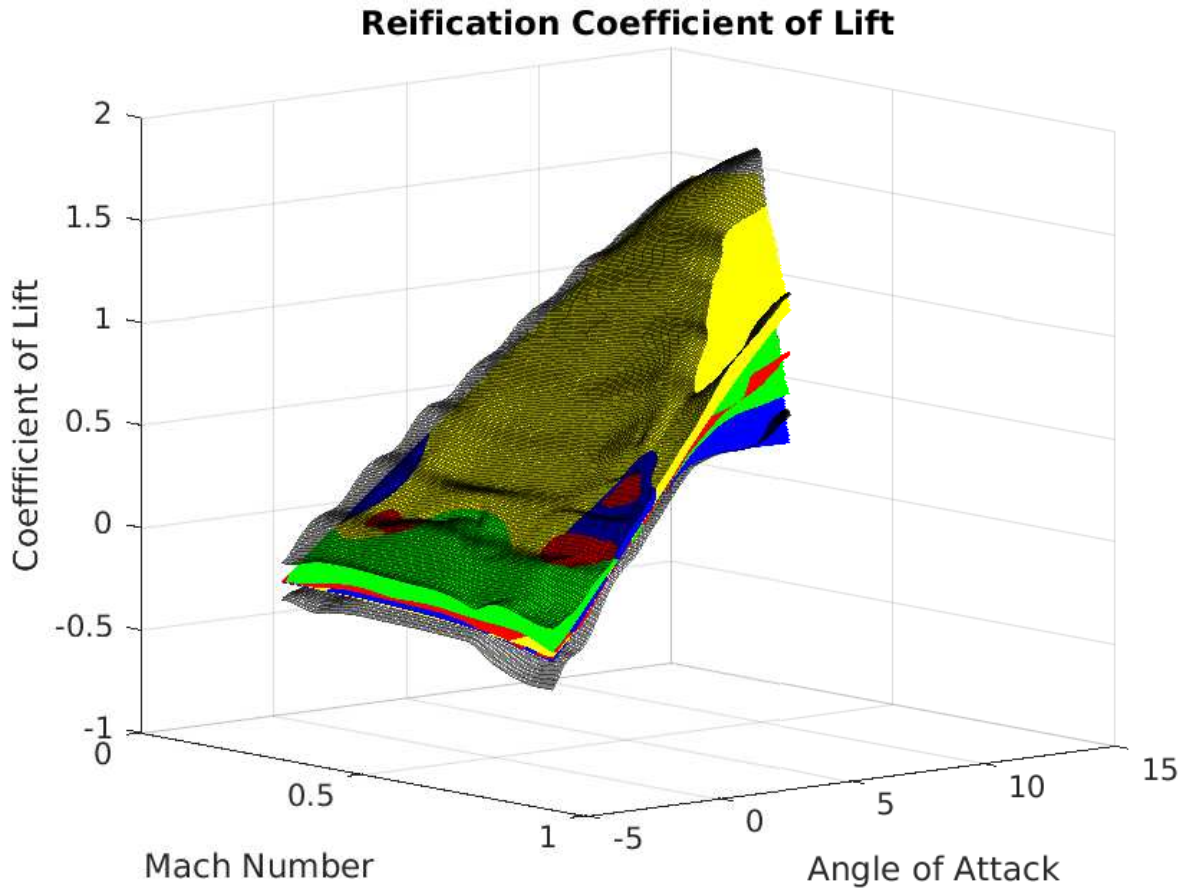


Figure 4.12: Results of the reification method for C_L . XFOIL is yellow, SU2 is blue, truth is green, the reification method mean is red, and its two sigma confidence interval is the black mesh.

The neighborhood of points around \mathbf{x}^* , $\mathcal{S}(\mathbf{x}^*)$, is different than that of the previous two examples due to the fact that \mathbf{x}^* is two dimensional. Also, the dimensions have very

different scales. It was chosen to select 10 points uniformly in both dimensions and create a square mesh of points around \mathbf{x}^* . The angle of attack dimension was sampled from $-0.05 \leq \mathbf{x}^* \leq 0.05$ to be consistent with the previous example, while the Mach number dimension was sampled from $-0.001935 \leq \mathbf{x}^* \leq 0.001935$. This number is was calculated by scaling 0.05 to the proportional value in the Mach number dimension.

The design space was discretized into a 200 by 200 grid of points. Algorithm 1 can be used to estimate the correlation at each of these 40,000 points in the discretized design space. This correlation is substituted into Eqs. 2.4 and 2.5 to estimate the mean and variance of the new fused estimate. Doing this over the entire design space yields Fig. 4.12, which shows the XFOIL (yellow), SU2 (blue), truth model (green), and the reification mean (red) and two sigma confidence interval (black mesh) for estimating the coefficient of lift. Unlike the previous demonstrations, notable trends are difficult to discern. For this reason plots of each method were not included. Performance of the method must be evaluated using the mean squared error.

Coefficient of Lift		Coefficient of Drag	
Fusion Method	MSE (10^{-2})	Fusion Method	MSE (10^{-5})
XFOIL	2.5385	XFOIL	16.057
SU2	0.7431	SU2	2.5407
No Correlation	0.8787	No Correlation	0.4241
Covariance Intersection	1.2240	Covariance Intersection	2.2288
Reification	0.2353	Reification	0.5608

Table 4.3: Mean squared error for XFOIL and SU2 without fusion, the no correlation case, the Covariance Intersection algorithm, and the reification approach.

Table 4.3 shows the mean squared error for each method. The mean squared error was calculated using a similar method as the previous examples. A square mesh of 40,000 points was created by uniformly sampling 200 points in each dimension. The difference

between the truth model and the fused estimate at each point is squared, and the mean of this value is the mean squared error.

The reification approach yields a significant reduction in mean squared error compared to the high fidelity SU2 model in both C_L and C_D , 68% and 78% respectively. Once again this represents a significant increase in accuracy for almost no increase in computational expense. It is especially interesting that we achieved this reduction in MSE for C_D by fusing a simulator with an order of magnitude poorer accuracy. Note the Covariance Intersection mean squared error is actually greater than XFOIL and SU2. This is due to the fact that this is a two dimensional case, so the algorithm incorporates data from both C_L and C_D . The reification approach is slightly worse than the zero correlation case for the drag prediction. This is due to the fact that the true value is always between XFOIL and SU2, so the ability for reification to predict when the two models are both skewed high or low is not utilized. This allows the zero correlation case to take a slight advantage in small areas of the design space. Overall, these results show that the reification method has the potential to offer significant increases in accuracy for situations where multiple models are available.

5. SUMMARY AND CONCLUSIONS*

We have presented a method for fusing information from multiple models of varying fidelity. Our model reification approach improves upon existing fusion methods by estimating correlation in cases where it is unknown and experimental data is inadequate. In three demonstration cases, the model reification approach yielded a significant improvement in predictive capability compared to using the models individually as well as existing fusion methodologies.

Our approach relied on several assumptions. Among those were that model inadequacy was quantified for any given model, that generating synthetic data by reifying models is a reasonable approach for enabling the estimation of error terms for simulation models, that a variance weighted average of correlation coefficient estimates is an appropriate method for fusing correlation information, that an appropriate neighborhood of the input spaces of the simulators can be identified for correlation estimation, and that the input spaces for different simulators are identical. We feel these assumptions were necessary to constrain the broad, complex subject of information fusion. The results support this conclusion and show that our reification approach has the potential to significantly improve information fusion from multifidelity information sources.

There are a few topics of future work we are actively researching. First, we are making modifications to mitigate ill-conditioned matrices as the models' outputs approach identical mean and variance. Second, we are working on a method to determine the optimal size of the neighborhood around a particular input configuration. Finally, we are working on methodology to sequentially construct the fused model by optimally sampling from

*Reprinted from "A Model Reification Approach to Fusing Information from Multifidelity Information Sources", 19th AIAA Non-Deterministic Approaches Conference, AIAA SciTech Forum, (AIAA 2017-1949), Copyright © 2017 by William D. Thomison and Douglas L. Allaire.

the available models. We seek to integrate this work into an importance weighted Gibbs sampling procedure from Ref. [35].

REFERENCES

- [1] G. E. P. Box, “Robustness in the strategy of scientific model building,” technical summary report, Wisconsin University - Madison Mathematics Research Center, 1979.
- [2] M. Kennedy and A. O’Hagan, “Bayesian calibration of computer models,” *J.R. Statist. Soc. B*, vol. 63, no. 3, pp. 425–464, 2001.
- [3] H. Boström, S. Andler, M. Brohede, R. Johansson, A. Karlsson, J. van Laere, L. Niklasson, M. Nilsson, A. Persson, and T. Ziemke, “On the definition of information fusion as a field of research,” 2007.
- [4] A. Mosleh and G. Apostolakis, “The assessment of probability distributions from expert opinions with an application to seismic fragility curves,” *Risk Analysis*, vol. 6, no. 4, pp. 447–461, 1986.
- [5] E. Zio and G. Apostolakis, “Two methods for the structured assessment of model uncertainty by experts in performance assessments of radioactive waste repositories,” *Reliability Engineering and System Safety*, vol. 54, no. 2-3, pp. 225–241, 1996.
- [6] J. Reinert and G. Apostolakis, “Including model uncertainty in risk-informed decision making,” *Annals of Nuclear Energy*, vol. 33, no. 4, pp. 354–369, 2006.
- [7] M. Riley and R. Grandhi, “Quantification of modeling uncertainty in aeroelastic analyses,” *Journal of Aircraft*, vol. 48, no. 3, pp. 866–873, 2011.
- [8] J. McFarland and B. Bichon, “Bayesian model averaging for reliability analysis,” *50th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, vol. 430, 2009.
- [9] E. Leamer, *Specification Searches: Ad Hoc Inference with Nonexperimental Data*. New York, NY: John Wiley & Sons, 1978.

- [10] D. Madigan and A. Raftery, “Model selection and accounting for model uncertainty in graphical models using occam’s window,” *American Statistical Association*, vol. 89, no. 428, pp. 1535–1546, 1994.
- [11] D. Draper, “Assessment and propagation of model uncertainty,” *Journal of the Royal Statistical Society Series B*, vol. 57, no. 1, pp. 45–97, 1995.
- [12] J. Hoeting, D. Madigan, A. Raftery, and C. Volinsky, “Bayesian model averaging: A tutorial,” *Statistical Science*, vol. 14, no. 4, pp. 382–417, 1999.
- [13] M. Clyde, “Model averaging.” In *Subjective and Objective Bayesian Statistics 2nd ed.* Chapter 13, Wiley-Interscience, 2003.
- [14] M. Clyde and E. George, “Model uncertainty,” *Statistical Science*, vol. 19, pp. 81–94, 2004.
- [15] S. Geisser, “A bayes approach for combining correlated estimates,” *Journal of the American Statistical Association*, vol. 60, pp. 602–607, 1965.
- [16] P. Morris, “Combining expert judgments: A bayesian approach,” *Management Science*, vol. 23, pp. 679–693, 1977.
- [17] R. Winkler, “Combining probability distributions from dependent information sources,” *Management Science*, vol. 27, no. 4, pp. 479–488, 1981.
- [18] S. Julier and J. Uhlmann, “A non-divergent estimation algorithm in the presence of unknown correlations.” In *proceedings of the American Control Conference*, pp. 2369-2373, 1997.
- [19] S. Julier and J. Uhlmann, “General decentralized data fusion with covariance intersection.” In D. Hall and J. Llinas, editor, *Handbook of Data Fusion*. CRC Press, Boca Raton FL, USA, 2001.

- [20] M. Goldstein and J. Rougier, “Reified bayesian modelling and inference for physical systems,” *Journal of Statistical Planning and Inference*, vol. 139, no. 3, pp. 1221–1239, 2006.
- [21] R. Rebba, S. Mahadevan, and S. Huang, “Validation and error estimation of computational models,” *Reliability Engineering and System Safety*, vol. 91, pp. 1390–1397, 2006.
- [22] P. Arendt, D. Apley, and W. Chen, “Quantification of model uncertainty: Calibration, model discrepancy, and identifiability,” *ASME Journal of Mechanical Design*, vol. 134, 2012.
- [23] S. Grime and H. F. Durrant-Whyte, “Data fusion in decentralized sensor networks,” *Control Engineering Practice*, vol. 2, no. 5, pp. 849–863, 1994.
- [24] S. W. Utete and H. F. Durrant-Whyte, “Network management in decentralized data fusion networks,” in *Photonics for Industrial Applications*, pp. 64–73, International Society for Optics and Photonics, 1994.
- [25] L. Chen, P. Arambel, and R. Mehra, “Fusion under unknown correlation: Covariance intersection as a special case.” In proceedings of the International Society of Information Fusion, pp. 905-912, 2002.
- [26] J. Gao and C. Harris, “Some remarks on kalman filters for the multisensor fusion,” *Information Fusion*, vol. 3, pp. 191–201, 2002.
- [27] C. Rasmussen and C. Williams, *Gaussian Processes for Machine Learning*. Boston, Massachusetts: The MIT Press, 2005.
- [28] A. O’Hagan and J. Kingman, “Curve fitting and optimal design for prediction,” *Journal of the Royal Statistical Society. Series B*, vol. 40, pp. 1–42, 1978.

- [29] M. Handcock and M. Stein, “A bayesian analysis of kriging,” *Technometrics*, vol. 35, pp. 403–410, 1993.
- [30] M. Drela, “Xfoil: An analysis and design system for low reynolds number airfoils,” in *Low Reynolds Number Aerodynamics*, pp. 1–12, Springer, 1989.
- [31] F. Palacios, M. R. Colonno, A. C. Aranake, A. Campos, S. R. Copeland, T. D. Economon, A. K. Lonkar, T. W. Lukaczyk, T. W. Taylor, and J. J. Alonso, “Stanford university unstructured (su2): An open-source integrated computational environment for multi-physics simulation and design,” *AIAA Paper*, vol. 287, p. 2013, 2013.
- [32] J. Thibert, M. Grandjacques, and L. Ohman, “Naca 0012 airfoil.”
- [33] C. Rumsey, “2d naca 0012 airfoil validation case,” *NASA Turbulence Modeling Resource*, 2013.
- [34] R. Barrett and A. Ning, “Comparison of airfoil precomputational analysis methods for optimization of wind turbine blades,” *IEEE Transactions on Sustainable Energy*, vol. 7, no. 3, p.1081-1088, 2016.
- [35] S. F. Ghoreishi and D. L. Allaire, “Compositional uncertainty analysis via importance weighted gibbs sampling for coupled multidisciplinary systems,” in *18th AIAA Non-Deterministic Approaches Conference*, p. 1443, 2016.