

A Model Reification Approach to Fusing Information from Multifidelity Information Sources

William D. Thomison* and Douglas Allaire†

*Department of Mechanical Engineering
Texas A&M University, College Station, TX 77840*

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 eliminates the observational data requirement. 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.

I. Introduction

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 uncertainty². 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. 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,

*Graduate Student, Department of Mechanical Engineering, AIAA Student Member

†Assistant Professor, Department of Mechanical Engineering, AIAA Member, dallaire@tamu.edu

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. A method is needed to fuse the information from both models to create a single, fused distribution for the quantities of interest.

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. Techniques available for this application include the adjustment factors approach⁴⁻⁷, Bayesian model averaging⁸⁻¹⁴, fusion under known correlation¹⁵⁻¹⁷, 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 2009²⁰. 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 synthetic data that is treated 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 II. Section III outlines our methodology. Demonstrations using three test problems are included in Section IV. Section V presents our conclusions drawn from this method.

II. 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 model inadequacy, an additive term $\delta(\mathbf{x})$ is included. This discrepancy term can vary dependent upon the inputs. The model can be represented by the equation

$$y = f(\mathbf{x}) = \bar{f}(\mathbf{x}) + \delta(\mathbf{x}), \quad (1)$$

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.

If the model simulates a complex process, it is difficult to reliably collect accurate data to be compared to 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 depends on the fidelity of the model. High fidelity models 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 two distributions. 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. Under this assumption it is possible to use information fusion methods under known or unknown correlation.

Accounting for correlation is critical 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 sources²³. 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 problem²⁴. 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' cameras.
2. Camera 1 sees an anomaly at location x . The machine vision system reports that one of the possibilities is 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. 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.

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. 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.

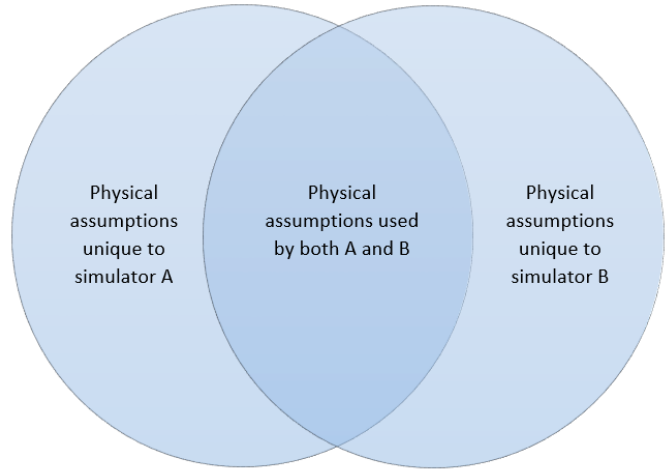


Figure 1. Relationship between two physics-based numerical simulations.

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. (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, $\bar{f}_1(\mathbf{x}^*) = f_1(\mathbf{x}^*) - \delta_1(\mathbf{x}^*)$ and $\bar{f}_2(\mathbf{x}^*) = f_2(\mathbf{x}^*) - \delta_2(\mathbf{x}^*)$. The fused models can be represented by the equation

$$y = k_1 f_1(\mathbf{x}^*) + k_2 f_2(\mathbf{x}^*), \quad (2)$$

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. (3) for k_1 and k_2 under the assumption $f_1(\mathbf{x}^*) \sim \mathcal{N}(\bar{f}_1(\mathbf{x}^*), \sigma_1^2)$ and $f_2(\mathbf{x}^*) \sim \mathcal{N}(\bar{f}_2(\mathbf{x}^*), \sigma_2^2)$.

$$\min_{\mathbf{k}} \mathbf{k}^T \Sigma \mathbf{k} \quad \text{subject to} \quad k_1 + k_2 = 1, \quad (3)$$

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

$$\mathbb{E}[y] = \frac{(\sigma_2^2 - \rho\sigma_1\sigma_2)\bar{f}_1(\mathbf{x}^*) + (\sigma_1^2 - \rho\sigma_1\sigma_2)\bar{f}_2(\mathbf{x}^*)}{\sigma_1^2 + \sigma_2^2 - 2\rho\sigma_1\sigma_2}, \quad (4)$$

and a variance of y given by

$$\text{Var}(y) = \frac{(1 - \rho^2)\sigma_1^2\sigma_2^2}{\sigma_1^2 + \sigma_2^2 - 2\rho\sigma_1\sigma_2}. \quad (5)$$

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

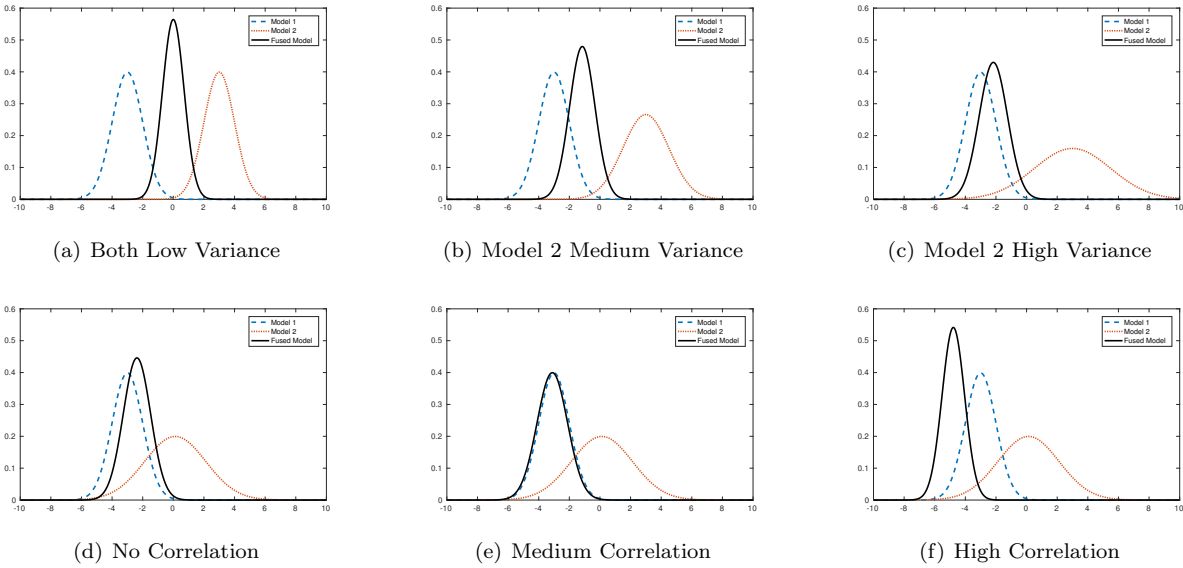


Figure 2. Trends in the fused estimate due to changes in variance and correlation coefficient. The fused model is the black solid line. Models 1 and 2 are blue and red, respectively.

An analysis of Eq. (4) and Eq. (5) reveals some interesting behaviors of this method as illustrated in Fig. 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 decreases. 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 too high or too low. 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 value 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 proves the generalized versions of Eq. (4) and Eq. (5), given by Eq. (6) and Eq. (7). The fused mean in the generalized approach is shown to be

$$\mathbb{E}[y] = \frac{\mathbf{e}^T \Sigma^{-1} \bar{\mathbf{f}}(\mathbf{x}^*)}{\mathbf{e}^T \Sigma^{-1} \mathbf{e}}, \quad (6)$$

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

$$\text{Var}(y) = \frac{1}{\mathbf{e}^T \Sigma^{-1} \mathbf{e}}. \quad (7)$$

II.B. 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 estimates²⁶. 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 space¹⁸. 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 ellipse²⁵. The ellipse is created using the inequality

$$\mathcal{B}_Q(c) \triangleq \{\mathbf{x} : \mathbf{x}^T Q^{-1} \mathbf{x} < c\}, \quad (8)$$

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. 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. 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

$$\tilde{\mathbf{P}}_{yy}^{-1} = \omega \tilde{\mathbf{P}}_{11}^{-1} + (1 - \omega) \tilde{\mathbf{P}}_{22}^{-1} \quad (9)$$

$$\tilde{\mathbf{P}}_{yy}^{-1} \mathbb{E}[y] = \omega \tilde{\mathbf{P}}_{11}^{-1} \bar{\mathbf{f}}_1(\mathbf{x}^*) + (1 - \omega) \tilde{\mathbf{P}}_{22}^{-1} \bar{\mathbf{f}}_2(\mathbf{x}^*), \quad (10)$$

where $\omega \in [0, 1]$ controls the weights applied to each model. This update ensures that the covariance ellipse of $\tilde{\mathbf{P}}_{yy}$ always encloses the intersection of $\tilde{\mathbf{P}}_{11}$ and $\tilde{\mathbf{P}}_{22}$, as shown in Fig. 4 for varying values of ω . ω is chosen such that the trace or determinant of $\tilde{\mathbf{P}}_{yy}$ is minimized. We chose to minimize the trace for our comparison to this approach.

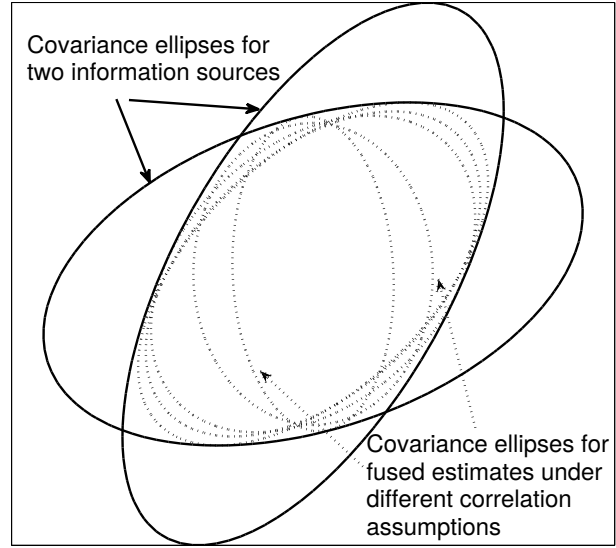


Figure 3. Geometric interpretation of the Covariance Intersection algorithm.

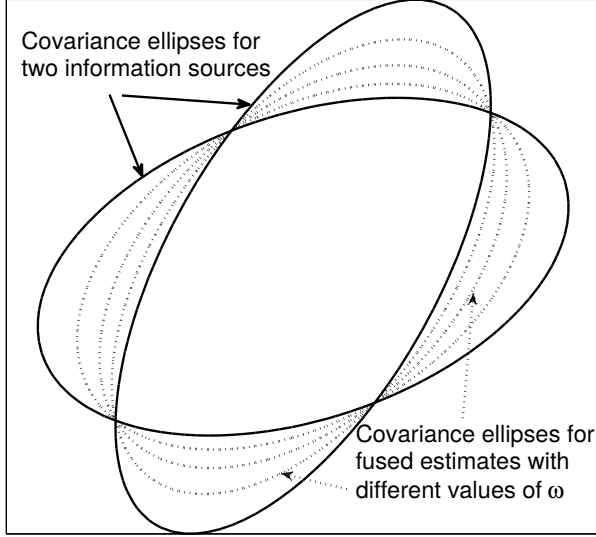


Figure 4. The covariance ellipse for any choice of ω encloses the intersection of the covariance ellipses of the information sources.

models as a “truth model.” This means that we assume the data generated by the model represent the true quantities. 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, $\delta_1(\mathbf{x}^*) \sim \mathcal{N}(0, \sigma_1^2)$ 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. (4) and (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

$$\tilde{f}_1(\mathbf{x}^*) = f_1(\mathbf{x}^*) - \bar{f}_1(\mathbf{x}^*) = \delta_1(\mathbf{x}^*) \quad (11)$$

$$\tilde{f}_2(\mathbf{x}^*) = f_2(\mathbf{x}^*) - \bar{f}_2(\mathbf{x}^*) \quad (12)$$

$$= \bar{f}_1(\mathbf{x}^*) - \bar{f}_2(\mathbf{x}^*) + \delta_1(\mathbf{x}^*), \quad (13)$$

where Eq. 13 follows from the fact that if we reify simulator 1, the model inadequacy of simulator 2 is given as $\delta(\mathbf{x}^*) = \bar{f}_1(\mathbf{x}^*) - \bar{f}_2(\mathbf{x}^*) + \delta_1(\mathbf{x}^*)$. The mean squared errors are then given as

$$\mathbb{E}[\tilde{f}_1(\mathbf{x}^*)^2] = \mathbb{E}[\delta_1(\mathbf{x}^*)] = \sigma_1^2 \quad (14)$$

$$\mathbb{E}[\tilde{f}_2(\mathbf{x}^*)^2] = \mathbb{E}[(\bar{f}_1(\mathbf{x}^*) - \bar{f}_2(\mathbf{x}^*))^2] + \mathbb{E}[\delta_1(\mathbf{x}^*)] \quad (15)$$

$$= (\bar{f}_1(\mathbf{x}^*) - \bar{f}_2(\mathbf{x}^*))^2 + \sigma_1^2, \quad (16)$$

and the covariance is given as

$$\mathbb{E}[\tilde{f}_1(\mathbf{x}^*)\tilde{f}_2(\mathbf{x}^*)] = \sigma_1^2. \quad (17)$$

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

$$\rho_1(\mathbf{x}^*) = \frac{\sigma_1^2}{\sigma_1\sigma_2} = \frac{\sigma_1}{\sqrt{((\bar{f}_1(\mathbf{x}^*) - \bar{f}_2(\mathbf{x}^*))^2 + \sigma_1^2)}}, \quad (18)$$

The subscript under the correlation coefficient denotes which model was reified. Under these assumptions, the other information available to access model fidelity is the model inadequacy. To account for the uncertainty

Due to the nature of the Covariance Intersection algorithm, it has a tendency to discard lower fidelity information sources to achieve the most conservative estimate possible in cases with scalar quantities 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.

III. Methodology

This section details our methodology for the complete process of fusing information from multiple models. We assume no observational data is available. 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

in which one is the highest fidelity, model 2 is also reified to estimate $\rho_2(\mathbf{x}^*)$. The average correlation between the two models can then be estimated using the variances as follows

$$\bar{\rho}(\mathbf{x}^*) = \frac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2} \rho_1(\mathbf{x}^*) + \frac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2} \rho_2(\mathbf{x}^*). \quad (19)$$

This average correlation can then be used in Eqs. (4) and (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 configuration, \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. (18), the new estimate for correlation after the reification of model 1 is

$$\begin{aligned} \rho_1(\mathcal{S}(\mathbf{x}^*)) &= \mathbb{E}_{\mathcal{S}(\mathbf{x}^*)}[\rho_1(\mathbf{x})] \\ &= \mathbb{E}_{\mathcal{S}(\mathbf{x}^*)} \left[\frac{\sigma_1}{\sqrt{((f_1(\mathbf{x}) - f_2(\mathbf{x}))^2 + \sigma_1^2)}} \right], \end{aligned} \quad (20)$$

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.

Algorithm 1: Correlation Estimation with Simulator 1 Reified

- 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 $[f_1^1(\mathbf{x}^1), \dots, f_1^m(\mathbf{x}^m)]$
 - 3: Calculate $\rho_1^i(\mathbf{x}^i) = \frac{\sigma_1}{\sqrt{((f_1(\mathbf{x}^i) - f_2(\mathbf{x}^i))^2 + \sigma_1^2)}$ 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 $\bar{\rho}(\mathcal{S}(\mathbf{x}^*)) = \frac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2} \rho_1(\mathcal{S}(\mathbf{x}^*)) + \frac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2} \rho_2(\mathcal{S}(\mathbf{x}^*))$
-

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 by simply adding the dependence on \mathbf{x} in Eqs. (16) and (20). 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.

IV. Demonstration

In this section we present the preliminary 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 is discussed in Section IV.B.

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 accuracy. 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 assess model fidelity. This is not possible in the case of an analytical test problem. Also, 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 estimate model inadequacy. However, it would be unrealistic to have complete knowledge of the model inadequacy at all points in the design space. To keep the demonstration as realistic as possible the discrepancy between the models and the true values was examined at only a few points in the design space. A Gaussian process was then employed to perform a regression over the design space using those values as training points.

Gaussian processes are 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-dimensional normal distribution where each set of inputs has a corresponding normal distribution that can be obtained using mean and covariance functions²⁷. Gaussian processes were first proposed as a regression technique by O’Hagan in 1978²⁸. 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 method²⁹. The benefit of using a Gaussian process is that it can model statistics 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.

We account for three types of uncertainty when performing this Gaussian process regression: model inadequacy, observation error, and code uncertainty. The model inadequacy is estimated by the difference between the true values and the model at the points chosen to interrogate. Observation error is added to these points to account for imperfection in measuring real-world processes. Code uncertainty is included between the selected data points. Gaussian process regression allows all three of these uncertainties to be combined and quantified for all points in the design space.

The discrepancy term can be biased or unbiased. 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 increase the variance of the model. While both may be used, 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 this example. 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.

IV.A. Analytical Example

We consider two simulators with a one-dimensional input. A “truth model” is given 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, the Covariance Intersection algorithm, and the assumption of zero correlation in the fusion process.

We assume the truth model is defined as

$$truth(x) = \sin(\mathbf{x}) + noise, \tag{21}$$

where $noise \sim \mathcal{N}(0, 0.1^2)$. Gaussian noise is added to simulate uncertainty due to observation error that would exist in a real-world problem. The two simulators are defined as

$$f_1(x) = (x - 0.1) - \frac{(x - 0.1)^3}{3!} + \frac{(x - 0.1)^5}{5!}$$

$$f_2(x) = (x + 0.3) - \frac{(x + 0.1)^3}{3!} + \frac{(x - 0.1)^5}{5!}.$$

These are simply modified versions of the Taylor series approximation of $\sin(x)$. The domain of x is limited to $-\pi \leq x \leq \pi$. An illustration of the truth model and two simulators is shown in Fig. 5. In this case, model 1 could be considered the high fidelity model. However, model 2 becomes much more accurate than model 1 when $x > 2$. 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 model discrepancy. Given the fact that no real-world truth models exist, perfectly quantifying the discrepancy in the model at every point would be unrealistic. Instead, we chose to uniformly select 20 points in the input space to calculate the error by taking the difference between the truth model and the simulator. This mimics a situation where a limited amount of test data is used to validate a simulator. This test data often has uncertainty due to observation error, so that was incorporated into the truth model. Code uncertainty is introduced in the areas between the 20 points where the models were not run.

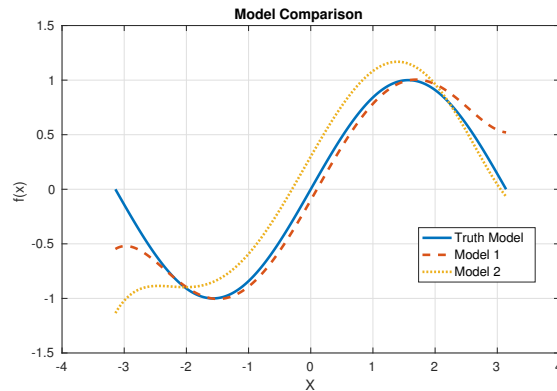


Figure 5. Output from each simulator along with the truth model.

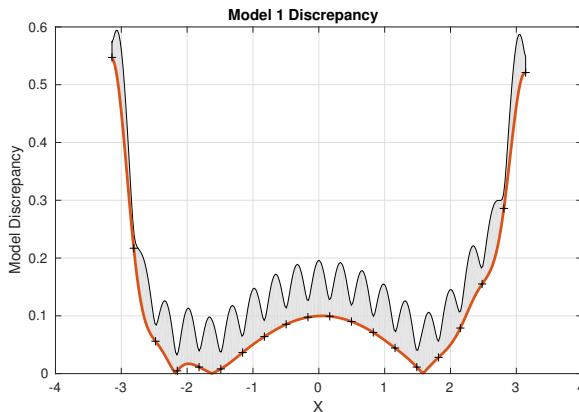


Figure 6. Plot of model 1 discrepancy with two sigma confidence intervals.

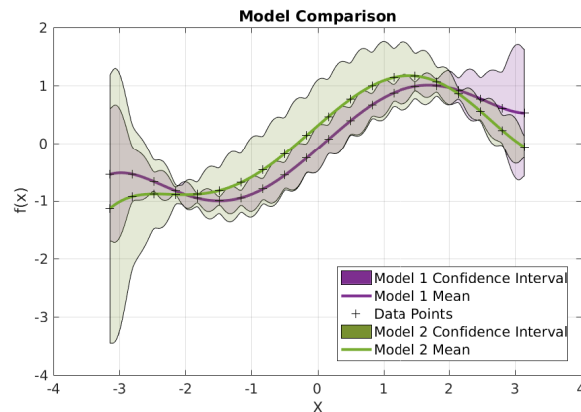


Figure 7. Mean and two sigma confidence interval of model 1 and model 2.

The construction of the unbiased estimation of model inadequacy begins with calculating the error at each of the 20 points selected previously, such that $error = |truth(x^*) - f(x^*)|$. Taking the absolute value creates the unbiased estimation. A Gaussian process regression is then performed using these error values as training points to estimate the model discrepancy at all points in the design space. Gaussian noise is added to compensate for the observation error uncertainty in the truth model, and code uncertainty is added between the observation points to produce the plot in Fig. 6. The gray shaded area represents the upper half of a two sigma confidence interval. It should be clarified that this is a plot of the uncertainty of model 1 and has its own standard deviation, σ_{GP} , due to the various types of uncertainty. The variance of model

1 is then defined as

$$\sigma_{model}^2 = |truth(\mathbf{x}^*) - f(\mathbf{x}^*)|^2 + \sigma_{GP}^2. \quad (22)$$

This process is done for both models to produce Fig. 7.

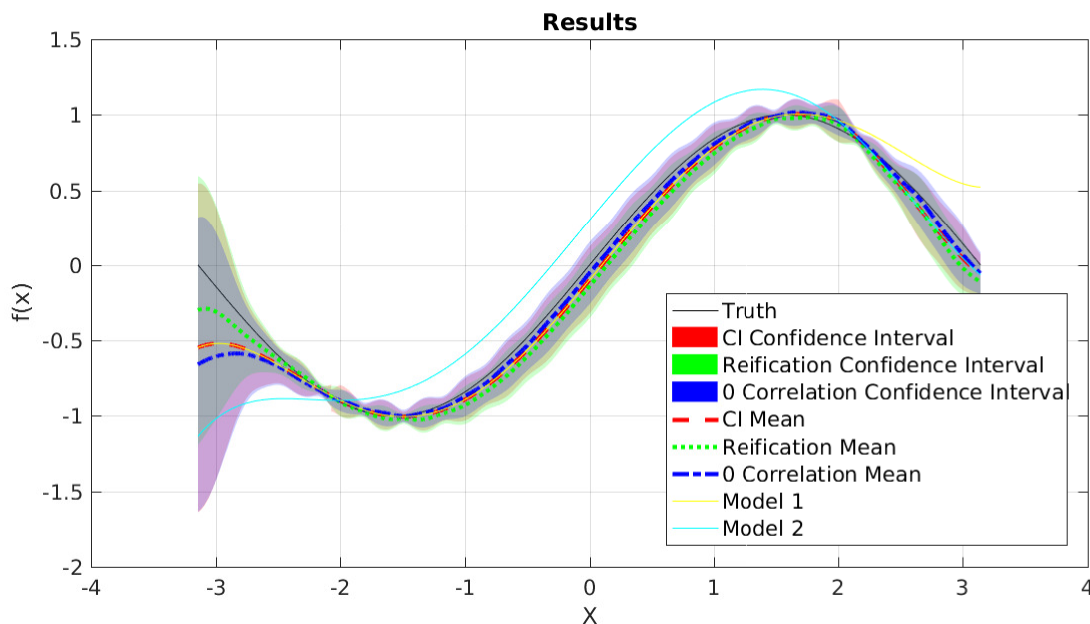


Figure 8. Mean and two sigma confidence interval of model 1 and model 2.

The last item needed in Algorithm 1 is $\mathcal{S}(\mathbf{x}^*)$, or the neighborhood of points around \mathbf{x}^* . This was chosen to be 10 points uniformly sampled from $-0.05 \leq \mathbf{x}^* \leq 0.05$. This can be modified depending on the degree of non-linearity of the design space. Algorithm 1 can now be performed to estimate the correlation at all points in the design space. This correlation is used in Eqs. (4) and (5) to estimate the mean and variance of the new fused estimate. Doing this over the entire design space yields Fig. 8, which compares the result to Covariance Intersection algorithm and fusion assuming zero correlation.

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 1. Mean squared error for models 1 and 2 without fusion, the no correlation case, the Covariance Intersection algorithm, and the reification approach.

There are several notable features illustrated in this plot. Due to this being a single dimension problem, Covariance Intersection always selects the model with the least variance at all points in the design space. While this does yield consistent results, it is not effectively leveraging information from both models. 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(f) where two correlated models allow for a prediction that is skewed beyond the higher fidelity model. 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. Overall, the reification approach performs very well in this problem as shown in Table 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).

IV.B. 1D CFD Demonstration

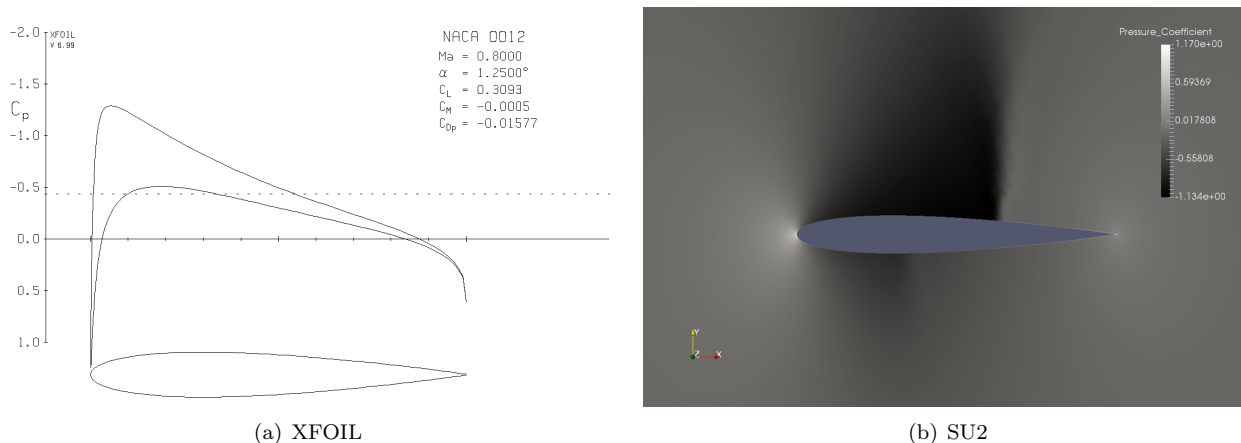


Figure 9. Example outputs of NACA 0012 airfoil from XFOIL and SU2.

This demonstration uses the computational fluid dynamics programs XFOIL³⁰ and SU2³¹ as the two simulators. The airfoil of interest is the NACA 0012, a common validation airfoil. The “truth” model used to validate the method and estimate model discrepancy is real-world wind tunnel data of the NACA 0012 airfoil^{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 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 drag³⁴. 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 values of Mach number and angle of attack. This accuracy comes with orders of magnitude increase in computational expense. Fig. 9 shows an example output of the two simulators that illustrates the difference in fidelity levels. To validate and quantify this expectation, 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 Covariance Intersection algorithm and fusion with zero correlation. A comparison between SU2, XFOIL, and the truth model is shown in Fig. 10. As expected, SU2 performs better than XFOIL at higher angle of attack.

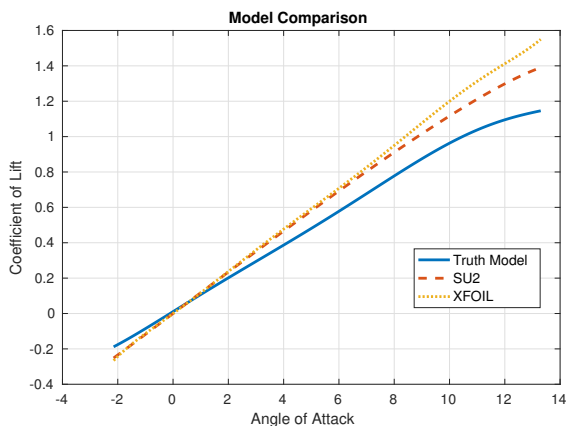


Figure 10. Coefficient of lift estimates from SU2, XFOIL, and wind tunnel data.

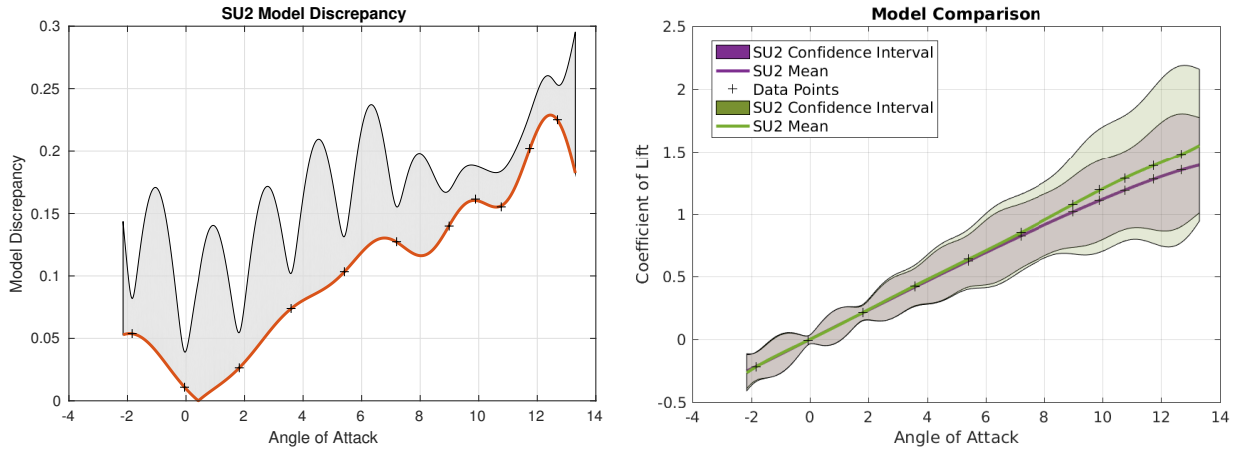


Figure 11. Gaussian process regression of model discrepancy of SU2. Figure 12. SU2 and XFOIL coefficient of lift predictions with two sigma confidence interval.

Following the same procedure as the previous demonstration, the model discrepancy is 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. 11. Following Eq. (22), the uncertainties were incorporated to produce Fig. 12.

Similar to the previous example, 10 points were sampled uniformly from $-0.05 \leq \mathbf{x}^* \leq 0.05$ for $\mathcal{S}(\mathbf{x}^*)$, or the neighborhood of points around \mathbf{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 \mathbf{x}^* . Algorithm 1 can be performed to estimate the correlation at all points in the design space. This correlation is used in Eqs. (4) and (5) to estimate the mean and variance of the new fused estimate. Doing this over the entire design space yields Fig. 13, 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 majority of the design space. This is again due to the property illustrated in Fig. 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 as can be seen in Fig. 12. A deeper exploration of this phenomena is a topic of future work.

Table 2 shows the mean squared error for each method as well as the individual simulators. The reification method yields a 66% reduction in error compared to the Covariance Intersection algorithm, which always uses the best model. This result shows that the reification method has the potential to offer significant increases in accuracy for situations where multiple models are available.

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

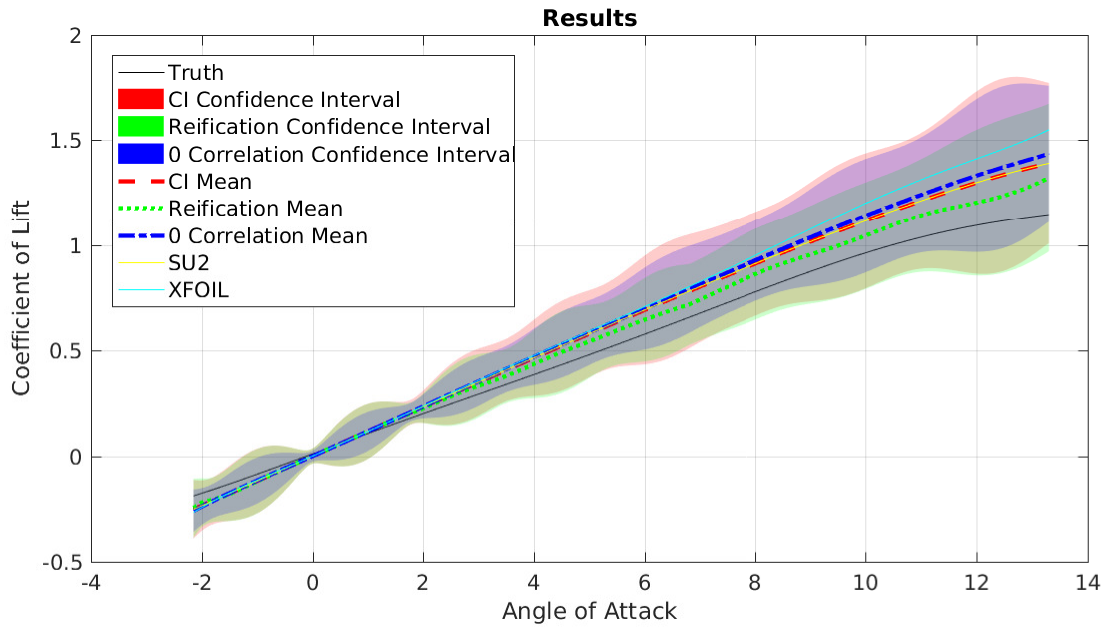


Figure 13. Results of no correlation case, Covariance Intersection algorithm, and the reification approach.

IV.C. 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 flows. 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 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.

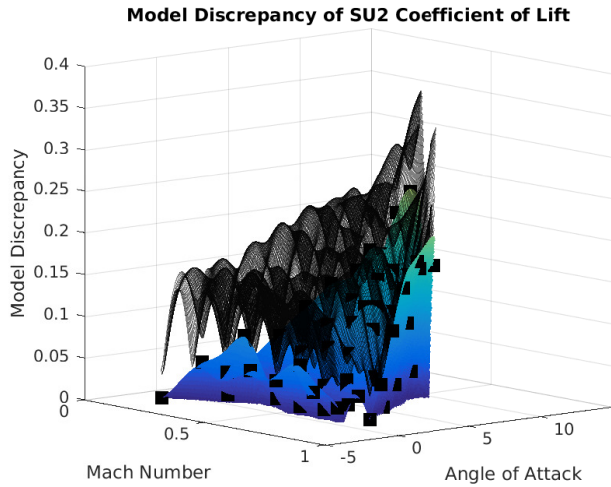


Figure 14. Gaussian process regression of model discrepancy of SU2 coefficient of lift with two sigma confidence interval mesh.

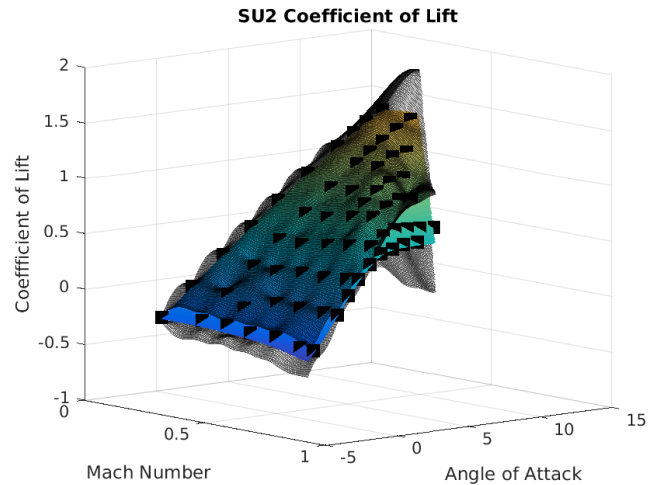


Figure 15. SU2 coefficient of lift predictions with two sigma confidence interval mesh.

Following the same procedure as the previous demonstration, the model discrepancy is estimated by taking the difference between the wind tunnel data and our models. 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 the SU2 coefficient of lift discrepancy plot in Fig. 14. Following Eq. (22), the uncertainties were incorporated to produce Fig. 15.

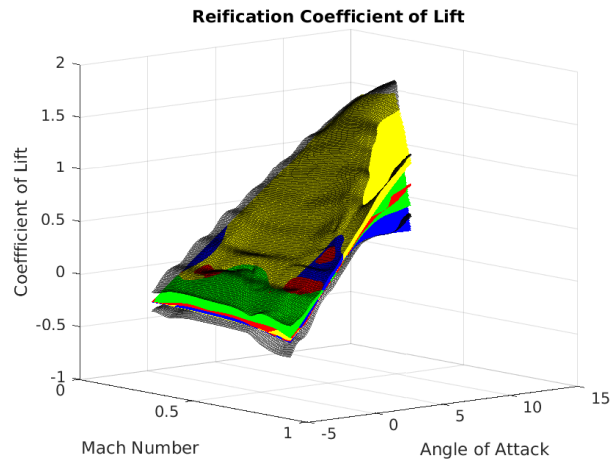


Figure 16. Results of the reification method for C_L . XFOIL is yellow, SU2 is blue, truth is green, the reification method mean is red, and its two sigma confidence interval is the black mesh.

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.

Table 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 error compared to the Covariance Intersection algorithm for both lift and drag. Note the Covariance Intersection mean squared error is actually greater than

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.

Algorithm 1 can be used to estimate the correlation at all points in the design space. This correlation is substituted into Eqs. (4) and (5) to estimate the mean and variance of the new fused estimate. Doing this over the entire design space yields Fig. 16, 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.

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 reification method has the potential to offer significant increases in accuracy for situations where multiple models are available.

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

V. Conclusion

We have presented a method for fusing information for multiple models of varying fidelity. This approach estimates the correlation coefficient between the models, if unknown, in a way that is consistent for any degree of correlation. 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.

Acknowledgment

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.

References

- ¹ Box, G. E. P., “Robustness in the Strategy of Scientific Model Building,” Technical summary report, Wisconsin University - Madison Mathematics Research Center, 1979.
- ² Kennedy, M. and O’Hagan, A., “Bayesian Calibration of Computer Models,” *J.R. Statist. Soc. B*, Vol. 63, No. 3, 2001, pp. 425–464.
- ³ Boström, H., Andler, S., Brohede, M., Johansson, R., Karlsson, A., van Laere, J., Niklasson, L., Nilsson, M., Persson, A., and Ziemke, T., “On the Definition of Information Fusion as a Field of Research,” 2007.
- ⁴ Mosleh, A. and Apostolakis, G., “The Assessment of Probability Distributions from Expert Opinions with an Application to Seismic Fragility Curves,” *Risk Analysis*, Vol. 6, No. 4, 1986, pp. 447–461.
- ⁵ Zio, E. and Apostolakis, G., “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, 1996, pp. 225–241.
- ⁶ Reinert, J. and Apostolakis, G., “Including Model Uncertainty in Risk-informed Decision Making,” *Annals of Nuclear Energy*, Vol. 33, No. 4, 2006, pp. 354–369.
- ⁷ Riley, M. and Grandhi, R., “Quantification of Modeling Uncertainty in Aeroelastic Analyses,” *Journal of Aircraft*, Vol. 48, No. 3, 2011, pp. 866–873.
- ⁸ McFarland, J. and Bichon, B., “Bayesian Model Averaging for Reliability Analysis,” *50th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, Vol. 430, 2009.
- ⁹ Leamer, E., *Specification Searches: Ad Hoc Inference with Nonexperimental Data*, John Wiley & Sons, New York, NY, 1978.
- ¹⁰ Madigan, D. and Raftery, A., “Model Selection and Accounting for Model Uncertainty in Graphical Models Using Occam’s Window,” *American Statistical Association*, Vol. 89, No. 428, 1994, pp. 1535–1546.
- ¹¹ Draper, D., “Assessment and Propagation of Model Uncertainty,” *Journal of the Royal Statistical Society Series B*, Vol. 57, No. 1, 1995, pp. 45–97.
- ¹² Hoeting, J., Madigan, D., Raftery, A., and Volinsky, C., “Bayesian Model Averaging: A Tutorial,” *Statistical Science*, Vol. 14, No. 4, 1999, pp. 382–417.
- ¹³ Clyde, M., “Model Averaging,” In *Subjective and Objective Bayesian Statistics* 2nd ed. Chapter 13, Wiley-Interscience, 2003.
- ¹⁴ Clyde, M. and George, E., “Model Uncertainty,” *Statistical Science*, Vol. 19, 2004, pp. 81–94.
- ¹⁵ Geisser, S., “A Bayes Approach for Combining Correlated Estimates,” *Journal of the American Statistical Association*, Vol. 60, 1965, pp. 602–607.
- ¹⁶ Morris, P., “Combining Expert Judgments: A Bayesian Approach,” *Management Science*, Vol. 23, 1977, pp. 679–693.
- ¹⁷ Winkler, R., “Combining Probability Distributions from Dependent Information Sources,” *Management Science*, Vol. 27, No. 4, 1981, pp. 479–488.
- ¹⁸ Julier, S. and Uhlmann, J., “A Non-Divergent Estimation Algorithm in the Presence of Unknown Correlations,” In *proceedings of the American Control Conference*, pp. 2369-2373, 1997.
- ¹⁹ Julier, S. and Uhlmann, J., “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 Goldstein, M. and Rougier, J., “Reified Bayesian Modelling and Inference for Physical Systems,” *Journal of Statistical Planning and Inference*, Vol. 139, No. 3, 2006, pp. 1221–1239.

- ²¹ Rebba, R., Mahadevan, S., and Huang, S., “Validation and Error Estimation of Computational Models,” *Reliability Engineering and System Safety*, Vol. 91, 2006, pp. 1390–1397.
- ²² Arendt, P., Apley, D., and Chen, W., “Quantification of Model Uncertainty: Calibration, Model Discrepancy, and Identifiability,” *ASME Journal of Mechanical Design*, Vol. 134, 2012.
- ²³ Grime, S. and Durrant-Whyte, H. F., “Data Fusion in Decentralized Sensor Networks,” *Control Engineering Practice*, Vol. 2, No. 5, 1994, pp. 849–863.
- ²⁴ Utete, S. W. and Durrant-Whyte, H. F., “Network Management in Decentralized Data Fusion Networks,” *Photonics for Industrial Applications*, International Society for Optics and Photonics, 1994, pp. 64–73.
- ²⁵ Chen, L., Arambel, P., and Mehra, R., “Fusion under Unknown Correlation: Covariance Intersection as a Special Case,” In proceedings of the International Society of Information Fusion, pp. 905-912, 2002.
- ²⁶ Gao, J. and Harris, C., “Some Remarks on Kalman Filters for the Multisensor Fusion,” *Information Fusion*, Vol. 3, 2002, pp. 191–201.
- ²⁷ Rasmussen, C. and C. Williams, *Gaussian Processes for Machine Learning*, The MIT Press, Boston, Massachusetts, 2005.
- ²⁸ O’Hagan, A. and Kingman, J., “Curve Fitting and Optimal Design for Prediction,” *Journal of the Royal Statistical Society. Series B*, Vol. 40, 1978, pp. 1–42.
- ²⁹ Handcock, M. and Stein, M., “A Bayesian Analysis of Kriging,” *Technometrics*, Vol. 35, 1993, pp. 403–410.
- ³⁰ Drela, M., “XFOIL: An Analysis and Design System for Low Reynolds Number Airfoils,” *Low Reynolds Number Aerodynamics*, Springer, 1989, pp. 1–12.
- ³¹ Palacios, F., Colonno, M. R., Aranake, A. C., Campos, A., Copeland, S. R., Economou, T. D., Lonkar, A. K., Lukaczyk, T. W., Taylor, T. W., and Alonso, J. J., “Stanford University Unstructured (SU2): An Open-source Integrated Computational Environment for Multi-physics Simulation and Design,” *AIAA Paper*, Vol. 287, 2013, pp. 2013.
- ³² Thibert, J., Grandjacques, M., and Ohman, L., “NACA 0012 Airfoil,” .
- ³³ Rumsey, C., “2D NACA 0012 Airfoil Validation Case,” *NASA Turbulence Modeling Resource*, 2013.
- ³⁴ Barrett, R. and Ning, A., “Comparison of Airfoil Precomputational Analysis Methods for Optimization of Wind Turbine Blades,” .