Verification, validation, uncertainty quantification and aggregation for engineering computational models in industrial applications

By

Andrew D. White

Dissertation

Submitted to the Faculty of the

Graduate School of Vanderbilt University

in partial fulfillment of the requirements

for the degree of

DOCTOR OF PHILOSOPHY

in

Civil Engineering

17 December 2022

Nashville, Tennessee

Approved:

Sankaran Mahadevan, Ph.D.

Douglas Adams, Ph.D.

Hiba Baroud, Ph.D.

Jason Valentine, Ph.D.

Alexander Karl, Ph.D.

Here I raise my Ebenezer
–Robert Robinson, 1758

ACKNOWLEDGMENTS

This dissertation is the culmination of the last five years of study, research, and the industrial application and development of VVUQ methods. Getting to this point would not have been possible without the many sacrificial contributors along the journey — too many to name everyone here.

First, my Family has been outstanding in every respect. Words fail me. Thank you for believing in me and for all that you mean to me. This really would not have been possible without your support, long-suffering, and love.

I would like to express my gratitude to my academic advisor Dr. Sankaran Mahadevan for his dedication to excellence in education, his knack for boiling down complex topics into as "simple as possible, but no simpler" to the benefit of his students and inspiring them to the same level of rigor mixed with simplicity and clarity. His mentorship has been valued, and his continual encouragement along the way was reassuring in the dimmest hours of my research. I am honored to be the 50th student he has advised over his 30+ year career at Vanderbilt! I would also like to thank his research Risk, Reliability, and Resilience Engineering research lab group whose research has significantly impacted developments in the field of VVUQ, and has provided a wealth of inspiration for my research. In particular, I have greatly benefited from the work of both recent alumnus and post-docs: Dr. Kyle Neal, William Sisson (PhD candidate), Dr. Pranav Karve, Dr. Paromita Nath, Dr. Yingxiao Kong, Dr. Saideep Nannapaneni, Dr. Shankar Sankararaman, Dr. Chenzhao Li, Dr. You Ling, Dr. Josh Mullins, Dr. John McFarland, and many more.

At Rolls-Royce, I have worked with my industrial PhD advisor Dr. Alexander Karl for over a decade. From his consultation and mentorship advising my Green Belt in Robust Design and pre-PhD work in VVUQ, Alex has always encouraged a continuous improvement mindset. He has been influential in the development of VVUQ capabilities within Rolls-Royce. My progress on the PhD was significantly impacted by his continual support and interest, long and constructive discussions about all things VVUQ, as well as co-authorship and reviews of my research papers.

In addition to Dr. Mahadevan and Dr. Karl, I am grateful for the support and review of my work from the rest of my academic committee Dr. Doug Adams, Dr. Hiba Baroud, and Dr. Jason Valentine. I also appreciate their support in my coursework.

My employer Rolls-Royce has made this financially and practically possible by allowing me the freedom to balance education and work, for which I am extremely grateful. Several people in my organization have helped me from the business case, funding, and guarding my time to align my research and work. A few of those involved over the course of my PhD (current and former colleagues) include: Mark Rhodes (head of the Indianapolis site at the time) who said to "make it happen" in response to my proposal; my current and previous managers Matt Miller, Dr. Loren Garrison and Dr. Boon Chui (and their managers particularly Mike Margetts, Brad Sumner, Brad Fanton) who "trusted me to deliver excellence", supported my endeavors, and encouraged my progress; and the R&T team who funded my work internally Lisa Teague, Phil Bastnagel, Alan Barta, Joe Krok, Reggie McGregor, and more.

Last but not least at Rolls-Royce, I want to share my appreciation for the depth of support from my functional area(s). Particularly, Dr. Jason Schmucker has contributed greatly to my work as a sounding board and contributor to my research as co-author on two of my papers. He has also been my MATLAB® guru go-to, and continues to play a key role in advancing VVUQ in Rolls-Royce. Pawel Chmielarczyk's Python/PyMC skills came to my rescue in my hour of need and has been a great colleague who is also helping to progress VVUQ methods at Rolls-Royce. Phil Owen and Dr. Peter Smout have promoted and supported my research through securing funding. I also want to thank my previous colleague Zach Grey for his contribution and general math wizardry that both confounded and inspired me, and for his contributions in our co-authored papers. There are too many more to list here, but so many have had only positive feedback and encouraged me in my work.

Finally, I feel that anyone with the stamina to read an entire dissertation is worthy of acknowledgment. And, if this stalwart readership includes those on my named reference list, I thank you for your contributions and offer this final disclaimer on the use and interpretation of your work, from which I am a grateful beneficiary: When one stands on the shoulders of giants, one may inadvertently cause unintended chafing of said shoulders as one reaches higher. With the anticipated forbearance of these generous giants, the author hopes that any such irritation caused may be easily overlooked as a byproduct of the learning process of a student. :)

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

Greek

α	Ratio of target and proposal distributions (Metropolis-Hastings MCMC)
$\boldsymbol{\delta}(\boldsymbol{x}, \boldsymbol{\phi})$	Model discrepancy function
$\boldsymbol{\delta}^{pc}(\boldsymbol{\phi})$	Model discrepancy function in PC space
$\boldsymbol{\epsilon}$	Model error which includes $\boldsymbol{\epsilon}_\delta$, $\boldsymbol{\epsilon}_h$, $\boldsymbol{\epsilon}_s$, $\boldsymbol{\epsilon}_p$, and potentially other sources of error
$\boldsymbol{\epsilon}_\delta$	Model discrepancy (bias) which may be modeled with a discrepancy function $\boldsymbol{\delta}(\boldsymbol{x}, \boldsymbol{\phi})$
$\boldsymbol{\epsilon}_d$	Measurement error
$\boldsymbol{\epsilon}_h$	Discretization error estimator of true error $\boldsymbol{\epsilon}_h^*$
$\boldsymbol{\epsilon}_h$	Model discretization error
$\boldsymbol{\epsilon}_p$	Truncation error due to use of dimension reduction methods, e.g., PCA
$\boldsymbol{\epsilon}_s$	Surrogate model error
$\boldsymbol{\epsilon}_{hg}$	Discretization error estimated using GP method (GPDE)
$\boldsymbol{\epsilon}_{hr}$	Recovery-based discretization error available in the FE model solution
γ	Ratio of the spatial and temporal order of accuracy of the numerical scheme
κ_k	The k^{th} eigenvalue in the diagonal of K
$\boldsymbol{\lambda}$	A vector containing the model reliability metric accuracy requirement for multiple outputs
λ	Accuracy requirement (tolerance) for the univariate model reliability metric
λ_M	Accuracy requirement (tolerance) for the multivariate model reliability metric (which incorporates $\boldsymbol{\lambda}$)
Ξ	Eigenvalues matrix formed by the eigendecomposition of C for active subspace
$\boldsymbol{\phi}$	Set of q discrepancy function parameter
$\boldsymbol{\psi}$	Set of r physics model and discrepancy function parameters
σ_D	Standard deviation of the expanded measurement distribution
σ_d	Standard deviation of the measurements (generic)
σ_d^*	Standard deviation of the 'true' measurements used to produce synthetic measurements
σ_s	Standard deviation of surrogate model error
σ_T	Standard deviation of the measurements (temperature)
σ_y	Standard deviations of the model outputs
σ_z	Standard deviation of the expanded model output distribution
$\sigma_{\hat{f}}$	Standard deviation of GP at the extrapolated estimate $\hat{f}$
σ_τ	Standard deviation of the measurements (time constant)
σ_{hg}	Standard deviation of the GPDE error estimate ϵ_{hg}

Σ	Covariance matrix of the likelihood function (includes sources of model and measurement uncertainty)
Σ^{pc}	Covariance matrix of the likelihood function in PC space
Σ_d^{pc}	Covariance matrix of the multivariate measurement uncertainty in PC space
Σ_D	Covariance matrix of the expanded measurement distribution
Σ_d	Covariance matrix of the multivariate measurement uncertainty
Σ_y	Covariance matrix of the model output distribution
Σ_z	Covariance matrix of the expanded model output distribution
Σ_{zz}	Covariance matrix of the expanded distribution $\boldsymbol{z} - \boldsymbol{z}'$
$\tilde{\Sigma}$	Covariance matrix combining measurement and model output distributions (validation)
σ_θ	Proposal distribution tuning parameter (Metropolis-Hastings MCMC)
$\boldsymbol{\theta}$	Set of p physics model parameters
$\boldsymbol{\theta}^*$	'True' parameter values used to produce synthetic measurements
τ	Time constant computed from the model/measurement transient output
$[\cdot]^b$	Indicates bias form of validation metrics, also uses expanded distributions z, D
$[\cdot]^c$	Indicates 'centered' (shape) form of validation metrics, also uses expanded distributions z, D
$[\cdot]^o$	Indicates use of expanded distributions z, D

Alpha

A	Area metric
$\hat{a}$	Observed order of accuracy of mesh refinement
a^*	Formal order of accuracy of mesh refinement
B_k	Number of elements in mesh level k
$b_{\hat{f}}$	Bias error between $\hat{f}$ and the model output at mesh level k
Cov$[\cdot]$	Covariance
C	Matrix that represents the expected value of the gradient outer product
c_θ	Proposal distribution tuning parameter (Metropolis-Hastings MCMC)
$\boldsymbol{D}$	Expanded measurement distribution for validation
E$[\cdot]$	Expectation
$\mathcal{F}$	Generic function that maps $\mathcal{F} : \mathbb{R}^p \rightarrow \mathbb{R}$
$\mathcal{F}_a$	Active subspace function that maps active variables to a scalar output $\mathcal{F}_a : \mathbb{R}^{p_a} \rightarrow \mathbb{R}$
f	Generic model output for discretization error discussion (Ch. 3)
$F(z)$	CDF of model outputs z
f^*, $\hat{f}$	True and estimated model output f for $h \rightarrow 0$ and $\Delta t \rightarrow 0$

$F_D(z)$ CDF of measurements D

F_s Safety factor used in the GCI

$\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta})$ Function representing the physics model

$\boldsymbol{g}_d(\boldsymbol{x})$ Function representing the physical measurement process

$\boldsymbol{g}_s(\boldsymbol{x}, \boldsymbol{\theta})$ Function representing the surrogate model used to replace the physics model

$\boldsymbol{h}(\boldsymbol{x})$ Known functional form representing missing physics in $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta})$

h Representation of element size (characteristic size for unstructured 2D mesh)

h_n Expansion factor for the accuracy requirement standard deviation for n-dimensional outputs

K Eigenvalues of the covariance matrix used in PCA (K is diagonal)

L^{pc} Likelihood function in PC space

$\boldsymbol{M}_j$ Distribution (across model samples) of MD's for measurement j

M_{ij} MD for model sample i and measurement j

$\mathcal{N}(\cdot, \cdot)$ Gaussian (normal) probability distribution

n Total number of model outputs at the n_ℓ measurement locations

n_ℓ Total number of measurement locations

n_c Number of characteristic features post-processed from the transient solution

n_{pc} Number of retained PCs of of the original n, where $n_{pc} \ll n$

n_{ts} Number of time steps in the transient FE solution

N Number of model runs (DOE or Monte Carlo)

N_d Number of measurement replicates

p Number of physics model parameters

$P(\cdot)$ Probability distribution

p_a Number of active variables (active subspace)

p_f Probability of failure

p_x Number of physics model parameters

P_{mh} Target distribution (Metropolis-Hastings MCMC)

q Number of discrepancy function parameters

r_{mh} Acceptance ratio (Metropolis-Hastings MCMC)

$\bar{R}_M$ Mean multivariate model reliability metric

r Total number of model and discrepancy function parameters ($r = p + q$)

R, R^o Model reliability metric

r_x, r_t Uniform mesh and time-step refinement ratios, respectively

R_{Mj}, R^o_{Mj} Multivariate model reliability metric for measurement j

Std$[\cdot]$	Standard devation
S_i^k	First order Sobol' index for θ_i and PC k
t	Time in seconds during transient square cycle
U	Eigenvectors of the covariance matrix used in PCA
U_1, U_2	Retained and truncated eigenvectors of U for PCA
Var$[\cdot]$	Variance
$\boldsymbol{w}_k$	Single eigenvector active subspace ($\boldsymbol{w}_k = W_a$), used to form active parameters $\boldsymbol{w}_k^T\boldsymbol{\theta}$ for PC k)
W, W_a	Eigenvector matrix formed by the eigendecomposition of C, where W_a is the active subspace
$\boldsymbol{X}$	Matrix of partial derivatives of $\boldsymbol{\mathcal{Y}}$ (aka Jacobian)
$\boldsymbol{X}^{pc}$	Matrix of partial derivatives of $\boldsymbol{\mathcal{Y}}^{pc}$ (aka Jacobian in PC space)
$\boldsymbol{x}$	Set of p_x physics model inputs
$\boldsymbol{x}^*$	'True' input values used to produce synthetic measurements
$\bar{\boldsymbol{y}}$	Mean model outputs
$\bar{\boldsymbol{y}}_d$	Mean measurements
$\hat{\boldsymbol{y}}^{pc}$	Model output transformed into PC space and truncated to n_{pc} PCs
$\boldsymbol{\mathcal{Y}}$	Discrepancy-augmented model outputs
$\boldsymbol{D}^{Vi}$	Measurements for model validation
$\boldsymbol{y}$	Matrix of model results including n outputs and N samples
$\boldsymbol{y}^{pc}$	Model output transformed into PC space, i.e., the principal components (PCs)
$\boldsymbol{y}_0$	Standardized model outputs
$\boldsymbol{y}_d$	Matrix of n measured outputs and N_d replicates
$\boldsymbol{y}_d^C$	Measurements for model calibration
$\boldsymbol{y}_d^V$	Measurements for model validation (replicate samples, before expansion)
T_{rm}	Rim-to-volume-weighted mean temperture (derived model output quantity)
$\boldsymbol{z}$	Expanded model output distribution for validation
$\boldsymbol{z}'$	Expanded model output distribution for validation translated to the mean of the measurement distribution
S_y	Matrix with N rows, each containing a copy of the standard deviations of the original outputs

Acronyms

ASME	American Society of Mechanical Engineers
CDF	Cumulative distribution function, also denoted $F(z)$
DOE	Design of experiments
FEM	Finite element (FE) method

GCI Grid convergence index

GP Gaussian process surrogate model method

GPDE Gaussian process disretization error

HTC Heat transfer coefficient

KOH Kennedy and O'Hagan discrepancy-augmented Bayesian calibration method

MCMC Markov chain Monte Carlo sampling method

MD Mahalanobis distance

MFE Model form error ('missing physics')

MTO Max take-off

PBA Probability Bounds Analysis

PC Principal component

PCA Principal component analysis

PDF Probability density function

PIRT Phenomena identification and ranking table

QoI Quantity of interest

RE Richardson extrapolation

SME Subject matter expert

VVUQ Verification, validation, and uncertainty quantification

CHAPTER 1

Introduction

No one believes the model except the analyst.

Everyone believes the measurements except the experimentalist.

–Unknown

1.1 Overview

In the design of complex systems, physics-based modeling and simulation are increasingly relied upon to explore, understand, and optimize. This optimization may be in terms of performance, cost, weight, reliability, and/or safety of the products they represent. A driving force behind the increased use of these models is to decrease the costs and timescales relative to physical test-focused research and development. However, the quality of decisions made using a model depends on how well it represents the actual physical system. Such issues require careful consideration in high-consequence systems when significant investment or human lives are at stake.

The goal of developing and exploiting physics models is to predict and understand the behaviors of an engineered system to an adequate level of accuracy. The models – conceptual, mathematical, computational – incorporate many simplifying assumptions and approximations, and thus introduce many possible sources of error and uncertainty. Therefore, physical testing remains an important source of truth to ensure the validity of the model-based design. However, the test and measurement of complex systems also provides imperfect information due to constraints in the representation of the full system, the operational environment, and measurement error and uncertainty.

These have driven advancements over the last few decades in rigorous methods for *verification* (estimation of numerical errors) and *validation* (estimation of physics errors). Due to the many sources of uncertainty in both the physics model and measurements used in the validation process, the topics of verification and validation (V&V) and *uncertainty quantification* (UQ) are increasingly considered together when assessing model accuracy and credibility [1–7]. For convenience in this research the acronym VVUQ is used to refer to the collection of steps and methods covered in verification, validation, uncertainty quantification, and related related areas such as surrogate modeling, dimension reduction, and model calibration. The process and methodologies of VVUQ seek to combine two imperfect sources of information – models and measurements – and derive decision-making value by addressing the uncertainty in both.

Despite significant advances in the many facets of VVUQ methodologies over the last few decades, studies illustrating applications to challenging, practical engineering problems have been limited. Published work tends

to focus on certain individual aspects of the process. Further, the application of a more holistic VVUQ process requires significant expertise to piece together the various methods and tools needed to complete all the steps. Therefore, the overarching goal of this research is to extend VVUQ methods to practical applications of high dimensional and multivariate engineering models. By applying these methods to develop a practical VVUQ framework (Chapter 7), sources of uncertainty will be identified and aggregated, while improving model quality and credibility through additional insights gained at each step of the process. Toward this end, a representative physics-based heat transfer model is used to guide the selection of methods and for demonstration. The model is used to predict metal temperatures of a gas turbine engine high-pressure turbine disc during operation.

When pursuing VVUQ for practical modeling and simulation, one must consider how the various sources of uncertainty will be aggregated so that the model prediction – the model's intended use – may be given meaningful error bounds. These sources may include aleatory (random, irreducible) uncertainty and epistemic (lack of knowledge, reducible) uncertainty. There are several approaches toward estimating uncertainty bounds that include classical confidence intervals [8], several non-probabilistic methods including interval analysis [9, 10], fuzzy theory [11], possibility theory [12], and probabilistic approaches such as probability bounds analysis (PBA) [3] and Bayesian methods [5, 13]. The probabilistic/Bayesian approach is used in this research due to its strength in the estimation of model parameter uncertainty and discrepancy through Bayesian inference, the ability to integrate probability-based validation metrics, and it enables a robust uncertainty aggregation process. Despite the maturity of probabilistic approaches, the application of VVUQ to complex and computationally expensive engineering simulations presents many challenges that have not been resolved. A few of these are addressed in this research, as outlined later in this section. Before introducing the research objectives, definitions for the steps of VVUQ are discussed.

1.2 Definitions for the VVUQ process

A process/framework for VVUQ, based on six primary steps, is introduced in Chapter 7 of this dissertation. Several of the individual steps are considered in depth in Chapters 2-6. Therefore, 'working definitions' for the steps of the proposed VVUQ process are given below, which are an amalgamation of several prior works [1, 2, 6, 7, 14] with some adaptation (and in some cases extension) for the purpose of this research. The relationship of these steps to the individual chapters is given in the next subsection.

Step 1. *Model definition*. This step ensures that a model is defined in support of a specific problem and stated requirements that are mutually agreed between a decision-maker, a subject matter expert (SME), analyst/practitioner, VVUQ team, and other stakeholders[1]. The physical phenomenon and model type are

[1]Depending on the application, additional stakeholders may be involved at this stage, or all of these may be a single person. In the remainder of this dissertation, it is assumed that the analyst/practitioner is an SME in the physics of interest (heat transfer) and has sufficient expertise to drive the VVUQ process.

selected by the SME with the goal of meeting the accuracy requirements while balancing cost and risk (model use risks, i.e., the risk of making decisions based on the results and conclusions of a model. See, e.g., [6, 14]). An example of the model definition process is given in Chapter 2.

Step 2. *Verification*. This is "the process of assessing software correctness and numerical accuracy of the solution to a given mathematical model" [2]. In the literature, model verification typically includes two activities: (i) *code verification*, which identifies coding mistakes in the computational model; (ii) *solution verification*, which estimates numerical errors that arise from the formulation of a discrete solution to the continuum mathematical model.

Step 3. *Model reduction.* This step is focused on developing a simplified representation of the physics model to speed up computationally intensive VVUQ simulations. Surrogate modeling and dimension reduction techniques are used to reduce the computational effort; both introduce additional numerical errors that must be quantified and included in the uncertainty aggregation process. (Note that reduced-order models commonly used in mechanics problems, such as aeroelasticity [15], use the same mathematical tools as the dimension reduction techniques pursued here; thus, the quantification of truncation error in reduced-order models can follow the same approach discussed w.r.t. dimension reduction in later chapters).

Step 4. *Calibration*. Physics models often have parameters that are not directly measurable; they need to be estimated indirectly by aligning model outputs to measured system outputs. This inverse process is mathematically challenging due to issues such as uncertainty in the measurements, model errors, and non-identifiability between parameters (e.g., different parameter sets may result in similar model outputs, depending on how the model is parameterized). In addition to finding a parameter set solution, this epistemic uncertainty source needs to be included in the uncertainty aggregation process.

Step 5. *Validation*. This is "the process of determining the degree to which a model is an accurate representation of the real world from the perspective of the intended uses of the model." [1, 16]. Validation often involves a comparison of model outputs to physical experimental measurements using a validation metric [2, 5, 17], and the decision-making process of whether the model is adequate for its intended use.

Step 6. *Prediction*. When a model is deemed adequate ('validated' for its intended use), the model is used to predict the quantity of interest (QoI). If the prediction is in a regime that has not been tested, it is extrapolation. The uncertainty due to this extrapolation should be considered in addition to all of the uncertainty quantified through the previous steps, in order to quantify 'error bounds' on the prediction of the QoI.

1.3 Research objectives and organization of the dissertation

The motivation for this research is to advance the application of VVUQ methodologies to real-world engineering models. Thus, Chapter 2 introduces a representative physics-based model to provide a context and realistic 'sandbox' in which to test and apply the methods proposed in this dissertation. Then, Chapters 3-7 address the five main research objectives, which are summarized below.

Chapter 3. *Solution verification for adaptively refined meshes*

The representative model in this research (Chapter 2) uses adaptive mesh and time-step refinement for the finite element solution. Recovery-based discretization error estimators are used to drive the refinement process. However, they are known to be inadequate representations of the magnitude of error that is required when performing VVUQ. Existing verification methods focus on uniform refinement to quantify this source of uncertainty, and are therefore incompatible with adaptive meshing. This chapter therefore develops an estimator for the adaptively refined FE solution for inclusion in an uncertainty aggregation framework.

Chapter 4. *Efficient calibration of physics-based models*

Physics-based models often require the calibration of unmeasurable model parameters to improve agreement between the model outputs and corresponding physical measurements. A popular choice for calibration in the context of VVUQ is Bayesian inference. The Monte Carlo based solution methods require a large number of model runs which can be prohibitive, and so the physics model is typically replaced by a faster surrogate model. Many available surrogate model methods support only univariate output and perform poorly with high-dimensional inputs. This chapter develops an efficient model calibration approach and novel surrogate model that operate within a transformed, dimension reduced subspace of the model's inputs and outputs.

Chapter 5. *Discrepancy modeling for model calibration with multivariate output*

Due to complexities of replicating real-world systems, physics-based models may exhibit bias (aka, *model discrepancy*) relative to corresponding measurements. This model bias is caused by model form error which results in uncertainty about predictions made with the model. Neglecting this bias during model calibration may result in over-fitting the physics model parameters and therefore poor predictive capability of the model. Treatment of model discrepancy during calibration with an additive discrepancy function/model has been widely studied on lower dimensional, and typically univariate, model outputs. This chapter investigates the advantages, challenges, and application of a model discrepancy function formulated within the transformed subspace of the multivariate model outputs.

Chapter 6. *Multi-metric validation under uncertainty for multivariate model outputs and limited measurements*

Model validation for real-world systems is challenging due to many sources of uncertainty, limited measurements, and multivariate model outputs. There is a wide array of possible validation metrics with which to assess the model. However, many of these are univariate and no single metric (univariate or multivariate) provides adequate information for decision-making. Therefore, the objective of this chapter is to adapt and extend three different validation metrics to (i) improve the decision-making process with respect to the validation assessment through a multi-metric approach, and (ii) extend a multivariate validation metric that enables an overall model assessment that is incorporated into the uncertainty aggregation framework in Chapter 7.

Chapter 7. *Uncertainty aggregation through model development and assessment towards prediction*

Uncertainty quantified throughout various analyses in the VVUQ process should be combined and carried forward when making predictions with the validated model. Several studies have tried to combine the contributions of different sources of error and uncertainty. However, gaps exist when attempting to apply these frameworks in industrial applications. This research therefore pursues a Bayesian paradigm for end-to-end aggregation of multiple sources of error and uncertainty, considering all the steps of verification, validation, and uncertainty quantification. The steps of the proposed framework/process are defined in the next section, and are demonstrated in Chapter 7 for a practical application (the model from Chapter 2).

Finally, a summary of the contributions of this dissertation and recommendations for future work are discussed in Chapter 8.

CHAPTER 2

Model definition (context and application of the research)

This is a sparring program.... It has the same basic rules, rules like gravity.

–Morpheus, The Matrix (1999)

2.1 An engineering physics model for application of the research methodologies

In order to demonstrate the proposed methods throughout this dissertation, a heat transfer physics model was selected that provides an adequate level of model complexity, while remaining manageable in terms of run time. The model represents the high-pressure turbine disc of a commercial aircraft turbofan engine, such as the one shown in Figure 2.1. The model is based on the finite element (FE) method [18], which is used extensively in the design of gas turbine engines [19–21]. For example, following verification, calibration, and validation, the FE model would be used to predict temperatures, which can be used for downstream calculation of quantities of interest such as thermally induced stress in the turbine and compressor discs [20], thermo-mechanically induced expansion of rotor and casings to determine blade tip clearance [21], and maximum component temperatures to guide material selection [19].

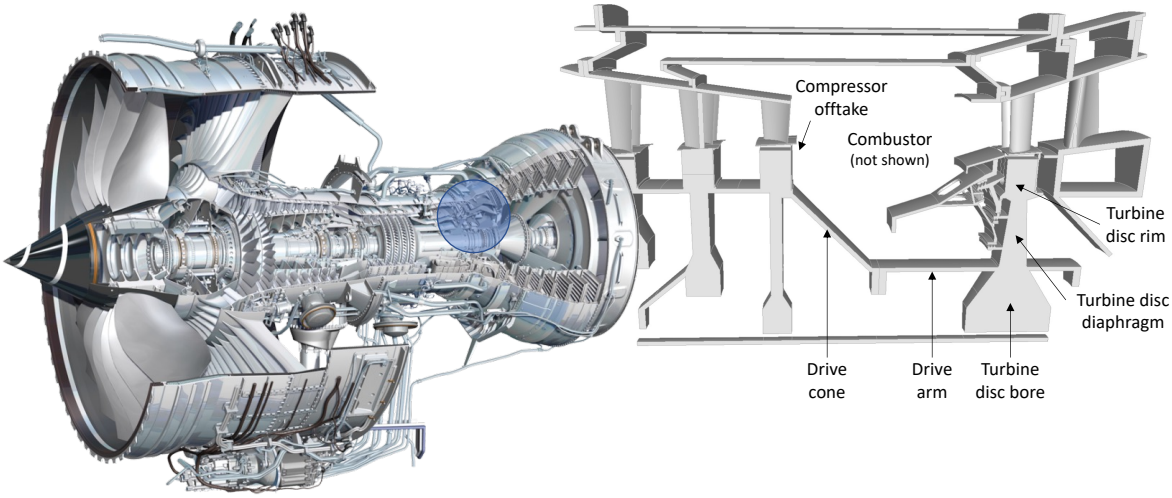


Figure 2.1: A typical gas turbine engine (left) and high pressure sub-system (right). The blue circle highlights the location of the high-pressure sub-system, which is comprised of a 3-stage compressor connected through a drive shaft to the turbine disc. The combustor (where the combustion process occurs) would be positioned between these components, but is not included in the diagram.

A sector cut through the core of the engine in Figure 2.1 (in the region highlighted by the blue circle) would look similar to the simplified 3D model shown on the right side of the figure, which shows the 'hot section' of the engine. The continuous combustion process of the gas turbine engine, the Brayton cycle, is achieved with

four main engine components, the first three of which are shown on the right-hand side of Figure 2.1: (1) an axial multi-stage compressor to provide compressed air, (2) a combustion chamber which mixes fuel with compressed air, and (3) turbine to extract the energy of the combustion process, (4) and a nozzle to accelerate the air and provide thrust [22].

The focus for heat transfer modeling is the turbine disc, which undergoes significant thermo-mechanical loading during engine operation due to high rotational speeds and heating/cooling induced during take-off, flight, and landing. Due to this, the turbine disc is considered a 'critical component' from a safety standpoint, and must undergo careful design, analysis (including modeling and simulation), and physical testing. Because of complexity in both modeling and physical testing within the engine environment, both must be used together to gain the best understanding of the design and ensure safe operation. The purpose of this chapter is to define this heat transfer model and its associated measurements, and thereby to also demonstrate the *model definition* step within the proposed VVUQ framework (Chapter 7).

2.1.1 Basic model requirements

In this application, the basic modeling requirements are to (i) develop an FE model for the prediction of the turbine disc temperatures for test running, (ii) perform the steps of VVUQ to ensure that the model is adequate for its intended use, (iii) predict rim-to-average disc response (as a preliminary indicator of stress sensitivity). There are three quantities of interest (QoI); the first two relate to the steady-state and transient temperature prediction in (i). The third QoI is the prediction of rim-to-average temperature in (iii). This QoI is the ultimate (primary) *intended use* of the model in this example. Thus, the steps in (ii) ensure the quality and accuracy of the model before making the final prediction. Chapters 3-6 focus on particular aspects of the VVUQ process, whereas Chapter 7 develops an end-to-end VVUQ framework for the purpose of uncertainty aggregation. This VVUQ framework is demonstrated for quantifying uncertainty in the prediction of the primary QoI.

Once the basic model requirements are defined, they are used to form a *conceptual model* that will enable predictions to the required level of accuracy. The conceptual model incorporates the assumptions and approximations regarding the physics of the system. A *mathematical model* is then produced based on the conceptual model, typically based on differential equations such as conservation of mass, momentum, energy, etc. Finally, these equations are translated into a numerical *computational model*, a discretized version that can be programmed into a computer code (e.g., FE model) [1, 2]. These modeling steps are briefly discussed in the next few sections in the context of the heat transfer model.

The FE model produces many outputs at many locations, considering the size of the mesh and the length of simulated engine running time. For the purpose of VVUQ, a subset of multivariate outputs at sensor locations is still sufficiently high to prove challenging for many existing statistical methods. Furthermore, the number of

model inputs and parameters can be prohibitively high (e.g., multiple boundary conditions, each with many inputs and parameters, inputs from other models, material inputs). Therefore, the objectives of the model definition step is to ensure that the right physics are modeled, but to balance complexity with sufficiency in meeting the requirements to an acceptable level of uncertainty.

2.2 Conceptual model

The *conceptual model* determines the physical phenomena that should be included and the level of model fidelity (e.g., 2D or 3D, steady-state or transient) that is required to meet the modeling objectives. The conceptual model is typically formulated by a subject matter expert (SME) by consideration of the QoI and the important physical phenomena that are expected to influence these QoI. These choices made by the SME result in the *model form*. Part of the purpose of this and the other steps within the VVUQ process is to ensure that the correct model form has been identified. In this section, the system and its operational environment are defined, then the phenomena are identified.

2.2.1 System description and operation

The first step to determining the physical phenomena involved is to describe the system and its operation. In the case of the turbine disc, model complexities arise due to the spatio-temporal nature of the analysis and the extreme environment and nonlinear behavior due to high rotational speeds and temperatures. As the engine is maneuvered through different operating conditions (or states) during aircraft flight, such as idle, max take off (MTO), and cruise, significant mechanical and thermal stresses occur in the turbine disc.

The system in Figure 2.1 is given further details during the conceptual modeling step, which in this case is the model geometry and an illustration of the air flows around the turbine disc (Figure 2.2), and the engine test cycle which defines the operating conditions (Figure 2.3). The model includes a 3-stage compressor and a single stage turbine. The combustor is not explicitly modeled in this work but sits within the space between these other two subsystems. The compressor and turbine are connected through the drive arm so that a portion of the extracted combustion energy drives the compressor. Outside of this primary cycle (the gaspath), secondary air flows are 'bled' off the compressor and used to purge cavities around the discs, purge rim cavities of the turbine disc, and supply air to internally cooled airfoils. The air temperature heats up significantly through the combustor and provides the heat source to the rim of the turbine disc. The compressor and turbine temperature profiles are indicated in the figure since they are key inputs to the model. They are also sources of uncertainty since it is difficult to accurately measure the temperature profile.

For the purpose of model calibration and validation, the engine testing includes a simplified running maneuver called a square cycle (Figure 2.3). The engine is run to idle and MTO conditions until stabilization (which occurs

at $t = 2000$ and $t = 6000$ seconds for idle, $t = 4000$ seconds for MTO), with rapid acceleration/deceleration maneuvers in between conditions (Figure 2.3, bottom). This provides both steady-state and transient temperature characteristics for evaluation of the model (Figure 2.3, top). Model results for this maneuver are the focus of this research; however, after completion of the VVUQ process, heat transfer model predictions would typically include running a simulated flight cycle, which could then be used for further stress analysis.

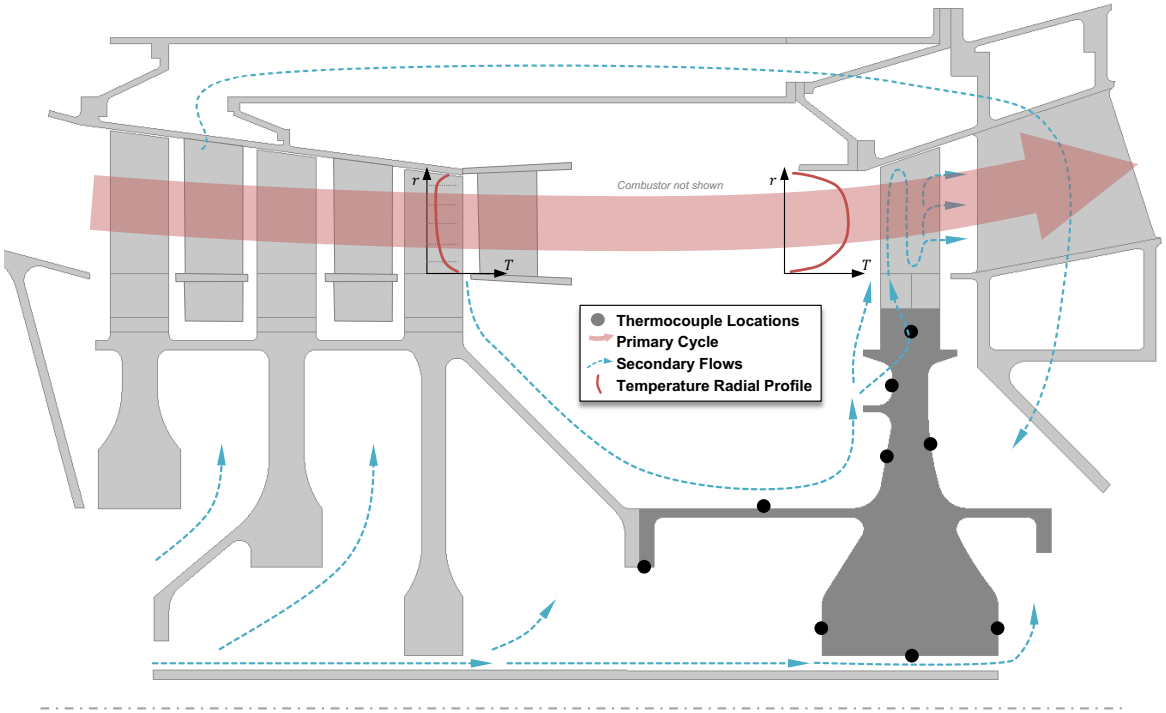


Figure 2.2: Turbine disc 2D axisymmetric FE model, primary and secondary flow paths, and thermocouple measurement locations at which results are extracted for the purpose of model calibration and validation.

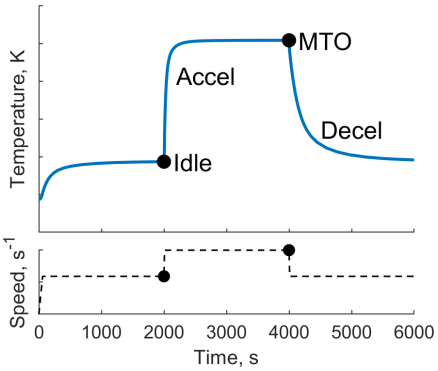


Figure 2.3: Turbine engine square cycle maneuver used for heat transfer model testing. Bottom: engine speed at idle and high power conditions. Top: typical transient temperature response.

2.2.2 Phenomena identification and ranking table (PIRT)

In this section, physical phenomena are selected and recorded for the turbine disc modeling. These are organized using the phenomena identification and ranking table (PIRT) [1, 2, 23] shown below in Figure 2.4. The heat sources for the turbine disc are from the hot gases from the combustor (indicated by the right-hand radial temperature profile in Figure 2.2) which result in conduction at the turbine disc rim (phenomena 1-3) and possible hot gas inflow around the rim (phenomena 4). Convection heat transfer (phenomena 5-7) occurs between the hot turbine disc and the cooling air from the compressor (indicated by the left-hand radial temperature profile in Figure 2.2). Other heat sources include radiation between components in within the engine (phenomena 8) and the frictional heating of the rotating disc due to drag from the surrounding air (phenomena 9). Note that some of these phenomena rely on the additional components included in the model to formulate appropriate boundary conditions. For example, the compressor is included since it supplies cooling air to the turbine, both around the forward side (left-hand) of the disc and around the bores, by bleeding off air from the primary cycle flow.

What is the expected impact of the individual physical phenomena on the QoI?

Ranking System: 1 Low Impact; 3 Moderate Impact; 9 High Impact

#	Physical Phenomena \ QoIs	Rim-to-volume-weighted average temperature (TRV)	Steady-State temperature response at Locations 1-9	Transient temperature response at Locations 1-9	Total
1	Conduction: Through rotating components based on material properties and temperature gradients	9	3	3	15
2	Conduction: Thermal contact resistance between turbine airfoil and disc	9	9	9	27
3	Convection: Heat source from flow path to airfoils/platforms	3	3	3	9
4	Convection: Heat source from flow path into internal cavities (rim inflow)	9	9	3	21
5	Convection: Forced (throughflow+rotation), between secondary flows in rotor-stator cavities	9	3	9	21
6	Convection: Buoyancy (temperature gradients & CF), between secondary flows in rotor-rotor cavities	3	3	3	9
7	Convection: Forced, internal slot for turbine cooling source air	9	3	3	15
8	Radiation: Heat source from combustor	1	3	1	5
9	Heat Generation: Frictional heating due to air drag on rotating components	3	3	1	7

Figure 2.4: Phenomena identification and ranking table (PIRT) records the identified physical phenomena involved and ranks them based on the impact to QoI using expert judgment.

The highest score across QoI (Figure 2.4, Total) represents the SME's importance ranking, which helps to prioritize modeling efforts. The level of confidence in the modeling of a given phenomenon may also be included to weight the scores [1]. This ranking is engineering judgment, and further work in the VVUQ process will confirm or correct these assumptions. For example, phenomena 2 comes out on top. However, it is later shown using sensitivity analysis in Chapter 7 that the model is insensitive to this particular assumption and the factor is removed. Of course, caution and possibly further understanding is warranted when this seeming inconsistency occurs, since this may be the result of assumptions in the computational model (Section 2.4).

2.3 Mathematical model

The physical phenomena identified in Figure 2.4 are then used to formulate a *mathematical model* of the problem, which translates the conceptual model into equation form and includes appropriately defined boundary conditions, initial conditions, and associated inputs and parameters that enable solution of these equations. In industrial settings, it is typical that the mathematical model has already been developed and embedded into commercial or in-house software. For the sake of illustration of the modeling process and explanation of important model parameters, the heat conduction problem is shown (as described by Zabaras [24], Chapter 17). The mathematical model is defined over domain $\mathcal{D}$, boundary Γ, and time interval $t \in [0, t_{max}]$ as

$$\rho C_p \frac{\partial T}{\partial t} = \nabla \cdot (k_m \nabla T) \qquad \text{in domain } \mathcal{D} \tag{2.1}$$

$$T(\boldsymbol{x}, t) = T_g \qquad \text{on boundary } \Gamma_g \tag{2.2}$$

$$k_m \frac{\partial T(\boldsymbol{x}, t)}{\partial \boldsymbol{n}} = q_h \qquad \text{on boundary } \Gamma_h \tag{2.3}$$

$$k_m \frac{\partial T(\boldsymbol{x}, t)}{\partial \boldsymbol{n}} = q_0 \qquad \text{on boundary } \Gamma_0 \tag{2.4}$$

where ρC_p is the volumetric heat capacity, k_m is the thermal conductivity of the disc material, and $\boldsymbol{n}$ is the vector normal to the surface. These material properties, along with Eq. 2.1, address phenomena 1 in Figure 2.4. Part of the boundary is specified with known heat flux q_h, e.g., phenomena 3 (heat source from the main flowpath), while other parts of the boundary have an unknown heat flux q_0, e.g., phenomena 5 (convection heat transfer). In the latter case, the unknown heat flux due to convection is written

$$k \frac{\partial T(\boldsymbol{x}, t)}{\partial \boldsymbol{n}} = q_0 = \theta h_c \big(T_\infty - T(\boldsymbol{x}, t)\big)\Big|_{\boldsymbol{x}=\Gamma_0} \tag{2.5}$$

where h_c is the heat transfer coefficient, θ is a scaling parameter, and T_∞ is the ambient temperature. The convection coefficient h_c may be defined either by using high-fidelity computational fluid dynamics (CFD) modeling or through lower-fidelity empirical modeling. In gas turbine heat transfer applications, the low-fidelity approach is common (likely informed with limited CFD) due to the need to reduce model run-times when performing transient simulations, e.g., during the phases of aircraft flight. A common empirical model used in this application is based on the relationship called the Nusselt number $Nu = h_c L/k_f = f(Re, Pr)$, which is a function of Reynolds number Re and Prandtl number Pr; L is characteristic length and k_f is the thermal conductivity of the fluid. For example, a rotating disc heat transfer correlation developed by Northrop and Owen [25] is,

$$Nu = 0.0197(b + 2.6)^{0.2} Pr^{0.6} Re^{0.8} \tag{2.6}$$

which is based on a power law $\Delta T = cr^b$, where c and b are constants and r is radius.

2.3.1 Uncertainty identification

Uncertainty is introduced through the use of the empirically-based approximations described above. *Parameters* such as θ in Eq. 2.5 are included to allow for adjustments during model calibration. Similar parameters may be applied to other aspects of the model boundary conditions, such as air friction heat generation. The adjustment of these parameters is accomplished through calibration methods such as least squares, maximum likelihood, maximum a posteriori [26, 27], and Bayesian inference [13, 28, 29]. The Bayesian approach is adopted in this research since it quantifies the uncertainty in the model parameters, expressed through prior and posterior probability distributions. The efficient implementation of Bayesian calibration is the topic of Chapters 4 and 5.

The model *inputs*, such as material properties, air properties, geometry, and known heat flux q_h, are defined directly based on either available measurements or results from other models. Both of these types of inputs will include uncertainty. For example, in a typical calibration problem it is assumed that the inputs are well-characterized through measurements during testing. In some cases however, the measurements may only be partially-characterized due to measurement limitations. An example of this situation is given in Chapter 7.

These sources of uncertainty (parameters and inputs) may be categorized as *aleatory* (random, irreducible) or *epistemic* (lack of knowledge, reducible). The model parameters are an epistemic source of uncertainty since obtaining additional measurements (or measurements with reduced uncertainty) for model calibration would reduce the posterior uncertainty. The inputs may involve both aleatory and epistemic sources, where an example of the latter was described for partially-characterized measurements.

A physics-based model may contain dozens or hundreds of inputs and parameters (variables). The job of the practitioner is to determine the subset of these that are significant sources of uncertainty. To start, engineering judgment is often required. Manual studies may be used to inform engineering judgment (varying a parameter to two different settings). Then, a more structured approach based on design of experiments (DOE) techniques such as screening designs, factorial designs, or space-filling DOE [30, 31] may be used. In DOE, the model inputs/parameters are varied through a meaningful range and the model is run many times to obtain a set of model input/parameter-output pairs. This data set enables further down-selection through input/parameter selection methods (e.g., a sensitivity analysis method is used in Section 7.3.3).

In the remainder of this dissertation, the set of p_x inputs and p parameters are denoted $\boldsymbol{x} \in \mathbb{R}^{p_x}$ and $\boldsymbol{\theta} \in \mathbb{R}^p$, respectively. In Chapter 5, a model discrepancy function with q additional parameters is introduced to capture the effect of model form error, resulting in a total parameter set of $r = p + q$ model and discrepancy parameters. In order to demonstrate the research objectives in Chapters 4 and 5, a small subset of representative model parameters were selected based on SME experience (Tables 4.1). For the VVUQ framework demonstration in Chapter 7,

a larger subset of model inputs and parameters is chosen (Table 7.2) and refined using sensitivity analysis to illustrate a more realistic uncertainty identification process.

2.4 Computational model

For typical engineering problems, the mathematical model is solved with numerical algorithms, i.e., with a *computational model*. As is common in many engineering applications, an existing in-house software code is used in this study. The model equations described in the previous section are discretized with the finite element method in space and finite difference in time using standard formulations, e.g., [18]. The code is capable of running solutions on rectangular or triangular 2D grids or the equivalent brick or tetrahedral 3D grids, but it has a built-in meshing capability which produces 3-node or 6-node triangles for 2D applications and 4-node or 12-node tetrahedral 3D elements.

In this application, the SME expects that a 2D axisymmetric model, as shown in Figure 2.2, is sufficient for predicting component metal temperatures of the turbine disc for the intended use of the results. Therefore, the built-in mesh generator's 6-noded triangular elements are used, and automatic mesh and time-stepping refinement are performed to a user-defined temperature accuracy. Additional details regarding mesh refinement accuracy and the estimation of discretization errors are discussed in Chapter 3 (An example of an automatically refined mesh is included in Figure 3.7b.) Model verification should be performed prior to calibration and validation.

Although the emphasis at this stage is on capturing the physical phenomena and turning that into a computational model, a surrogate model [32, 33] is typically needed to perform the many model runs required for VVUQ methods. The surrogate model approach is therefore considered along with model output processing and measurements in the next sub-sections.

2.4.1 Multivariate model output

The FE model solution produces many outputs: spatially across the set of nodes defining the model and temporally through the square cycle (Figure 2.3). However, for the purpose of model calibration and validation, a subset of the spatial output (model temperature output at all nodes) is only post-processed at n_ℓ discrete thermocouple locations, which are indicated on Figure 2.2 (in the figure, $n_\ell = 9$; measurements are discussed in the next section). The temperature output at these locations is a time series similar to the one shown in Figure 2.3 (top), which contains n_{ts} time steps (n_{ts} ranges from 120-170 steps in this research). Despite this initial reduction from many nodes to n_ℓ locations, the output dimension could still be up to $n = n_\ell \times n_{ts} = 1530$.

As noted above, a surrogate model is required to speed up the model runtimes. Since surrogate models usually map the inputs/parameters to a single output, separate surrogate models must be defined for each of these n outputs. Some surrogate models are computationally inexpensive to train, but training n models may still be

expensive. An alternative is to train a time-based surrogate such as 'long short-term memory' [34], but this method is computationally expensive due to the amount of data required for training.

However, due to the autocorrelation that is present in the time series output at each individual location, the dimension of the time series output can be reduced by extracting n_c meaningful 'features' of the time-dependent response (reducing from $\mathbb{R}^{n_{ts}} \rightarrow \mathbb{R}^{n_c}$). This is a form of *model reduction* (see Section 7.3.3), which helps to reduce the computational burden for the purpose of VVUQ analysis. Two approaches to defining these features are used in this research[1], as discussed next.

The first approach, used in Chapter 4 and 5, is to post-process the transient output at each thermocouple location to extract $n_c = 4$ characteristic quantities: stabilized idle temperature (T^{idle}), stabilized maximum take-off temperature (T^{mto}), heating rate (τ^{mto}), and cooling rate (τ^{idle}). The last two represent time constants that are computed by solving for τ in

$$T(t) = T_{start} + (T_{end} - T_{start})\big[1 - \exp(-t/\tau)\big] \tag{2.7}$$

by finding t and $T(t)$ such that $\big(T(t) - T_{start}\big)/\big(T_{end} - T_{start}\big) \approx 0.632$. This value implies $t = \tau$ since $1 - \exp(-t/\tau) \approx 0.632$. In Eq. 2.7, T_{start} and T_{end} represent the idle and MTO stabilized temperatures (or vice versa, depending on whether evaluating the heating or cooling case). The four extracted quantities at the thermocouple positions are grouped into the multivariate model output vector $\boldsymbol{y}$, where $n = n_\ell \times n_c$. Since the VVUQ process relies on DOE and Monte Carlo methods, in this research we treat $\boldsymbol{y} \in \mathbb{R}^{N \times n}$ as a matrix of N rows of the model runs/samples.

The second approach, used in Chapter 7, is to simply extract results at a reduced set of manually selected time instants. The time instants include the two stabilized temperature time points ($t = 2000$ and $t = 4000$ seconds for idle and MTO, respectively) are selected as above to characterize the steady-state response. Then an additional 12 time instants are selected during the transient response to characterize the heating and cooling rates. Thus, in this approach, the multivariate model output now has $n_c = 14$ (and again the total output size is $n = n_\ell \times n_c$).

2.5 Test measurements for model calibration and validation

It has been noted in sections above that thermocouple measurements are made during physical testing for the purpose of calibrating and validating the heat transfer model. The time series produced from the measurements at each of the n_ℓ locations are processed in the same way as the model results processing described in the previous section. Thus, the measured outputs are denoted $\boldsymbol{y}_d \in \mathbb{R}^{N_d \times n}$, where as above $n = n_\ell \times n_c$ (using either of the two approaches to extract n_c quantities per location). N_d is the number of measurement replicates. In the

[1] An alternative approach is to apply PCA to the entire time series [35, 36] or to the combined spatio-temporal model outputs [37], although the reduced outputs size n_c is useful when considering the validation problem in physical space (Chapter 6).

heat transfer model example, due to the expense of a single engine test, there is unlikely to be engine-to-engine replicates. However, the engine is fitted with two thermocouples at each of the n_ℓ locations, spaced circumferentially 180^o apart. This is done primarily for redundancy against thermocouple failure. Due to symmetry the two measurements may also be considered 'replicate' measurements. Thus, $N_d = 2$ in general in this research, unless otherwise noted.

Due to the limited number of replicates (samples), there is additional uncertainty in the measured outputs regarding the true mean value. There are also too few samples from which to estimate the measurement uncertainty. Therefore, it is assumed in this research that a 'known' (prescribed) measurement uncertainty is available, based on previous information or experience, in the form of a standard deviation σ_d. This value describes the random and systematic errors of a given measurement system, as might be derived using an engineering standard such as ASME PTC 19.1-2018 [38]. In lieu of further information, we assume zero-mean Gaussian-distributed errors that are independent. Thus, for multivariate outputs $\boldsymbol{y}_d$ this measurement uncertainty is described by a covariance matrix $\Sigma_d = \text{diag}[\sigma_{d1}^2, \ldots, \sigma_{dn}^2]$, where each output may have a different prescribed measurement uncertainty.

For the examples in this dissertation, synthetic measurements are derived by choosing a set of 'true' input and parameter values, $\boldsymbol{x}^*$ and $\boldsymbol{\theta}^*$, respectively. The model is run, the outputs are post-processed as described in the previous section, and then zero-mean Gaussian measurement uncertainty is added σ_d^* to obtain N_d replicates.

In Chapters 4 and 5, the focus is model calibration only. In Chapter 7, calibration and validation are both performed. It is a commonly agreed principle in model definition (e.g., machine learning context [39] or the physics-based model context [1]) that these two activities should be performed using separate datasets in order to better assess the predictive capability of the model. For many engineering applications including the present one, it is challenging to obtain a single dataset, let alone two. There is therefore a spectrum of possible calibration/validation measurement scenarios that result in varying levels of validation quality [40] and add to the complexity of the VVUQ process. A full discussion of this topic is considered beyond the scope of the present work, but a few example scenarios of practical interest are given below in Table 2.1.

Table 2.1: Calibration and validation measurement scenarios

Scenario	Measurements	Physics Model
1	1 engine / test	1 model
2	1 engine, 2 sequential tests	1 model
3	2 engines / tests, different engine configs.	2 model configs., same governing eqns.
4	2 engines / tests, nominally identical engines	1 model

(configs = geometry, boundary conditions, loading, etc.)

If there is sufficient instrumentation in the first scenario, a cross-validation [39] approach may be possible, considering that there are multiple measurement locations. For example, if leave-one-out cross validation were

used, this would be carried out through a repeated calibration process where individual measurement locations are removed from the dataset one at a time. The resulting set of calibrated model parameters could be summarized either through a parameter posterior weighting [41] (see Section 7.3.5.3) or simply combined through averaging.

In the second scenario, the same engine is run through two different test cycles, for example, the square cycle discussed above and another modified set of engine maneuvers. In this case, calibration may be performed using measurements from the first maneuver and validation using the measurements from the second. In the third and fourth scenarios, one test is used for calibration and the second test for validation. The fourth scenario provides the best[2] information since both tests relate most closely to the design configuration. In gas turbine engine testing, it is likely that calibration and validation will need to be performed under one of the first two scenarios, given the expense and time to test an engine. However, for the purposes of demonstrating the VVUQ framework in Chapter 7, it is assumed that sufficient measurements exist such that the fourth scenario applies.

2.6 Conclusion

The model defined in this section is used in the remainder of this dissertation. In Section 1.2 we referred to this process as the *model definition* step of the VVUQ framework, which is to be discussed in Chapter 7. Although the framework is presented as a series of sequential steps to simplify the explanation, model definition is an iterative process as feedback from later VVUQ steps is obtained. For example, after an initial model is selected and defined, solution verification should be performed (Chapter 3). The solution verification process itself is dependent on the selected model and input/parameter settings. After the verification step, the calibration (Chapter 4) and/or validation (Chapter 6) steps may indicate that the model form is incorrect and that the model should be revised. Thus, the verification step must be revisited. However, through this iterative process of model refinement, the model errors and sources of uncertainty are systematically identified and reduced, resulting in better model predictions.

[2]Of course, none of these scenarios provides perfect information. Even if two engines are tested it does not provide a statistical sample of the population. And, there is additional uncertainty in extrapolation from ground testing to prediction under flight conditions. These issues of measurement quality/accuracy and extrapolation require further consideration in future research.

CHAPTER 3

Solution verification for adaptively refined meshes

3.1 Introduction

In the context of computational models, verification is "the process of assessing software correctness and numerical accuracy of the solution to a given mathematical model" [2]. In the model V&V literature, this is split into code verification and solution verification[1]. *Code verification* seeks to identify errors in the computational computer code that describes the model, and often includes comparison against known solutions and benchmark models. In this research, the perspective is from that of a code user who builds a model using an existing in-house or commercial code. Thus, for the current discussion, it is assumed that the code developer has performed code verification[2]. *Solution verification* involves the estimation of numerical errors in a specific application of the code and includes discretization errors (time and space), iterative convergence errors, and round-off errors. Of these, the most significant contributor is discretization errors [3]. Therefore, this chapter focus on the estimation of this quantity for the purpose of uncertainty aggregation.

Discretization occurs in the process of solving the underlying mathematical equations (that describe a continuum) using numerical methods that are solved with a computer, which requires solutions at a finite number of 'grid points' throughout the physical domain. Discretization error is the difference between this discretized solution and the exact solution, and it is estimated for the purpose of uncertainty quantification or for guiding automatic mesh refinement [42]. Methods for estimating discretization error are either *a priori* which are based on the mathematical problem statement or *a posteriori* which are based on the model solution on one or more numerical grids. A posteriori estimators are considered more useful for estimating the magnitude of the numerical solution errors [43], and may be further subdivided into extrapolation-based and finite element-based error estimators [42, 44]. One of the most widely used extrapolation-based methods in engineering is Richardson extrapolation (RE), which is introduced further in the next section. A common finite element-based method is the recovery method, which computes gradients from the solution on a single grid to estimate mesh errors at each node throughout the mesh [42, 43, 45]. Extrapolation-based approaches are considered more accurate than finite element-based methods for the purposes of uncertainty quantification [16, 46], but the latter are useful for adaptive mesh refinement.

An important motivation for developing discretization error estimators is that simply using the relative difference between two solutions with different element sizes may be misleading [2]. The reasoning for this is

[1] As noted in [5], estimating the errors in the surrogate model is another form of verification. This is addressed in the proposed framework under model reduction, see Sections 2.4.1 and 7.3.3.

[2] Of course, this must not be simply assumed, but evidence should be supplied by the code developer that code verification has been completed. For further discussion on code verification, see [2, 42]

illustrated in Figure 3.1, which shows the difficulty of relating the relative error to the true error. The relative error $\epsilon_{h,rel}$ between two solutions f_1 and f_2 will only approximate the true error ϵ_h^* as the solutions asymptotically approach the true continuum solution f^* (h is used to generically represent element size). In general, $\epsilon_{h,rel} \neq \epsilon_h^*$ and should only be used when there are no other options available (this is referred to as the "emergency method" in [47], in which case it is recommended to multiply the result by a safety factor of 3).

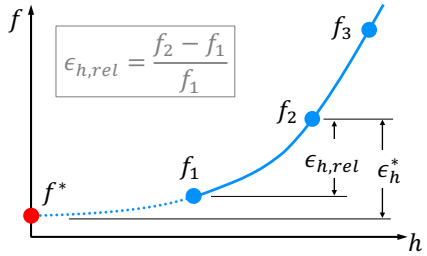


Figure 3.1: Illustration of how relative error $\epsilon_{h,rel}$ may give a misleading estimate of true error ϵ_h^*.

[48]

3.1.1 Richardson extrapolation

Richardson extrapolation is the prevailing method in the VVUQ literature for estimating discretization error [2, 5, 16, 23, 42, 48–50]. Adapting the notation in [42], the standard RE method assumes that the discretization error $f_k - f^*$ for mesh level k takes the form of a power series expansion as a function of element size h_k,

$$f_k - f^* = c_1 h_k + c_2 h_k^2 + HOT \tag{3.1}$$

$$f_k - \hat{f} = c_1 h_k + c_2 h_k^2 \tag{3.2}$$

where c_1 and c_2 are coefficients and HOT in Eq. 3.1 represents higher order terms that have been truncated. Additional terms may be retained for greater accuracy, but this requires additional mesh levels to determine the additional coefficients c_i. The results from two mesh levels are used to solve Eq. 3.2 for the estimated exact solution $\hat{f} \approx f^*$. The HOT are negligible when h_k is within the so-called *asymptotic range*, i.e., the HOT are relatively small compared to the retained terms. It is further assumed in RE that the mesh levels are defined using *systematic mesh refinement*, which relies on a uniform refinement ratio (not necessarily a uniform mesh) and "consistent quality" across the mesh levels [2].

In RE, it is important to show that the *observed order of accuracy* (an outcome of a mesh refinement study) is equivalent to the *formal order of accuracy* of the discretization scheme. The formal order of accuracy is "the value of the exponent of the leading term of the power series expansion of the truncation error" [1] of the discretized model equations. The observed order of accuracy is estimated using the solution for $\hat{f}$ in Eq. 3.2 as follows.

Three uniformly refined meshes (fine=1, medium=2, and coarse=3) are generated using a constant refinement ratio, $r_x = h_{k+1}/h_k$, which is $r_x = h_2/h_1 = h_3/h_2$ for the three mesh case.

For an unstructured mesh, this refinement ratio may be computed from the number of elements of each mesh, $r_x = (B_{k+1}/B_k)^{1/d}$, where B_k is the number of elements in mesh level k, and $d = 1$, 2 or 3 depending on the dimension of the model. The model is run on each of the three meshes to obtain the solution outputs f_k at selected locations around the domain, or as a global function of the model. The observed order of accuracy for each output is computed

$$\hat{a} = \frac{ln\left(\frac{f_3 - f_2}{f_2 - f_1}\right)}{ln(r_x)} \tag{3.3}$$

and the RE estimate of the exact continuum solution is,

$$f^* \approx \hat{f} = f_1 + \frac{f_1 - f_2}{r_x^{\hat{a}} - 1} \tag{3.4}$$

To determine an error estimate for uncertainty quantification, the grid convergence index (GCI) was proposed by Roache [49]. This is computed as the difference between $\hat{f}$ and a given mesh solution f_k,

$$GCI = F_s|\hat{f} - f_k| \tag{3.5}$$

where F_s is a factor of safety. If the observed order of accuracy is similar to the formal order of accuracy $F_s = 1.25$ is suggested, otherwise $F_s = 3$. It is also recommended to limit the range of the observed order of accuracy in the computation of this bound to $0.5 \leq \hat{a} \leq a^*$ the where a^* is the formal order of accuracy [2].

3.1.2 Challenges with Richardson extrapolation

In practice, the standard RE methodology can be quite challenging due to the small element size required (thus high computational expense) to achieve the asymptotic range [46]. Due to this, it is also difficult to show that the observed order of accuracy is similar to the formal order. Oscillatory (non-monotonic) convergence due to coarser meshes indicates that the asymptotic range has not been achieved, and estimation of errors for coarse meshes is not well understood for RE [2]. Another significant issue is the requirement of uniform refinement, which precludes the use of adaptive meshing. Besides loss of the advantages of adaptive mesh refinement (which have fewer elements and are therefore more efficient in computational effort and storage space), uniform refinement is challenging to apply to structured meshes and is even more difficult in unstructured meshes with irregular geometry. Some meshing tools do not offer sufficient control over element size to enforce uniformity, or may only allow uniform refinement for a uniform mesh, which results in higher mesh density than is required throughout much of the mesh. Since adaptive refinement is incompatible with RE, the GCI error estimator is also not applicable.

Several researchers have proposed improvements to RE to partially alleviate these issues. Kammer et al [51] replaced the power series with rational functions which require fewer terms (therefore, mesh levels) to achieve a similar level of error (the reference above also considers a joint surrogate for discretization error and physics model). Eça and Hoekstra [46] proposed a least squares RE and GCI to reduce the effects of scatter in the RE solution. Logan and Nitta [52] discuss ten approaches related to RE, some of which are aimed at dealing with non-monotonic convergence. Thomas et al [53] suggest the use of 'windows' and 'downscaling', which takes a region of the mesh and refines semi-locally to enable uniform refinement such that asymptotic behavior may be achieved. Rangavajhala et al [54] proposed fitting a Gaussian process (GP) model to the grid refinement results and then using this fit to estimate the error as element size is reduced.

While several of the approaches for discretization error estimation in the previous section require an initial assumption on the functional form (e.g., Eq. 3.2), GPs are more flexible and their form is learned from the available data [55], which in this case are solutions f_k from the mesh refinement study. In addition, the GP naturally offers error bounds on predicted values. A potential downside in the use of GP is that it is generally considered to perform poorly in extrapolation, and extrapolation from finite h to $h = 0$ is the goal of discretization error estimation methods. However, GP is adopted in this chapter, and as discussed in the next section, the extrapolation in this application is 'bounded' in a practical way.

The use of the GP alleviates the practical challenges in RE for the present application in two ways. First, the work by Rangavajhala estimated discretization error for multidisciplinary meshes and mesh interfaces using uniform refinement. In this chapter, the use of GP is extended to the case of estimation of discretization errors for adaptively refined meshes for use within an uncertainty aggregation framework. Second, even for uniform element size refinement, the meshes cannot be easily confirmed to achieve the asymptotic range due to practical constraints of the mesh tool and geometry. Therefore, by using GP, this constraint is relaxed.

3.2 Methodology

The basic approach proposed in this section is to first fit a GP[3] to a model output f_k for several levels k of adaptively refined meshes, then extrapolate the GP toward the estimated continuum solution $\hat{f}$. At this extrapolated point, the GP variance $\sigma^2_{\hat{f}}$ and bias relative to mesh level k, $b_{\hat{f}} = \hat{f} - f_k$, are use to define an error estimator referred to in this research as the GP-based discretization error (GPDE) estimate, or ϵ_{hg}. This result could be used directly in place of ϵ_h, however, there is additional uncertainty in the discretization error due to model input/parameter uncertainty. Thus, in a final step ϵ_{hg} is further scaled to incorporate this additional source of uncertainty.

The code used for heat transfer analysis of the model in Chapter 2 is an in-house finite element (FE) analysis

[3]Implemented with MATLAB® function `fitrgp` with defaults, including the ARD squared exponential covariance function [56]

tool at Rolls-Royce with adaptive mesh and time stepping capability. The tool provides access to the results of the finite element recovery-based error estimator ϵ_{hr} that is used for automatic refinement (and is available at all time steps and spatial locations in the FE solution). However, as noted in the introduction, finite element based methods are not considered useful for quantifying the magnitude of the discretization errors. Therefore, the goal of this section is to develop a new discretization error estimator ϵ_h for this adaptively refined mesh. Rather than using the estimate of f^* directly (as in [5]), the resulting ϵ_h is treated as a source of uncertainty within the VVUQ framework presented later in Chapter 7.

This section proceeds as follows. Section 3.2.1 discusses how mesh levels are defined using the mesh tool, both for the case of an adaptively refined mesh and a uniformly refined mesh. Then, Section 3.2.2 outlines the proposed GP-based approach to estimate discretization errors. This approach is applied to both a uniformly refined mesh in Section 3.2.3.1 (for comparison to the RE/GCI-based method), and an adaptively refined mesh in Section 3.2.3.2 (which is the main focus of this chapter, for the estimation of discretization error used in the VVUQ framework of Chapter 7). Finally, Section 3.3 shows how additional error due to input/parameter uncertainty is incorporated into the final discretization error result ϵ_h. The chapter is concluded in Section 3.4.

3.2.1 Setting mesh levels

Discretization error estimates are generated in the next section for the two different mesh refinement studies: uniform refinement and adaptive refinement. The adaptive mesh and time-stepping routines of the in-house code are controlled by error limits that are set by the user. In the following discussion, these error limits are referred to as E_x and E_t for spatial and temporal adaptive refinement, respectively. They are defined in temperature units (Kelvin). The uniformly refined mesh levels may be defined in the in-house code by turning off the adaptive meshing and specifying a characteristic element size h, which is applied uniformly across the mesh. However, the adaptive time-step cannot be turned off, thus the uniform refinement only applies to the spatial aspect and E_t must also be specified, as described next.

To address the relationship between E_x and E_t (in the adaptive case) and h and E_t (in the uniform case) during refinement, the approach by Richards [57] is extended for application to these error limits. Richards related the uniform time-stepping refinement ratio $r_t = \Delta t_k / \Delta t_{k+1}$ to the uniform spatial refinement ratio as $r_t = r_x^\gamma$, where γ is the ratio of the spatial and temporal formal order of accuracy of the numerical schemes ($\gamma = 2$ in this problem). For the present work, the extension is to define r_t using the adaptive time-step error limit settings for two mesh levels, $r_t = E_t^{k+1} / E_t^k$.

In order to perform an adaptive refinement study, mesh levels are defined as follows (the uniform refinement study is similarly defined, except that E_x^k is replaced by h_k and adaptive spatial refinement is turned off):

Step i. Choose desired spatial refinement ratio r_x

Step ii. Generate an initial mesh with mesh settings E_x^1 and E_t^1

Step iii. Compute the number of elements B_1

Step iv. Choose E_x^{k+1} and r_x, then generate a new mesh with $E_t^{k+1} = E_t^k \cdot r_x^2$

Step v. Compute the number of elements B_{k+1}

Step vi. Repeat Steps iii-v to achieve $(B_k/B_{k+1})^{1/2}$ equivalent to the chosen r_x in Step i

Step vii. Repeat Steps iii-vi to obtain the desired number of mesh levels

3.2.2 GP based discretization error estimation (GPDE)

Using the series of meshes defined above, the proposed GPDE approach to estimation of discretization error is carried out as follows for either the adaptive or uniform refinement studies. The process is described for a single model output, but is repeated for the number of outputs.

Step 1. Define a series of refined meshes by modifying error limits E_x^k and E_t^k (adaptive study) or h_k and E_t^k (uniform study), as described in Section 3.2.1.

Step 2. Fit a GP to the model solutions from each of these meshes (for each output of interest) as a function of E_x^k (adaptive) or h_k. The GPs assume a zero-mean basis function, (as in [54]).

Step 3. Estimate the exact solution approximation $\hat{f}$ by extrapolating to $E_x = 0$ (or $h = 0$ for the uniform study), along with its standard deviation $\sigma_{\hat{f}}$ (based on GP uncertainty) and the bias $b_k = \hat{f} - f_k$.

Step 4. Combine these values into the GPDE,

$$\epsilon_{hg} \sim \mathcal{N}\left(0, \sigma_{\hat{f}}^2 + b_k^2\right) \tag{3.6}$$

which is a zero-mean Gaussian distribution with variance $\sigma_{hg}^2 := \sigma_{\hat{f}}^2 + b_k^2$.

3.2.3 Demonstration of GPDE

In the next two subsections, GPDE is demonstrated first for the case of three uniformly refined mesh levels (Section 3.2.3.1), then for five adaptively refined mesh levels (Chapter 3.2.3.2) using the heat transfer model from Section 2.4. The geometry is shown in Figure 3.2 with $n_\ell = 9$ numbered output locations for reference in the results (and a typical adaptively refined mesh). Discretization errors will be computed for each location.

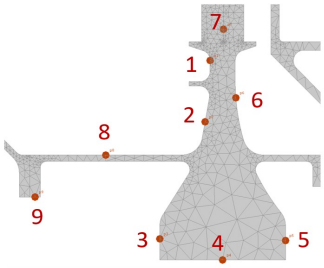


Figure 3.2: Turbine disc thermal model discretization error estimation locations (thermocouple positions).

3.2.3.1 GPDE for uniform refinement compared to Richardson extrapolation

Mesh settings were iteratively modified to achieve the refinement ratio $r_x \approx 1.62$ ($r_t \approx 2.62$) which results in the three mesh level settings for h and E_t listed in Table 3.1. FE solutions for each mesh were computed and outputs were extracted at the 9 thermocouple locations at time $t = 2010$ seconds[4]. Then, GP models were fit to the three solutions as a function of h, which are shown in Figure 3.3.

Table 3.1: Uniform refinement discretization error study settings for element size h and adaptive time step error limit E_t.

Mesh Level	units	1	2	3
Characteristic element size, h	mm	3.04	4.91	7.96
Temporal error limit, E_t	K	0.73	1.91	5

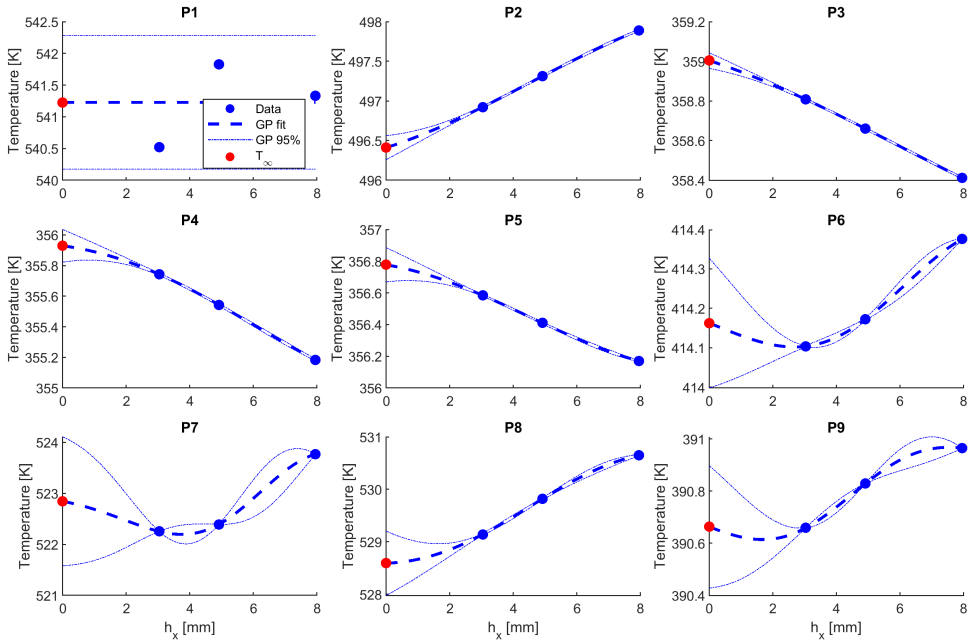


Figure 3.3: GP fit of temperature as a function of average element size [mm] for uniformly refined meshes at locations P1-P9 shown in Figure 3.2. The red point is $\hat{f}$ at $h = 0$.

[4]This time was chosen since it occurs during the transient maneuver of the square cycle (Figure 2.3), where errors are near their worst.

In cases where the GPs in Figure 3.3 are less well-defined in the extrapolation region $h \to 0$ (e.g., P1, P6, P7, P9), it is observed that the estimate for $\hat{f}$ (red point) drifts towards the mean of the three mesh level values f_k and its standard deviation grows toward the standard deviation of the three values f_k. This 'saturation' is most obvious and complete in the case of P1, which fails to find a good fit due to lack of a clear trend. Although tighter uncertainty bounds is preferred, this 'failure' mechanism is a reasonable default when there is no clear convergence toward $\hat{f}$. In the RE/GCI approach, such behavior precludes the use of the estimate altogether.

The GPDE estimate Eq. 3.6 was computed from these results at mesh level 2 and compared to the RE-based GCI (computed using the same meshes) and ϵ_{hr} in Figure 3.4. The RE computations did not achieve agreement between the observed and formal order of convergence[5] due to challenges with achieving refinement in the asymptotic range (location P8 is at a relatively thin structure which prevents uniform refinement; P1 is near boundary conditions that induced high temperature gradients). The recovery based solution (ϵ_{hr}) results in a substantially error than either the GP or GCI approach, which may supports the guidance in [16, 46] that recovery based discretization error estimators are not useful in magnitude for VVUQ. Finally, the comparison between GCI and GPDE is not perfect, but since the GCI results were not within the asymptotic range, it was decided they were satisfactory enough to carry forward with the adaptively refined mesh study. Future work should consider applying this approach to simpler geometry in order to demonstrate observed order of accuracy.

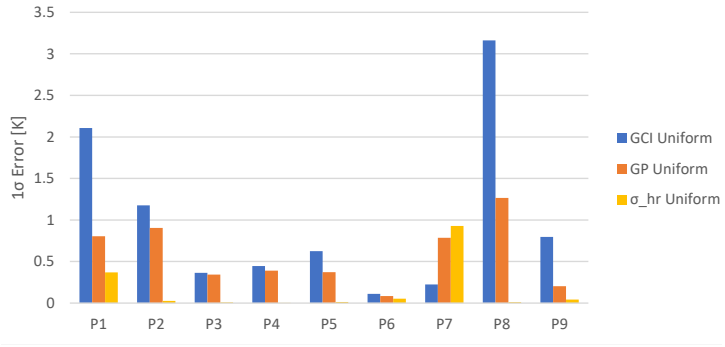


Figure 3.4: Comparison of three discretization error estimators for uniformly refined mesh: RE-based GCI, GP, and the recovery-based error estimator standard deviation σ_{hr}. These represent 1σ errors ($F_s = 1$).

3.2.3.2 GPDE for adaptive refinement

Next, to perform the adaptive refinement discretization study, five cases were defined. Based on SME experience, an upper bound of $E_x = 5$ K and lower bound of $E_x = 1$ K were chosen, resulting in a spatial refinement ratio $r_x \approx 1.5$ and temporal refinement ratio $r_t = r_x^2 \approx 2.24$. The settings for E_x and E_t are listed in Table 3.2. A minimum of $E_t = 0.45$ K was set to avoid excessive time steps. The FE solutions were obtained at the locations P1-P9 and GP models fit as a function of E_x. These GPs are are shown in Figure 3.5 along with the estimated $\hat{f}$

[5]Thus, the GCI approach suggests that a safety factor of $F_s = 3$ is applied when using the results for UQ studies. However, for this comparisons below no safety factor is applied ($F_s = 1$).

(red dot), obtained by extrapolation to $E_x = 0$.

Table 3.2: Adaptive refinement discretization error study with adaptive error limit settings.

Mesh Level	units	1	2	3	4	5
Spatial error limit, E_x	K	1	1.50	2.24	3.34	5
Temporal error limit, E_t	K	0.45	1.01	2.25	5.03	11.25

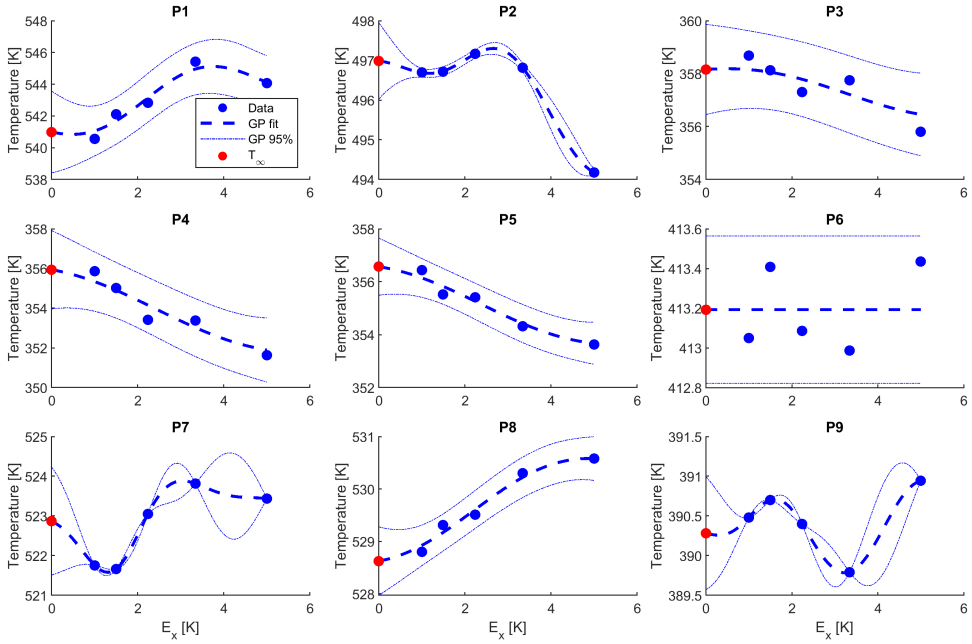


Figure 3.5: GP fit of temperature as a function of adaptive refinement accuracy [K] for adaptively refined meshes at locations P1-P9 shown in Figure 3.2. The red point is $\hat{f}$ at $E_x = 0$.

The GP bias ($b_k = \hat{f} - f_k$), GP standard deviation ($\sigma_{\hat{f}}$), and the GPDE standard deviation (σ_{hg}) are tabulated in the first three rows of Table 3.3. The third mesh level ($k = 3$) is chosen for the purpose of demonstration in this study (in practical situations the finest mesh may not be chosen in order to balance runtime and accuracy). For comparison, the table includes the standard deviation and range of f_k for the five mesh levels. At most of the locations (except P6), the GP's standard deviation is smaller than the 5-solution standard deviation ($\sigma_{\hat{f}} < \text{Std}[f_k]$), indicating that the extrapolation has not reached saturation.

Table 3.3: GP discretization error study results for adaptive refinement (in Kelvin, at $t = 2010$ sec.).

Location →	1	2	3	4	5	6	7	8	9
GP mean bias, $b_3 = \hat{f} - f_3$	1.84	0.18	0.85	2.52	1.15	0.11	0.18	0.89	0.12
GP std. dev., $\sigma_{\hat{f}}$	1.31	0.49	0.87	1.01	0.55	0.19	0.69	0.33	0.36
GPDE std. dev., σ_{hg}	2.26	0.52	1.21	2.71	1.28	0.22	0.72	0.95	0.38
5-solution std. dev.	1.66	1.08	0.98	1.47	0.98	0.19	0.88	0.65	0.39
5-solution range	4.87	3.00	2.88	4.24	2.80	0.45	2.16	1.78	1.15

3.2.3.3 Comparison of GPDE for uniform and adaptive refinement

The uniform and adaptive refinement study results are compared in Figure 3.6. The adaptive results (Figure 3.5) show a higher level of non-monotonicity relative to uniform refinement (Figure 3.3), as one might expect due to local refinement effects. This also results in larger GPDE than the uniform case at several locations, as Figure 3.6 highlights. These results are again compared to ϵ_{hr}, which again show its potential limitations as an estimator for use in UQ. However, as is shown in the next section, due to its availability throughout the FE solution domain (space and time), it will be used to obtain the additional uncertainty due to parameter dependence.

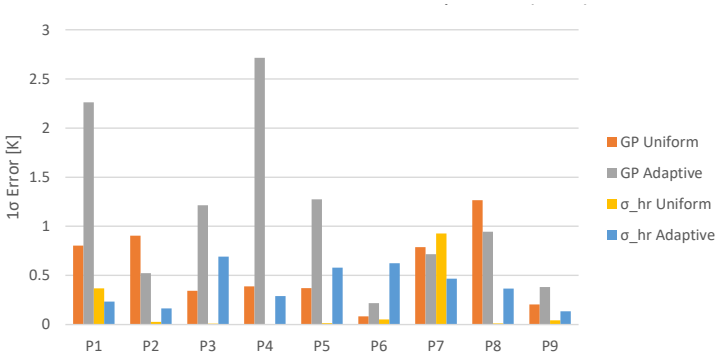


Figure 3.6: Comparison of uniform and adaptive refinement discretization errors for GP and the recovery-based error estimator standard deviation σ_{hr}. These represent 1σ errors ($F_s = 1$).

The medium (mid-level) meshes from the uniform and adaptive refinement studies are shown in Figure 3.7. While the adapted mesh errors are higher in some locations, the number of elements is substantially smaller. This reflects the trade-off between analysis speed and accuracy that the user must choose between. The adaptively refined mesh errors could be reduced by choosing mesh level 1 or 2 (rather than 3).

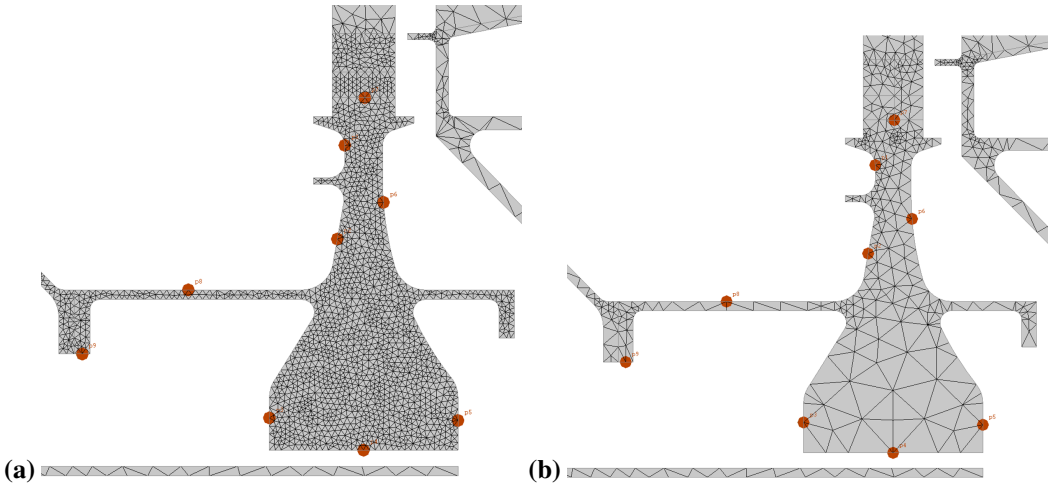


Figure 3.7: Comparison of (a) uniformly refined and (b) adaptively refined meshes. The medium (middle) mesh level is shown for each from the two studies.

3.3 Discretization errors and uncertainty aggregation

Now that the GPDE is computed for the adaptively refined meshes, it may be included in the estimate of total discretization error ϵ_h, which will be used later in this dissertation for the purpose of uncertainty aggregation. Thus, the final step is to consider the additional uncertainty due to variation in the model inputs/parameters. To account for variability in the inputs/parameters using the GPDE approach directly would be computationally expensive: new GPs would need to be re-fit across five meshes for every set of considered input/parameter combinations (e.g., using DOE). Instead, we propose to obtain this variability using the recovery-based error $\epsilon_{hr} = \mathcal{N}(0, \sigma_{hr}^2)$, which is available at all solution points as previously noted[6]. Thus, the total discretization error ϵ_h combines the recovery-based estimated parameter/input variability contribution ϵ_{hr} with the GPDE estimate ϵ_{hg} as follows (for each output):

Step 1. Obtain the variance from the GPDE estimate, σ_{hg}^2 (Eq. 3.6) and FE solution σ_{hr}^2.

Step 2. Define an average scaling factor across the n_ℓ output locations,

$$\beta = \frac{1}{n_\ell} \sum_{i=1}^{n_\ell} \frac{\sigma_{hg,i}}{\sigma_{hr,i}} \tag{3.7}$$

Step 3. Perform a DOE over model inputs/parameters and compute the variance of the recovery-based estimate ϵ_{hr} computed for each DOE run (i.e., the variance of σ_{hr}). Let the the total variance in the recovery-based solution be defined as follows, where the second term is from the DOE results

$$\tilde{\sigma}_h^2 = \sigma_{hr}^2 + \text{Var}[\sigma_{hr}] \tag{3.8}$$

Step 4. Define the total discretization error as a zero-mean Gaussian with a scaled standard deviation that combines the result of Step 2 and Step 3, i.e., $\sigma_h = \beta\tilde{\sigma}_h$,

$$\epsilon_h \sim \mathcal{N}(0, \sigma_h^2) \tag{3.9}$$

The results for these steps are summarized in Table 3.4. The DOE variation $\text{Var}[\sigma_{hr}]$ contributed an additional 22% of the nominal σ_{hr}, on average over the nine locations. Also, $\beta \approx 3.72$ in the study.

Table 3.4: Discretization error standard deviation σ_h, estimated from the GPDE and FEA recovery-based errors (in Kelvin, at $t = 2010$ sec.).

Std. Dev. ↓	Location →	1	2	3	4	5	6	7	8	9
GPDE	σ_{hg}	2.26	0.52	1.21	2.71	1.28	0.22	0.72	0.95	0.38
FEA error	σ_{hr}	0.23	0.17	0.69	0.29	0.58	0.62	0.47	0.37	0.13
DOE error	$\text{Std}[\sigma_{hr}]$	0.08	0.1	0.16	0.11	0.15	0.13	0.09	0.02	0.04
FEA & DOE error	$\tilde{\sigma}_h$	0.25	0.16	0.71	0.31	0.60	0.64	0.48	0.37	0.14
Total discr. error	σ_h	0.92	0.61	2.64	1.15	2.22	2.37	1.77	1.36	0.52

[6]The value reported from the FE tool is interpreted as a standard deviation, σ_{hr}, based on the tool's documentation

3.4 Conclusion

Discretization error estimation is fraught with significant challenges. The prevailing RE-based methods have stringent requirements that may preclude their usefulness in many practical applications due to the need for the mesh to be in the asymptotic range, and when adaptive refinement is used. The estimates are sensitive to the model definition including input/parameter uncertainty, which means that the estimated discretization errors based on a given set of inputs/parameters is not a deterministic quantity. The GP-based approach explored in this chapter shows promise as a pragmatic way forward, allowing additional flexibility for adaptively refined meshes in both space and time. The incorporation of additional uncertainty for variation of the inputs and parameters was also demonstrated.

Benefits of using the GP for the estimation of discretization errors are that specifying a functional form (e.g., polynomial) is avoided and monotonicity (i.e., the asymptotic range) is not mandatory, as is the case for Richardson extrapolation-based methods. There is no strict requirement of refinement within the asymptotic range. It was also shown that the extrapolation properties of the GP (with a zero-mean basis function) 'saturates' to the mean and standard deviation of the solutions f_k of the k mesh levels. This is a sort of fail-safe that, at worst, reflects the variability in the obtained solutions. It is easy to check if this has occurred and the analyst may decide whether performing additional mesh levels would be advantageous.

The results shown in this chapter provide an estimate at the selected time instant ($t = 2010$ seconds). This could be repeated over all solution time steps if required. However, to simplify the application of this error in a conservative manner, results at this time instant are applied to all time steps. This discretization error estimate is used later (Chapter 7) during calibration, validation, and prediction by randomly sampling the normal distribution in Eq. 3.9.

Future improvements to the discretization error estimation process include:

- To gain further confidence in the application of the GPDE approach, it is recommended that a rigorous comparison to RE is performed on simpler geometries with structured/unstructured-uniformly refined, and unstructured-adaptive meshes. The study should consider element sizes from within the asymptotic range to sizes outside the asymptotic range (more practical element sizes, as investigated in this chapter).

- The proposed approach addresses errors at the thermocouple locations. Improved propagation across space and time could be developed by taking advantage of ϵ_{hr} which is available at all nodes and time steps.

- The GP fits were based on a zero-mean basis function, which gave favorable properties when extrapolation reached saturation. However, it is worth investigating a hybrid approach: fit a GP with alternative mean functions, for example one based on the power series used in RE. Alternative kernel functions may also be considered.

- The methodology for quantifying the uncertainty in the discretization error estimate due to dependence on inputs/parameters (which are uncertain) relied on the recovery-based errors available in the FE solution DOE results. Where computational effort allows, the GPDE could be computed during each DOE run. In that case, the surrogate and discretization error GP may be combined as in [51].

CHAPTER 4

Efficient calibration of physics-based models[1]

4.1 Introduction

Physics-based engineering models are subject to many sources of uncertainty, including model form (Section 2.2), inputs and parameters (Section 2.3.1), discretization errors (Chapter 3), and measurements due to limited samples or sparsity (Section 2.5). In this chapter, we develop a computationally efficient approach to model calibration within a Bayesian framework, using the heat transfer model example from Chapter 2.

Model calibration is an inverse problem which may be accomplished with methods such as ordinary least squares (OLS), maximum likelihood estimate (MLE), maximum a posteriori (MAP) estimation, or Tikhonov regularization [26, 27]. All of these result in the best point estimate through some form of optimization. By contrast, the Bayesian calibration [28] approach results in parameter estimates in terms of probability distributions that reflect epistemic uncertainty due to other sources of uncertainty (e.g., measurement uncertainty) and parameter unidentifiability. In this chapter, therefore, Bayesian calibration is applied to the turbine disc model introduced in Chapter 2 so that sources of uncertainty will be accounted for and the resulting parameter probability distribution may be used for further analysis, as discussed in Chapter 7.

Bayesian calibration has been demonstrated by researchers in the context of spatially varying parameters [59], reliability analysis [60], simplified heat transfer modeling [61], gas turbine compressor heat transfer modeling [62], nuclear fuel performance modeling [35], multi-fidelity modeling [63], hypersonic flight model calibration including Active Subspace [64, 65]. Much of this work has used the pioneering framework of Kennedy and O'Hagan [13, 66]. Bayesian calibration is based on Bayes' rule, $P(\boldsymbol{\theta}|\boldsymbol{y}_d) \propto P(\boldsymbol{y}_d|\boldsymbol{\theta})P(\boldsymbol{\theta})$, which states that the product of the *prior* probability of the parameters $P(\boldsymbol{\theta})$ and a *likelihood* function $L(\boldsymbol{\theta})$ is proportional to the *posterior* probability of the parameters $P(\boldsymbol{\theta}|\boldsymbol{y}_d)$, where the notation indicates that the posterior is conditioned on the measurements $\boldsymbol{y}_d$. This will be discussed in more detail in Section 4.4.1.

Bayesian calibration often relies on Markov Chain Monte Carlo (MCMC) sampling [67–69], which draws samples from the joint parameter posterior probability distribution. Each of the thousands-to-millions of MCMC samples requires an evaluation of the heat transfer model; thus, as noted in Chapter 2, it is common to replace the physics model with a fast-running surrogate model to reduce the computational effort. The surrogate model can be 'trained' using smaller set of model runs. These runs are typically based on input/parameter settings defined using a space-filling design of experiments (DOE) [30], such as Latin Hypercube Sampling (LHS) [31].

This chapter considers problems with high-dimensional inputs and outputs, and proposes a novel dimension-

[1] Adapted from [58]

reduced surrogate modeling approach to achieve computational efficiency in Bayesian model calibration. The proposed surrogate modeling approach involves three primary steps. First, since the thermal model outputs are multivariate and correlated [35, 59], a dimension reduction method called *principal component analysis* (PCA) [70–72] is used to transform the correlated model outputs to an uncorrelated set of outputs (which are linear combinations of the original outputs). In the transformed space, the dataset can be represented with substantially fewer dimensions than the original outputs, while preserving most of the information in the dataset (through the variance) [73]. This new set of lower-dimensional outputs, which are called *principal components* (PCs), offer three benefits: (i) Reducing the dimensionality of the outputs means that fewer surrogate models are required. (ii) Most surrogate modeling methods require a scalar response. Since the outputs are uncorrelated, any of the scalar output-based methods may be used. (iii) For Bayesian calibration, the now uncorrelated outputs also simplify the computation of the likelihood function to the product of statistically independent distributions, based on the multiplication rule of probability theory [74].

Second, another dimension reduction method called *active subspace* (AS) is applied to the dataset's gradient-space [64, 65, 75] to form a reduced set of *active variables* from linear combinations of the original model inputs/parameters. In this chapter, only the model parameters are varied (model inputs are treated deterministically), thus AS forms a reduced set of 'active parameters'. A single active parameter is chosen in this case for each PC output, although additional active parameters may be required depending on model complexity.

Finally, regression models (quadratic is used in this research) are fit to the PC-AS transformed dataset, which maps one active parameter to one PC. Thus, the overall surrogate model is formed by the collection of these 1-to-1 mappings and we refer to this as the PC-AS surrogate model.

After formulating the PC-AS surrogate model, we demonstrate Bayesian calibration of a gas turbine disc FE thermal model, followed by forward propagation of the parameter uncertainties using standard Monte Carlo simulation. These steps are summarized as follows, and shown in Figure 4.1. The nomenclature and terminology will be clarified in subsequent sections.

Step 1. Select FE model parameters and prior probability distributions
Step 2. Run FE model through a space-filling DOE to generate outputs
Step 3. Perform PCA on the FE outputs to remove correlations and for dimension reduction
Step 4. Perform AS to reduce input dimension
Step 5. Generate the PC-AS surrogate models
Step 6. Measure physical outputs
Step 7. Transform measurements to the PC-space of the model
Step 8. Formulate the likelihood function in the PC-space
Step 9. Solve for parameter posteriors using MCMC
Step 10. Propagate parameter posteriors through the model to obtain output uncertainty

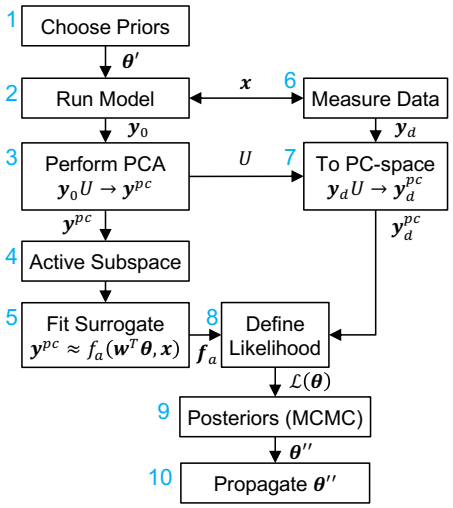


Figure 4.1: Proposed calibration methodology

The rest of this chapter is organized as follows: Section 4.2 discusses the FE model (steps 1 and 2) and measurements (steps 6 and 7) used in the application. Section 4.3 discusses the model transformations and resulting PC-AS surrogate model (steps 3-5). Section 4.4 demonstrates Bayesian calibration of the FE model using the PC-AS surrogate model (steps 8-10). Conclusions and future work are discussed in Section 4.5.

4.2 Model definition

The methodology is presented using the heat transfer model dataset presented in Chapter 2. The specific model details for this chapter are described in this section before defining the dataset for the demonstration of the PC-AS surrogate and its use in the model calibration example. Figure 4.2 repeats Figure 2.2 except that a set of parameters are indicated for the analysis in this chapter and the number of output locations is $n_\ell = 12$ (three additional thermocouples on the compressor). As discussed in Chapter 2.2, a model variable down-selection process is typically required before calibration to determine the driving variables and ensure that the calibration process is tractable. However, in order to focus this chapter on the proposed methodology, a small subset of the possible model parameters $\boldsymbol{\theta} \in \mathbb{R}^p$ were selected. These $p = 5$ parameters are also listed in Table 4.1.

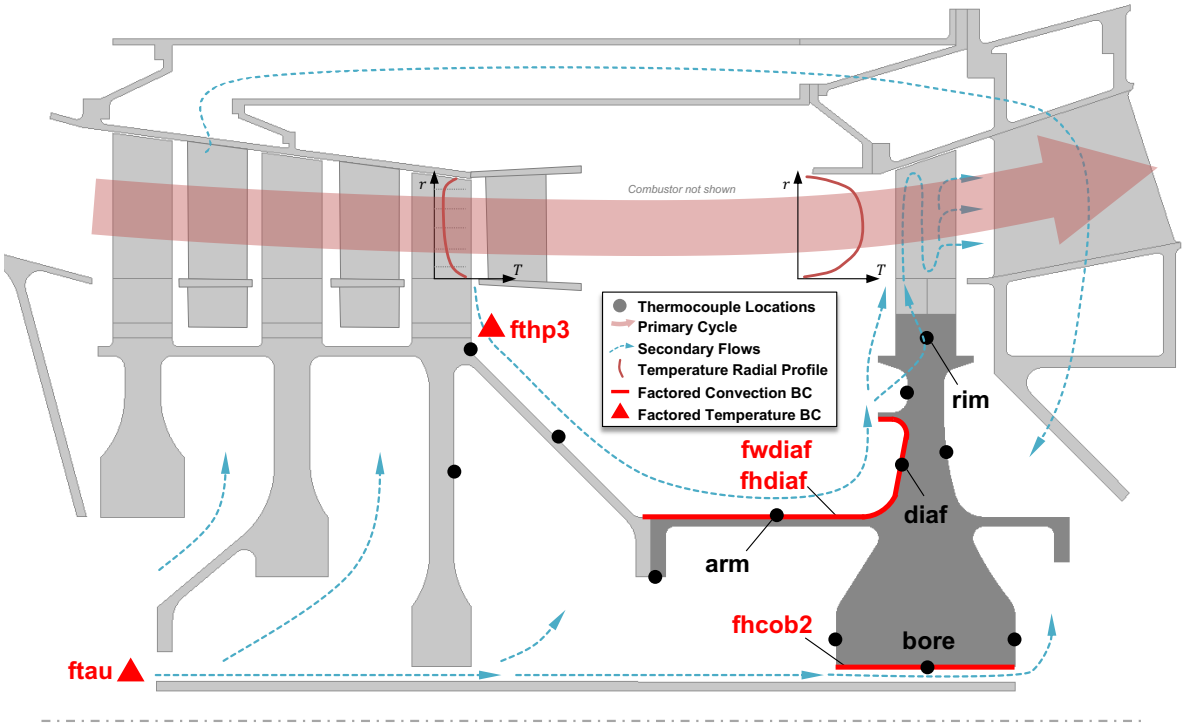


Figure 4.2: Heat transfer FE model showing selected model parameters (in red), and thermocouple locations used in calibration (results will be shown in Section 4.4.4 for the four named thermocouple locations)

The parameter set includes two heat transfer coefficient (HTC) factors, `fhdiaf` applied to a free disc HTC and `fhcob2` applied to the disc bore, which uses an HTC that is a combination of free disc and duct flow. Factor `fwdiaf` is applied to a frictional heating (windage) HTC. Two boundary temperature factors `ftau` and `fthp3` are used to calibrate the source temperatures feeding into the secondary air cavities around the disc. The factor `ftau` modifies the time constant of the air supplied to the compressor disc bores, which is calibrated due to uncertainties from missing engine geometry. Factor `fthp3` modifies the temperature of the air drawn off the hub of the final compressor rotor, which is calibrated due to uncertainties in temperature profile measurements near the hub. These five parameters are given the ranges shown in Table 4.1 $[\theta_{min}, \theta_{max}]$ for the purpose of performing the DOE. These ranges are also used to define uniform prior distributions during Bayesian calibration (see Section 4.4). The range is based on engineering experience. The posterior parameter distributions may provide feedback in the event that selection of this initial choice was poor, e.g., if the posterior results in a tight distribution around the boundary of the uniform prior. Finally, in order to demonstrate the methodology, the column of 'true' values $\boldsymbol{\theta}^*$ in Table 4.1 is used to generate model outputs which are used as synthetic measurements.

Table 4.1: Calibration parameters

	Name	Description	$[\theta_{min}, \theta_{max}]$	θ^*
θ_1	`fhcob2`	HTC, cob/bore ID	$[0.5, 2.0]$	1.7
θ_2	`fhdiaf`	HTC, fwd. diaphragm	$[0.5, 2.0]$	0.7
θ_3	`ftau`	Temp, time lag	$[0.5, 2.5]$	2.0
θ_4	`fthp3`	Temp, compr. offtake	$[0.8, 1.2]$	1.1
θ_5	`fwdiaf`	Windage, fwd. diaphragm	$[0.5, 2.0]$	1.5

4.2.1 Heat transfer model outputs

The multivariate output of the FE model was described in Section 2.4.1. In this chapter, the 6-noded triangular element mesh contains 10,071 nodes (710 for the turbine disc only, the darker gray domain in Figure 4.2). However, measurements $\boldsymbol{y}_d$ are only available at the $n_\ell = 12$ discrete locations and so model outputs $\boldsymbol{y}$ are extracted from this set of locations for the purpose of calibration. The thermocouple locations are selected based on both practical considerations and parameter sensitivities. An engine test may have thousands of instrumentation channels but only a small subset of these will be dedicated to any single component. The thermocouples must be routed through the engine and out to the test stand, which requires a telemetry system for rotating components. Ideally, the locations are also optimized based on parameter sensitivities [59], but this process is not included in this research.

The time step histories for four representative thermocouple locations (out of the $n_\ell = 12$) are plotted in Figure 4.3 to illustrate how the turbine disc temperatures change through the engine test square cycle. These time series contain $n_{ts} = 123$ time steps each, resulting in total output dimension $n = n_\ell \times n_{ts} = 1476$. As described in Section 2.4.1, this is reduced to $n = n_\ell \times n_c$ by defining $n_c = 4$ characteristic quantities: two stabilized temperatures (T^{mto} and T^{idle}) and two calculated time constant quantities (τ^{mto} represents the heating rate of change and τ^{idle} represents the cooling rate). The time constants are defined from Eq. 2.7. Thus, the reduced output set is comprised of $n = n_\ell \times n_c = 12 \times 4 = 48$ outputs (rather than 1476 outputs).

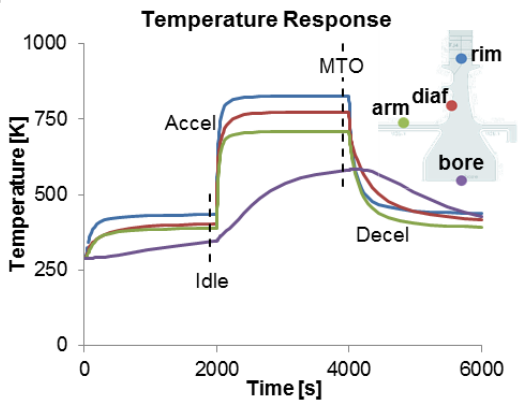


Figure 4.3: Representative FE model transient temperature outputs for four selected thermocouple locations (square cycle).

Then, a LHS DOE was performed to obtain $N = 200$ model runs for the creation of the surrogate model, which provides generous sampling given the five model parameters. In this case, the model runtime was fast (13 seconds per run of 16 models run in parallel, one model run per computer core). After calculating the time constants, the n outputs from the DOE are combined into the output matrix $\boldsymbol{y}$ for further analysis,

$$\boldsymbol{y} = [\boldsymbol{y}_1, \boldsymbol{y}_2, ..., \boldsymbol{y}_n] = \begin{bmatrix} T_1^{mto} & T_1^{idle} & \tau_1^{mto} & \tau_1^{idle} \\ \vdots & \vdots & \vdots & \vdots \\ T_N^{mto} & T_N^{idle} & \tau_N^{mto} & \tau_N^{idle} \end{bmatrix} \in \mathbb{R}^{N \times n} \tag{4.1}$$

It was noted in Section 2.4 that solution verification must be performed prior to calibration. The FE code used in this analysis has adaptive mesh and time-stepping refinement capability to meet a chosen error threshold. However, for the present work a manual verification study was carried out by refining the mesh to a regular mesh sizing of approximately 10 mm edge lengths, relative to a turbine disc radius of 300 mm. The spatial errors were determined to be generally less than 1 K and the worst transient error at the rim thermocouple position was 1.4 K. Temporal accuracy convergence was estimated to be less than 4 K with back-to-back model runs at refinement thresholds of 5 K and 1 K. For the purpose of this study, these errors were deemed acceptable and were not carried forward into the calibration step. A more rigorous approach is considered in Chapter 3 and this discretization error is treated as a source of epistemic uncertainty within the VVUQ aggregation framework described in Chapter 7.

4.2.2 Synthetic test measurements

The process introduced in Section 2.5 is used to generate synthetic measurements using the known parameters values θ^* shown in Table 4.1 and known σ_d^*. Results are first extracted at the $n_\ell = 12$ thermocouple locations and then post-processed to obtain the same $n = 48$ outputs as the model output described in the previous section. Since there are two types of outputs, the σ_d^* measurement uncertainty is comprised of two values called σ_T^* for temperature and σ_τ^* for time constant. The temperature errors were defined as zero-mean Gaussian i.i.d. measurement error with a standard deviation $\sigma_T^* = 3$ K. The time constant measurement error σ_τ^* was related to the temperature uncertainty using a first order Taylor series approximation on the relationship $\tau = g(T)$ from Eq. 2.7. At at a given σ_T^* and mean temperature μ_T^* the first order Taylor series approximation is,

$$(\sigma_\tau^*)^2 = \text{Var}[\tau] \approx \left(g'(\mu_T^*)\right)^2 \left(\sigma_T^*\right)^2 \tag{4.2}$$

As discussed in the last section, the overall number of measurement locations is selected based on practical considerations and the potential information gain from the experiment. For the purpose of demonstration in this example, the selected 12 locations did not include any optimization. In addition to the locations in Figure 4.2,

it is typical in gas turbine engine testing to replicate measurements at these locations in the tangential direction (into the page in Figure 4.2), as a safeguard against thermocouple failure. Often, only two replicates $N_d = 2$ are available at a given axial-radial position due to the cost of instrumentation and measurement channel limitations. Since synthetic data is used in this analysis, $N_d = 2$ will be compared with $N_d = 10$.

4.3 Efficient surrogate modeling

The basic approach to the PC-AS model was described in Section 4.1 and is now described in detail using the dataset generated from the DOE in Section 4.2.

4.3.1 Output transformation with principal component analysis (PCA)

Principal component analysis (PCA) is a matrix transformation that 'rotates' the dataset to derive new outputs, called principal components (PCs). These new outputs are linear combinations of the original outputs that maximize variance in each of the new directions. This is done by solving the finite eigenvalue problem of the covariance matrix of the dataset [70, 71],

$$\Sigma_y = \frac{1}{N-1} \boldsymbol{y}_0^T \boldsymbol{y}_0 = UKU^T \tag{4.3}$$

where $\boldsymbol{y}_0$ is a standardized version of the model outputs (outputs have their mean subtracted and are normalized by their standard deviation), $U \in \mathbb{R}^{n \times n}$ (PCA coefficients) is a matrix containing the eigenvectors, and K is a diagonal matrix (also $n \times n$) containing the eigenvalues. These eigenvalues are the variances of the PCs. The PCs are formed from the eigenvalue-ordered columns of

$$\boldsymbol{y}^{pc} = \boldsymbol{y}_0 U \tag{4.4}$$

where the first PC (column) has the largest variance. The first PC therefore represents the largest proportion of the data. Back-transformation is accomplished by matrix multiplication by the transpose of U, $\boldsymbol{y}_0 = \boldsymbol{y}^{pc} U^T$.

In typical applications of this method, dimension reduction is achieved by splitting the eigenvector/eigenvalues into two sets. The first set retains n_{pc} eigenvectors in $U_1 \in \mathbb{R}^{n \times n_{pc}}$ which form the retained PCs $\hat{\boldsymbol{y}}^{pc} = \boldsymbol{y}_0 U_1 \in \mathbb{R}^{N \times n_{pc}}$. The second set $U_2 \in \mathbb{R}^{n \times (n - n_{pc})}$ represents the discarded (truncated) PCs $\boldsymbol{y}_0 U_2$. Often, the reduction is significant such that $n_{pc} \ll n$, where n_{pc} must be chosen heuristically either using the *variance explained*, or by computing the *truncation error*. These are computed next for this example.

The percentage of variance explained is the sum of variances of retained PCs to the total variance (see relative variance 'Rel Var' in Figure 4.4). If the eigenvalues exhibit a rapid decrease, as shown after the fifth PC in Figure 4.4, we truncate to $n_{pc} < n$. In this example, the compression of the output space to $n_{pc} = 5$ retains 99.5% of the original total variance (i.e., the sum of the first 5 percentage contribution values, shown red in Figure 4.4). These

$U_1 \in \mathbb{R}^{n \times n_{pc}}$ form a new orthonormal basis for the reduced n_{pc}-dimension subspace over the $\mathbb{R}^n$ output space, which become the PCs,

Truncation error is the difference between the original dataset and the truncated dataset, which is reconstructed to recover the approximate original outputs. Reconstruction is achieved by back-transforming the truncated set $\hat{\boldsymbol{y}}^{pc}$ to the original space, $\hat{\boldsymbol{y}}_0 = \hat{\boldsymbol{y}}^{pc} U_1^T$, where $U_1 \in \mathbb{R}^{N \times n_{pc}}$ is the reduced set of eigenvectors. Then, the error is computed as $(\boldsymbol{y}_0 - \hat{\boldsymbol{y}}_0) \odot S_y$, where $S_y = [\boldsymbol{\sigma}'_{y1}, \boldsymbol{\sigma}'_{y2}, \ldots, \boldsymbol{\sigma}'_{yn}] \in \mathbb{R}^{N \times n}$ is a matrix with N rows, each containing a copy of the standard deviations of the original outputs, and $\odot$ is the Hadamard product (element-wise multiplication).

Finally, since calibration will be performed in the PC-space, the test data is also transformed to the PC-space of the model. The data $\boldsymbol{y}_d$ are first standardized with the model output column means and standard deviations to obtain standardized $\boldsymbol{y}_{d0}$, then transformed using the eigenvectors from the model transformation similar to Eq. 4.4)

$$\boldsymbol{y}_d^{pc} = \boldsymbol{y}_{d0} U \tag{4.5}$$

which may again be truncated by using U_1 in place of U.

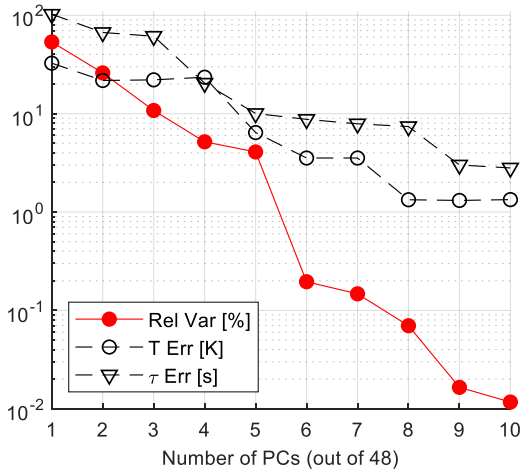


Figure 4.4: Percentage contribution of PCs to the total variance (red), and maximum output reconstruction error of temperatures (T, K) and time constants (τ, s).

4.3.2 Input transformation with active subspace (AS)

The transformed model outputs result in a mapping $\mathcal{F} : \mathbb{R}^p \to \mathbb{R}$ of the parameter set $\boldsymbol{\theta} \in \mathbb{R}^p$ to a given PC ($\boldsymbol{y}_k^{pc}$), to which the AS method [65] is applied ($\mathcal{F}$ is used here for a generic function to simplify notation in this section). If this mapping $\mathcal{F}$ is differentiable and square integrable, a symmetric positive semi-definite matrix may

be defined with the eigenvalue decomposition,

$$C = \int_{\boldsymbol{\theta}} \nabla \mathcal{F}(\boldsymbol{\theta}) \nabla \mathcal{F}(\boldsymbol{\theta})^T \rho(\boldsymbol{\theta}) d\boldsymbol{\theta} = W \Xi W^T \tag{4.6}$$

where $W = [w_1, \ldots, w_p] \in \mathbb{R}^{p \times p}$ is an orthogonal matrix of the eigenvectors, $\Xi = \text{diag}[\xi_1, \ldots, \xi_p] \in \mathbb{R}^{p \times p}$ are magnitude-ordered eigenvalues, $\rho(\boldsymbol{\theta})$ is the sampling density. The parameters $\boldsymbol{\theta}$ are scaled to a $[-1, 1]^p$ hypercube. Since LHS has been used, $\rho(\boldsymbol{\theta})$ is interpreted as a uniform distribution of the random variables $\boldsymbol{\theta}$, as in prior work [76–79]. Another interpretation of Eq. 4.6, based on the definition of expectation $\text{E}[x] = \int x \rho(x) dx$ [80], is that,

$$C = \text{E}\big[\nabla \mathcal{F} \nabla \mathcal{F}^T\big] \tag{4.7}$$

which is to say, C is the expected value (average) of the gradient outer product. The eigenspace of C defines important directions in the domain of $\mathcal{F}$. Ordering the eigenpairs $[\xi_i, \boldsymbol{w}_i]$ of this result in decreasing magnitude of ξ_i indicates that $\boldsymbol{w}_1$ is the most important direction, followed by $\boldsymbol{w}_2$, etc. Identifying these important directions has the potential for dramatic computational implications when considering quadrature rules for integration [81], optimization to minimize or maximize $\mathcal{F}$ [76–79, 82], or approximation of $\mathcal{F}$ [76, 78, 79, 82, 83]. By means analogous to PCA above, a heuristic based on eigenvalue magnitudes is used to truncate to $p_a < p$ such that a p_a-dimensional partition of W, the active subspace $W_a = [\boldsymbol{w}_1, \ldots, \boldsymbol{w}_a]$, captures the majority of the change in the function. The remaining columns of W are the inactive subspace. The function approximation $\mathcal{F}_a : \mathbb{R}^{p_a} \to \mathbb{R}$ is called a ridge approximation over the active subspace [84],

$$\mathcal{F}(\boldsymbol{\theta}) \approx \mathcal{F}_a(W_a^T \boldsymbol{\theta}) \tag{4.8}$$

Plots of the model parameters and PC outputs based on Eq. 4.8 are known as *shadow plots* [85]. In the case that strong trends over the first one or two important directions are observed [75–78, 82, 83], the ability to visualize how the function changes in these important directions allows better selection of the most appropriate type of approximation, i.e., providing strong empirical evidence that a function is predominately linear, quadratic, or more complicated. Moreover, this change of variables is simply p_a linear combinations of the p original parameters. Therefore, the entries of the eigenvectors may be considered as weights indicating the importance of a particular parameter ordered by the corresponding eigenvalue. Thus, the magnitude of the entries of the first eigenvector offers a sensitivity analysis. More information on the sensitivity analysis interpretations can be found in [86].

The definition of Eq. 4.6 depends on gradients, which are often not available in standard FE tools. Therefore, gradient approximations will be based on the space-filling model samples. In this problem, a global linear gradient approximation is assumed for a single eigenvector (see Algorithm 1.3 in [65]). This eigenvector $\boldsymbol{w}_1$ is calculated

from the linear gradients $\boldsymbol{b}$ as $\boldsymbol{w}_1 = \boldsymbol{b}/||\boldsymbol{b}||$, resulting in active parameters $\boldsymbol{w}_k^T\boldsymbol{\theta}$, where the subscript of this single eigenvector is repurposed from here on to represent PC k.

4.3.3 The resulting PC-AS surrogate model

By this combination of PCA and AS, the surrogate modeling problem has been significantly simplified: the 5-input and 48-output problem is compressed into five 1-input and 1-output quadratic surrogate models. Furthermore, AS dimension reduction makes visualization of the surrogate models feasible as shown by the five shadow plots in Figure 4.5. The shadow plots are the PC-space outputs $\boldsymbol{y}_k^{pc}$ plotted as a function of the active subspace parameters $\boldsymbol{w}_k^T\boldsymbol{\theta}$ (blue dots), which are fit with quadratic polynomial regression surrogate models (red curves). The measurements are also transformed into the model's PC-space using Eq. 4.5 (green horizontal lines). The goal of calibration is to find the best values of $\boldsymbol{w}_k^T\boldsymbol{\theta}$ which result in a surrogate model output equivalent to the data, i.e. the intersection of the green and red lines for all k PC-AS shown in Figure 4.5. Re-fitting the PC-AS surrogate model for different numbers of N (retaining 50 test points) and n_{pc} resulted in $R^2 > 0.95$ for $N > 20$ and $n_{pc} \geq 5$.

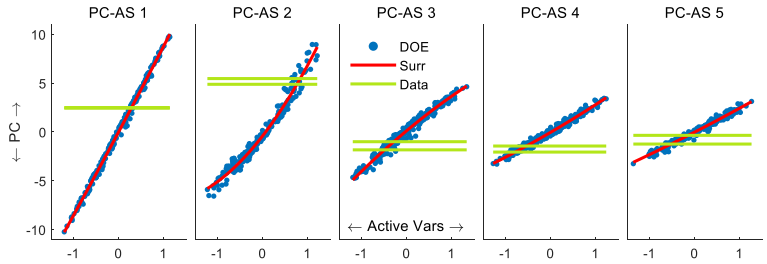


Figure 4.5: Active subspace shadow plots for the first 5 PCs (blue dots), surrogate model fits (red line), and measurements in PC-space (green lines).

4.3.4 Sensitivity analysis of PCs vs eigenvector

Another benefit discussed in Section 4.3.2 is that the eigenvectors of the active parameters provide first-order parameter sensitivity analysis for each PC. These eigenvectors are plotted in Figure 4.6 and are compared to first order Sobol' indices. The first order Sobol' index for θ_i and PC k was generated using a modularized sample-based method [87],

$$S_i^k = \frac{\text{Var}\left[\text{E}[\boldsymbol{y}_k^{pc}|\theta_i]\right]}{\text{Var}[\boldsymbol{y}_k^{pc}]} \tag{4.9}$$

where the expectation is taken over samples within bins of each θ_i to determine the variance of $\boldsymbol{y}_k^{pc}$ due to θ_i. Both approaches use the available 200 LHS samples from the FE model. The magnitude of these two first order sensitivity measures are in good agreement, but the AS approach requires no additional calculation.

Note that in both methods, the sensitivity results are computed for outputs in PC-space. Alternatively, sensi-

tivity results could be calculated for all physical outputs. Calculating sensitivities for all outputs results in many conflicting relationships when the objective is dimension reduction, i.e., parameters that are important to one output may not be important to another output. By computing sensitivity results for the PCs, the affect of a parameter is considered across all outputs. And, in this case there are only 5 PCs versus 48 physical outputs, potentially improving interpretability of the results. How to best use the sensitivity analysis of multivariate outputs merits further investigation (e.g., [88, 89], particularly as the dimensionality of the output space increases (i.e. more thermocouples). An approach toward combining these into a single Pareto chart of importance ranking is shown later in Figure 7.7. Furthermore, interaction effects may be of interest, which requires the calculation of higher order Sobol' indices.

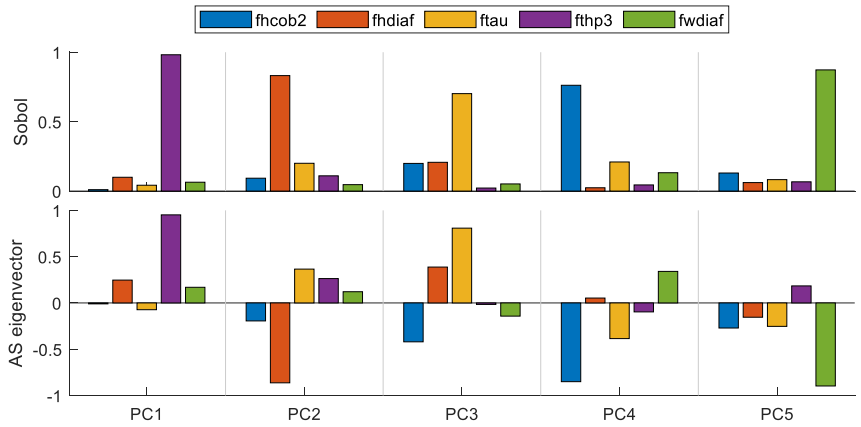


Figure 4.6: First-order Sobol' indices (top) compared to active subspace eigenvectors (bottom) for each PC.

4.4 Application of the PC-AS surrogate in Bayesian calibration

Model calibration is an inverse problem that seeks to align the model outputs $\boldsymbol{y}$ with the measurements $\boldsymbol{y}_d$ by adjusting model parameters $\boldsymbol{\theta}$. This process is illustrated in Figure 4.7. Traditional 'model matching' of complex engineering models has often been either a manual process or optimization-based process [26, 27] that seeks to reduce model to measurement errors below an experience-based tolerance c, e.g., $|y - y_d| \leq c$. These approaches generally do not account for other sources of model uncertainty and result in a point estimate for the parameters. Furthermore, they are challenging when $\boldsymbol{y}$ is multivariate.

As stated in the introduction, the Bayesian approach is used in this research since it treats the parameters as probability distributions (i.e., replaces $\boldsymbol{\theta}$ with $P(\boldsymbol{\theta})$), which enables integration of the calibration process with the other VVUQ process steps (Chapter 7) and allows for incorporating other sources of model, input, and measurement uncertainty. The sources of uncertainty will be reflected in the posterior probability of the parameters. For example, generally, more data and better data (lower measurement uncertainty) will reduce the parameter uncertainties, providing a trade-off between cost of data collection and model uncertainty reduction. The affects

of the number of measurement replicates is considered in the study below.

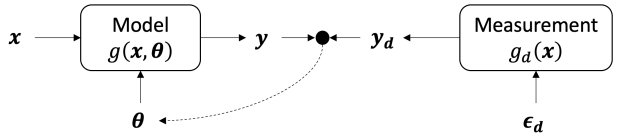


Figure 4.7: Diagram of the model calibration process, where the dashed arrow represents calibration by adjusting $\boldsymbol{\theta}$ to improve agreement between model $\boldsymbol{y} = \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta})$ and measurements $\boldsymbol{y}_d = \boldsymbol{g}_d(\boldsymbol{x})$.

4.4.1 Bayes' rule and setting up the calibration problem

Bayes' Rule relates the parameter posterior distribution $P(\boldsymbol{\theta}|\boldsymbol{y}_d)$ to the prior distribution $P(\boldsymbol{\theta})$ and the likelihood $P(\boldsymbol{y}_d|\boldsymbol{\theta}) = L(\boldsymbol{\theta})$, which is proportional to the probability of observing the measurements given a set of parameters. This relationship is formally stated,

$$P(\boldsymbol{\theta}|\boldsymbol{y}_d) = \frac{P(\boldsymbol{y}_d|\boldsymbol{\theta})P(\boldsymbol{\theta})}{\int P(\boldsymbol{y}_d|\boldsymbol{\theta})P(\boldsymbol{\theta})d\boldsymbol{\theta}} \tag{4.10}$$

The integral in the denominator is a normalizing constant and is difficult to calculate for medium to high N_d. Fortunately, Markov Chain Monte Carlo (MCMC) solution methods [67–69, 90, 91] are based on the proportionality of this expression, $P(\boldsymbol{\theta}|\boldsymbol{y}_d) \propto L(\boldsymbol{\theta})P(\boldsymbol{\theta})$, and so the task only requires that the likelihood and prior distributions are specified. An MCMC method will be discussed briefly in the next section.

The likelihood definition depends on the model assumptions and sources of uncertainty involved. Similar to Figure 4.7, the model output $\boldsymbol{y}_d = \boldsymbol{g}_d(\boldsymbol{x})$ is compared to the measured data $\boldsymbol{y} = \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta})$, including sources of uncertainty as [13, 59, 60],

$$\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta}) + \boldsymbol{\epsilon}(\boldsymbol{x}) = \boldsymbol{g}_d(\boldsymbol{x}) + \boldsymbol{\epsilon}_d(\boldsymbol{x}) \tag{4.11}$$

where $\boldsymbol{\epsilon}(\boldsymbol{x})$ represents model error and $\boldsymbol{\epsilon}_d(\boldsymbol{x})$ is measurement error, which may also depend on $\boldsymbol{x}$. Both of these may be a function of the inputs. The model error may contain several sources of uncertainty, $\boldsymbol{\epsilon} = \boldsymbol{\epsilon}_\delta + \boldsymbol{\epsilon}_h + \boldsymbol{\epsilon}_s + \boldsymbol{\epsilon}_p$, where $\boldsymbol{\epsilon}_\delta = \delta(\boldsymbol{x}, \boldsymbol{\phi})$ is model discrepancy due to model form error, $\boldsymbol{\epsilon}_h$ represents discretization error, $\boldsymbol{\epsilon}_s$ is surrogate model error, and $\boldsymbol{\epsilon}_p$ is truncation error due to the use of dimension reduction techniques such as active subspace and PCA. Discretization errors were the focus of Chapter 3, and the extension of discrepancy functions [13, 61, 92, 93] to the PC space is the focus of Chapter 5. For the purpose of this chapter, $\boldsymbol{\epsilon}_s$ (error in PC-AS) and $\boldsymbol{\epsilon}_d$ are included during calibration. The other sources of error are discussed in more depth in Chapters 5 and 7.

Measurement errors are defined as discussed in Section 2.5, where $\boldsymbol{\epsilon}_d \sim \mathcal{N}(0, \Sigma_d)$, and Σ_d is a diagonal matrix with n entries. In this chapter, two types of output are used in the calibration. Thus, the first half of the diagonal entries of Σ_d are for temperature measurement (σ_T^2) and the other half are for the time constant uncertainty (σ_τ^2), where the latter are derived from Eq. 4.2. Note that in some Bayesian calibration applications where there is

sufficient measurement data (e.g., [59, 61]), σ_T and σ_τ would be calibrated along with $\boldsymbol{\theta}$. However, in the gas turbine heat transfer application, there will typically not be repeated testing from which the measurement may be estimated.

Next, the problem in Eq. 4.11 is recast into the PC-space to use the PC-AS surrogate model. First, the physics model $\boldsymbol{y}$ is replaced with the PC-AS surrogate $\boldsymbol{y} \approx \boldsymbol{g}_s(\boldsymbol{x}, \boldsymbol{\theta}) + \epsilon_s$, where a Gaussian approximation is made for the surrogate model errors $\epsilon_s \sim \mathcal{N}(\mathbf{0}, \sigma_s)$ from the residuals of the fit in Figure 4.5. Next, the measurements are transformed to PC space according to Eq. 4.5. Then, since PCA is a linear transformation, the covariance matrix of the measurements Σ_d transformed into PC space is also Gaussian [94] with $\Sigma_d^{pc} = U^T[\text{diag}(\boldsymbol{\sigma}_y)^{-2}\,\Sigma_d]U$. The factor $\text{diag}(\boldsymbol{\sigma}_y)^{-2}$ is included due to the use of standardized outputs (Section 4.3.1). Thus,

$$\boldsymbol{g}_s(x, \theta) + \epsilon_s = \boldsymbol{y}_d^{pc}(\boldsymbol{x}) + \boldsymbol{\epsilon}_d^{pc} \tag{4.12}$$

Based on these errors, a zero-mean[2] multivariate Gaussian likelihood function is assumed with covariance $\Sigma^{pc} = \sigma_s^2 I + \Sigma_d^{pc}$, where $I \in \mathbb{R}^{n \times n}$ is the identity matrix. Assuming there are N_d replicate measurements, and n_{pc} uncorrelated PCs, the overall likelihood is the product of multivariate Gaussian distributions

$$L^{pc}(\boldsymbol{\theta}) \sim \frac{1}{\sqrt{(2\pi)^{n_{pc}}|\Sigma^{pc}|}} \exp\left(-\frac{1}{2}\left(\boldsymbol{y}^{pc} - \boldsymbol{y}_d^{pc}\right)^T \left(\Sigma^{pc}\right)^{-1} \left(\boldsymbol{y}^{pc} - \boldsymbol{y}_d^{pc}\right)\right) \tag{4.13}$$

Finally, the joint probability distribution of the parameter priors $P(\boldsymbol{\theta})$ is defined as uniform distributions with limits based on Table 4.1.

4.4.2 Markov chain Monte Carlo sampling

Various methods are available for the solution of Bayes' Rule, depending on the complexity of the posterior distribution. In this chapter, we use a standard Metropolis-Hastings MCMC [28, 90, 91] sampling approach and compare the result to the maximum a posteriori (MAP) point estimate. The MAP is found by maximizing the right hand side of Eq. 4.10, which amounts to maximizing the proportion, $P(\boldsymbol{\theta}|\boldsymbol{y}_d) \propto L(\boldsymbol{\theta})P(\boldsymbol{\theta})$. On the other hand, the Metropolis-Hastings algorithm generates samples from the right hand side of Eq. 4.10, which is known as the target distribution in MCMC literature, $P_{mh}(\boldsymbol{\theta}) := L(\boldsymbol{\theta})P(\boldsymbol{\theta})$. Since the form of the target distribution is not known a priori, it is sampled indirectly using a proposal distribution Q_{mh} with each sample conditioned on the previous sample, i.e. $Q_{mh}(\theta_i|\theta_{i-1})$. This generates a Markov chain of "one-step memory" samples that are either accepted or rejected based on the acceptance ratio,

$$r_{mh} = \min\left(1, \frac{P_{mh}(\theta_i)}{P_{mh}(\theta_{i-1})} \cdot \frac{Q_{mh}(\theta_{i-1}|\theta_i)}{Q_{mh}(\theta_i|\theta_{i-1})}\right) = (1, \alpha) \tag{4.14}$$

[2]This assumes there is no discrepancy model to account for model error.

If $\alpha \geq 1$, a new sample is drawn; if $\alpha < 1$, α is compared with a new random number to determine whether to accept or reject a sample. With sufficient samples the chain converges to the posterior distribution. In this chapter, we select a Gaussian proposal $Q_{mh}(\theta_i|\theta_{i-1}) = \mathcal{N}(\theta_{i-1}, \sigma_\theta)$, which results in $\alpha = P_{mh}(\theta_{new})/P_{mh}(\theta)$ in Eq. 4.14 due to symmetry. This form of the Metropolis-Hastings algorithm is known as Random Walk Metropolis [90] and each sample is centered on the previous sample in the chain. The proposal distribution standard deviation is used to tune the algorithm to the specific problem, and in this analysis was specified as a fraction of the width of the parameter prior, $\sigma_\theta = c_\theta(\theta_{max} - \theta_{min})$. Tuning was based on targeting an acceptance rate of 30%, based on suggestions in the literature, e.g., 10-50% [91] or 25-40% [90].

4.4.3 Bayesian calibration results

Next, using the approach outlined above, Bayesian calibration is performed for four cases. These cases considered different measurement scenarios by varying the number of measurements replicates N_d and the level of measurement uncertainty, both the 'true' value σ_T^* and the prescribed uncertainty σ_T since the two may differ in practical problems. The four cases are listed in Table 4.2 as a triplet in the header row, where the measurement uncertainty is given in Kelvin (results in the table are discussed further below).

Table 4.2: Calibration posterior parameter results compared to θ_i^* (σ_T in units K)

Case (σ_T^*, σ_T, N_d)		**I** (0,1,10)	**II** (3,1,10)	**III** (3,3,10)	**IV** (3,3,2)
`fhcob2`	MAP	1.72	1.78	1.78	1.63
$\theta_1^* = 1.7$	μ	1.72	1.78	1.77	1.55
	σ/μ	0.02	0.02	0.06	0.12
`fhdiaf`	MAP	0.68	0.70	0.70	0.75
$\theta_2^* = 0.7$	μ	0.68	0.70	0.70	0.72
	σ/μ	0.03	0.03	0.06	0.11
`ftau`	MAP	1.94	2.13	2.13	2.31
$\theta_3^* = 2.0$	μ	1.94	2.13	2.11	2.09
	σ/μ	0.04	0.03	0.09	0.13
`fthp3`	MAP	1.09	1.10	1.10	1.10
$\theta_4^* = 1.1$	μ	1.09	1.10	1.10	1.10
	σ/μ	0.01	0.01	0.01	0.03
`fwdiaf`	MAP	1.57	1.46	1.46	1.28
$\theta_5^* = 1.5$	μ	1.57	1.46	1.46	1.25
	σ/μ	0.03	0.03	0.08	0.20

In Case I, synthetic measurement noise $\sigma_T^* = 0$ K, representing perfect data; thus, Case I represents a lower bound of posterior uncertainty among the four cases. Case II demonstrates the result when this measurement noise is non-zero. In both Case I and II, the assumed measurement noise in the likelihood σ_T is set low ($\sigma_T = 1$K) to avoid division by zero. Case III assumes an increase in the prescribed uncertainty σ_T, relative to Case II, so

that it is equal to the true measurement noise. Case IV, compared to Case III, demonstrates the impact of the low-replicate measurements that are typically available in a gas turbine heat transfer application by assuming $N_d = 2$.

The calibration results (posteriors) are summarized in Table 4.2 and the marginal posterior distributions are plotted in Figure 4.8 for each of these cases. The table includes the MAP, the marginal posterior mean (μ), and marginal posterior coefficient of variation (CoV$= \sigma/\mu$). For comparison, the 'true' solution θ_i^* is shown for each parameter (and is indicated in Figure 4.8 as a black triangle). The MAP for each case was calculated to use as the starting point of each MCMC solution and these results are also included in the table.

The results for Case I and II show that Case I has the least bias in the posterior mean value relative to the true solution. The measurement noise added in Case II results in a small bias relative to the true solution in some of the parameters, but otherwise the difference between these first two cases is small.

Next, the increase in σ_T for Case III doubles or triples the CoV for all parameters except `fthp3`, but does not change the mean value of the distributions. `fthp3` was shown to be the most sensitive parameter in output PC1 (Figure 4.6), so this parameter is least sensitive to the calibration assumptions in the four cases. Since σ_T is not calibrated in this application, it is clear from this result that this quantity must be carefully defined to obtain accurate posterior parameter results, or more measurements are required.

Finally, Case IV decreases the number of replicate measurements to $N_d = 2$. As expected, compared to Case III with $N_d = 10$, the uncertainty and the bias increase in the calibrated parameter posteriors due to reduced 'information' available for calibration. The importance of good prior definitions increases as the number of measurements decrease. Also, because gas turbine tests typically have a low N_d, it might be advantageous to optimize thermocouple locations [59] to improve posterior results.

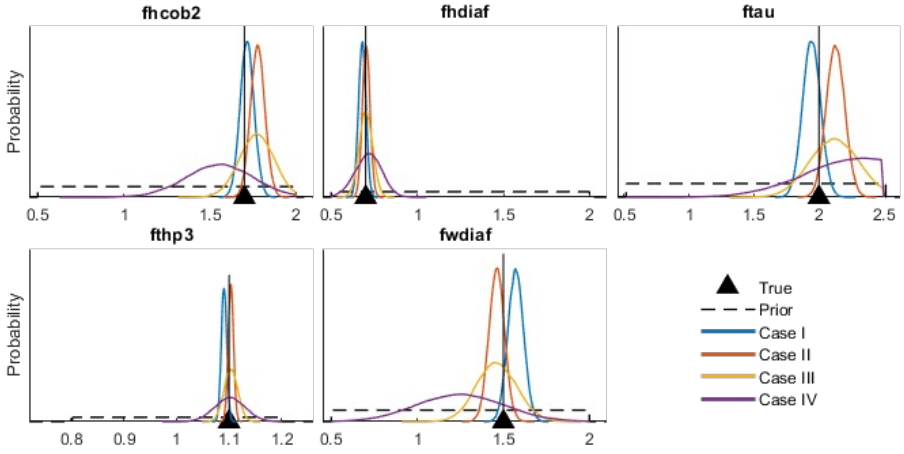


Figure 4.8: Parameter posterior marginal distributions from the four MCMC cases.

4.4.4 Forward propagation of the parameter posteriors

The updated parameter uncertainty represented by the MCMC samples for the four cases was propagated through the surrogate model to obtain the model output uncertainty. Marginal distributions are shown in Figure 4.9. The parameter priors are independent, but some correlation emerges during calibration which is included when running the samples through the surrogate model. For example, the two largest pairwise linear correlation coefficients calculated for Case IV were $\rho_{fthp3,fwdiaf} = -0.76$, followed by $\rho_{ftau,fhdiaf} = 0.54$ (where $\rho_{X,Y}$ is the correlation coefficient between two random variables X and Y). Fig. 8 shows four rows representing the four quantities extracted from the transient response (T^{mto}, T^{idle},τ^{mto}, and τ^{idle}) and four columns representing the four locations from the right hand side of Figure 4.3 (rim, bore, diaf, and arm). These locations are shown to demonstrate the response; the other locations are omitted due to space constraints. The 10 measurements are shown as black dots on the horizontal axis, with the two-measurement subset of Case IV shown in red.

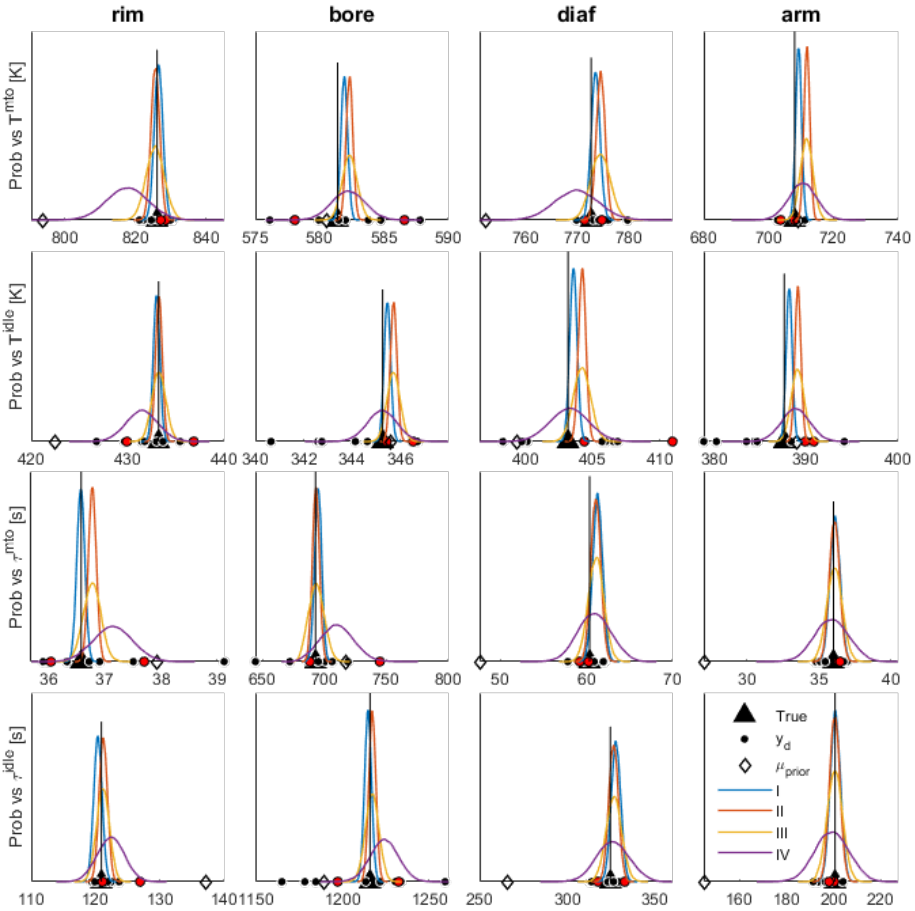


Figure 4.9: Model output posterior predictions showing the four extracted output quantities (rows) and the four selected thermocouple locations (columns). Red y_d are the $N_d = 2$ subset used for Case IV.

The outputs follow similar trends as observed in the parameters, comparing the different cases. The posterior variance of each output was significantly smaller than the prior in all cases. In Figure 4.9, the posterior mean is

observed to be much closer to the true value than the prior mean. Case I shows the least bias of the posterior mean relative to the true value, which is defined by running the model at θ^*. The noise in the measurements (Case II) and the increased variance (Case III) are clearly propagated into the results and show the same trends. Case IV has an increased bias since only two measurements are available (indicated by the two red dots).

4.4.5 Summary of the analysis process

Finally, to summarize the analysis process for each step shown in Figure 4.1, the application to the heat transfer model was as follows:

Step 1. Selected 5 parameters with uniform priors based on engineering experience (Table 4.1).

Step 2. The FE model was run 200 times based on LHS DOE. Four outputs (temperatures and time constants) were used to characterize the 123 time-step transient temperature response at each thermocouple location. These 4 outputs and the 12 thermocouple locations shown in Figure 4.3 result in 48 total outputs.

Step 3. The 48 outputs were transformed with PCA (Eq. 4.3) to an uncorrelated space, which also allowed dimension reduction. The 5 PCs with the largest eigenvalues were selected from the original 48 PCs, capturing 99.5% of the total variance.

Step 4. Active Subspace (Eq. 4.6) dimension reduction for the inputs was performed using the 5 individual PCs and the 5 model parameters, resulting in a set of 5 PC-AS univariate sample sets.

Step 5. The 5 sample sets were fit with quadratic polynomials to form the PC-AS surrogate (Figure 4.5).

Step 6. Synthetic measurements were derived as discussed in Section 4.2.

Step 7. The synthetic measurements were transformed into the model's PC-space for calibration.

Step 8. The likelihood was defined within the PC-space assuming a multivariate normal due to Eq. 4.12.

Step 9. The MCMC calculations were performed with a total of 535,000 accepted samples from the surrogate model, of which 50,000 were retained as posterior samples after 1:10 thinning and a burn-in of the first 35,000 samples. Convergence was considered successful with autocorrelation decay within 100 samples and an acceptance rate of 30%. Thus, in total, almost 2 million samples were generated, taking only 1-2 minutes on a laptop with an Intel Core i7 2.7GHz and 16GB of memory (Matlab® R2018b). The resulting posterior marginals are shown in Figure 4.8

Step 10. The 50,000 posterior samples were propagated through the surrogate model, to obtain the results shown in Figure 4.9.

4.5 Conclusion

This chapter has shown how the use of PCA and active subspace together enables a robust PC-AS surrogate modeling method by reducing the dimensionality of the problem. The PC-AS surrogate model enables efficient Bayesian calibration solutions by MCMC, which was demonstrated on a typical gas turbine engine disc heat

transfer model. In the context of the VVUQ process, the efficiency of calibration using surrogate models frees the engineer to spend more time improving the model physics and less time performing model calibration by manual parameter tuning. Because surrogate-enabled Bayesian approach accounts for data and model uncertainties, it provides a richer simulation output for developing robust engineering designs. Incorporating model discrepancy into the calibration step is considered next in Chapter 5. Including parameter uncertainty within the broader context of uncertainty aggregation is treated in Chapter 7, which develops an end-to-end VVUQ framework.

Future research pursuits motivated by this chapter include:

- The PC-AS surrogate model accuracy was high in the studied example ($R^2 = 0.95$ for $N > 20$ samples). However, in the next chapter, where additional PCs are necessary for the addition of a discrepancy function, it was found that PC-AS accuracy was not as good when $n_{pc} > 5$. Similar challenges were noted in [95]. These issues may stem from the simplification to a single active variable and linear gradient estimation. Further investigation is required for where PC-AS may be most applicable and whether the use of more advanced AS definitions (more active variables, improved gradient estimation) could increase the generality of the approach. Even if accuracy is not as high as other methods, it offers potential advantages due to the computationally inexpensive surrogate training relative to other methods such as GP and neural networks.

- The ability to handle more general transient maneuvers (perhaps through including all time points in the PCA [59]), through the use of dynamic active subspace methods [96], or more advanced time-dependent surrogate modeling would be advantageous.

- Optimization of sensor placement [59] could be used to identify better measurement locations for maximizing the information gained from sparse measurements.

- The use of sensitivity analysis to guide calibration parameter selection in the case of multivariate outputs is not straightforward [88, 89]. In this chapter, Sobol' indices were computed for PCs rather than physical outputs, which seems to offer advantages in down-selecting parameters for multivariate outputs[3]. Other works have taken advantage of this to some extent [95, 97], however, more general guidance is needed on how to combine information across PCs.

[3]One application is demonstrated in the next chapter in Section 5.4.2.

CHAPTER 5

Discrepancy modeling for model calibration with multivariate output[1]

5.1 Introduction

In Chapter 4, the use of Bayesian calibration was focused on model parameter estimation. This chapter focuses on the quantification of *model form error* (MFE, also known as 'missing physics') which was not addressed in Chapter 4. MFE may result in an observed *discrepancy* (also referred to as bias) between the model outputs and measured values. Besides simply correcting the physics modeling before calibrating the model [99] (which may not always be possible in complex models), researchers have proposed alternative approaches for correcting the model: (i) augment the model outputs with a discrepancy function [13, 100], (ii) modify the model parameters during calibration (augment the parameters) [101], (iii) modify the model parameter distribution after calibration by using quantified validation results [5, 41], or (iv) augment the governing differential equations with model form error terms [102, 103]. In all of these approaches, the objective is to either improve the model predictions or to obtain a more accurate estimate of physically meaningful model parameters [93, 102, 104–106].

The second and third approaches result in expansion of the model parameter uncertainty. This may not enable the model correction to address model bias. The fourth approach may be applied either within the governing equations [102] or indirectly through a recently published 'black box' approach [103]. In this work, we focus on the first method of correction (augmenting the output with a discrepancy term), which has received the most attention due to the original work by Kennedy and O'Hagan [13] (KOH). The KOH approach simultaneously calibrates the physics model parameters along with an additive *discrepancy function*. The physics model is replaced with a Gaussian process (GP) [55, 107] surrogate model to speed up the solution method and a second GP is used to represent the discrepancy function. Derivative works include engineering applications and variations on this idea considering both univariate [99, 100, 108] and multivariate model output [36, 59, 61, 109, 110].

Despite the popularity of the additive discrepancy approach, there are challenges in its application: (1) definition of the discrepancy function and its priors [92, 93], (2) non-identifiability of the physics model and discrepancy function parameters [92, 93, 110–112], (3) solution methodology (simultaneous vs. modular) [108, 113], (4) application to multivariate model outputs [36, 61, 109], and (5) calibration with the discrepancy function makes the final model dependent on the discrepancy function. Additive discrepancy functions are typically phenomenological since the missing physics is unknown, therefore making it difficult to generalize for predictions [101], particularly when the measurements are sparse.

The first and second challenges are closely related since the definition of the discrepancy function will impact

[1] Adapted from [98]

the identifiability of the parameters. Thus, non-identifiability is an important topic in this chapter. There are two forms of non-identifiability[2] [92, 115, 116]: *structural non-identifiability* is related to the model parameterization, independent from the measurements, in which the influence of two or more parameters on the model output may be indistinguishable [26, 115]; *practical non-identifiability* relates to the quality and quantity of the measurements [92, 116]. Sahu and Gelfand [117] note that structural non-identifiability does not prohibit Bayesian learning, but the resulting parameter posteriors are not separable. As a simple example, consider the calibration of the parameters $\boldsymbol{\theta}$ in $y = (\theta_1 + \theta_2)x$ for data $(\boldsymbol{x}_d, \boldsymbol{y}_d)$. The parameters are clearly non-identifiable. If Bayesian inference is used to find the posterior distribution $P(\boldsymbol{\theta}|\boldsymbol{y}_d)$, the marginal distributions will exhibit a lack of convergence, but their combination $\tilde{\theta} = \theta_1 + \theta_2$ will produce a meaningful result.

The third challenge relates to whether the discrepancy function parameters are calibrated at the same time as the physics model parameters or separately in a 'modular' fashion. The latter has been widely used to overcome the non-identifiability issue [106, 108, 110, 113, 114, 118–121]. However, in the modular approach the error between the prior prediction and measurements is used as the starting point for calibration of the discrepancy function. This may bias the discrepancy function parameters towards this 'prior' (which may or may not be desired). In this work, we favor a simultaneous approach where admitted by the problem. Techniques to improve the simultaneous solution are to define more informative priors [61, 93, 100, 122] (i.e., add more information to the problem [26, 117]), or propose alternative (potentially simpler [92]) non-GP surrogate models of the discrepancy function [36, 99].

Regarding the fourth challenge, multivariate applications of KOH have used PCA (Section 4.3.1) or similar methods to reduce the dimensionality of the surrogate models [59, 61, 109]. However, in these works Bayesian calibration was performed in the physical space (PCA is used only for reducing the number of surrogate models). Another approach is to perform model calibration within PC space by transforming both the model outputs (or surrogate model) and measurements into this space, which offers dimension reduction for the likelihood. This was explored in [36, 58, 123]. However, only [36] considered including a discrepancy function in PC space, which in that case was a simple zero-mean Gaussian.

To the authors' knowledge, there have been no previous investigations of model discrepancy functions in PC space. Therefore, in this work we adopt the approach of calibration in the PC space to explore alternative discrepancy functions. We then investigate the potential benefits of the formation and calibration of discrepancy functions within this subspace for addressing the dimensionality challenge and the previous three challenges. The fifth challenge stated above is not fully considered here. Instead, preliminary thoughts are given on how discrepancy functions may be useful for model diagnosis.

[2] A related issue is that the model output may exhibit a lack of sensitivity to certain parameters [114], meaning that they are non-identifiable, though the converse is not necessarily true [115]

The remaining sections are organized as follows. Section 5.2 provides a few mathematical and notational preliminaries. Section 5.3 discusses the methodology for calibration, selection, and evaluation of discrepancy functions in PC space. An illustrative model is used to highlight key challenges and considerations. Then, the methodology is applied to a gas turbine engine heat transfer model calibration problem in Section 5.4 and the chapter is concluded in Section 5.5.

5.2 Background

The Bayesian calibration problem is considered in the next section with additional parameters from a discrepancy function. Then, an overview of the non-identifiability assessment used in this chapter is presented in Section 5.2.2. As in Chapter 4, the calibration problem is again considered in PC space. PCA was introduced in Section 4.3.1. Recall that the PCs are formed by the eigenvalue decomposition of the model's covariance matrix Σ_y, resulting in eigenvectors U and eigenvalues κ_k in the diagonal of K.

5.2.1 Bayesian calibration with additive discrepancy

The calibration problem was described in Section 4.4.1 (Eq. 4.11) by equating the model to the measurements along with their respective sources of uncertainty. Measurements were defined as zero-mean Gaussian distributions with covariance $\Sigma_d = \text{diag}(\sigma_{d1}^2, \sigma_{d2}^2, \cdots, \sigma_{dn}^2)$, defined based on guidance from a measurement expert. Model errors $\boldsymbol{\epsilon}(\boldsymbol{x})$ include several sources, $\boldsymbol{\epsilon} = \boldsymbol{\epsilon}_\delta + \boldsymbol{\epsilon}_h + \boldsymbol{\epsilon}_s + \boldsymbol{\epsilon}_p$, that were previously defined. In this chapter, we focus on the first of these, which we define as a model *discrepancy function* $\boldsymbol{\epsilon}_\delta := \boldsymbol{\delta}(\boldsymbol{x}, \boldsymbol{\phi})$, which is due to model form error and has parameters $\boldsymbol{\phi} \in \mathbb{R}^q$. A GP surrogate model is used in Section 5.4 and so $\boldsymbol{\epsilon}_s$ is also included as a zero-mean Gaussian with covariance Σ_s. The remaining sources of model error are not included for the present purposes (there will be more discussion on this in Chapter 7).

When considering non-identifiability in subsequent analysis (e.g., Section 5.2.2), the combined (corrected) model is also written [92],

$$\boldsymbol{\mathcal{Y}} = [\mathcal{Y}_1, \ldots, \mathcal{Y}_n] := \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta}) + \boldsymbol{\delta}(\boldsymbol{x}, \boldsymbol{\phi}) \tag{5.1}$$

where will group the physics model and discrepancy function parameters into $\boldsymbol{\psi} := \{\boldsymbol{\theta}, \boldsymbol{\phi}\} \in \mathbb{R}^r$ where $r := p+q$. Using this notation, Bayes' theorem is $P(\boldsymbol{\psi}|\boldsymbol{y}_d) = P(\boldsymbol{y}_d|\boldsymbol{\psi})P(\boldsymbol{\psi})/P(\boldsymbol{y}_d)$. For convenience later in this chapter, the condensed notation for prior $\boldsymbol{\psi}' := P(\boldsymbol{\psi})$, posterior $\boldsymbol{\psi}'' := P(\boldsymbol{\psi}|\boldsymbol{y}_d)$, and likelihood $L(\boldsymbol{\psi}) := P(\boldsymbol{y}_d|\boldsymbol{\psi})$ are also used. Due to the nature of measurement and surrogate modeling errors, we again make the common choice [13, 100, 105, 109] of a Gaussian likelihood, $L(\boldsymbol{\psi}) \sim \mathcal{N}(\boldsymbol{0}, \Sigma)$, which in multivariate form with discrepancy is

expressed

$$L(\boldsymbol{\psi}) \sim \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} \exp\left(-\frac{1}{2} \left(\boldsymbol{y} - \boldsymbol{y}_d + \bar{\boldsymbol{\delta}} \right)^T \Sigma^{-1} \left(\boldsymbol{y} - \boldsymbol{y}_d + \bar{\boldsymbol{\delta}} \right) \right) \tag{5.2}$$

where the covariance matrix is $\Sigma := \Sigma_d + \Sigma_s$. It is assumed that each of these error sources are independent resulting in a diagonal Σ. The mean of the discrepancy function is denoted $\bar{\boldsymbol{\delta}} := E\left[\boldsymbol{\delta}(\boldsymbol{x}, \boldsymbol{\phi})\right]$. In Section 5.3.1, this Bayesian calibration problem is transformed into PC space.

5.2.2 Assessing non-identifiability

There are several ways to assess non-identifiability in inverse problems. For example, *profile likelihood* (Raue et al and Kreutz [116, 124]) computes the MLE of all parameters as a function of one parameter ψ_i fixed at several values through its support, the profile likelihood is computed: $\text{argmax}_{\psi_{i \neq j}} L(\boldsymbol{\psi}_{\sim i})$. This is repeated for each parameter and non-identifiability is assessed from the flatness of the generated profiles. Alternatively, Arendt et al [112] focuses on the posterior parameter marginal variance reduction relative to the prior, $\text{cov}[P(\psi_i|\boldsymbol{y}_d)]/\text{cov}[P(\psi_i)]$. A third option, which we use in this chapter similar to [92], is based on inspection of the rank of the Jacobian (or sensitivity matrix [26]). We therefore refer to this as the *rank-based method*. The Jacobian arises in the inverse solution of linear or linearized systems, such as the inverse solution by Gauss linearization [26]. For the multivariate combined outputs[3] from Eq. 5.1, the linearization around $\hat{\boldsymbol{\psi}}$ is,

$$\begin{bmatrix} \mathcal{Y}_1 \\ \vdots \\ \mathcal{Y}_n \end{bmatrix}_{\boldsymbol{\psi}} \approx \begin{bmatrix} \mathcal{Y}_1 \\ \vdots \\ \mathcal{Y}_n \end{bmatrix}_{\hat{\boldsymbol{\psi}}} + \underbrace{\begin{bmatrix} \nabla \mathcal{Y}_1 |_{\hat{\boldsymbol{\psi}}}^T \\ \vdots \\ \nabla \mathcal{Y}_n |_{\hat{\boldsymbol{\psi}}}^T \end{bmatrix}}_{\boldsymbol{X}} (\boldsymbol{\psi} - \hat{\boldsymbol{\psi}}) \tag{5.3}$$

$$\boldsymbol{\mathcal{Y}} \approx \hat{\boldsymbol{\mathcal{Y}}} + \boldsymbol{X}(\boldsymbol{\psi} - \hat{\boldsymbol{\psi}}) \tag{5.4}$$

where the Jacobian $\boldsymbol{X}$ is formed by partial derivatives $\nabla \mathcal{Y}_k$ at $\hat{\boldsymbol{\psi}}$ with $\nabla = [\partial/\partial\psi_1, \cdots, \partial/\partial\psi_r]^T$, and $\hat{\boldsymbol{\mathcal{Y}}}$ represents $\boldsymbol{\mathcal{Y}}$ evaluated at $\hat{\boldsymbol{\psi}}$. Equating the model approximation to the measurements, $\boldsymbol{\mathcal{Y}}_d \approx \hat{\boldsymbol{\mathcal{Y}}} + \boldsymbol{X}(\boldsymbol{\psi} - \hat{\boldsymbol{\psi}})$, the parameters are solved in the least squares sense according to the normal equations,

$$\boldsymbol{\psi} - \hat{\boldsymbol{\psi}} = (\boldsymbol{X}^T \boldsymbol{X})^{-1} \boldsymbol{X}^T (\boldsymbol{\mathcal{Y}}_d - \hat{\boldsymbol{\mathcal{Y}}}) \tag{5.5}$$

for which $(\boldsymbol{X}^T \boldsymbol{X})^{-1}$ clearly must exist, requiring also that $\boldsymbol{X}$ has columns that are linearly independent. Thus, non-identifiability is present when there are linearly dependent columns, or equivalently, $|\boldsymbol{X}^T \boldsymbol{X}| = 0$ or $\text{rank}(\boldsymbol{X}) < r$. In practice, the discriminant may not need to be exactly zero for there to be non-identifiability. As a further

[3] Note that for the identifiability assessment in Eq. 5.3, etc., $\boldsymbol{\mathcal{Y}}$ has been transposed implicitly to simplify notation. The dimensions are clear from the context in the remainder of the chapter.

step to the check above, Ling and Mahadevan [92] suggested an algorithm to iteratively remove parameters from the problem to discover which contribute to non-identifiability.

The rank-based approach relies on the invertibility of $\boldsymbol{X}^T\boldsymbol{X}$, stronger prior information improves identifiability [26, 125]. This has been observed in the application of KOH [61, 93, 100, 126]. One way to see this is by comparing OLS (Eq. 5.5) or MLE (replace $\boldsymbol{X}^T\boldsymbol{X}$ with $\boldsymbol{X}^T\Sigma_d^{-1}\boldsymbol{X}$ in Eq. 5.5) to the MAP estimate [26],

$$\boldsymbol{\psi} - \hat{\boldsymbol{\psi}} = (\boldsymbol{X}^T\Sigma_d^{-1}\boldsymbol{X} + \boldsymbol{V}^{-1})^{-1}\boldsymbol{X}^T(\boldsymbol{\mathcal{Y}}_d - \hat{\boldsymbol{\mathcal{Y}}}) \tag{5.6}$$

where $\boldsymbol{V}$ is the covariance of the priors. Thus, $\boldsymbol{V}^{-1}$, which is diagonal due to the assumption of independent priors, adds a regularizing 'ridge' to the diagonal of $\boldsymbol{X}^T\Sigma_d^{-1}\boldsymbol{X} = \frac{1}{\sigma_d^2}\boldsymbol{X}^T\boldsymbol{X}$. If the priors are normally distributed as $\mathcal{N}(0, \tau^2)$, this is equivalent to ridge regression (or Tikhonov regularization) with $\boldsymbol{V}^{-1} = \lambda\boldsymbol{I}$, where $\lambda = \sigma_d^2/\tau^2$ [125]. In this case, the solution may exist when $n < r$ [26]. Although this is also true for Bayesian inference, in the application of Section 5.4, we assume weakly informative (uniform) priors and therefore only use $(\boldsymbol{X}^T\boldsymbol{X})^{-1}$ as the basis of the number of feasible parameters $\boldsymbol{\psi} = \{\boldsymbol{\theta}, \boldsymbol{\phi}\}$ that can be calibrated.

5.3 Methodology

In this section, the calibration problem and non-identifiability assessments from Section 5.2 are transformed into PC space in Sections 5.3.1 and 5.3.2.1. Then, appropriate discrepancy functions are selected in Section 5.3.2.3 to address the challenges raised in Section 5.1 regarding the KOH approach with additive discrepancy. Evaluation criteria for the selection discrepancy functions are established in Section 5.3.3. An illustrative model is introduced to highlight several of the challenges and opportunities before applying the approach to a heat transfer model in Section 5.4.

5.3.1 Bayesian calibration problem in PC space

In order to consider model discrepancy in PC space during calibration, the problem is cast into PC space as shown in Chapter 4.4.1. Assuming that $\boldsymbol{y} = \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta})$ and $\boldsymbol{y}_d = \boldsymbol{g}_d(\boldsymbol{x})$ are first standardized[4], the model outputs are transformed as $\boldsymbol{y}^{pc} = \boldsymbol{y}U$, measurements as $\boldsymbol{y}_d^{pc} = \boldsymbol{y}_dU$, and model discrepancy $\boldsymbol{\delta}^{pc}(\boldsymbol{\phi}) = \boldsymbol{\delta}(\boldsymbol{x}, \boldsymbol{\phi})U$. Defining measurement uncertainty $\boldsymbol{\epsilon}_d^{pc} \sim \mathcal{N}(0, \Sigma_d^{pc})$, Eq. 4.11 in PC space becomes

$$\boldsymbol{y}^{pc}(\boldsymbol{\theta}) + \boldsymbol{\delta}^{pc}(\boldsymbol{\phi}) = \boldsymbol{y}_d^{pc} + \boldsymbol{\epsilon}_d^{pc} \tag{5.7}$$

PCA is a linear transformation, thus covariance is transformed $\Sigma_d^{pc} = U^T[\text{diag}(\boldsymbol{\sigma}_y)^{-2}\ \Sigma_d]U$, where the factor $\text{diag}(\boldsymbol{\sigma}_y)^{-2}$ is for standardization, and the likelihood from Eq. 5.2 remains multivariate normal [94] in the PC

[4]To simplify notation, the subscript 0 denoting standardization is omitted.

space

$$L^{pc}(\boldsymbol{\theta}, \boldsymbol{\phi}) \sim \frac{1}{\sqrt{(2\pi)^{n_{pc}}|\Sigma^{pc}|}} \exp\left(-\frac{1}{2}\left(\boldsymbol{y}^{pc} - \boldsymbol{y}_d^{pc} + \bar{\boldsymbol{\delta}}^{pc}\right)^T (\Sigma^{pc})^{-1}\left(\boldsymbol{y}^{pc} - \boldsymbol{y}_d^{pc} + \bar{\boldsymbol{\delta}}^{pc}\right)\right) \tag{5.8}$$

with a covariance matrix $\Sigma^{pc} = \Sigma_d^{pc} + \Sigma_s^{pc}$. The discrepancy mean value is denoted $\bar{\boldsymbol{\delta}}^{pc} = \mathrm{E}\left[\boldsymbol{\delta}(\boldsymbol{\phi})\right]$. If there are i.i.d. replicate measurements $\boldsymbol{y}_{dj}^{pc}$ for $j = 1 \ldots N_d$, then Eq. 5.8 becomes a product over these replicates.

5.3.2 Selecting discrepancy functions in PC space

The selection of discrepancy functions within the PC space is guided by limitations on the total number of parameters $r = p + q$ (identifiability condition) and limitations within the PC space on the functional form.

5.3.2.1 Non-identifiability assessment in PC space

First, the rank-based identifiability approach from Section 5.2.2 is transformed into the PC space. The combined model output $\mathcal{Y}_k$ in PC space is $\mathcal{Y}_k^{pc} := \mathcal{Y}_k U$. Based on linearity of the derivative and PCA transformation [127], the Jacobian in PC space at $\hat{\boldsymbol{\psi}}$ is therefore defined,

$$\boldsymbol{X}^{pc} = U^T \begin{bmatrix} \nabla\mathcal{Y}_1\big|^T \\ \vdots \\ \nabla\mathcal{Y}_n\big|^T \end{bmatrix}_{\hat{\psi}} = \left[\begin{array}{ccc|ccc} \frac{\partial\mathcal{Y}_1}{\partial\theta_1} & \cdots & \frac{\partial\mathcal{Y}_1}{\partial\theta_q} & \frac{\partial\mathcal{Y}_1}{\partial\phi_1} & \cdots & \frac{\partial\mathcal{Y}_1}{\partial\phi_p} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial\mathcal{Y}_{n_{pc}}}{\partial\theta_1} & \cdots & \frac{\partial\mathcal{Y}_{n_{pc}}}{\partial\theta_p} & \frac{\partial\mathcal{Y}_{n_{pc}}}{\partial\phi_1} & \cdots & \frac{\partial\mathcal{Y}_{n_{pc}}}{\partial\phi_q} \end{array}\right]_{\hat{\psi}} \in \mathbb{R}^{n_{pc}\times r} \tag{5.9}$$

As previously shown Section 4.3.1, the application of PCA allows for truncation to $n_{pc} \ll n$ outputs, based on the decreasing magnitude of the eigenvalues in K. Therefore, the number of rows in $\boldsymbol{X}^{pc}$ is significantly less than $\boldsymbol{X}$. In other words, the amount of 'information' contained in $\boldsymbol{y}$ is actually less than its n dimensions due to correlation between the outputs [70] (this is the basic idea behind data compression). To ensure an identifiable solution for the combined set of model parameters $\boldsymbol{\psi} \in \mathbb{R}^r$ requires that $r \leq n_{pc}$ and the r columns of $\boldsymbol{X}^{pc}$ must also be linearly independent, i.e., $\mathrm{rank}(\boldsymbol{X}^{pc}) = r$. The result, under the assumption of weakly informative priors, is that n_{pc} restricts the number of 'free' parameters that are available for defining the discrepancy function in the PC space. This is demonstrated with the illustrative model in Section 5.3.4.4.

5.3.2.2 Functional dependence in PC space

In physical space, it may be possible to determine an expected trend function or to define a GP that is dependent on $\boldsymbol{x}$, potentially improving the discrepancy fit and reducing the number of parameters required[5]. However, since the PCs are linear combinations of the multivariate outputs through the multiplication with U, the dependence on inputs $\boldsymbol{x}$ is implicit (note that $\boldsymbol{x}$ is omitted from Eq. 5.7 since it encodes different output locations and/or

[5]Developing a trend function may be challenging when the measurements are sparse, as is the case in the examples later in this section (e.g., Figure 5.1) and the heat transfer model in Section 5.4.

quantities). To show this more concretely, consider the transformation of a discrepancy function $\boldsymbol{\delta}(\boldsymbol{x}, \boldsymbol{\phi})$ when $n = 4$. The resulting discrepancy vector δ_i^{pc} is no longer dependent on $\boldsymbol{x}$,

$$\boldsymbol{\delta}^{pc} = \boldsymbol{\delta}(\boldsymbol{x}, \boldsymbol{\phi})U = \begin{bmatrix} \delta_1(x_1, \phi_1) \\ \delta_2(x_2, \phi_2) \\ \delta_3(x_3, \phi_3) \\ \delta_4(x_4, \phi_4) \end{bmatrix}^T \begin{bmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ u_{21} & u_{22} & u_{23} & u_{24} \\ u_{31} & u_{32} & u_{33} & u_{34} \\ u_{41} & u_{42} & u_{43} & u_{44} \end{bmatrix} = \begin{bmatrix} \delta_1^{pc}(\phi_1) \\ \delta_2^{pc}(\phi_2) \\ \delta_3^{pc}(\phi_3) \\ \delta_4^{pc}(\phi_4) \end{bmatrix}^T \tag{5.10}$$

The discrepancy functions are therefore limited to dependence on $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$.

5.3.2.3 Selected discrepancy functions

Based on the two limitations above, the options for defining the form of the discrepancy function are reduced. On the other hand, these limitations may be seen as advantages since they simplify the definition of the discrepancy functions and allow for a simultaneous calibration process. In view of these considerations, we limit our investigation to the following functional forms for the subsequent analysis (a simplified model in Section 5.3.4.4 and the heat transfer model in Section 5.4),

$$\boldsymbol{\delta}_i^{pc}(\boldsymbol{\theta}, \boldsymbol{\phi}) = \phi_{i0} + \sum_{j \in q^*} \phi_{ij}\theta_j \tag{5.11}$$

where $j \leq p$ (recall that p is the number of physics model parameters). In order to choose among possible terms to limit the number of total parameters, the importance of each θ_j on the model output will first be assessed using sensitivity analysis. This assessment results in several possible discrepancy functions which are then evaluated against criteria established in the next section.

5.3.3 Evaluating the selected discrepancy functions

To assess the selected discrepancy function, three convergence criteria are specified in this chapter. We refer to the 'true' solution as $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta}^*) + \boldsymbol{h}(\boldsymbol{x})$, where $\boldsymbol{\theta}^*$ are a set of 'true' (known) model parameters and $\boldsymbol{h}(\boldsymbol{x})$ is a known functional form representing missing physics in $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta})$ (the physics model). As was noted in Section 5.1, the goal of including a discrepancy function is either to obtain an unbiased estimate of the parameters or to improve model predictions. The first criteria below is minimized along with parameter bias (relative to the true solution), while the third criteria addresses improved (corrected) predictions *at the measurement locations*[6] The second criteria considers the impact on the underlying (uncorrected) physics model predictions.

[6]Prediction errors here are only evaluated against the calibration dataset. However, since synthetic measurements are used, the calibration and validation data would be from the same population.

Criteria 1. The posterior parameters should converge toward the true parameters,
$\mathrm{E}\big[\boldsymbol{\theta}''\big] \approx \boldsymbol{\theta}^*$

Criteria 2. The uncorrected posterior predictions should converge toward the model output evaluated at $\boldsymbol{\theta}^*$,
$\mathrm{E}\big[\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta}'')\big] \approx \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta}^*)$

Criteria 3. The discrepancy-corrected posterior predictions should converge toward the true solution,
$\mathrm{E}\big[\boldsymbol{\mathcal{Y}}(\boldsymbol{\psi}'')\big] = \mathrm{E}\big[\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta}'') + \boldsymbol{\delta}(\boldsymbol{x}, \boldsymbol{\phi}'')\big] \approx \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta}^*) + \boldsymbol{h}(\boldsymbol{x})$

Of course, these criteria are only appropriate for the evaluation in this chapter, in which the true solution is known. Relative to an unknown true solution (i.e., the real world), these criteria are not applicable but one may instead (i) test candidate models for parameter non-identifiability, (ii) check uncorrected model posterior variance, (iii) compare predictions using the posteriors and corrected model to a 'hold out' validation dataset if available. However, the intent of this chapter is to determine whether the PC space approach works for a known situation. These criteria will be illustrated with simple model in Section 5.3.4 and the heat transfer model in Section 5.4, along with the following numerical metrics for the latter.

Performance metrics are employed to evaluate the above criteria when assessing discrepancy formulations for the heat transfer model (Section 5.4). Since the model calibration is performed in PC space, a natural metric for the evaluation of Criteria 2 and 3 (related to the model outputs) is the PCA area metric [128]. This metric is based on (i) converting the model outputs and measurements into PC space, (ii) applying the univariate area metric [129] to each PC separately (Eq. 5.12), then (iii) combining these individual PC area metrics by weighting them by their corresponding eigenvalues (Eq. 5.13). This metric is summarized in the following equations

$$A_k = \int_{-\infty}^{\infty} \big|F_k(\xi) - F_{dk}(\xi)\big| d\xi \tag{5.12}$$

$$A^{pc} = \frac{1}{\mathrm{Tr}(K)} \sum_{k=1}^{n_{pc}} \kappa_k A_k \tag{5.13}$$

where $F_k(\xi)$ and $F_{dk}(\xi)$ are the cumulative distributions functions (CDF) of the model outputs and measurements in PC space ($\boldsymbol{y}^{pc}$ and $\boldsymbol{y}_d^{pc}$), ξ is a dummy variable for integration, and κ_k is the eigenvalue of the k^{th} PC.

Assessing Criteria 1 requires a different metric due to the lack of correlation in the prior parameters (independent uniform marginals are prescribed). In this case, the parameter posteriors $\pi(\boldsymbol{\theta})$ are compared to a reference solution $\pi_o(\boldsymbol{\theta})$ using the Kullback-Leibler divergence [130]. In this chapter, the multivariate normal approximation [131] is used for computational convenience,

$$\begin{aligned} D_{KL}(\pi||\pi_o) &= \sum_{x \in \mathcal{X}} \pi(x) \log \frac{\pi_o(x)}{\pi(x)} \\ &\approx \frac{1}{2}\left(\mathrm{Tr}(\Sigma_{\pi_o}^{-1}\Sigma_\pi) + (\mu_{\pi_o} - \mu_\pi)^T \Sigma_{\pi_o}^{-1}(\mu_{\pi_o} - \mu_\pi) - p + \log \frac{\mathrm{Det}\ \Sigma_{\pi_o}}{\mathrm{Det}\ \Sigma_\pi} \right) \end{aligned} \tag{5.14}$$

The reference solution $\pi_o(\boldsymbol{\theta})$ is generated by calibrating to synthetic data generated using the true parameter set $\boldsymbol{\theta}^*$ without prescribed model discrepancy. This represents the best solution possible for $P(\boldsymbol{\theta}|\boldsymbol{y}_d)$ given the assumptions in the defined likelihood (Eq. 5.8).

5.3.4 An illustrative model

Before application to the heat transfer model in Section 5.4, several of the above points are illustrated using a simple model. The model is defined with missing physics and first calibrated without including a discrepancy function. Next, a discrepancy function with too many parameters is added without consideration of non-identifiability. Then, an identifiable discrepancy function is calibrated. Finally, the model dataset is transformed into PC space before carrying out the calibration as proposed above.

Consider a 'multivariate' output $\boldsymbol{y} := \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta}) \in \mathbb{R}^4$ that has parameters $\boldsymbol{\theta} = [\theta_0, \theta_1, \theta_2]$. The model is to be calibrated using the data collected at four sensor locations S1, S2, S3, and S4 along the x coordinate,

$$\boldsymbol{y} := \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta}) = \theta_0 + \theta_1 \boldsymbol{x} + \theta_2 \boldsymbol{x}^3 \tag{5.15}$$

The parameter priors $\boldsymbol{\theta}' = P(\boldsymbol{\theta})$ are specified as uniform distributions centered on $\bar{\boldsymbol{\theta}} = [1, 2, 5]^T$ with bounds $\theta_0 \in [0, 2]$, $\theta_1 \in [1, 3]$, and $\theta_2 \in [2, 8]$, respectively. Corresponding measurements at the four sensor locations are generated synthetically by first defining a true function with $\boldsymbol{\theta}^* = [1, 2, 5]^T$ and $\boldsymbol{h}(\boldsymbol{x}) := -sin(6\boldsymbol{x})$,

$$\boldsymbol{y}_{true} := \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta}^*) + \boldsymbol{h}(\boldsymbol{x}) = 1 + 2\boldsymbol{x} + 5\boldsymbol{x}^3 - sin(6\boldsymbol{x}) \tag{5.16}$$

Then, $N_d = 30$ random samples are generated with zero-mean Gaussian noise, which has covariance Σ^* where the diagonal elements are defined $(\sigma^*)^2 = 0.25^2$,

$$\boldsymbol{y}_d := \boldsymbol{g}_d(\boldsymbol{x}) = \boldsymbol{y}_{true} + \mathcal{N}(0, \Sigma^*) \tag{5.17}$$

The model and measurements for the case calibrated without a discrepancy function are plotted in Figure 5.1a, which shares a common layout with the next several figures. The $N = 100$ samples of the prior predictive distribution $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta}')$ are shown as light gray lines (and points) and the $N_d = 30$ measurement samples are shown as black points at the sensor locations. The uncorrected (blue) and corrected (red) true outputs used to generate the data are indicated for comparison to the calibrated solution, which is discussed next.

The models were calibrated to the measurements using a standard Metropolis random walk MCMC algorithm [132]. A total of 5×10^4 posterior samples were retained after 2.5×10^4 samples of 'burn-in' and a 1 in 5 thinning to reduce the autocorrelation between samples. The marginal sample traces from the MCMC computations of the

first case are shown in Figure 5.1 column (c). The posterior samples represent $\boldsymbol{\theta}'' = P(\boldsymbol{\theta}|\boldsymbol{y}_d)$ and are summarized in (d) as histograms and bivariate scatter plots. Propagating the posterior samples through the model produces the posterior prediction $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta}'')$, shown in panel (a) and at the four sensor locations in row (b).

5.3.4.1 No discrepancy function (original space)

The model calibrated with no discrepancy function is shown in Figure 5.1. A few observations can be made regarding this model fit to motivate the need for a model discrepancy function. The most obvious is that the posteriors represented in the histograms of Figure 5.1 (d) do not agree with the true parameters, or $\boldsymbol{\theta}'' \neq \boldsymbol{\theta}^*$. This also results in the posterior prediction disagreeing with the true solution, $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta}'') \neq \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta}^*)$. As observed in panels (a,b), the posterior predictions (blue) do not align with the blue dashed reference lines. Another observation is the uncertainty of the posteriors do not reflect the model form error (MFE), resulting in posteriors that are 'strong and wrong' (which violates Criteria 1, Section 5.3.3). Furthermore, with limited measurements along the x-axis, there is insufficient information for discerning the functional form of $\boldsymbol{\delta}(\boldsymbol{x}, \boldsymbol{\phi})$. As we will see in Section 5.4, this issue is even more difficult in the engineering application in which $\boldsymbol{x}$ represents non-trivial 2D geometry locations and different engine operating conditions. Discerning the form of the missing physics from the model errors directly may also be misleading, depending on whether the prior model under-fits (as in the present case) or over-fits the measurements (if the model were over-fit, prior residuals would be near zero).

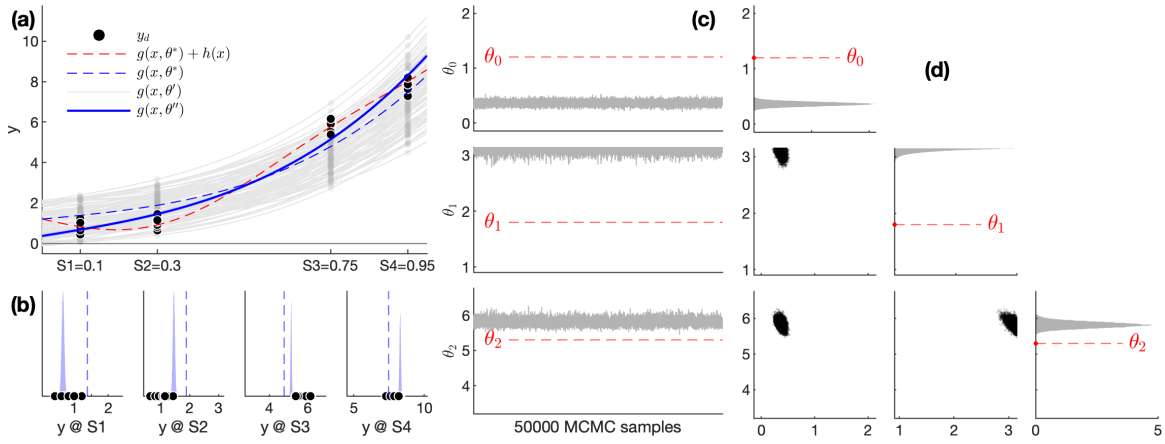


Figure 5.1: Calibration without the discrepancy term: (a) Models shown for the prior prediction $g(\boldsymbol{x}, \boldsymbol{\theta}')$, posterior prediction $g(\boldsymbol{x}, \boldsymbol{\theta}'')$, true solution $g(\boldsymbol{x}, \boldsymbol{\theta}^*) + h(\boldsymbol{x})$, and the model evaluated at the true parameters $g(\boldsymbol{x}, \boldsymbol{\theta}^*)$. (b) Measurements and model predictions histograms at each sensor (the horizontal axis limits are set to the range of the prior prediction). Part (c) and (d) summarize the MCMC posterior parameter samples: (c) Posterior sample traces. (d) This lower triangular grid of plots includes histograms of the posterior samples on the diagonal and bivariate scatter plots between pairs of parameters (axis limits are set to the range of the priors). The red dashed lines indicate the true parameters values.

5.3.4.2 Non-identifiable discrepancy function (original space)

Next, an additive discrepancy function is defined by including a simple unknown constant to each output location,

$$\boldsymbol{\delta}(\boldsymbol{x},\boldsymbol{\phi}) := \begin{cases} \phi_0, & x = x_1 \\ \phi_1, & x = x_2 \\ \phi_2, & x = x_3 \\ \phi_3, & x = x_4 \end{cases} \tag{5.18}$$

The result of calibrating with this discrepancy function is shown in Figure 5.2. The mean results of $\boldsymbol{g}(\boldsymbol{x},\boldsymbol{\theta}'')$ and $\boldsymbol{g}(\boldsymbol{x},\boldsymbol{\phi}'') + \boldsymbol{\delta}(\boldsymbol{\phi}'')$ improve considerably, but at the cost of greater prediction uncertainty, which is represented by the blue shading (99^{th} percentiles) around $\boldsymbol{g}(\boldsymbol{x},\boldsymbol{\theta}'')$ in (a). Increased uncertainty is due to non-identifiability. In particular, θ_1 and θ_2 are ill-defined (see histograms of $\boldsymbol{\theta}''$ in panel (d)). Thus, introducing too much flexibility results in non-identifiability since there are now seven parameters to calibrate with the same amount of data. This is evident from the posterior marginal distributions as pointed out by Arendt [112].

The rank-based approach (Section 5.2.2) was also performed to assess the discrepancy function in Eq. 5.18 for non-identifiability. In this case, the terms $\frac{\partial \delta_k}{\partial \phi_j}|_{k=j} = 1$, and are otherwise zero. The resulting Jacobian $\boldsymbol{X}$ is

$$\boldsymbol{X} = \begin{bmatrix} \nabla \mathcal{Y}_1 \big|^T \\ \vdots \\ \nabla \mathcal{Y}_n \big|^T \end{bmatrix}_{\hat{\psi}} = \left[\begin{array}{ccc|ccc} \frac{\partial g_1}{\partial \theta_0} & \frac{\partial g_1}{\partial \theta_1} & \frac{\partial g_1}{\partial \theta_2} & \frac{\partial \delta_1}{\partial \phi_0} & \cdots & \frac{\partial \delta_1}{\partial \phi_3} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial g_4}{\partial \theta_0} & \frac{\partial g_4}{\partial \theta_1} & \frac{\partial g_4}{\partial \theta_2} & \frac{\partial \delta_4}{\partial \phi_0} & \cdots & \frac{\partial \delta_4}{\partial \phi_3} \end{array}\right]_{\hat{\psi}} = \left[\begin{array}{ccc|cccc} 1 & x_1 & x_1^3 & 1 & 0 & 0 & 0 \\ 1 & x_2 & x_2^3 & 0 & 1 & 0 & 0 \\ 1 & x_3 & x_3^3 & 0 & 0 & 1 & 0 \\ 1 & x_4 & x_4^3 & 0 & 0 & 0 & 1 \end{array}\right] \tag{5.19}$$

where $g_k = g(x_k,\boldsymbol{\theta})$ represents a single output. The vertical rule in the matrix indicates two blocks that separate $\mathcal{Y}_k$ into dependence on $\boldsymbol{\theta}$ and dependence on $\boldsymbol{\phi}$. For this discrepancy function, the result is $\mathrm{rank}(\boldsymbol{X}) = 4$ which is less than the total number of parameters r. This model and dataset can only afford a single additional discrepancy parameter.

Another observation in Figure 5.2 is that the corrected posterior solution agrees very well with the true solution, i.e., the red prediction histograms are centered on the red dashed true solution line in panel (b). However, there is large uncertainty in the uncorrected model, $\boldsymbol{g}(\boldsymbol{x},\boldsymbol{\theta}'')$. The added flexibility of the discrepancy function enables the corrected solution to produce good results, but if the intent is to rely on $\boldsymbol{g}(\boldsymbol{x},\boldsymbol{\theta}'')$ for making future predictions, the result has relatively high uncertainty in this case.

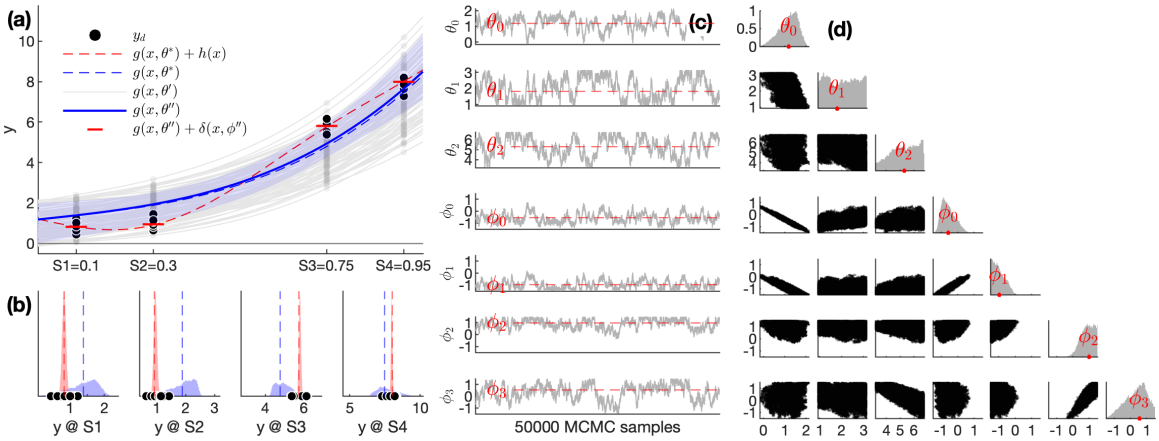


Figure 5.2: Calibration with a partially identifiable discrepancy constant added to each output

5.3.4.3 Identifiable discrepancy function (original space)

In order to reduce the number of parameters and therefore reduce the non-identifiability observed in Figure 5.2, information may be added by means of a functional form (dependent on $\boldsymbol{x}$). The sparse measurements do not clearly indicate an appropriate functional form for the trend in discrepancy shown in Figure 5.1 or 5.2. However, to illustrate how an informative discrepancy model performs with fewer parameters, we choose $\boldsymbol{\delta}(\boldsymbol{x}, \boldsymbol{\theta}) = \phi_0 sin(6\boldsymbol{x})$, which agrees perfectly with the true $\boldsymbol{h}(\boldsymbol{x})$ up to a multiplicative constant. This is identifiable since $\frac{\partial \delta_k}{\partial \phi} = sin(6x_k)$ results in a column $[0.56, 0.97, -0.98, -0.55]^T$ which would replace the right four columns of $\boldsymbol{X}$ in Eq. 5.19, resulting in full $\text{rank}(\boldsymbol{X}) = 4 = r$. The result of the calibration is shown in Figure 5.3. The results are favorable since: $\boldsymbol{\theta}'' \approx \boldsymbol{\theta}^*$ in panels (c,d), the posterior predictions $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta}'') \approx \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta}^*)$, and $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta}'') + \boldsymbol{\delta}(\boldsymbol{x}, \boldsymbol{\phi}'') \approx \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta}^*) + \boldsymbol{h}(\boldsymbol{x})$ in panels (a,b). That is, the criteria from Section 5.3.3 are met.

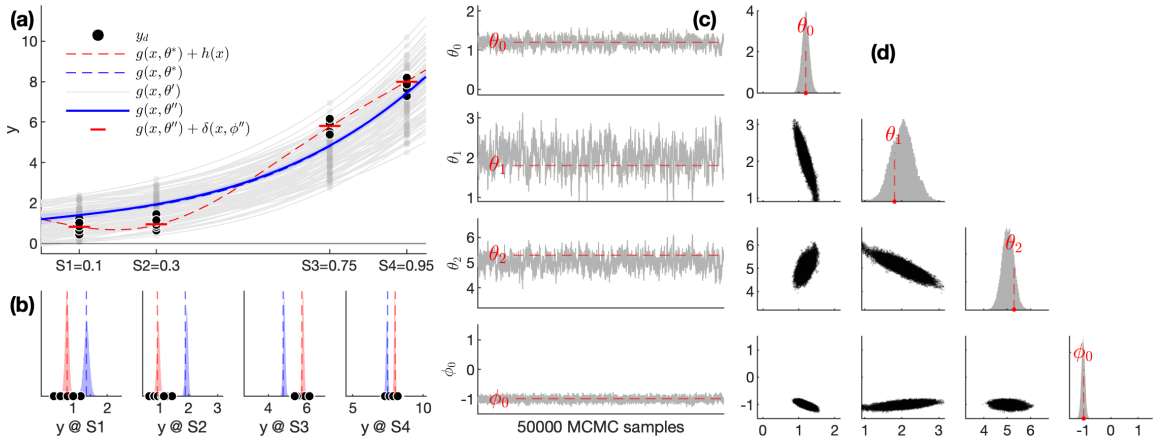


Figure 5.3: Calibration with an identifiable discrepancy function equivalent to the true MFE (up to unknown parameter ϕ_0), $\delta(x, \phi) = \phi_0 sin(6x)$

5.3.4.4 Discrepancy function in PC space (non-identifiable)

A final calibration is performed by transforming the model outputs (Eq. 5.15) at the four sensor locations into PC space, as described in Section 5.3.1. The resulting eigenvalues contribute 85.2%, 14.2%, %0.6, and 0% to the total variance explained. This suggests that two PCs should be retained. If a constant discrepancy function is included, $\boldsymbol{\delta}^{pc}(\boldsymbol{\phi}) = \boldsymbol{\phi}$, the problem definition is non-identifiable since $\text{rank}(\boldsymbol{X}^{pc}) = 3 < r$ and the system is therefore under-determined. The results also indicate this with substantial prediction uncertainty, as shown in Figure 5.4. For example, see the strong correlation between parameters θ_0 and ϕ_0 in the bivariate plot of Part (d). This example also shows that calibration in PC space will only work if there is sufficient model complexity to require enough PCs such that additional discrepancy parameters may be calibrated. For simpler models like this, PCA is not advantageous.

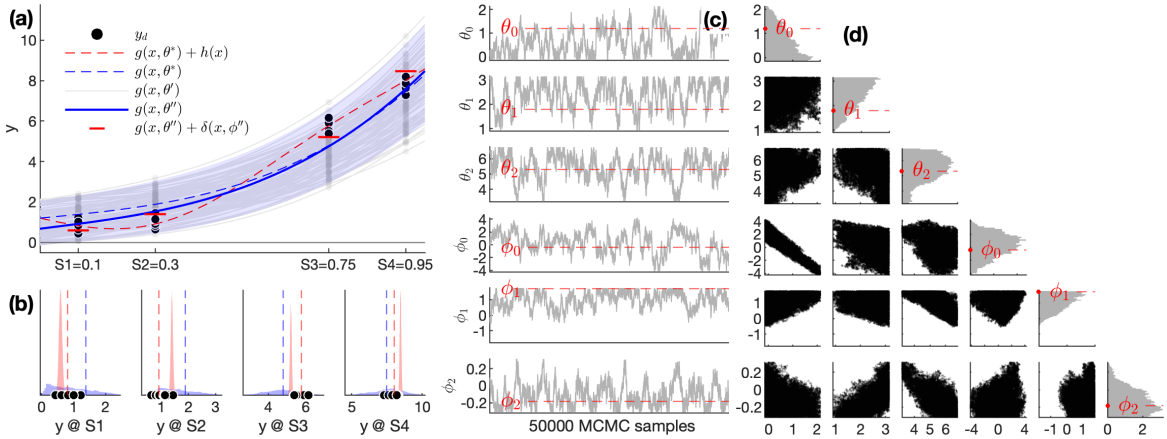


Figure 5.4: Calibration with a non-identifiable constant PC space discrepancy function, $\boldsymbol{\delta}^{pc}(\boldsymbol{\phi}) = \boldsymbol{\phi}$

5.3.5 Use of the calibrated discrepancy

The introduction discussed five challenges in the application of the additive discrepancy approach to dealing with MFE. The fifth was related to how the results with additive discrepancy should be used after calibration. It was also noted that the purpose of calibration with an additive discrepancy function is to either improve the model predictions or to obtain a more accurate estimate of physically meaningful model parameters [93, 102, 104–106]. When accurate parameters is the goal, Figure 5.3 shows that including an accurate discrepancy function improves the parameter estimate. Conversely, substantial error in the parameter estimate is incurred if discrepancy is neglected (Figure 5.1). However, if correction of the model outputs is the goal, there are two concerns. First, the uncorrected model will still have significant error as shown in the histograms of Figure 5.2b. The uncorrected model and its uncertainty lie mostly outside the distribution of measurements. Second, the corrected model (including the discrepancy function) will only be applicable within the domain of the 'training data', that is, at measurement locations and for the inputs at which the discrepancy function was trained.

Many practical calibration applications will be based on limited measured locations, and therefore using the discrepancy function to correct other outputs that were not included during calibration (interpolation or extrapolation) is unlikely to be trustworthy. It is similarly risky to interpolate or extrapolate the model to other input settings that were not used during calibration, due to limited measurements. Therefore, when this methodology is applied to a physics model with limited measurements, we propose using the discrepancy function as a *diagnostic tool* rather than for the purpose of correcting the model predictions. We suggest two practical uses:

1. *Parameter shifts*: Calibrate the model with and without the discrepancy function. Look for shifts in the parameter posteriors between the two results. Parameters that change substantially may reveal related model physics that require further improvement.

2. *Discrepancy magnitude*: Calibrate the model with a discrepancy function and compare the magnitude $|\delta_k|$ (or $|\delta_i^{pc}|$) to a positive threshold value δ_{thr}. There are two possibilities: (i) $|\delta_k| < \delta_{thr}$ indicating that the calibrated parameters $\boldsymbol{\theta}$ may be used directly while ignoring the discrepancy; (ii) $|\delta_k| > \delta_{thr}$ indicating that the physics model must be refined to reduce the MFE.

In the application that follows, we briefly demonstrate these diagnostic approaches.

5.4 Application to the heat transfer model

The discrepancy functions defined in this chapter are now applied to the 2D axisymmetric heat transfer model presented in Chapter 2. The general set up of the calibration problem is largely the same as Chapter 4, with a few modifications as described earlier in this chapter. The model is shown in Figure 5.5 with $n_\ell = 9$ numbered measurement locations (the compressor locations are omitted in this chapter) and indicated boundary condition calibration factors $\boldsymbol{\theta}$, which are also listed in Table 4.1. Additional discrepancy function parameters are described in Section 5.4.2, which discusses the selection of appropriate discrepancy functions.

The multivariate model outputs are processed as described in Chapter 2.4 and 4.2.1, the transient multivariate model outputs are dimension-reduced by first extracting 4 features (T^{mto}, T^{idle}, τ^{mto}, τ^{idle}) at each thermocouple position and then performing PCA on the dataset $\boldsymbol{y} \in \mathbb{R}^{N \times n}$. The first 5 of these PCs contained 50.0%, 31.3%, 10.8%, 5.0%, and 2.6% of the variance, respectively, and the remaining 4 total 0.3%.

Based on the findings in Section 5.3.2.1, the non-identifiability assessment of the Jacobian suggests we aim to retain as many PCs as possible[7]. It was found that the PC-AS models used in Chapter 4 do not capture the small PC values as well as PC-GP (GP fit to PCs) surrogate models, especially for these smaller PCs (this identified an

[7]This may seem counter-intuitive to the use of dimension reduction. However, the goal of dimension reduction is to make surrogate modeling more feasible. Furthermore, as noted in Section 5.3.2.1, adding more outputs would not necessarily increase the amount of information (and thus PCs) due to the correlation between model outputs.

area for further research on the PC-AS). Furthermore, the GPs failed to fit after the first 9 PCs, and so $n_{pc} = 9$ PCs were retained.

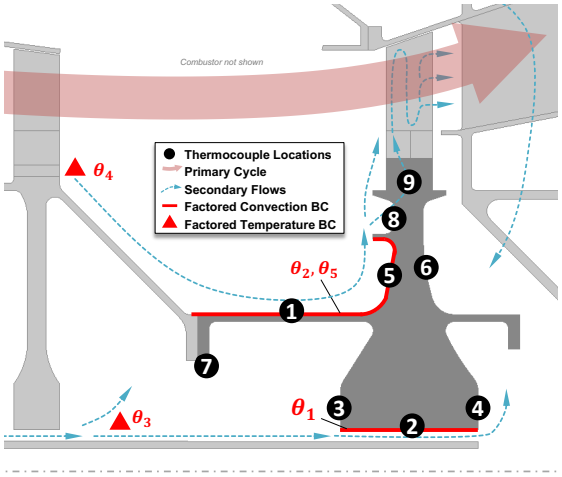


Figure 5.5: Turbine disc heat transfer model, parameters, and measurement locations[8]

5.4.1 Synthetic measurements and model form error

Measurements were generated synthetically for this chapter similar to the approach in Chapter 4. The FE model was run with the 'true' parameter values in Table 4.1. Then, Gaussian noise with $\sigma^* = 3$ Kelvin was added to the temperatures to produce $N_d = 2$ replicate samples for each output. Similar noise is included for the time constant outputs by propagation of σ^* as described in Section 4.2.2.

For this study, two measurement datasets were created as a way[9] to represent a case without model form error (MFE) and a case where MFE is present due to additional physics $\boldsymbol{h}(\boldsymbol{x})$. To simplify the notation in this section $\boldsymbol{C}_1$ and $\boldsymbol{C}_2$ are used (instead of $\boldsymbol{y}_d$) for the two cases, respectively. These cases are defined similar to Eq. 5.16 and 5.17,

$$\boldsymbol{C}_1 := \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta}^*) + \mathcal{N}(\boldsymbol{0}, \Sigma^*) \qquad \text{(model has no MFE)} \tag{5.20}$$

$$\boldsymbol{C}_2 := \boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta}^*) + \boldsymbol{h}(\boldsymbol{x}) + \mathcal{N}(\boldsymbol{0}, \Sigma^*) \qquad \text{(model with MFE)} \tag{5.21}$$

In this example, $\boldsymbol{h}(\boldsymbol{x})$ represents an added radiation heat transfer boundary condition[10] that was applied along the red line indicated for θ_2 and θ_5 in Figure 5.5. For the comparison of selected model discrepancy functions, $\boldsymbol{C}_2$ represents the primary dataset. Calibration of the original model against $\boldsymbol{C}_2$ represents the case with MFE. Calibration against $\boldsymbol{C}_1$ serves as a datum solution since there is no MFE. Figure 5.6 shows $\boldsymbol{C}_1$ (left) and the

[8] The location number order in Chapters 5 and 6 are the same, but inadvertently different than the ordering in Chapters 3 and 7.
[9] To simplify the analysis, two measurement datasets were created rather than creating two different models.
[10] This location also includes boundary conditions for convection and heat generation from air friction.

difference $\boldsymbol{C}_2 - \boldsymbol{C}_1$ (right) at the MTO stabilized condition.

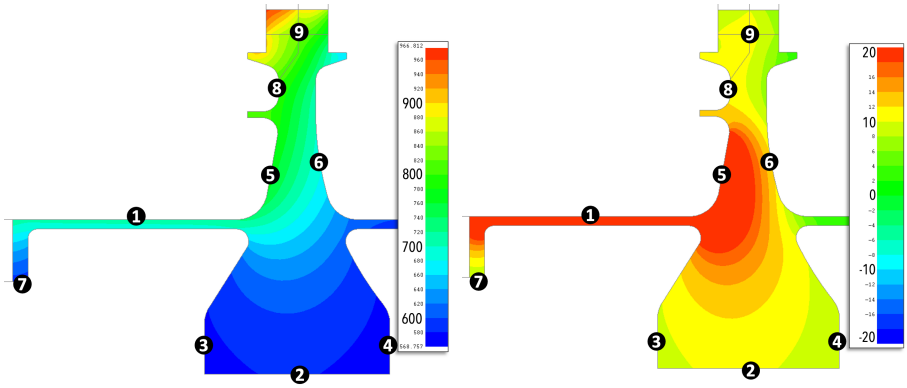


Figure 5.6: Temperature contours for $\boldsymbol{C}_1$ (left) and 'true' temperature discrepancy $\boldsymbol{h}(\boldsymbol{x}) = \boldsymbol{C}_2 - \boldsymbol{C}_1$ (right) in Kelvin at MTO. The difference contours represent the 'true' value of $\boldsymbol{h}(\boldsymbol{x})$ over the entire domain, which is only observed as $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta}) - \boldsymbol{C}_2 \approx \boldsymbol{h}(\boldsymbol{x})$ at the 9 measurement locations.

5.4.2 Selecting discrepancy functions

In this section, several discrepancy function cases are defined for further analysis based on the findings in Section 5.3.2, including non-identifiability and functional dependence. The discrepancy functions are down-selected from the general linear forms defined in Eq. 5.11 based on the sensitivity of model parameters $\boldsymbol{\theta}$ (determined by sensitivity analysis) and an assessment of non-identifiability. Then, a number of cases are defined in Table 5.1 to compare the performance for a different number of linear terms, the number of PCs to which discrepancy is applied, and the assumptions regarding priors. The table also shows the results of computing the metrics defined in Section 5.3.3.

It was found in the previous section that up to $n_{pc} = 9$ PCs could be retained given the information in the dataset. Since there are $p = 5$ model parameters, the number of discrepancy function parameters is limited to $q \leq 4$, in order to ensure that the solution is identifiable (recall $r = p + q$). Since we are limited to $q \leq 4$ terms, only terms in Eq. 5.11 that include important parameters are retained, where importance is determined through global sensitivity analysis. The same efficient first-order Sobol' index method [87] used in Chapter 4 is also adopted in this chapter to make use of the available 200 DOE samples, however, the indices are recomputed since the number of outputs was reduced from $n_\ell = 12$ to $n_\ell = 9$. Separate sets of Sobol' indices were computed for each PC[11] (since the sensitivity analysis method is w.r.t. a single dependent variable) and are shown in Figure 5.7. The results were computed 5000 times based on bootstrapping[125] of the 200 original samples in order to estimate sampling error (due to the limited number of samples). Thus, the bar with error whiskers in Figure 5.7 represents the bootstrap mean and one standard deviation, respectively.

[11]Note that, by performing sensitivity analysis using the PCs rather than the 36 physical outputs, a more concise summary of important parameters is given. For example, since parameters θ_2 and θ_4 are important factors in the first two PCs, and these two PCs explain 81.3% of the model variance, these two factors are likely the most important parameters overall, followed by θ_1, etc.

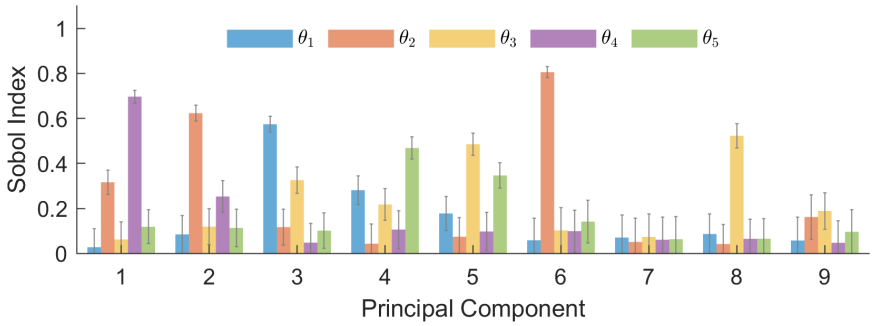


Figure 5.7: First order Sobol' indices for each of the first 9 out of 36 PCs including $\pm$1std bootstrap error bars.

The results in Figure 5.7 indicate that there are only one or two significant model parameters per PC for most of the PCs. This information was used to reduce the number of terms from Eq. 5.11 to include only these significant model parameters,

$$\boldsymbol{\delta}^{pc}(\boldsymbol{\theta}) = \begin{cases} \phi_{10} + \phi_{12}\theta_2 + \phi_{14}\theta_4 & \text{PC1} \\ \phi_{20} + \phi_{22}\theta_2 + \phi_{24}\theta_4 & \text{PC2} \\ \phi_{30} + \phi_{31}\theta_1 + \phi_{33}\theta_3 & \text{PC3} \\ \phi_{40} + \phi_{41}\theta_1 + \phi_{45}\theta_5 & \text{PC4} \\ 0 & \text{otherwise} \end{cases} \tag{5.22}$$

Finally, subsets of the terms in Eq. 5.22 were selected as cases for further investigation, where each case has $q \leq 4$ parameters ϕ_{ij}. The selected subsets/cases are tabulated in Table 5.1 further below. For example, one case includes the first constant term ϕ_{i0} on the first four PCs (Case 24), another includes the first two terms $\phi_{i0} + \phi_{ij}\theta_j$ applied to the first 2 PCs (Case 29 and 37), etc. The remaining cases are described below Eq. 5.23.

To assess the discrepancy functions for identifiability based on Eq. 5.9, the partial derivatives for the model parameters were estimated using finite differences around the nominal values, $\hat{\boldsymbol{\psi}} = [1, 1, 1, 1, 1, 0, 0, 0, 0]^T$. An example of how non-identifiability is assessed is shown in Eq. 5.23 alongside the partial derivatives of the linear discrepancy models $\partial \mathcal{Y}_i / \partial \phi_{i0} = 1$ and $\partial \mathcal{Y}_i / \partial \phi_{ij} = \theta_j$. This Jacobian represents Case 24 with ϕ_{i0} on the first four PCs (equivalently, the same Jacobian would result if $\phi_{ij}\theta_j$ were used for the first four PCs, since $\theta_j = 1$), and is a full-rank and invertible matrix, i.e., $\text{rank}(\boldsymbol{X}^{pc}) = 9$, suggesting it is structurally identifiable. The other cases can be evaluated similarly.

$$\boldsymbol{X}^{pc} = \begin{bmatrix} \nabla \mathcal{Y}_1\big|^T \\ \vdots \\ \nabla \mathcal{Y}_9\big|^T \end{bmatrix}_{\hat{\psi}} = \left[\begin{array}{rrrrr|cccc} -0.349 & 3.092 & -0.988 & 39.965 & 1.704 & 1 & 0 & 0 & 0 \\ -1.142 & -7.507 & 2.975 & 10.357 & 1.613 & 0 & 1 & 0 & 0 \\ -2.817 & 1.993 & 3.453 & -2.174 & -1.304 & 0 & 0 & 1 & 0 \\ -2.962 & 0.416 & -1.536 & 0.373 & 1.702 & 0 & 0 & 0 & 1 \\ -0.891 & -0.438 & -0.789 & 1.727 & -2.999 & 0 & 0 & 0 & 0 \\ -0.053 & -0.145 & -0.120 & -0.825 & -0.143 & 0 & 0 & 0 & 0 \\ -0.483 & -0.432 & 0.242 & -1.785 & -0.408 & 0 & 0 & 0 & 0 \\ 0.096 & 0.132 & 0.011 & -0.483 & 0.062 & 0 & 0 & 0 & 0 \\ -0.052 & 0.200 & 0.042 & -0.012 & -0.045 & 0 & 0 & 0 & 0 \end{array}\right]_{\hat{\psi}} \tag{5.23}$$

Having used the identifiability assessment to narrow the range of possible discrepancy function cases, we now describe the cases selected for further analysis in Table 5.1. The cases are grouped into four 'Sets' in order to highlight features of the results. Set 1 represents calibration without a discrepancy function. Set 2 cases are calibrated with a discrepancy function but there is no MFE. Set 3 cases are calibrated with a simple additive scalar discrepancy. Set 4 considers calibration with different function types (including GP) and the parameter priors. The 'Case' numbers are for reference in the discussion below. The 'MFE?' column indicates whether MFE is present (recall from Section 5.4.1 that $\boldsymbol{C}_1 =$ no MFE or $\boldsymbol{C}_2 =$ MFE). Under the heading 'Discrepancy Function', $\bar{\phi}'_{ij}$ represents the choice of prior, δ_i^{pc} is the discrepancy functional form for PC$_i$ using terms from Eq. 5.22, and n_δ is the number of PCs to which a discrepancy function was applied. The priors are Gaussian with two choices of mean: zero and non-zero. The non-zero mean was determined using the following steps:

1. Compute the prior model's discrepancy $\bar{y}_d - y'_i$, where y'_i is the prior prediction for output i.
2. Transform this discrepancy to PC space.
3. Fit linear regression models to this discrepancy as a function of $\boldsymbol{\theta}$ to obtain coefficients of selected ϕ_{ij}.
4. Use the resulting fit's mean coefficients $E\big[\phi'_{ij}\big] = \bar{\phi}'_{ij}$ to be the prior mean of the discrepancy parameters.
5. Use the standard error (SE) of the fit to set the prior standard deviation as $b\times$SE ($b = 10$ was found to work well in most cases, but $b = 1$ was used for the more flexible discrepancy functions in Case 29 and 37).

For cases that have prior $\bar{\phi}'_{ij} = 0$ (zero mean), the prior standard deviation was computed as in step (5). To illustrate how a more flexible function compares to the simple linear functions, Case 39 is based on fitting a GP model to $\bar{\delta}_i^{pc}$ as a function of $\boldsymbol{\theta}$. The remaining columns present evaluation metric results (from Section 5.3.3) which are discussed in the next section.

Table 5.1: Selected discrepancy function cases and posterior metrics

			Discrepancy Function			Posterior Metrics		
	Case[12]	MFE?	$\bar{\phi}'_{ij}$	δ_i^{pc}	n_δ	$\boldsymbol{\theta}''$	$\boldsymbol{y}^{pc}(\boldsymbol{\theta}'')$	$\boldsymbol{\mathcal{Y}}^{pc}(\boldsymbol{\psi}'')$
Set 1	1	N	–	–	0	0	9	–
	21	Y	–	–	0	69	59	–
Set 2	3	N	0	ϕ_{i0}	2	51	46	77
	11	N	$\neq 0$	ϕ_{i0}	2	556	108	76
Set 3	22	Y	0	ϕ_{i0}	1	49	39	13
	23	Y	0	ϕ_{i0}	2	91	56	14
	24	Y	0	ϕ_{i0}	4	245	73	14
Set 4	23	Y	0	ϕ_{i0}	2	91	56	14
	26	Y	0	$\phi_{ij}\theta_j$	2	60	44	13
	29	Y	0	$\phi_{i0} + \phi_{ij}\theta_j$	2	20	30	15
	31	Y	$\neq 0$	ϕ_{i0}	2	687	128	13
	34	Y	$\neq 0$	$\phi_{ij}\theta_j$	2	1017	137	13
	37	Y	$\neq 0$	$\phi_{i0} + \phi_{ij}\theta_j$	2	1394	163	12
	39	Y	$\neq 0$	GP	2	1165	175	3

5.4.3 Evaluating the discrepancy functions

The performance results are shown in Table 5.1 under columns 'Posterior Metrics'. The first column is the result of computing the Kullback-Leibler divergence (Eq. 5.14) between the posteriors $\boldsymbol{\theta}''$ of the given case and the reference case (Case 1) posteriors, as described in Section 5.3.3. The second column is the result of the PCA area metric (Eq. 5.12) of the comparison between the uncorrected posterior prediction $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta}'')$ and the true solution $\boldsymbol{y}^{pc}(\boldsymbol{\theta}^*)$. The third column is the result of computing the PCA area metric between the corrected posterior prediction $\boldsymbol{\mathcal{Y}}^{pc}(\boldsymbol{\psi}'') := \boldsymbol{y}^{pc}(\boldsymbol{\theta}'') + \boldsymbol{\delta}^{pc}(\boldsymbol{\phi}'')$ and the true solution $\boldsymbol{y}^{pc}(\boldsymbol{\theta}^*) + \boldsymbol{h}^{pc}$, where $\boldsymbol{h}^{pc}$ is the PC transform of $\boldsymbol{h}(\boldsymbol{x})$.

The third column shows the largest result for Set 2 where there is no MFE but a discrepancy function is used anyway; in this case, the discrepancy term only adds noise to the results. For the remaining cases (Set 3 and 4), there is low uncertainty in the corrected prediction due to the flexibility of the discrepancy function, even when there is large uncertainty in the marginal distributions. Thus, the corrected output does not provide the primary feedback on the calibration performance of the discrepancy function. Therefore, in this analysis we focus on the *uncorrected* posterior prediction results in the column labeled $\boldsymbol{y}^{pc}(\boldsymbol{\theta}'')$ (but still wish to see low values in the third column).

The next several paragraphs discuss the results of the first two columns. Figures 5.8, 5.9, and 5.10 (see further below) plot the marginal posterior distributions for the parameters, PCs 1 to 4, and four physical outputs, respectively. Only Set 1 and Set 4 are plotted due to space constraints.

[12] Several cases were considered but only a few are discussed in this chapter, which is why the case numbers are non-consecutive

Set 1 Cases: calibration without a discrepancy function

Case 1 serves as an ideal reference (no MFE) solution only for the purpose of evaluating the parameter performance using the Kullback-Leibler. In contrast, Case 21 includes MFE like most of the other cases, but no discrepancy function. So Case 21 serves as the reference solution against which to compare discrepancy function performance. As Figure 5.8 shows, the posterior parameters of Case 21 are biased (see especially θ_4), and the posterior predictions in Figures 5.9 and 5.10 are also biased away from the true output (red dashed line). In other words, ignoring discrepancy may result in both biased parameters, and it does not guarantee an unbiased model output.

Set 2 Cases: calibration with a discrepancy function (no MFE)

These cases compare the two options for discrepancy function priors when the constant discrepancy function ϕ_{i0} is used but there is not actually any MFE. The expected result is that their metrics should resemble Case 1 (ideal) and that the calibrated discrepancy term is near zero, if it has discovered the fact that there is no MFE. Case 3 has a zero-mean prior and Case 11 has a non-zero mean prior centered on the initial model discrepancy, $\bar{\boldsymbol{y}}_d - \boldsymbol{y}'$. However, both cases showed a worse result relative to Case 1, although the metric values for Case 3 are less than Case 21 and significantly less than Case 11. These cases suggest that the non-zero prior performs poorly. This same trend is observed in Set 4.

Set 3 Cases: calibration with an additive scalar discrepancy

This set considers the behavior of the additive scalar discrepancy function when it is applied to $n_\delta = 1$, 2, or 4 PCs. Interestingly, the best results occur when this discrepancy function is only applied to 1 PC. This result may be due to increased practical identifiability due to fewer parameters.

Set 4 Cases: calibration with different function types and priors

Lastly, Set 4 compares combinations of function type and discrepancy parameter priors when applied to 2 PCs ($n_\delta = 2$). The trend from Set 2 is again observed in the metrics of Table 5.1, that is, the non-zero priors for Cases 31, 34, 37, and 39 result in poor agreement with the true solution. This can also be observed as bias in the posterior parameters in Figure 5.8 and larger posterior prediction uncertainty in Figures 5.9 and 5.10.

Case 29 is the best performing overall, based on the metrics and plotted distributions (Cases 22 and 26 follow closely in second and third place). The fact that there is still bias for θ_4 in Figure 5.8 may be due to the use of the smaller prior standard deviation ($b \times SE$, with $b = 1$). However, the posterior prediction plots (Figures 5.9 and 5.10) show that this reduced prior also reduces the posterior variance.

Case 22 is the second best performing. It includes just a single constant discrepancy term on PC1. This good performance may be related to the nature of the true MFE $\boldsymbol{h}(\boldsymbol{x})$, i.e., radiation. The radiation boundary condition

affects the overall steady-state temperature levels. The eigenvector of PC1 have the largest terms for the steady-state temperature output. Therefore, if the MFE were instead something that impacted the transient output (such as missing physics related to convection around the disc), a constant adder may be insufficient as a discrepancy model.

Case 39 is one of the two worst performing. This case uses the GP discrepancy function. Poor performance is likely due to fitting the GP around the prior prediction discrepancy. Interestingly, despite large uncertainty in the discrepancy function parameters, the added flexibility allows the metric for $\mathcal{Y}^{pc}(\psi'')$ (the discrepancy-corrected model output) to be extremely low. This illustrates the potential issues of relying on the discrepancy-corrected solution when making predictions.

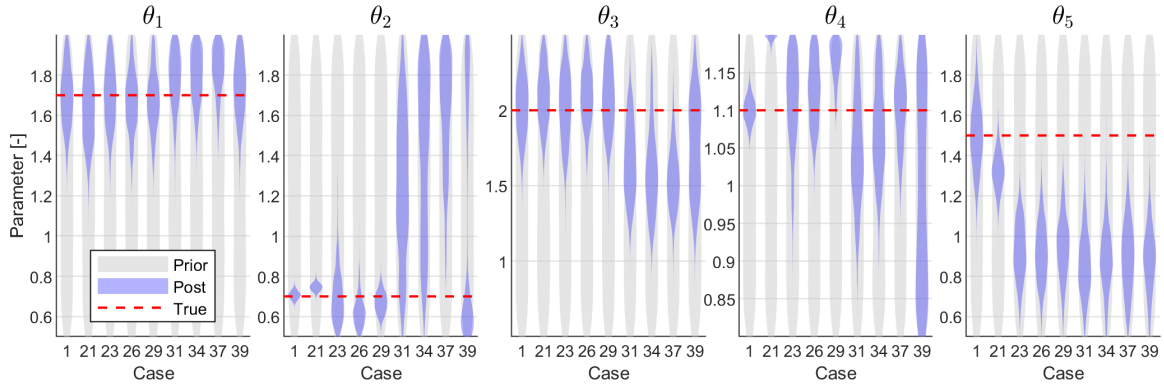


Figure 5.8: Parameter prior and posterior marginal distributions compared to the true value $\boldsymbol{\theta}^*$ (Set 1 & 4)

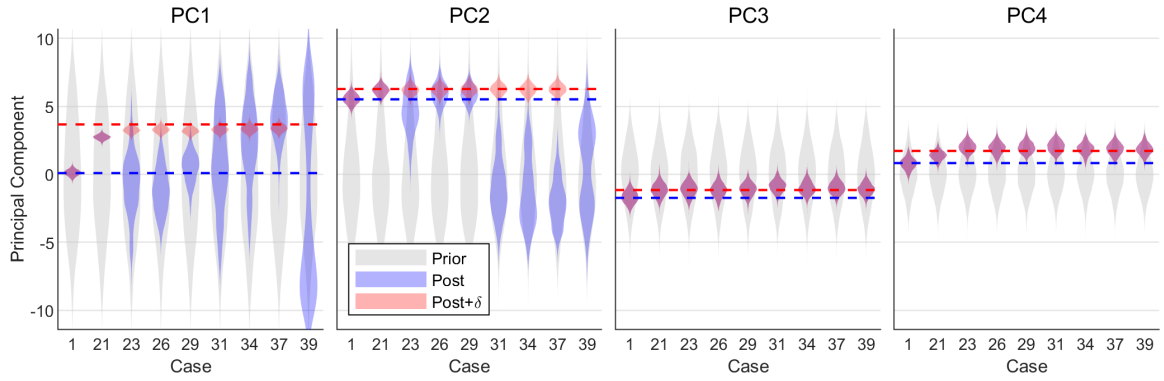


Figure 5.9: Model outputs marginal distributions in PC space for prior predictions, posterior predictions, and corrected posterior predictions. The two posterior solutions are assessed against the true values $\boldsymbol{y}^{pc}(\boldsymbol{\theta}^*)$ (blue dashed line) and $\boldsymbol{y}^{pc}(\boldsymbol{\theta}^*) + \boldsymbol{h}^{pc}$ (red dashed line), respectively. Note that a discrepancy function is applied to PC1 and PC2 only, so the remaining PCs remain unchanged as expected. (Set 1 & 4)

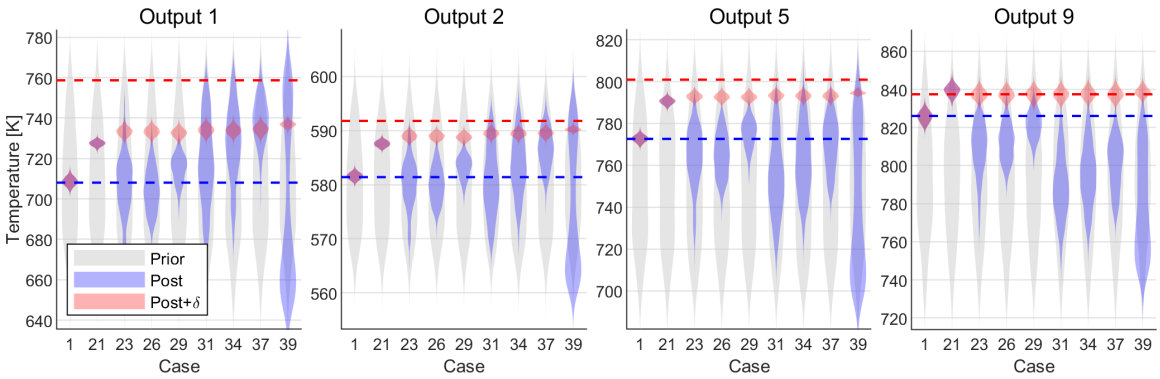


Figure 5.10: Model output marginal distributions in physical space for prior predictions, posterior predictions, and corrected posterior predictions. The two posterior solutions are assessed against the true values $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta}^*)$ (blue dashed line) and $\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta}^*) + \boldsymbol{h}(\boldsymbol{x})$ (red dashed line), respectively. (Set 1 & 4)

5.4.4 Use of the calibrated discrepancy for physics model diagnosis

It was stated in Section 5.1 that we aimed to investigate how model discrepancy influences predictions. It was further shown in Section 5.3 through the simple example that making predictions that are corrected by a discrepancy function is not straightforward due to interpolation/extrapolation issues (discrepancy functions are defined only at sensor locations). Therefore, it was proposed to instead use model discrepancy for diagnostic purposes in this application: perform a back-to-back calibration of the model with and without a model discrepancy function included, then review the shifts in the posteriors and posterior predictions to look for boundary conditions that contribute to the source of missing physics in the model.

First, we assume that a good discrepancy function has been obtained, such as Case 29. Then, for example, the shift seen for θ_4 and θ_5 when comparing Case 21 (no discrepancy function) to Case 29 (includes a discrepancy function), which are the most significant parameters that drive output in the region with MFE. For the model outputs, Figure 5.10 Case 29 shows that Output 1 corrected solution (red distribution) is still far from the measurement (red dashed line), but the correction performs better at Outputs 2, 5, and 9. This suggests the MFE is nearest the region of Output 1 (which is indeed where the model was in error).

5.5 Conclusion

The main contribution of this chapter is the definition of additive discrepancy functions for multivariate models within the PC space of the model outputs. This work included the identification of possible functional forms in the PC space through a rank-based identifiability assessment and through performing term selection using multivariate sensitivity analysis. The proposed methodology was partially demonstrated for a simple model and then further extended to a multivariate output (36 outputs) heat transfer model. These demonstrations highlighted the dichotomy of the importance of including a model discrepancy term to avoid bias in the calibrated model parameters and the challenges presented when considering multivariate model output.

In the engineering application example, the best discrepancy functional form was found to be a simple linear function applied to just the first two PCs (Case 29). The simplest discrepancy function option of adding a constant term to only the first PC (Case 22) performed nearly as well as or better than more complex functions. The results also showed that the zero-mean priors for model discrepancy performed better than priors based on the initial prior model discrepancy, which simplifies definition of the priors. These results show that a model discrepancy term can be included for multivariate models using simple function definitions, enabling simultaneous calibration with the model parameters to reduce model parameter bias. This behavior was demonstrated in the results for the most important parameter (θ_4). However, for other parameters that had less influence on the model output (e.g., θ_5), the practical non-identifiability may have interfered such that the bias was made worse.

In Section 5.4.4, the challenges with using the calibrated discrepancy functions during model predictions were discussed. An approach to using model discrepancy as a diagnostic tool was recommended. The recommended approach is to perform a back-to-back calibration with and without the discrepancy function and then compare the two sets of results to determine where the model physics may require refinement. This was demonstrated for the heat transfer application.

There are several areas for further investigation.

- Model refinements may not be an option due to time or practical constraints. In this case, rather than using the back-to-back results described above for diagnosis, the parameter posteriors from both sets of results could be used as an additional source of epistemic uncertainty when making model predictions.

- Other types of missing physics (e.g., flow conditions around the turbine disc rather than the radiation used in the example) and how they influence the calibration performance of the proposed discrepancy functions should be considered.

- Inclusion of more of the original time series in the data set before performing PCA [36] may improve surrogate model results. Adding more data to the PCA transformation might increase the number of meaningful PCs and consequently require an increased number of discrepancy parameters (although, it was noted that, due to correlations, this is likely to have diminishing returns).

- Finally, Subramanian and Mahadevan demonstrate an alternative to the model output discrepancy approach with intrusive [133] and non-intrusive [103] methods that estimate model form error in the governing equations. The benefit is that it facilitates a more rigorous approach for prediction/extrapolation (e.g., for locations in the model other than the sensor positions). This approach could be compared for the heat transfer application to determine the best option when considering complex engineering models.

CHAPTER 6

Multi-metric validation under uncertainty for multivariate model outputs and limited measurements[1]

6.1 Introduction

The main focus of this chapter is model *validation assessment*, which is part of the validation step of the VVUQ framework introduced in Section 1.2. In the framework, model validation is performed after verification (Chapter 3) and calibration (Chapter 4 and 5), but before prediction (Chapter 7). Details regarding the VVUQ process and how these steps interact is the focus of the next chapter.

Given the challenge of modeling the complexity of real-world systems, there have been significant efforts towards improved processes for assessing whether a physics model is 'valid' for its intended use in light of the many sources of uncertainty that affect the model prediction. Several standards and guidelines have been developed in recent years, e.g., [1, 16, 134, 135], that are aimed at model verification and validation (V&V) and uncertainty quantification (UQ), and more recently at assessing *model credibility* [6, 14], which relates to whether or how much the model can be trusted for the intended application. "The concern of V&V is to assess the accuracy of a computational simulation" [16] and involves code verification, solution verification, and model validation. As previously discussed, UQ is aimed at estimating the uncertainty arising from different sources in the models and measurements, propagating and aggregating the multiple types of uncertainty through the model to estimate the uncertainty in the model prediction. It is worth noting that V&V and UQ are not independent processes; each contributes to the other (thus the use of the combined acronym VVUQ). These sources of uncertainty must also be addressed during model validation, as discussed in this chapter.

Model validation is defined within the ASME standards and guides as, "the process of determining the degree to which a model is an accurate representation of the real world from the perspective of the intended uses of the model" [1, 16]. The objective is two-fold: make a decision about the model's adequacy based on a *validation assessment*, and to evaluate the extent of extrapolation from the model's validation performance to its intended use [2]. The first part, *validation assessment*, involves conducting validation experiments and computing quantitative *validation metrics* to assess the difference between model prediction and experimental measurements. A decision about the appropriateness of the model for making predictions may be aided by setting an appropriate validation metric threshold. The second part, *evaluation for intended use*, typically involves prediction, which inherently results in some amount of extrapolation [129] from the experimental condition to the usage condition. This chapter focuses on validation metrics. Setting thresholds and extrapolation are not considered. It is recognized that both of these issues regarding intended use are challenging and often application specific [1].

[1] Adapted from [17]

Despite the research on validation metrics over the past few decades, challenges remain when applying these to practical engineering systems. There is limited guidance in the standards on how the validation assessment should be performed and used in subsequent analysis. For example, the ASME guide for computational fluid dynamics and heat transfer (V&V 20-2009) [16] provides a clear process for computing validation uncertainty and an associated model error interval, but it is not evident how the use of dimension reduction techniques should be addressed, or how correlated parameter uncertainty (e.g., derived through Bayesian calibration as in Chapter 4) should be incorporated, or how the resulting error interval should be applied to model predictions (Chapter 7). A similar guide for computational solid mechanics (V&V 10-2019) offers valuable insights to the V&V process but only provides reference to a few scalar validation metrics, states that validation metrics are an active area of research, and does not clarify how they should be used to influence model use.

This chapter therefore seeks to advance the utility of existing validation metrics (Section 6.2) for engineering applications in three directions: (i) a multi-metric approach to the validation assessment of models with multivariate outputs, (ii) consideration of limited number of measurement samples, (iii) improved interpretability of metric results (Section 6.3). The extended metrics are demonstrated on a 2D numerical example (Section 6.3.4) and the gas turbine engine heat transfer model (Chapter 2) with multiple outputs (Section 6.4). In the latter problem, the multiple outputs are the same physical quantity but at different locations, at steady state; extension to transient multivariate model output is considered in Chapter 7. The chapter's findings and conclusions are summarized in Section 6.5.

6.2 Background

Validation metrics must be capable of quantitatively assessing the difference between measurement and model prediction, both of which are uncertain. Sources of model uncertainty include numerical discretization errors, input uncertainty, model parameter uncertainty, and model form error. Additionally, the computational methods used in UQ often rely on replacing the time-consuming physics models by faster surrogate models, which adds another source of model uncertainty. Sources of measurement uncertainty include instrumentation calibration, installation effects, experimental conditions, operator error, and sample size error due to limited measurements (which may be quantified with statistical methods [38, 136] and incorporated into the validation assessment [137]).

Several quantitative validation metrics have been proposed in the literature and a few of these are discussed below. More work has been done for univariate model output, although several multivariate metrics have been considered. Assessing multiple physics model outputs simultaneously is a key point of interest in this chapter to address correlation among the outputs. Also, much of the existing work on validation metrics has assumed that 'fully characterized' [138] validation experiments have been performed and that there is sufficient replication (repeated measurements) to quantify the measurement uncertainty. These assumptions are often not met in practice

and therefore the deployment of the existing validation metrics for practical applications can be challenging.

6.2.1 Features of interest for the selection of validation metrics

The engineering literature notes desirable properties that validation metrics should possess [2, 5, 139–142] and ways to classify types of validation metrics [7, 138, 143–147]. Due to the many available metrics and varied applications, these properties and classification criteria are not always consistent or comprehensive, thus making the metric selection process challenging. For example, Gardner et al [147] classify metrics into two main categories: probabilistic ratios (f-divergences) and probabilistic differences (integral probability metrics). Riedmaier et al [7] instead categorize metrics based on the validation quantity of interest (deterministic vs distributional) and the metric result (Boolean, probabilistic, or real-valued). However, neither of these classifications consider multivariate validation metrics, which combine the validation assessment across many model outputs and result in a single metric value.

The following requirements are considered important for the selection of validation metrics in this chapter:

Requirement 1. The validation metric should allow for both univariate and multivariate assessments.

Requirement 2. The validation metric should address uncertainty in model predictions and measurements.

Requirement 3. The validation metric should produce an output that is both interpretable by the end-user to assist decision-making and applicable for downstream use, such as model selection or model prediction.

Requirement 4. The validation metric should incorporate correlations among model outputs during multivariate evaluation.

As pointed out in [147], a single metric may not meet all of these objectives. For example, in the first requirement, a multivariate metric provides an overall summary assessment of model validation over multiple outputs, whereas a univariate metric assesses individual outputs and may be more useful for model diagnosis. Therefore, the discussion below (the next two sections) considers several metrics in the selection process and groups them into univariate and multivariate metrics.

6.2.2 Univariate validation metrics

A few of the more popular univariate metrics in the V&V literature were compared by Liu et al [143] and Ling and Mahadevan [138]: classical hypothesis test p-values [118, 119, 148], Bayes' factor [149], area metric [2, 129], 'frequentist' or confidence interval [139, 146] (Liu et al only), and model reliability metric [150] (Ling and Mahadevan only). All of these address requirements 1 (univariate) and 2.

The confidence interval approach provides bounds on a distribution statistic (such as the mean), but does not compare the entire distribution. The classical hypothesis test result is pass/fail, whereas metrics with probability-

based output provides a "degree of validity" [142] that may also be used to carry the validation result into the predictions [5] or to be used in measurement resource allocation [137]. Examples of metric outputs that are probability-based include Bayes' factor and the model reliability metric. The Bayes' factor requires the definition of a "prior model probability" which is often difficult to prescribe. Alternatively, the model reliability metric does not require a prior model, and it is interpretable since it may be related to the engineering units of the validation quantity of interest through an *accuracy requirement* or 'tolerance'. It evaluates the probability that the difference between the model prediction and measurement is greater than this accuracy requirement. Thus, the model reliability metric addresses both aspects of requirement 3 and is selected for further extension in this chapter. It is further introduced below (Section 6.2.2.2).

Another validation metric approach is to compare the cumulative distribution functions (CDFs) of the model output and measurements. The area metric (introduced in Section 5.3.3) does this by computing the absolute difference between the CDFs, which may be analytical or empirical. The ability to assess empirical distributions means that it does not depend on distribution assumptions [2] and can easily be used with sample-based distributions. The metric's resulting area is in units of the validation quantity of interest, $1 \times y = y$ [139], making it interpretable. For example, the user may be able to relate the metric output to existing engineering tolerances in the same units. This metric therefore addresses the interpretability aspect of requirement 3 and is introduced below.

6.2.2.1 Area metric

Let the model output distribution be denoted y and the measurements y_d. The area metric compares the mismatch between corresponding CDFs $F(y)$ and $F_d(y)$, respectively, as

$$A = \int_{-\infty}^{\infty} \Big| F(y) - F_d(y) \Big| dy \tag{6.1}$$

When $A = 0$, there is "no evidence for a mismatch" between the two distributions [2]. The metric is unbounded for positive values, $A \geq 0$. For cases of significant bias such that the CDFs do not overlap, the area metric result is approximately equal to the difference between distribution means [151], i.e., $A \approx |\bar{y} - \bar{y}_d|$.

6.2.2.2 Model reliability metric

The model reliability metric computes the probability that $|y - y_d|$ is within a specified accuracy requirement[2] λ, which may be written as follows where $P(\cdot)$ represents probability

$$R = P\big(|y - y_d| < \lambda\big) \tag{6.2}$$

Thus λ relates the metric result to engineering units, but in contrast to the area metric, the model reliability metric's output R is a probability value between 0 and 1.

6.2.3 Multivariate validation metrics

Multivariate metrics evaluate the differences between model and measurement for many outputs simultaneously and return a single numerical result. A few available metrics are briefly introduced. The Kullback-Liebler (K-L) divergence and the related Jensen-Shannon distance may be computed for either univariate or multivariate model output [145, 152]. The K-L divergence is unbounded and non-symmetric. The Jensen-Shannon distance is derived from the K-L divergence to address these issues, but both metrics result in units that are non-intuitive, and they are computationally challenging in higher dimensions.

Several multivariate extensions have been proposed for the area metric described in the previous section, such as U-pooling [129], PIT area metric [153], and PCA area metric [128]. All the extensions first combine the outputs and then apply the univariate area metric to this combined 'univariate' result. U-pooling uses the concept of the univariate probability integral transform (PIT) theorem [8] to transform measurement samples into a common 'u-space' (probability space). This 'pooled' CDF is compared to a uniform CDF using the area metric, which then reflects the level of disagreement between model response and measurements. However, this procedure does not address correlation between the outputs. Therefore, Li et al [153] used a multivariate PIT that accounts for these correlations. This method is challenging to implement since it requires estimating joint CDFs of the model outputs, which may not be possible for high-dimensional model outputs [128]. For this reason, Li and Lu [128] proposed a PCA-based area metric, which combines the multivariate outputs by first transforming them into an uncorrelated principal component space, and then aggregating these uncorrelated outputs through a weighting based on their eigenvalues. This metric was used for model-to-model comparisons to assess discrepancy function performance in Chapter 5. It was shown that the PCA area metric is straightforward to compute (Eq. 5.13), but the resulting units are non-intuitive (they are eigen-value weighted linear combinations of the original model outputs), and therefore setting thresholds is more challenging. The PCA area metric may be more useful for

[2]The accuracy requirement is a tolerance on the difference $y - y_d$, whereas a threshold on the resulting probability R may still be needed by the decision-maker. Thresholds are beyond the scope of this dissertation. However, Ling and Mahadevan [138] related the threshold probability value of this metric to thresholds in p-value, z-test, and t-test (under the assumption of Gaussian model output and measurement uncertainty).

relative comparisons, as in the previous chapter, where a threshold is not required. In this chapter this metric is not pursued since the present objective is to evaluate a single model to measurements. Thus, these multivariate extensions to the area metric do not address the need for an interpretable metric that also accounts for correlation among multivariate model outputs. For the purpose of this chapter, it was therefore concluded not to use a multivariate area metric in the multi-metric approach (an alternative is considered in the next section).

Finally, the univariate model reliability metric has been extended to a multivariate model reliability metric [141]. Similar to the univariate version, it compares the difference between model output and measurements to an accuracy requirement. The multivariate Mahalanobis distance (MD) [154] is used for this purpose since it accounts for correlation between outputs. This property of the multivariate model reliability metric, when combined with the univariate metrics, means that all four of the above requirements are met. Therefore, this metric has been chosen for further extension, along with the univariate area metric and univariate model reliability metric in Section 6.3, and its mathematical definition is given below.

6.2.3.1 Multivariate model reliability metric

The multivariate model reliability metric compares multivariate distributions of the model $\boldsymbol{y}$ and measurements $\boldsymbol{y}_d$ using the MD, denoted $\boldsymbol{M}_j$ for measurement j. The MD is compared to a multivariate accuracy requirement,

$$R_{Mj} = P\big(\boldsymbol{M}_j \leq \lambda_M\big) \tag{6.3}$$

$$\lambda_M = \sqrt{\boldsymbol{\lambda}^T \tilde{\Sigma}^{-1} \boldsymbol{\lambda}} \tag{6.4}$$

where λ_M combines the univariate requirements for n outputs $\boldsymbol{\lambda} = [\lambda_1, \lambda_2, \cdots, \lambda_n]$ and transforms this vector into the MD space using the covariance matrix $\tilde{\Sigma} = \Sigma_y + \Sigma_d$, where Σ_y and Σ_d are the covariance matrices of the model outputs and measurements, respectively [141]. This λ_M (with $\tilde{\Sigma}$) defines a hyperellipse, which is shown below in the numerical examples (Section 6.3.4). The MD is computed between the j^{th} measurement replicate $\boldsymbol{y}_{dj}$ and i^{th} model sample $\boldsymbol{y}_i$, also using $\tilde{\Sigma}$ as

$$M_{ij} = \sqrt{(\boldsymbol{y}_i - \boldsymbol{y}_{dj})^T \tilde{\Sigma}^{-1} (\boldsymbol{y}_i - \boldsymbol{y}_{dj})} \tag{6.5}$$

in which M_{ij} is a scalar value and $\boldsymbol{M}_j$ is the distribution (over the model samples) for measurement replicate j. A value of R_{Mj} is computed for each replicate, thus the distribution of R_M (over the measurement samples) is due to measurement uncertainty. An example of this distribution is shown in Figure 6.1. If measurement uncertainty is reduced or more measurements are available (which reduces the sampling error), the uncertainty in R_M likewise decreases. As noted above, this uncertainty may then be reflected in the post-validation model predictions as

shown in [5, 137]. For the purpose of the examples in this chapter, we simply report the mean of this distribution $\bar{R}_M$ by integrating out measurement uncertainty. Use of the full distribution for modifying the posteriors for prediction is demonstrated in Chapter 7 (see Section 7.3.5.3).

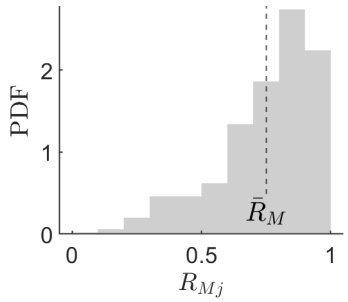


Figure 6.1: An example histogram distribution of R_M for multiple measurement distribution samples j. The mean $\bar{R}_M$ is indicated by the vertical dashed line.

6.3 Methodology

The objective of this section is to extend the area metric, model reliability metric, and multivariate model reliability metric. As previously stated, the intent is to perform the validation assessment simultaneously with multiple metrics (these two types) to meet the metric requirements in Section 6.2.1 and thereby improve the decision made from the assessment. Section 6.3.1 introduces the distributions that are compared in the validation assessment, which is schematically illustrated in Figure 6.2. Then, issues with the existing metrics are highlighted and extensions are proposed in Sections 6.3.2 and 6.3.3. A simple 2D numerical example is used to demonstrate these extensions in Section 6.3.4 before application to the heat transfer model in Section 6.4.

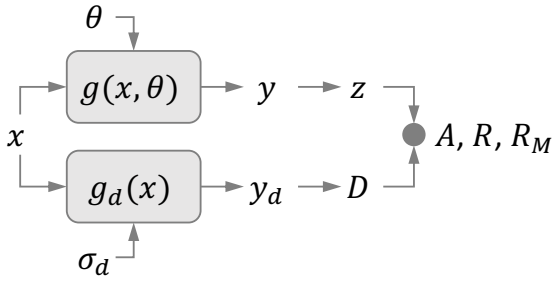


Figure 6.2: The validation process results in computed metrics A, R, and R_M (and their extensions). The model $g(x,\theta)$ and measurement $g_d(x)$ include sources of uncertainty that affect the validation assessment.

6.3.1 Distributions used in the validation comparison

In the context of VVUQ, the validation assessment compares a model output $y = g(x,\theta)$ to a measurement $y_d = g_d(x)$ as shown in Figure 6.2, both of which include various sources of uncertainty. The impact of these sources of uncertainty on the form of the probability distributions y and y_d are discussed in this section and

expanded distributions are defined as z and D, respectively.

6.3.1.1 Measurement distribution

The measurements include the system output quantity y_d used for validation and system inputs x that define that state/condition during test (the inputs are also used in defining the model). Both types of measurements include uncertainty that affect the validation assessment. In this section, we focus on how the system output measurement distribution is defined. It is assumed that the input uncertainty is reflected in both y and y_d.

It is common in engineering applications that there are few measurement samples (replicates) at a given sensor location, perhaps only one or two. Thus, measurement variability cannot be computed directly from the test samples but must be supplied independently ('prescribed') based on previous information or experience. This is represented by a standard deviation[3] σ_d that represents the combination of random and systematic uncertainty of a given measurement system [38]. In the multivariate case this is defined $\Sigma_d = \text{diag}[\sigma_{d1}^2, \cdots, \sigma_{dn}^2]$. It is assumed that the distribution of measurements takes the form of a multivariate Gaussian[4]:

$$\boldsymbol{D} \sim \mathcal{N}\left(\bar{\boldsymbol{y}}_d, \Sigma_D\right) \tag{6.6}$$

where the combined variance is $\Sigma_D = \text{diag}[\sigma_d^2 + \sigma_d^2/N_d, \cdots, \sigma_d^2 + \sigma_d^2/N_d]$, which includes uncertainty in the sample mean. The result in Eq. 6.6 therefore describes the approximate population of expected measurement samples (and not just the mean value).

6.3.1.2 Model output distribution

As shown in Figure 6.2, the model output distribution y includes the contribution of model parameter uncertainty, model form error, numerical errors, and uncertainty in the inputs. It is assumed that the distribution y (and multivariate $\boldsymbol{y}$) is sample-based as the result of uncertainty propagation using Monte Carlo simulation methods. For example, parameter uncertainty may be characterized through Markov chain Monte Carlo (MCMC) during Bayesian inference [58] (however, the validation approach proposed in this chapter is not dependent on the use of Bayesian inference). To compare to the distribution $\boldsymbol{D}$, the zero-mean Gaussian measurement uncertainty with variance Σ_d is included with the model output as

$$\boldsymbol{z} \sim \boldsymbol{y} + \mathcal{N}(\boldsymbol{0}, \Sigma_d) \tag{6.7}$$

[3] The terminology 'standard uncertainty' is used in [38].

[4] Mullins et al [137] treated error in the mean value for model validation using the t-distribution, but we use Gaussian since the t-distribution is undefined for a single degree of freedom.

which has covariance $\Sigma_z = \Sigma_y + \Sigma_d$. The motivation for this addition to the model output is that we are predicting the full distribution of what we would measure, if there were enough measurements [28]. This is explained through the following four scenarios discussed below and schematically shown in Figure 6.3. For this discussion, assume (i) there is no input uncertainty, (ii) but there is uncertainty in the measured output (characterized by prescribed σ_d), and (iii) there is either one or very few measurement samples.

(a) The first scenario compares a single measurement sample y_d to a deterministic model output y. The model output is perfectly known since there is no input uncertainty or model uncertainty.

(b) This scenario adds to the single measurement in (a) zero-mean Gaussian measurement uncertainty to represent the measurement distribution D (sampling error is ignored). In order to compare the model prediction to D, the same measurement uncertainty must be added to y, resulting in z.

(c) This scenario is the same as (b) except that sampling error is accounted for. D is now represented by the mean $\bar{y}_d$ (which could still be a single measurement) and an increased variance. The model distribution z remains the same since sampling error does not apply.

(d) In this final scenario, D is the same as in (c) but now z incorporates sources of model uncertainty. This scenario represents the final result for Eq. 6.6 and 6.7 used in the validation assessment in this chapter.

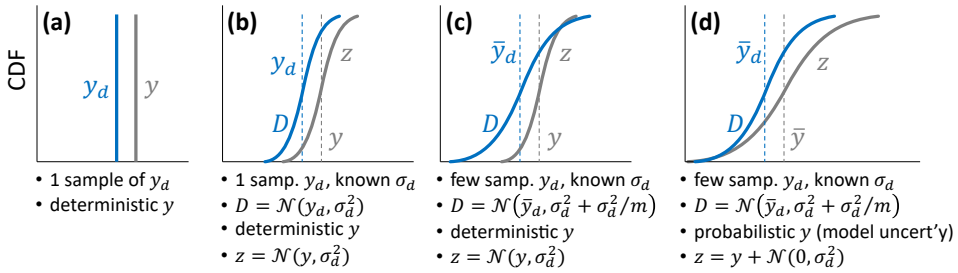


Figure 6.3: Scenarios to motivate the use of the expanded distributions D and z in the validation comparison.

Further motivation for comparing $\boldsymbol{D}$ to $\boldsymbol{z}$ in the validation assessment may be seen by considering issues with the existing validation metrics, which are discussed below separately for the area metric (in Section 6.3.2.1) and the model reliability metric (in Section 6.3.3.1).

6.3.2 Extensions to the area metric

There are two issues related to the original area metric that we seek to address in this chapter. The first relates to the discussion above when comparing the model CDF to the measurement CDF based on limited samples. Second, there is confounding between errors in distribution bias and shape when comparing the two CDFs.

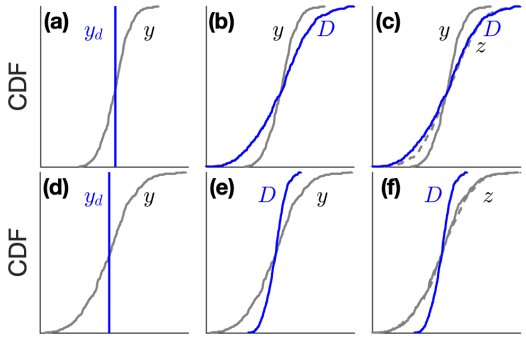
6.3.2.1 Limited measurement samples

This first issue was the initial motivation for the definitions of D and z given in the previous section. When the measurement CDF is based on limited measurements, the area metric is less informative. For instance, consider

computing the area metric between a single measurement y_d compared to a model output distribution y. Two cases are shown in Figure 6.4a/d where the case in the bottom row has a larger model output variance. In either case, it is unclear whether there is agreement in the distribution shapes [129] and only a comparison of mean values can performed. In Figure 6.4b/e the measurement y_d is expanded to D (Eq. 6.6), which shows that the model output and measurement uncertainty are different. However, the comparison is still inconclusive since the model output distribution may have low uncertainty (for example, if the model and input uncertainty are low). In fact, comparison to z in Figure 6.4c shows that the model and measurement distributions are not substantially different, whereas the opposite conclusion is drawn in Figure 6.4f. By expanding the two sets of CDFs the comparison using the area metric is clarified, so Eq. 6.1 becomes:

$$A^o = \int_{-\infty}^{\infty} \Big| F(z) - F_D(z) \Big| dz \tag{6.8}$$

where this may be written A_k^o for the k^{th} output of a multivariate output.



Top row: Var$[D]$ $\sim$ Var$[z]$, Bottom row: Var$[D]$ $<$ Var$[z]$

Figure 6.4: Illustration of how limited measurement samples results in an ambiguous distribution comparison. By incorporating the prescribed σ_d to compare D to z (dashed), the validation assessment improves.

6.3.2.2 Confounding of results between distribution bias and shape

Second, even if there is no sampling error, it is not clear whether a non-zero area metric result is due to difference in distribution bias or shape, i.e., the two are confounded. Figure 6.5 shows a similar comparison to the previous two figures, where this time a single measurement is shown only to simplify the comparison. The main point of this figure is to show that the same area metric result is possible where one is due to bias (a) and the other is due to increased variance in the model output (b). The same outcome could also be shown when there are many measurements.

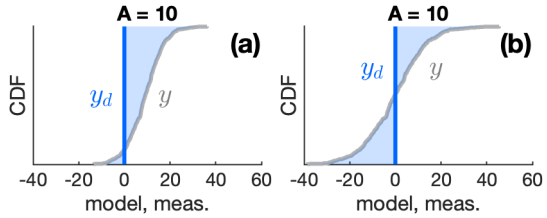


Figure 6.5: Illustration of confounding in the area metric where two different scenarios result in $A = 10$, even though (a) is due to bias and (b) is due to increased uncertainty.

To address the confounding issue in the area metric, two further extensions are proposed for the original area metric, denoted A^b and A^c. The first provides information regarding model bias. A^b is computed in Eq. 6.9 similar to A^o (Eq. 6.8), except with the absolute value removed from the integrand,

$$A^b = \int_{-\infty}^{\infty} \Big(F(z) - F_D(z) \Big) dz \tag{6.9}$$

which may be written A^b_k as noted above for multivariate output. This simple modification assesses the model bias by integrating both the positive and negative areas, rather than their absolute values. If the result sums to zero ($A^b \approx 0$), this gives an indication that there is no bias between the distributions (even if their shapes are different). A positive or negative result indicates the direction of model output bias, e.g., if both outputs are positive, a positive result for A^b means that that model is over-predicting relative to the measurements. If there is no overlap between the distributions, the result of A^b will be equivalent to A^o (except that A^o is always positive).

The second modification is A^c which assesses whether the distribution shapes are different (after 'centering' the two distributions to remove bias by shifting the model output distribution at its mean to the mean of the measurements). It is evaluated as in Eq. 6.10 similar to A^o after shifting the model output distribution. The centered distribution is denoted z' and its CDF is $F'(z)$,

$$A^c = \int_{-\infty}^{\infty} \Big| F'(z) - F_D(z) \Big| dz \tag{6.10}$$

Again, this may be written A^c_k for multivariate output. These extensions will be demonstrated in Section 6.3.4 with a bivariate output.

The components of the expanded area metric may be implemented and interpreted as follows. A^o is computed first; if it is high, this indicates significant error between the model output and measurement. The metrics A^b and A^c may then be compared. If A^b is larger than A^c, it indicates that the error is mostly due to bias, and vice versa.

6.3.3 Extensions to the model reliability metrics

Next, the issues identified above (limited measurement samples and confounding between distribution bias and shape) also apply to the model reliability metric but are treated in a slightly different ways due to the way the metric is computed. These are described in Sections 6.3.2.1 and 6.3.3.2. There is a third issue for the model reliability metric discussed in Section 6.3.3.3, which is how to set the accuracy requirement in both the univariate and multivariate case.

6.3.3.1 Limited measurement samples

When the model reliability metric is computed using model output and measurement samples (rather than analytically), a probability value is generated at each measurement sample, i.e., R_j and R_{Mj} for univariate and multivariate, respectively. First considering the univariate case, when there are limited measurement samples (and measurement uncertainty is not accounted for), the result may be more optimistic since the distribution of $y - y_d$ has variance σ_y^2 rather than $\sigma_y^2 + \sigma_d^2$.

The multivariate case is similarly affected, but when model outputs are strongly correlated the computation of $\tilde{\Sigma}^{-1}$ in Eq. 6.3 may be ill-conditioned and/or result in sensitivity to model bias due to a distorted accuracy requirement hyperellipse. This distorted hyperellipse is shown for a 2D case in Figure 6.6a with $N_d = 1$ measurement and strongly correlated outputs, resulting in $R_{Mj} = 0$. Figure 6.6b shows a similar case but now with $N_d = 2$ measurement samples (the mean is indicated by the triangle). This case shows the sensitivity to bias in the case of strong correlation since $R_{M1} = 0$ (sample at [-5,5]) and $R_{M2} = 1$ (sample at [5,-5]), leading to an overall ambiguous result (as noted above, this metric is evaluated at each measurement sample with the accuracy ellipse is re-centered at the measurement). The expanded distributions proposed in Section 6.3.1 improve these results by comparing D to z, and adjusting the hyperellipse for λ_M based on the redefined covariance matrix $\tilde{\Sigma} = \Sigma_D + \Sigma_z$ (Figure 6.6c). The metric in Eq. 6.3 is now defined

$$\begin{aligned} M_{ij}^o &= \sqrt{(\boldsymbol{z}_i - \boldsymbol{D}_j)^T \tilde{\Sigma}^{-1} (\boldsymbol{z}_i - \boldsymbol{D}_j)} \\ R_{Mj}^o &= P\big(\boldsymbol{M}_j \leq \lambda_M\big) \end{aligned} \tag{6.11}$$

The univariate equivalent to Eq. 6.2 based on $\sigma^2 = \sigma_D^2 + \sigma_z^2$ is

$$R^o = P\big(|z - D| < \lambda\big) \tag{6.12}$$

which is also written R_k^o when indexing through multivariate outputs.

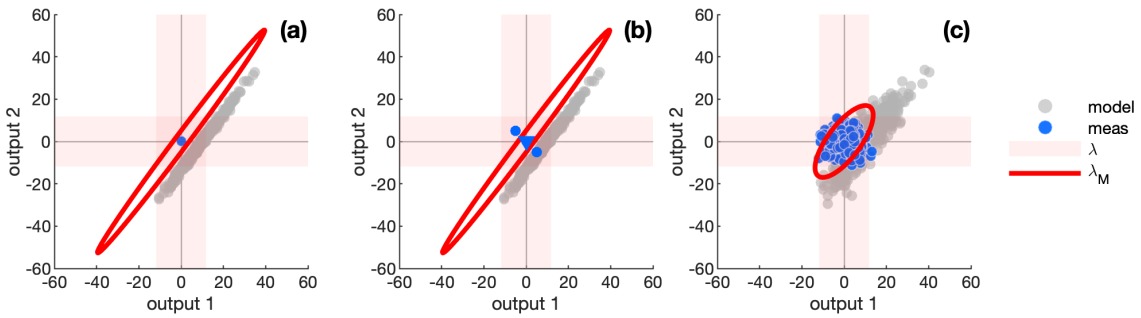


Figure 6.6: Illustration of the issues when there are limited measurement samples and high correlation in the multivariate metric: (a) one sample, (b) two samples (triangle = mean), (c) evaluation based on z and D. Accuracy ellipse is shown at the mean value, but is centered on individual measurements during the calculation.

6.3.3.2 Confounding of results between distribution bias and shape

Similar to the area metric, the model reliability metric also suffers from confounding between distribution bias and shape. This is illustrated in Figure 6.7, which shows two different scenarios for the distribution of $z - D$ that both result in probability $R = 0.5$. The model in scenario (a) is biased, whereas scenario (b) is unbiased but has more uncertainty. A similar situation applies to the multivariate case.

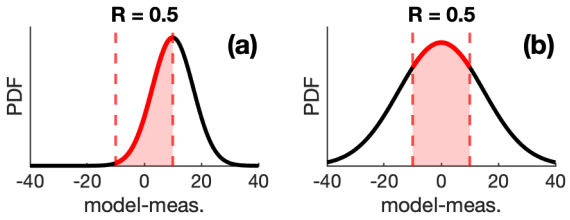


Figure 6.7: Illustration of confounding in the model reliability metric where two different scenarios result in probability $R = 0.5$, even though (a) is due to bias and (b) is due to increased uncertainty.

To address this issue of confounding, two extensions are made to the model reliability metrics, similar to those made for the area metrics. The extended metrics for bias and distribution shape are denoted R^b and R^c (univariate) and R^b_M and R^c_M (multivariate), respectively. In order to reduce duplication, only the multivariate reliability metric is discussed; however, the univariate metric is obtained in a similar manner.

Due to the way the reliability metric is computed, the approach is slightly different to the area metric modification when considering bias. In order to isolate the bias effect, R^b_{Mj} evaluates the model reliability metric based on the model output, and a shifted copy of the model output $\boldsymbol{z} - \boldsymbol{z}'$, where $\boldsymbol{z}' = [z'_1, \cdots, z'_k, \cdots, z'_n]$ represents the model output distribution shifted to the mean of the measurement distribution. This eliminates any differences due to distribution shape while maintaining the original correlation structure of the model outputs. Since the distribution $\boldsymbol{z} - \boldsymbol{z}'$ is compared rather than $\boldsymbol{z} - \boldsymbol{D}$, a revised covariance matrix $\Sigma_{zz} = \Sigma_z + \Sigma_z$ is required in Eq. 6.13 to compute both the Mahalanobis distance M^b_j and the modified accuracy requirement λ^b_M. Thus, R^b_{Mj} is

defined

$$M_j^b = \sqrt{(\boldsymbol{z} - \boldsymbol{z}')^T \Sigma_{zz}^{-1} (\boldsymbol{z} - \boldsymbol{z}')}$$
$$R_{Mj}^b = P\big(M_j^b \leq \lambda_M^b\big) \tag{6.13}$$

Thus, R_M^b will evaluate to a higher probability value when the distributions are unbiased and this will diminish as bias increases. When correlation between the outputs is strong, the metric will be sensitive to the direction of bias, i.e., when bias occurs in directions that are not aligned with the covariance structure of the data (this is illustrated in the numerical example below, Figure 6.10).

Next, R_M^c is defined to evaluate the difference due to distribution shape by removing the difference between the means of the two distributions. This is achieved by evaluating the distribution $\boldsymbol{z}' - \boldsymbol{D}$, where the model output distribution is again shifted ('centered') to the measurement distribution mean (although R_M^b also uses $\boldsymbol{z}'$, R_M^c compares this shifted distribution to the measurements $\boldsymbol{D}$ rather than $\boldsymbol{z}$). This modification is similar to the approach used in the area metric and is defined as

$$M_j^c = \sqrt{(\boldsymbol{z}' - \boldsymbol{D})^T \tilde{\Sigma}^{-1} (\boldsymbol{z}' - \boldsymbol{D})}$$
$$R_{Mj}^c = P\big(M_j^c \leq \lambda_M\big) \tag{6.14}$$

where the distribution of $\boldsymbol{z}' - \boldsymbol{D}$ has the same covariance $\tilde{\Sigma}$ as R_M^o and therefore uses the same accuracy requirement λ_M. Note that R_M^c will evaluate to the same result as R_M^o if there is no bias present.

Regarding interpretation of the extended model reliability metrics, when the probability value of R_M^o is low it indicates error in the model relative to the measurements. If either R_M^b or R_M^c are lower than the other, the lower of the two indicates whether bias or shape is driving the decrease in R_M^o. If both are low, then the error is likely a combination of both bias and distribution shape differences.

6.3.3.3 Setting the accuracy requirement (univariate)

The model reliability metric accuracy requirement represents a tolerance on the model to measurement error. This tolerance is defined by the user in the same units as the validation quantity of interest, which makes the metric result interpretable by the user. However, although a user may have some sense for how to relate this to an existing deterministic comparison tolerance, it may not be clear how it should be defined in the presence of uncertainty. In previous work, setting the accuracy requirement for the model reliability metric was left to the decision-maker. We propose specifying the accuracy requirement using probability intervals that are based on the provided/estimated measurement uncertainty that was discussed in Section 6.3.1.1. Note that this is different than a *threshold* probability value of R^o or R_M^o, which the decision-maker will still need to determine. Thresholds for the model reliability metric and area metric require further work and are not addressed in this chapter. However,

once the model reliability metric's accuracy requirement is established, the decision-maker is more informed about how to set an appropriate threshold for the probability value.

To define the accuracy requirement, we start with the univariate case with distribution $z - D$, which has a total variance $\sigma_z^2 + \sigma_D^2$. Larger values of this variance will result in a smaller probability that $z - D$ falls within λ. Therefore, it is reasonable to set the value of λ based on an appropriate reference value for the variance. For this reference variance, we propose to use the measurement uncertainty as $\sigma_D^2 + \sigma_D^2 = 2\sigma_D^2$, where the factor of 2 is due to the comparison of two distributions. Making this assumption suggests that a good metric value should result when the model and measurement distributions are similar, which is the basis on which the area metric operates (i.e., the area metric returns $A \approx 0$ when the two distributions are similar). Finally, using $2\sigma_D^2$ by itself is equivalent to setting λ at the one standard deviation-level. Therefore, an expansion factor h_1 is applied to achieve the desired target probability level. For example, in the Gaussian case, $h_1 = 1.96$ provides a two-standard deviation accuracy such that perfect agreement will return $R = 0.95$. Therefore, the accuracy requirement is defined as

$$\lambda = h_n \sqrt{2\sigma_D^2} = h_n \sqrt{2\left(\frac{\sigma_d^2}{N_d} + \sigma_d^2\right)} \tag{6.15}$$

where $n = 1$ represents a univariate output ($n > 1$ is considered in the multivariate extension, Section 6.3.3.4) and σ_D was previously defined in Section 6.3.1 as having variance $\sigma_d^2 + \sigma_d^2/N_d$, the second term of which addresses the sampling error in the computed mean value. Also note that the above result must be modified when computing R^b since the distribution is altered to estimate bias. In that case, the appropriate reference variance is $2\sigma_z^2$ and the accuracy requirement is $\lambda^b = h_n \sqrt{2\sigma_z^2}$ (again, with $n = 1$).

6.3.3.4 Setting the accuracy requirement (multivariate)

Next we consider the extension of Eq. 6.15 to the multivariate reliability metric accuracy requirement. Recall that from Section 6.2.3.1 that the multivariate accuracy requirement is $\lambda_M = \sqrt{\boldsymbol{\lambda}^T \tilde{\Sigma}^{-1} \boldsymbol{\lambda}}$, into which Eq. 6.15 may be inserted. However, there are two required adjustments needed in the n-dimensional case to achieve the desired probability level. The corrected multivariate accuracy requirement is denoted λ'_M:

$$\lambda'_M = \frac{h_n / h_1}{\sqrt{\boldsymbol{\sigma}^T \tilde{\Sigma}^{-1} \boldsymbol{\sigma}}} \sqrt{\boldsymbol{\lambda}^T \tilde{\Sigma}^{-1} \boldsymbol{\lambda}} \tag{6.16}$$

where $\boldsymbol{\sigma}^2 = diag(\tilde{\Sigma})$. The first factor, h_n/h_1 simply replaces the univariate factor h_1 (that is already included in Eq. 6.15) with the n-dimensional equivalent expansion factor h_n ($n > 1$) based on results from Wang et al [155]. Their work shows that to achieve a probability of 0.95, the univariate value $h_1 = 1.96$ must increase in higher dimensions, e.g., $h_2 = 2.4477$, $h_3 = 2.7955$, etc.

The second factor, $\sqrt{\boldsymbol{\sigma}^T \tilde{\Sigma}^{-1} \boldsymbol{\sigma}}$, accounts for the projection of vector $\boldsymbol{\lambda}$ into a single hyperellipse of radius

λ'_M. A simple 2D example illustrates why this is required. Let the accuracy requirements be the same in each dimension. Also, let the model output have equal variances σ in each dimension. Computing the original λ_M (Eq. 6.4) results in,

$$\lambda_M = \sqrt{\boldsymbol{\lambda}^T \tilde{\Sigma}^{-1} \boldsymbol{\lambda}} = \sqrt{\begin{bmatrix} \lambda & \lambda \end{bmatrix} \begin{bmatrix} 1/\sigma^2 & 0 \\ 0 & 1/\sigma^2 \end{bmatrix} \begin{bmatrix} \lambda \\ \lambda \end{bmatrix}} = \sqrt{2}\frac{\lambda}{\sigma} \tag{6.17}$$

The $\sqrt{2}$ in this result is eliminated when dividing this additional factor since it can be shown that, $\sqrt{\boldsymbol{\sigma}^T \tilde{\Sigma}^{-1} \boldsymbol{\sigma}} = \sqrt{2}$. When $\tilde{\Sigma}$ includes non-zero covariance terms, $\sqrt{\boldsymbol{\sigma}^T \tilde{\Sigma}^{-1} \boldsymbol{\sigma}} < \sqrt{n}$ but the same cancellation occurs.

A simple bivariate dataset is used in the next section to demonstrate the extended area metric and model reliability metrics. This is followed by a higher-dimensional example for the heat transfer model in Section 6.4.

6.3.4 Bivariate example of the extended metrics

Three model distribution cases are defined in Table 6.1 for comparison to the same measurement distribution to show the behavior of the extended area metric and model reliability metrics. There are two main objectives in this demonstration: Case I is used to compare results across the different metrics, and Case II and III demonstrate the behavior of the multivariate model reliability metric (which addresses correlation between the outputs). However, results are shown for all metrics in Table 6.2 for the three cases.

Table 6.1 defines the three cases as bivariate Gaussian distributions for the model outputs and measurements, which only differ in the model mean value $\bar{\boldsymbol{y}} = [\bar{y}_1, \bar{y}_2]$. The model outputs have standard deviations $\boldsymbol{\sigma}_y = [\sigma_{y1}, \sigma_{y2}]$ and correlation coefficient ρ. The measurements are similarly defined, and it is assumed that there are $N_d = 2$ replicate measurements. $N = 500$ samples were used to define model outputs $\boldsymbol{z}$ and the expanded measurement distribution $\boldsymbol{D}$.

The bivariate distributions for Case I are shown in Figure 6.8a and the corresponding marginal distributions are plotted in Figure 6.9a/d. Figure 6.8b/c show the modified distributions that are used to compute the extended metrics (Sections 6.3.2 and 6.3.3). Figure 6.10a/b show distributions corresponding to Case II and III (similar to the Case I in Figure 6.8a). Each case is discussed further in the next two subsections.

Table 6.1: Definition of bivariate distribution example Cases I-III ($N = 500$) for the model and measurements.

	Model outputs ($\boldsymbol{y}, \boldsymbol{z}$)			Measurements ($\boldsymbol{y}_d, \boldsymbol{D}$)			
	$\bar{\boldsymbol{y}}$	$\boldsymbol{\sigma}_y$	ρ	$\bar{\boldsymbol{y}}_d$	$\boldsymbol{\sigma}_d$	ρ_d	N_d
Case I	$[10, 0]$	$[7.5, 10]$	0.9	$[0, 0]$	$[3, 3]$	0	2
Case II	$[10, 10]$	$[7.5, 10]$	0.9	$[0, 0]$	$[3, 3]$	0	2
Case III	$[10, -10]$	$[7.5, 10]$	0.9	$[0, 0]$	$[3, 3]$	0	2

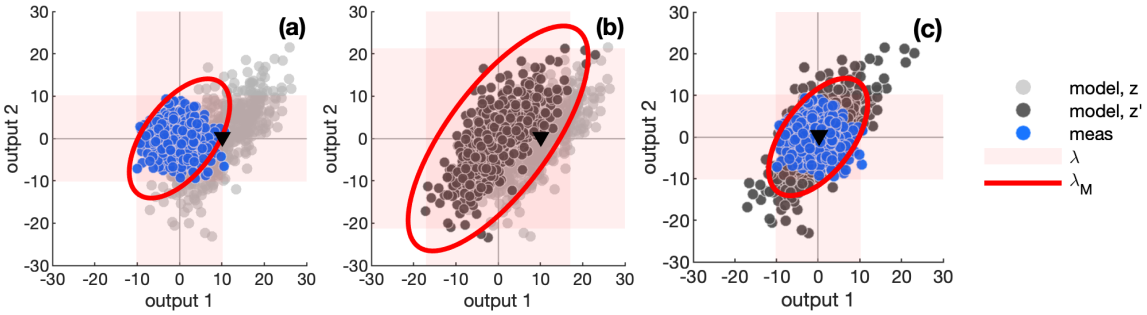


Figure 6.8: Bivariate example Case I (a) distributions $\boldsymbol{z}$ and $\boldsymbol{D}$ used to compute R^o, R^o_M. (b) distributions $\boldsymbol{z}$ and $\boldsymbol{z}'$ used to compute R^b, R^b_M, (c) distributions $\boldsymbol{z}'$ and $\boldsymbol{D}$ used to compute R^c, R^c_M. Accuracy requirements for the univariate are shown at $\pm\lambda$ and multivariate as a red ellipse based on λ_M. Black triangle = model output mean.

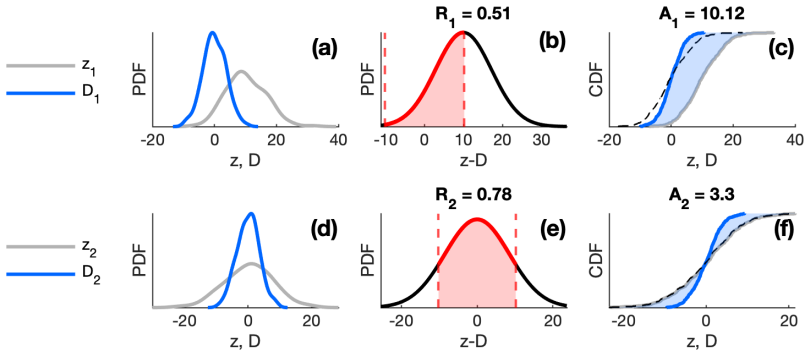


Figure 6.9: Bivariate example Case I (a,d) marginal distributions of $\boldsymbol{z}$ and $\boldsymbol{D}$ from Table 6.1. (b,e) $z - D$ PDF and model reliability metrics (c,f) z and D CDFs and area metrics. The dashed lines in (c,f) are used for computing A^c.

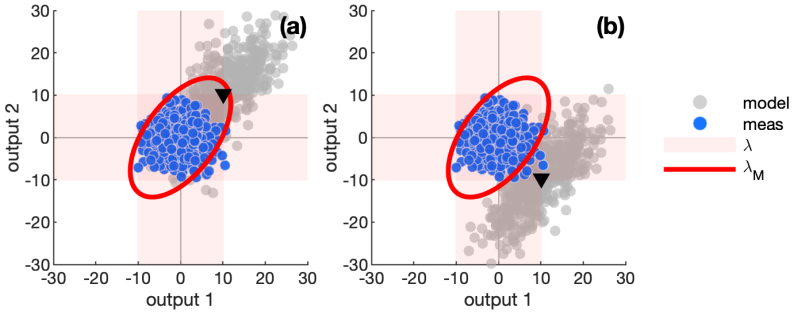


Figure 6.10: Bivariate example (a) Case II with model bias $\bar{\boldsymbol{y}} = [10, 10]$ that is approximately aligned to the direction of model output correlation. (b) Case III with model bias $\bar{\boldsymbol{y}} = [10, -10]$ that is approximately orthogonal with model output correlation. Black triangle = model output mean.

6.3.4.1 Case I univariate metric results for the bivariate model

The marginal distributions of the Case I distribution are plotted in Figure 6.9a/d. The corresponding difference distribution $z - D$ and the computed model reliability metric based on Eq. 6.12 is shown in Figure 6.9b/e. The two CDFs are shown in 6.9c/f, along with the computed area metric based on Eq. 6.8. The computed metric results are summarized in Table 6.2 under Case I 'original'. For the model reliability metric, $\lambda_1 = \lambda_2 \approx 10.2$

(computed from Eq. 6.15 based on $N_d = 2$, $\sigma_d = 3$, and $h_1 = 1.96$).

The related Case I bias and shape metrics are also computed using the appropriate definitions above. As shown for Case I in Table 6.1, there is a 10 unit bias in z_1. This bias is reflected in the computed metrics for output 1, since the magnitudes of the bias metrics are 'worse' than those of the shape metrics. For instance, model reliability metric results are $R_1^b = 0.78$ and $R_1^c = 0.84$, which suggests that bias is to blame for lowering the probability of the original metric $R_1^o = 0.51$. Similarly, the area metrics $A_1^b = 10.1$ compared to $A_1^c = 2.2$ show that bias is the cause of the $A_1^o = 10.1$ result (since bias is significant, A_1^b and A_1^o are equivalent). A low value of A_1^c suggests there is only a small difference due to distribution shape. In the second output z_2, the results are favorable since model reliability metric probability is high $R_2^o = 0.95$ (this is the target probability as defined in Eq. 6.15 with $h_1 = 1.96$) and the area metric result is very low $A_2^o = 0.3 \approx 0$. The assessment is therefore given both in terms of probability and engineering units which helps with interpreting the results and relating it to the magnitude of the physical output. However, the univariate results should be considered along with the multivariate output to ensure correlation is considered in the assessment.

Table 6.2: Bivariate example validation metric results. Case I: results for distributions in Figure 6.8 (original metrics computed for univariate case in Figure 6.9). Case II & III: results for distributions in Figure 6.10 (a) and (b), respectively.

	Case I: $\bar{\boldsymbol{y}} = [10, 0]$			Case II: $\bar{\boldsymbol{y}} = [10, -10]$			Case III: $\bar{\boldsymbol{y}} = [10, 10]$		
Metric	original	b (bias)	c (shape)	original	b (bias)	c (shape)	original	b (bias)	c (shape)
R_1	0.51	0.78	0.84	0.51	0.78	0.84	0.51	0.78	0.84
R_2	0.78	0.95	0.78	0.48	0.84	0.78	0.52	0.86	0.78
R_M	0.37	0.70	0.75	0.43	0.83	0.75	0.10	0.29	0.75
A_1	10.1	10.1	2.2	10.1	10.1	2.2	10.1	10.1	2.2
A_2	3.3	0.3	3.3	10.3	10.3	3.3	9.7	-9.7	3.3

6.3.4.2 Case I, II, and III multivariate metric results for the bivariate model

As previously discussed, the multivariate model reliability metric is useful for providing an overall sense of the model validation assessment and it includes the effects of correlation between outputs. In Case I, the metric results in slightly lower value of $R_M^b = 0.70$ than $R_M^c = 0.75$, indicating that bias is a more significant factor than distribution shape in lowering the original metric R_M^o.

In order to highlight the affects of correlation during the validation assessment, Case II and III were produced using the same model and measurements defined in Table 6.1 except for changes in the imposed model bias. Case II shown in Figure 6.10a has model bias that is aligned with correlation of the dataset. Thus, even though there is now bias in both dimensions, the overall value of R_M increases relative to Case I (from 0.37 to 0.43). Conversely, Case III shown in Figure 6.10b has model bias that is not aligned with the correlation of the dataset, returning $R_M = 0.1$. This is the lowest between the three cases and is worse since the model outputs are biased and the

measurements are not in agreement with the data set's correlation structure.

To summarize the methodology, we have selected and extended the area metric and model reliability metrics in a way that together address the desirable features given in Section 6.2.1. Namely, the effect of limited measurements on metric interpretation was improved through the proposed resampling process; output correlation is considered through the Mahalanobis distance-based model reliability metric; and the interpretability of the metrics' results are improved through augmenting the original metrics with additional versions showing the contributions of bias and distribution shape. By using both metric types together, a model may be diagnosed at individual outputs (or at individual measurement locations) and also given an overall validation assessment.

6.4 Application to the heat transfer model

The proposed methodology is demonstrated in this section on the gas turbine disc heat transfer model that was introduced in Chapter 2. The 2D axisymmetric finite element (FE) model geometry is again shown for reference in Figure 6.11. It includes the same $n_\ell = 9$ thermocouple positions and numbering used in Figure 5.5. These represent measurement positions in the corresponding physical engine, which is used for validation testing. Results were extracted from the thermal FE solution at these same locations to perform the validation assessment.

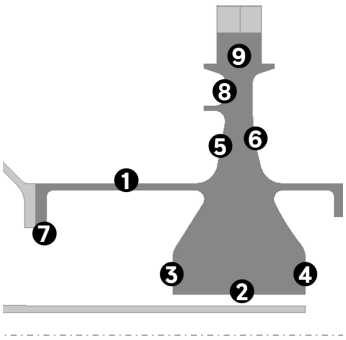


Figure 6.11: Turbine disc 2D axisymmetric FE heat transfer model geometry showing temperature output locations that correspond to thermocouple measurement positions in the physical test.

Typically for model calibration and/or validation, the FE model would be run transiently through several engine conditions (speeds) to evaluate both stabilized temperatures and time-dependent heating and cooling rates. The validation metrics described in this chapter are applied to the stabilized conditions to demonstrate the approach. Thus, when considering the multivariate model response, only spatial correlations are included in this chapter. A feature-based approach for the temporal component was used for model calibration in Chapter 4 [58] and Chapter 5 [98] that could also be used for the validation comparison (by compare time-constants). An alternative approach is demonstrated in Chapter 7 where results from the transient response are extracted at multiple time instants (thus the outputs are all temperature). Other authors have explored an extension of reliability metrics to dynamics applications as discussed in [156]. However, validation of more complicated transient response remains

an area for further work (e.g., when the transient response is not from a square cycle) and is beyond the present scope.

Furthermore, we clarify that model calibration is not a prerequisite for the application of the validation metrics proposed in this chapter. However, the model in this case has been calibrated using one set of measurements (top row of Table 6.4) and therefore separate measurements are used for validation. A Bayesian calibration approach was used, (similar to Chapters 4 and 5), and therefore the model outputs $\boldsymbol{y}$ represent posterior prediction samples.

6.4.1 Predicted model outputs

To demonstrate the methodology, stabilized MTO temperatures were predicted assuming a model that has uncertainty in the response. These predictions are given in Table 6.3. As discussed in Section 6.3.1, the resampled model predictions $\boldsymbol{z}$ (rather than $\boldsymbol{y}$) are used in the proposed validation process. The correlation coefficients between the outputs of both $\boldsymbol{y}$ and $\boldsymbol{z}$ are shown in Figure 6.12 (a) and (c), respectively. To illustrate the behavior of a specific set of outputs, scatter plots of outputs 8 and 9 are also shown in panel (b) for $\boldsymbol{y}$ and (d) for $\boldsymbol{z}$. The high correlation between locations 8 and 9 in panel (b) may cause challenges in the matrix inversion for the multivariate model reliability metric. This has been alleviated by use of the predictive distribution, as shown in (d), due to the additional diagonal terms included in $\tilde{\Sigma}$.

Table 6.3: Summary of model predictions used in validation. Calibrated model outputs are $\boldsymbol{y}$ and the corresponding posterior predictive distribution is $\boldsymbol{z}$ with $\sigma_d = 4.3$K. Calibration measurements are shown in Table 6.4.

		MTO Predicted Temperatures by location [Kelvin]								
	statistic	1	2	3	4	5	6	7	8	9
$\boldsymbol{y}$	$\bar{y}$	707.7	581.2	574.4	571.1	770.0	687.5	591.0	793.0	820.8
	σ_y	5.3	1.6	1.4	1.3	4.6	2.2	2.7	6.3	6.8
$\boldsymbol{z}$	$\bar{z}$	707.8	581.1	574.4	571.2	770.3	687.5	591.1	793.1	820.9
	σ_z	6.8	4.5	4.4	4.5	6.2	4.8	5.0	7.4	8.0

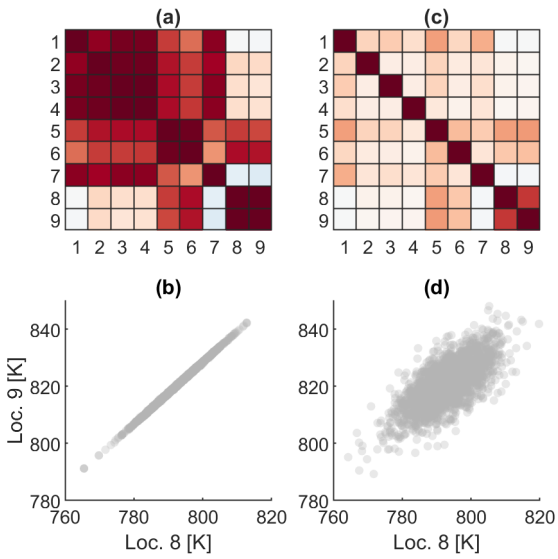


Figure 6.12: Calibrated model output correlation. The scatter plots show correlation coefficient for $\boldsymbol{y}$ (a) and $\boldsymbol{z}$ (c). The corresponding scatter plots for outputs 8 and 9 are also shown for $\boldsymbol{y}$ (b) and $\boldsymbol{z}$ (d).

6.4.2 System response measurements

The measurements used in this demonstration are summarized in Table 6.4. The model was calibrated to $N_d = 4$ synthetic replicate measurements $\boldsymbol{y}_d^C$ (with Gaussian noise of 1% added). The sample standard deviation of these calibration measurements is listed in Table 6.4.

To explore the behavior of the validation metrics, several synthetic validation datasets are compared (they are given set numbers, e.g., $\boldsymbol{D}^{V1}$ is set 1, where the expanded $\boldsymbol{D}$ is used, as defined in Section 6.3). In the validation shown here, it is assumed there is a single replicate ($N_d = 1$) due to instrumentation limitations, but $\sigma_d \approx 4.3K$ was provided by measurement experts ($\lambda \approx 16.7$K based on Eq. 6.15). To facilitate demonstration of the metrics, the different model validation cases are generated by modifying the synthetic measurements rather than modifying the model.

The validation baseline (no bias) measurement set is $\boldsymbol{D}^{V0}$. The next two validation cases $\boldsymbol{D}^{V1}$ and $\boldsymbol{D}^{V2}$ have 10K bias imparted at outputs 8 and 9 near the disc rim (see Figure 6.11) to show the effect of 'orthogonal' (opposite sign) and 'aligned' (same sign) bias, similar to the 2D example shown in Figure 6.10. Orthogonal bias is demonstrated in $\boldsymbol{D}^{V1}$ and aligned bias in $\boldsymbol{D}^{V2}$. The next validation case $\boldsymbol{D}^{V3}$ imparts bias at all outputs of +10K to demonstrate the effects of bias at a greater number of outputs. Finally, case $\boldsymbol{D}^{V4}$ has the same mean as $\boldsymbol{D}^{V1}$, but the assumed measurement uncertainty was reduced by one half to demonstrate a more significant difference in the distribution shapes between model and measurements (which also means that $\lambda \approx 8.3$K for this case).

Table 6.4: Calibration and validation measurements summary. Calibration and validation data are based on different tests. Sufficient replication of the calibration test allowed for estimation of the standard deviation. The validation data standard deviation was provided by expert opinion due to inadequate replication (the same value is used for cases $\boldsymbol{D}^{V0} - \boldsymbol{D}^{V3}$). Bold text indicates differences in the validation cases relative to the baseline $\boldsymbol{D}^{V0}$.

			MTO Measured Temperatures by location [Kelvin]								
Meas. Set	statistic	N_d	1	2	3	4	5	6	7	8	9
$\boldsymbol{y}_d^C$	μ	4	705.9	574.2	571.7	573.8	768.0	690.8	594.4	789.6	826.7
	σ	4	2.7	10.3	6.6	2.6	12.1	4.8	11.4	6.9	10.9
$\boldsymbol{D}^{V0}$	μ	1	708.1	581.4	574.5	571.3	772.6	689.0	590.6	797.8	826.1
$\boldsymbol{D}^{V1}$	μ	1	708.1	581.4	574.5	571.3	772.6	689.0	590.6	**807.8**	**816.1**
$\boldsymbol{D}^{V2}$	μ	1	708.1	581.4	574.5	571.3	772.6	689.0	590.6	**807.8**	**836.1**
$\boldsymbol{D}^{V3}$	μ	1	**718.1**	**591.4**	**584.5**	**581.3**	**782.6**	**699.0**	**600.6**	**807.8**	**836.1**
	σ	*	4.3	4.3	4.3	4.3	4.3	4.3	4.3	4.3	4.3
$\boldsymbol{D}^{V4}$	μ	1	708.1	581.4	574.5	571.3	772.6	689.0	590.6	797.8	826.1
	σ	*	**2.1**	**2.1**	**2.1**	**2.1**	**2.1**	**2.1**	**2.1**	**2.1**	**2.1**

6.4.3 Validation assessments and discussion

In this section, the proposed multi-metric validation approach presented in Section 6.3 is applied to the heat transfer model datasets. Results are compared first for the model reliability metrics and then for the area metric. The cases are organized such that Figure 6.13 (model reliability) corresponds to Figure 6.15 (area metric), and these consider imposed bias in just two of the rim thermocouples (cases $\boldsymbol{D}^{V1}$ and $\boldsymbol{D}^{V2}$). Similarly, Figure 6.14 (model reliability) corresponds to Figure 6.16 (area metric), and these cases show the effect of a bias ($\boldsymbol{D}^{V3}$) and a shape difference ($\boldsymbol{D}^{V4}$) across all locations.

6.4.3.1 Model reliability metrics

The model reliability metric results in Figures 6.13 and 6.14 include the univariate and multivariate results in three panels, one per case. Panel (a) of both figures is the baseline case $\boldsymbol{D}^{V0}$. In each panel, the horizontal axis includes univariate outputs 1-9 and a final entry for the multivariate output labeled 'M'. The vertical axis is probability. Based on the accuracy requirement definition (Eq. 6.15), a probability $R = 0.95$ implies perfect agreement between the model predictions and measurements, which is indicated by the red line (where R generically represents the reliability metric probability result, and R_k and R_M are generic univariate and multivariate results). In this example, $N_d = 1$ which increases λ, giving the results a larger target, and showing the importance of good estimates of measurement uncertainty when replication is low. If there is a large disparity in the covariance or if the model bias is non-zero, as will typically be the case, then the result is $R < 0.95$ and a threshold should be defined to determine how low is acceptable. For the purpose of this application, the decision-maker treats $\boldsymbol{D}^{V0}$ as producing acceptable model validation results (this may represent historical test data and models) and sets the threshold to a probability value of $T_R = 0.8$.

Figure 6.13 compares $\boldsymbol{D}^{V1}$ (b) and $\boldsymbol{D}^{V2}$ (c), to which additional bias was introduced at the rim locations 8 and 9, relative to $\boldsymbol{D}^{V0}$. In $\boldsymbol{D}^{V1}$, the bias has opposite sign[5] (location 8 is -10K and location 9 is +10K). Since location 8 and 9 are in close proximity (Figure 6.11) they are highly correlated (Figure 6.12). The bias in this case is therefore 'orthogonal' to the correlation and R_8^b and R_M^b both show values outside of T_R, indicating that the error is due to bias (R_9^b is satisfactory due to offsetting errors). In contrast, case $\boldsymbol{D}^{V2}$ panel (c) shows results when both locations 8 and 9 are given the same bias of +10K, which is consistent with the strong positive correlation between the outputs. Thus, while we see both R_8^b and $R_9^b < T_R$, it is observed that the multivariate result is $R_M^b > T_R$ for $\boldsymbol{D}^{V2}$, which is a better outcome than in $\boldsymbol{D}^{V1}$ panel (b). Since only the bias was modified, R^c metrics are unaffected relative to $\boldsymbol{D}^{V0}$. The decision-maker's interpretation of these results will, of course, depend on the intended use of the model. For example, if the results are used to determine thermal stresses, it may be more important to achieve consistency in correlation (captured via R_M) than getting the absolute temperature level at a given location (captured by R_k) below a bias threshold.

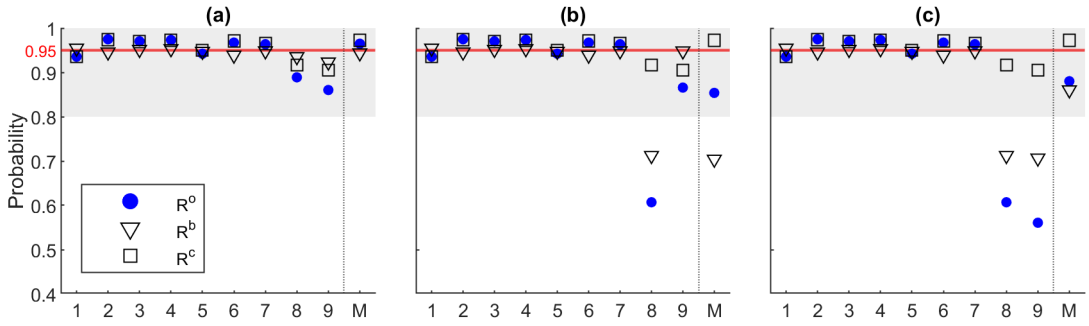


Figure 6.13: Model reliability metric results for Cases $\boldsymbol{D}^{V0}$ (a), $\boldsymbol{D}^{V1}$ (b), and $\boldsymbol{D}^{V2}$ (c) as defined in Table 6.4

Figure 6.14 compares $\boldsymbol{D}^{V3}$ (b) and $\boldsymbol{D}^{V4}$ (c) to show the impact of bias at more locations ($\boldsymbol{D}^{V3}$) and the impact of a greater disparity in variance ($\boldsymbol{D}^{V4}$). Validation results in panel (b) show that bias imposed at all locations results in a low metric. All but location 1 fall outside of the acceptance limit T_R. It is not unexpected that the result will be low since the model overall performs poorly and not all of the correlations shown in Figure 6.12 are positive. Since only bias is present in the case shown in panel (b), R^c results are acceptable. Validation results in panel (c) show case $\boldsymbol{D}^{V4}$, which has no imposed bias, but the measurement uncertainty assumption was reduced to $0.5\sigma_d$ so that model variance was higher in a relative sense. This results in in failed R^c for locations 1, 8, and 9.

[5]This bias is relative to the measurements; it could be considered model bias by changing the signs to +10K and -10K, respectively

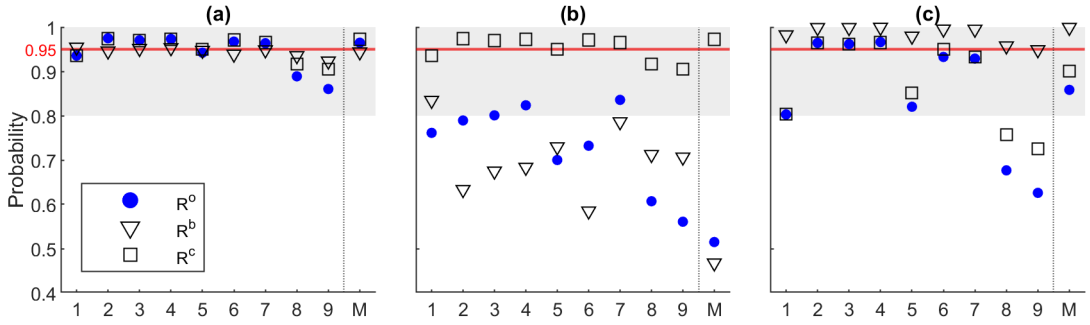


Figure 6.14: Model reliability metric results for Cases $\boldsymbol{D}^{V0}$ (a), $\boldsymbol{D}^{V3}$ (b), and $\boldsymbol{D}^{V4}$ (c) as defined in Table 6.4

6.4.3.2 Area metrics

The area metric results in Figures 6.15 and 6.16 are based on the same comparisons made with the model reliability metrics. In this case, there is no multivariate metric. Along with the proposed area metrics, the mean difference between the model response and measurements ($\bar{\boldsymbol{y}} - \bar{\boldsymbol{y}}_d$) is included. It is observed that this difference aligns closely with the bias metric A^b as noted before. Note that, bias is imposed in Table 6.4 relative to the measurements, but these plots are shown as $y - y_d$ out of convention, giving the opposite sign (typically, one thinks of bias in the model rather than measurements). Furthermore, the decision-maker's threshold of $T_A = 12$ K is included for reference (shown as gray bands).

Figure 6.15 (a) plots $\boldsymbol{D}^{V0}$, $\boldsymbol{D}^{V1}$ (b), and $\boldsymbol{D}^{V2}$ (c). Similar to the results for the model reliability metric, panel (b) shows that the proposed area metric for bias A_8^b indicates poor model performance, and the sign of the metric agrees with with $y - y_d$ (as compared to A^o which is always positive). Panel (c) shows that A_8^b and A_9^b are both biased in the same direction. If one considers only these univariate cases, it would be tempting to consider the result of panel (b) as superior to panel (c) – just the opposite as concluded from the model reliability metric. Thus, considering both metrics together provides a clearer overall perspective on the quality of the model validation results for the decision-maker.

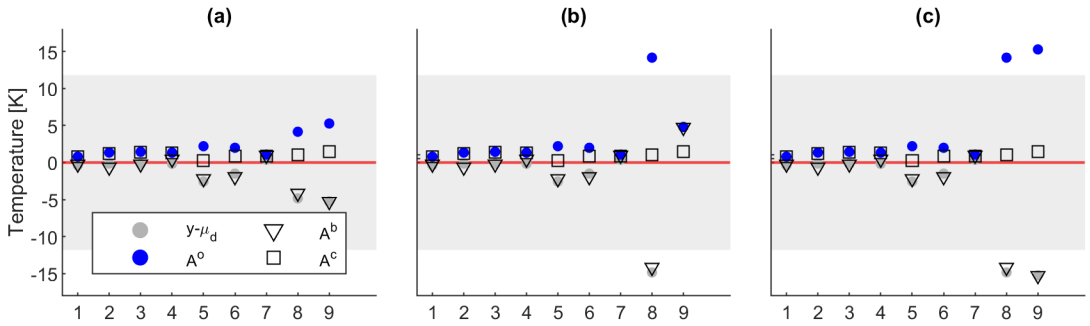


Figure 6.15: Area metric results for Cases $\boldsymbol{D}^{V0}$ (a), $\boldsymbol{D}^{V1}$ (b), and $\boldsymbol{D}^{V2}$ (c) as defined in Table 6.4

Figure 6.16 (b) shows conclusions similar to the univariate model reliability metrics for case $\boldsymbol{D}^{V3}$ where

all outputs are biased with +10K. There are several outputs (1-4, and 7) that are on the edge of the threshold, while locations 5, 6, 8, and 9 have all exceeded T_A. Panel (c) shows case $\boldsymbol{D}^{V4}$ (with an increased disparity in the variance). In contrast to the model reliability metric conclusions in Figure 6.14, the area metric indicates an acceptable outcome. It appears to have a lower sensitivity to distribution shape differences, when compared to the model reliability metric, which showed results below the threshold for outputs 8 and 9.

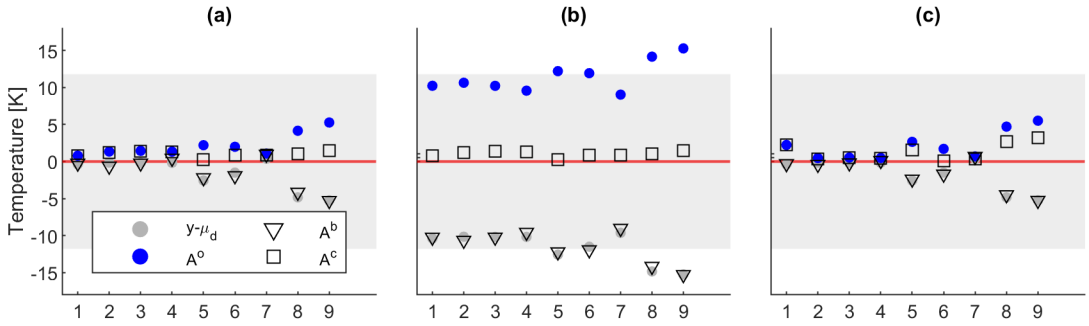


Figure 6.16: Area metric results for Cases $\boldsymbol{D}^{V0}$ (a), $\boldsymbol{D}^{V3}$ (b), and $\boldsymbol{D}^{V4}$ (c) as defined in Table 6.4

6.5 Conclusion

The topic of quantitative model validation has grown in importance as engineering decision-making increasingly relies on computer simulation, especially for complex systems. However, in practice, quantitative approaches have generally seen limited use, or the decision-making benefits have not yet been shown to be of significant value. Considering these challenges, in this chapter we have selected and extended two validation metrics (area and univariate/multivariate model reliability) that together address the desirable metric features that were listed in Section 6.2.1. We have addressed the effect of sample size on the metric interpretation through the a resampling procedure (Section 6.3.1). The output correlation was considered by the use of the Mahalanobis distance-based model reliability metric (Section 6.3.3). The interpretability of the results are improved through augmenting the original metrics with additional versions that indicate the contributions bias and distribution shape (Section 6.3.2 and 6.3.3). The proposed extensions to the validation metrics are demonstrated on a simple numerical example in Section 6.3.4.

Finally, the extended area metric and model reliability metrics were used together for the validation assessment of a gas turbine FE model. The overall assessment using the multivariate metric was augmented with the assessment at individual locations, allowing problem areas in the model to be identified. By using both the area metric and univariate model reliability metric, this location-by-location diagnosis may be given in terms of metric outputs in engineering units or probability, improving interpretability. From the area metric, the user has a more intuitive understanding of the error magnitude relative to the physical quantity. The probability-based model reliability metric provides a physically meaningful interpretation (with probability values bounded between 0 and 1) to

summarize the individual and the overall model validation result (using the multivariate model reliability metric). The probability result for the multivariate metric may also be used for further work within a VVUQ framework, for propagating the validation result to predictions with the validated model [5]. This is demonstrated using the VVUQ framework developed in Chapter 7.

Two potential application-specific issues may arise when using the proposed multivariate metric and should be considered by the practitioner. First, there is a practical trade-off between the magnitude of the correlation (from $\mathrm{Cov}[\boldsymbol{y}]$) on one hand and the magnitude of measurement uncertainty σ_d on the other. The goal of introducing the multivariate model reliability metric is to capture output correlation, which is achieved by using the Mahalanobis distance (MD). However, it was noted in Section 6.6 that extreme correlation may cause numerical issues or practical issues (such as metric sensitivity to limited samples, shown in Figure 6.6a/b). By adding measurement uncertainty when defining $\boldsymbol{z}$ (Section 6.3.1), these numerical and practical issues are reduced (see Figure 6.6c and Figure 6.12). However, if measurement uncertainty is relatively high, it will tend to 'wash out' these correlations between outputs in $\boldsymbol{z}$. In the limit of increasing measurement uncertainty, the computed MD effectively becomes equivalent to a scaled Euclidean distance, because the covariance matrix becomes more and more diagonal (small covariance terms).

The second issue is related to the use of covariance when computing the MD. While the proposed metrics do not make any assumption about the distributions of model output and measurement, it should be considered whether covariance is a good 'statistic' for characterizing the distance from the distribution when the model output distribution is non-normal. This is analogous to the use of variance to characterize a distribution in the univariate case. If the model output distribution is very non-normal and 'skewed' (which may be determined by various multivariate normality tests, e.g. [157]), it is possible that the use of covariance is inappropriate or that transformations [158] should be made to the model output before evaluating the multivariate metric. This is an advanced statistical topic that is outside the scope of this dissertation.

In addition to further exploration and developing more specific guidance on the two issues raised above, other directions for future research include setting metric thresholds and applying the multivariate metric to higher-dimensional datasets. The time-dependent output was not explicitly included in this chapter and, although it is advanced further in Chapter 7, it deserves further attention. Since the multivariate metric provides an overall assessment of the model, it could be investigated for application to model selection [159–161] and sensor placement optimization [59, 162, 163].

CHAPTER 7

Uncertainty aggregation through model development and assessment towards prediction

Expect everything, and the unexpected will never happen.

–Norton Juster, *The Phantom Tollbooth*

7.1 Introduction

It was discussed at the beginning of this dissertation that there has been significant development of rigorous verification and validation (V&V) and uncertainty quantification (UQ) methodologies in the last 20 years. Despite significant advances in the many facets of VVUQ methodologies, studies illustrating applications to challenging, practical engineering problems have been limited. Demonstrations that exist tend to focus on certain individual aspects of the process, and application of a more holistic process requires significant expertise to piece together the methods and tools in all the steps. Therefore, the goal of this chapter is to pursue *uncertainty aggregation* (the accumulation of uncertainty) through all the steps of VVUQ that were introduced in Section 1.2, and apply the methodology to the multivariate predictions of the computational model introduced in Chapter 2. The intent of the methodology is to not only identify and aggregate the uncertainty sources, but also to provide several points of feedback for the purpose of improving the model. In this way, information and uncertainty from the model and test measurements are accounted for when assessing the accuracy and usefulness of the model.

The chapter is organized as follows. Section 7.2 introduces a few existing uncertainty aggregation studies and identifies gaps that are addressed in this research. Section 7.3 presents a proposed VVUQ framework with application to the heat transfer model from Chapter 2. Conclusions and recommendations for further work are discussed in Section 7.4.

7.2 Background

The process and methodologies of VVUQ seek to combine two imperfect sources of information – models and measurements – and derive decision-making value by addressing the uncertainty in both. Individual steps of the VVUQ process have been studied by many authors, as recently reviewed by Riedmaier et al [7], resulting in a "heterogeneous landscape" of methodologies. Most of the methodologies are focused on individual aspects of model V&V and/or UQ and only a few consider the aggregation of uncertainty in a comprehensive VVUQ process. Two of the more complete methodologies for uncertainty aggregation as noted by Riedmaier et al are the "Frequentist" approach (which we refer to as the probability bounds analysis, PBA) by Roy and Oberkampf [3] and the "Bayesian" approach by Sankararaman and Mahadevan [5]. Several organizations have also published

guides and standards for V&V of models to improve the quality and credibility of engineering computational models [1, 6, 14, 16, 40, 134, 164]. The guidance is generally useful but is often high-level, leaving the practitioner to fill in the gaps in their own application in order to select among competing techniques for individual VVUQ steps. Furthermore, important concepts are not addressed in these guides and standards, such as the uncertainty resulting from model calibration and the errors that arise from the use of surrogate models to speed up the VVUQ analysis.

Drawing from the above references, four steps are common in VVUQ analysis: *model definition* (which includes the conceptual model, mathematical model, and computational model), *verification*, *validation*, and *prediction*. A fifth important step is *model calibration*. A sixth step, *model reduction*, is also included in the proposed framework to more clearly include the model simplifications that are typically required, such as surrogate modeling and dimension reduction. These six steps of VVUQ were defined in Section 1.2 and several have already been discussed in detail in previous chapters. In this chapter, the six steps are used first to benchmark previous VVUQ studies and then are demonstrated in the next section using the heat transfer model from Chapter 2. The steps and associated substeps for the framework developed in this chapter are shown in Figure 7.1 in Section 7.3.

The two VVUQ studies mentioned above (PBA and Bayesian approaches) and two of the ASME standards are selected in this chapter for comparison against the six VVUQ steps to highlight common gaps that must be addressed for the application of a VVUQ process. The first engineering standard considered, ASME V&V10 [165] (solid mechanics), was published in 2006 and was among the earlier guidance documents on the topic of V&V[1]. This was revised in 2019 as an official standard [1] and provides broad guidance on several VVUQ topics, focusing the most on model definition, verification, and validation. It briefly discusses key aspects of model calibration but offers limited guidance on specific methods and does not address uncertainty aggregation. The second engineering standard, ASME V&V20 [16] (CFD and heat transfer), was published in 2009 and focuses primarily on V&V. It is more prescriptive in the methodology than V&V10, deriving 'validation uncertainty' by combining numerical error, input uncertainty, and measurement uncertainty using an approaches established by the metrology community [136], e.g., sum of squares of 'standard uncertainties'. Thus, there is some aggregation towards bounding the validation result, but the sum of squares may not always be applicable for other sources of uncertainty (such as model calibration parameter uncertainty, dimension reduction errors, or surrogate model errors). It is also not clear how to incorporate these other sources of uncertainty into the final predictions.

The third VVUQ study is the probability bounds approach [3], which combines aleatory and epistemic sources of uncertainty through a probability box (p-box). This work along with [2] covers the aspects of model definition, verification, validation, and prediction. In the probability bounds approach, aggregation of uncertainty is achieved by expanding (adding to) the p-box (using the results of the area metric [129]), which treats each uncertainty

[1] Others include the 1998 AIAA guide for CFD [134] and 1996 DOD instructions [164]

source as independent (correlation is not accounted for). The width of the p-box define an error interval for model predictions, with some guidance on how to adjust this for extrapolation. Model calibration parameter uncertainty is not discussed, but could be included as an additional independent source of uncertainty. Applications and extensions include [23, 128, 153, 166–168]

The fourth VVUQ study is a Bayesian network approach [5] for uncertainty aggregation across verification, calibration, validation, and prediction. Various sources of uncertainty were addressed including input uncertainty, numerical errors (verification and surrogate model), and model parameter uncertainty (quantified through Bayesian calibration). These sources of uncertainty were combined with a probabilistic validation metric to quantify the prediction uncertainty. While this paper did not focus on model definition or how to perform verification for more complex models, it nevertheless incorporates the broadest set of VVUQ steps. Furthermore, the Bayesian approach is able to account for appropriate correlations between sources of uncertainty. This Bayesian approach has also been extended to multivariate validation metrics [141], sparse and imprecise measurements and mixed aleatory and epistemic input uncertainty sources [60], inclusion of model discrepancy [13, 98], and overall integration of verification, calibration, validation, and prediction [5].

Based on extensions to the Bayesian framework noted above, a similar Bayesian approach is adopted in this research and extended to address model definition, dimension reduction, adaptive meshing errors, the presence of both aleatory and epistemic uncertainty in the model inputs, and multivariate model validation. This chapter demonstrates the framework as an end-to-end application to the heat transfer model.

7.3 The proposed framework

The proposed VVUQ framework is shown schematically in Figure 7.1, which forms the outline of this section. The next six subsections discuss each of the main steps, which are indicated by the dark blue boxes, and demonstrate them using the heat transfer model. The red boxes indicate inputs to each step and the light blue box is the end result along with the prediction. Process sub-steps (gray) represent tasks around VVUQ which feed into the goal of this chapter, which is to aggregate uncertainty from one step to the next (green boxes). The arrows indicate the general flow of the process, but in practice there are feedback loops throughout that are omitted for simplicity.

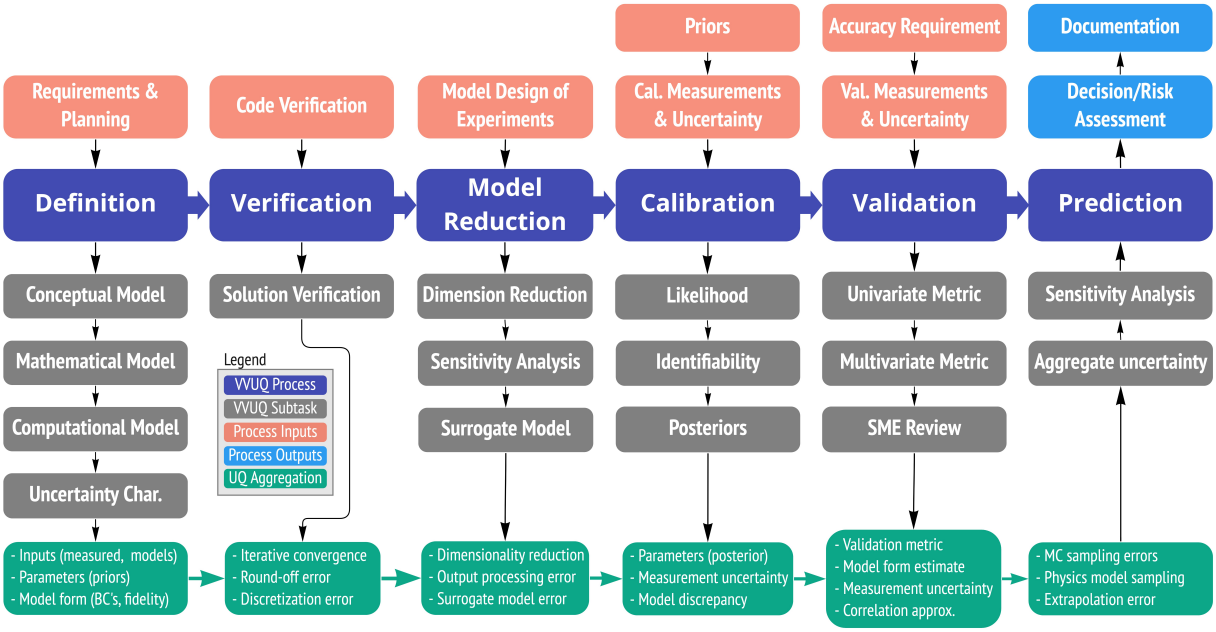


Figure 7.1: The uncertainty aggregation framework developed as part of this research.

7.3.1 Model Definition

The model definition process was outlined in Section 2.1 for the heat transfer model. The start of this process considers the purpose ("intended use") of the model and its output quantities of interest (QoI). Decision-making regarding these QoI is affected by the prediction accuracy of the model. Thus, estimation of 'error bounds' on the predicted QoI is the ultimate purpose of performing VVUQ analysis.

7.3.1.1 Requirements

Section 2.1.1 listed modeling requirements which lead to the identification of physical phenomena involved and the importance of these relative to the QoI. Besides focusing only on the final QoI, the VVUQ process should consider additional requirements through each of the framework steps listed in 7.1. These provide more granular and measurable criteria that lead to opportunities for model diagnosis by the analyst and improved credibility for the decision-maker. Failure to meet any of these individual requirements suggests that further model improvements are necessary, thus guarding against the use of inadequate models. However, once the requirements have been satisfactorily met, both the analyst and decision-maker gain better assurance in the accuracy of the model for its intended use.

A model's intended use may be the direct prediction of a QoI, or the model's predictions may be used as inputs to another model (whose output is the primary QoI) [1, 5]. In the present heat transfer application, the model represents a gas turbine engine disc in and the thermal environment it operates in. Its intended use is to

predict disc metal temperatures that will be used as inputs to a separate stress model of the turbine disc, which in turn is used to estimate part life (usage limits in service). For the sake of demonstration in this chapter, the rim-to-volume-weighted mean temperature T_{rm} is used as a low-fidelity prediction of stress, since T_{rm} is known to correlate well with stress. Therefore, T_{rm} is considered to be the primary QoI and temperature at the measured locations is the secondary QoI.

For the heat transfer model, the accuracy requirements on the QoI are further subdivided into requirements for the individual steps in the VVUQ framework in Table 7.1. The first requirement relates to setting a limit on T_{rm}, after accounting for the aggregated sources of uncertainty in the prediction. The requirement ensures that the probability of failure (exceeding a threshold value $T^*_{rm} = 105$ Kelvin) is 'acceptably low', which is defined here to be $p_f = 0.02$.

Table 7.1: Representative VVUQ requirements for the heat transfer model

#	Requirement	Description	Framework step
1	$p_f = P(\|T_{rm}\| < T^*_{rm}) < 0.02$	Rim-to-volume-weighted mean temp. T_{rm}	Prediction
2	$R = P(\|y_i - y_{d,i}\| < \lambda_i) > 0.8$	Univariate temperature output y_i	Validation
3	$R_M = P(\boldsymbol{M} < \lambda_M) > 0.8$	Multivariate temperature outputs $\boldsymbol{y}$	Validation
4	Bayesian inference of $\boldsymbol{\theta}$	Refine parameter uncertainty $P(\boldsymbol{\theta})$	Calibration
5	$\epsilon_h + \epsilon_p + \epsilon_s \leq 5$ K	Combined numerical errors	Verification & Model Reduction

The second and third requirements relate to the validation assessment of the temperature predictions (the secondary QoI), evaluated with the model reliability validation metric [17, 141, 150]. The validation process proposed in this dissertation was discussed in detail in Chapter 6, which considers multiple metrics that address both the univariate and multivariate output of the model (which was described in Section 2.4.1). Thus, the second requirement considers the temperature output at individual measurement locations, y_i where i indexes over outputs (that include different time instants and spatial locations). This univariate requirement states that the difference between model and measurements $y_i - y_{d,i}$ must be less than an accuracy tolerance λ_i. The accuracy requirement represents the allowable difference in the mean values. The third requirement is the multivariate equivalent of the second, comparing multiple model outputs $\boldsymbol{y}$ to corresponding measurements. The multivariate metric uses the Mahalanobis distance $\boldsymbol{M}$ to account for correlation between the outputs. Another advantage of the multivariate metric is that it condenses the validation assessment across many outputs to a single metric, which can be used for post-validation predictions as shown later in this chapter (Section 7.3.6). The multivariate accuracy requirement is defined as a vector containing the univariate requirements $\boldsymbol{\lambda} = [\lambda_1, \ldots, \lambda_n]$.

The fourth requirement in this example states that calibration is to be performed using Bayesian inference methods [58] to refine (and ideally, reduce) the prior parameter uncertainty $P(\boldsymbol{\theta})$. This also includes plotting of the distributions generated during the MCMC solution discussed in Section 4.4, which provide additional insights

into the model behavior and quality. Further detail on this requirement is possible, including setting convergence criteria for the MCMC algorithms used in Bayesian inference [28, 29], or setting requirements on the uncertainty of the posterior model parameters. In the latter case, failing to meet the uncertainty requirement may result in additional modeling or testing to reduce this source of epistemic uncertainty. However, in this example, it is simply required that Bayesian inference be used to quantify model parameter uncertainty. The fifth and final requirement stated in Table 7.1 is that the combined numerical errors from discretization (ϵ_h), dimension reduction (ϵ_p), and surrogate modeling (ϵ_s) should be less than 5 Kelvin in the predicted output across time and space. These are quantified during the verification and model reduction steps.

7.3.1.2 Conceptual, mathematical, and computational model

As noted above, there are two sets of requirements in the modeling process, when considering VVUQ: modeling requirements and accuracy requirements. The latter were discussed in the previous section, whereas Section 2.1.1 listed modeling requirements. These lead to the development of the *conceptual model*. Section 2.2 demonstrates this process by using a phenomena identification and ranking table (PIRT, Figure 2.4), which organizes the important phenomena identified by SME. Importance is ranked with respect to the selected QoI, which were listed on the right hand side of the table. The conceptual model is one of the most important steps in the definition of the model, since incorrect assumptions lead to *model form error* that may not be revealed until physical testing (usually expensive) has been performed. All subsequent modeling choices are dependent on these initial assumptions. In addition, the SME should consider how the system will operate in both the validation domain and prediction domain. There may be insufficient information for certain aspects of the modeling such that alternative models must be compared during the VVUQ process. This process is referred to as *model selection* [159–161]. Although the process considered here enables model selection, it is beyond the present scope. The conceptual model described in Section 2.2 showed the model geometry and expected environment/boundary conditions around the turbine disc. The operation of the engine was also described, which defines required boundary conditions and time-dependence that are considered in the next step of the model development process. Operation considered in this example is the test cycle (square cycle, Figure 2.3). In general the model will also be run through several flight cycles to predict transient temperatures and stresses during flight, although this is omitted in this discussion.

The conceptual modeling assumptions are then turned into a *mathematical model* using physical laws expressed through differential equations with appropriately specified boundary conditions (e.g., Eqs. 2.1 to 2.4). An example of these equations for the heat transfer model was shown in Section 2.3. The mathematical model includes a number of inputs and parameters that may be unknown or only partially defined from existing 'text book' data. Refinement of these may be part of the test plan, including smaller tests that estimate key model inputs (using uncertainty quantification methods) or model calibration which refines the model parameters using

measurements of the system outputs. Finally, the *computational model* (Section 2.4) is developed by discretization the mathematical model using various numerical methods. For the heat transfer model, the finite element (FE) method [18] is used and is implemented in an in-house code.

7.3.1.3 Uncertainty identification

In the last section, it was mentioned that the mathematical model contains a number of inputs ($\boldsymbol{x} \in \mathbb{R}^{p_x}$) and parameters ($\boldsymbol{\theta} \in \mathbb{R}^{p}$) that the SME may not be able to fully quantify. Thus, they are sources of uncertainty due either to aleatory measurement uncertainty or epistemic uncertainty due to limited measurements, lack of measurements, or are unmeasurable quantities (such as some model parameters). Due to the size and complexity of many models, the total number of inputs and parameters may be in the hundreds or more. This requires SME guidance to narrow down to a more manageable set (see Section 7.3.3.2). Even so, identifying these and gathering estimates of their uncertainty requires substantial effort (Eek [169] documents a real-world example where this is by far the most significant time cost in the VVUQ project). Due to time constraints or limited information for the problem at hand, historical or reference information may be obtained where new measurements cannot be obtained. Since the Bayesian approach is used, these sources of uncertainty are described through probability distributions.

For the heat transfer model example, several sources of uncertainty are listed in Table 7.2; the list is not exhaustive, but is used to illustrate the framework. The first column separates these into inputs $\boldsymbol{x}$ and parameters $\boldsymbol{\theta}$, along with an index for reference. Each row relates to one of the numbered phenomena in the PIRT (Figure 2.4) and provides a description, variable name used in the computational model, the the assumed probability distribution. For inputs, these are propagated through the model. For the parameters, they represent prior distributions which will be refined during calibration. The rightmost column lists an assumed 'true' value for the creation of synthetic measurements, which are defined in Section 2.4.1 and below. In this example, the $p_x = 4$ input distributions are approximated as Gaussian, centered on the mean of the measurements (3 of the 4 are scaled from 0). A description of what these inputs represent is given next, below the table. The $p = 14$ parameters consist of a thermal contact conductance (`fcond`), 11 heat transfer correlation (HTC) factors, a heat generation model factor (`fwdrout`), and a radiation emissivity (`frad`). Since these represent sources of epistemic uncertainty, they are given uniform distributions to reflect the limited knowledge of their values ahead of testing. However, they are also given bounds based on SME experience.

Table 7.2: Source of uncertainty and their characterization.

$\boldsymbol{x}$	Phenom.	Uncertainty source: Inputs	Name	Distribution	$\boldsymbol{x}^*$
x_1	4	Compr. offtake air temp. (drive cone supply)	`fthp3`	$\sim \mathcal{N}(0, 0.01)$	0
x_2	4	Compr. exit, station T30 (drive cone supply)	`ft30`	$\sim \mathcal{N}(1, 0.005)$	1
x_3	4	Compr. inlet, station T26 (bore supply)	`ft26`[2]	$\sim \mathcal{N}(0, 0.005)$	0
x_4	4	Turbine inlet, station T42 (front rim supply)	`ft42`	$\sim \mathcal{N}(0, 0.005)$	0
$\boldsymbol{\theta}$	Phenom.	Uncertainty source: Parameters	Name	Distribution	$\boldsymbol{\theta}^*$
θ_1	2	Thermal conductance (blade to disc)	`fcond`	$\sim \mathcal{U}(0, 5)$	1
θ_2	5	HTC on outer drive cone cavity	`fhdcc`	$\sim \mathcal{U}(0.5, 2)$	1.1
θ_3	5	HTC on inner drive cone cavity flange	`fhflng`	$\sim \mathcal{U}(0.5, 2)$	1.7
θ_4	6	HTC on inner drive cone	`fhcone`	$\sim \mathcal{U}(0.5, 2)$	0.5
θ_5	5	HTC on HPT bore front & drive arm	`fhboref`	$\sim \mathcal{U}(0.5, 2)$	1.9
θ_6	5	HTC on HPT bore ID	`fhbore`[3]	$\sim \mathcal{U}(0.5, 2)$	0.7
θ_7	5	HTC on HPT bore rear	`fhborer`	$\sim \mathcal{U}(0.5, 2)$	1.3
θ_8	5	HTC on HPT diaphragm, front	`fhdiaf`	$\sim \mathcal{U}(0.5, 2)$	0.6
θ_9	5	HTC on HPT diaphragm, rear	`fhdiar`	$\sim \mathcal{U}(0.5, 2)$	1.6
θ_{10}	5	HTC on HPT preswirler exit	`fhprsw`	$\sim \mathcal{U}(0.5, 2)$	1.3
θ_{11}	5	HTC on HPT rim, front	`fhrimf`	$\sim \mathcal{U}(0.5, 2)$	1.9
θ_{12}	7	HTC in disc slot	`fhslot`	$\sim \mathcal{U}(0.5, 2)$	0.5
θ_{13}	9	Air friction heat generation (drive cone)	`fwdrout`[4]	$\sim \mathcal{U}(0.5, 2)$	1.5
θ_{14}	8	Radiation from combustor (emissivity)	`frad`	$\sim \mathcal{U}(0, 0.3)$	0.2

The model inputs are defined directly based on either available measurements or other models. Inputs received from other models will include errors, which fall under epistemic uncertainty. The inputs that are measured may include both aleatory and epistemic uncertainty. In a typical calibration problem, it is often assumed that the inputs are well-characterized through measurements during testing. In some cases, they may only be partially characterized if the measurement capability is limited, and therefore may also include epistemic uncertainty.

In the present application, the selected inputs are considered partially characterized (i.e., they include epistemic and aleatory uncertainty) measurements of air temperature that supply the inner cavity secondary flows. This supply air is bled from the primary cycle air flow at the high-pressure compressor inner flowpath radius, as shown in Figure 7.2, and directed to the inner cavities for cooling purposes. The primary cycle air temperature radial profile has higher temperature at the inner and outer radii due to aerodynamic losses. The radial profile is schematically shown in Figure 7.2 (axes, upper left), along with the measurement locations indicated by dotted horizontal lines five radii. The data is sparse, with respect to resolving the temperature profile which supplies air into the inner cavities, resulting in epistemic uncertainty regarding this corresponding model input. This source of epistemic uncertainty in the temperature boundary condition could be reduced through more advanced measurement methods [170] or estimated [171] and included with the uncertainty sources in the heat transfer model. For the present work, the latter course is taken. This is defined as input `fthp3` in Table 7.2. Three other inputs are similarly defined for the purpose of illustration in this chapter.

[2] In Chapters 4 and 5, a parameter controlling the transient rate of this temperature source is modeled instead (called `ftau`).
[3] `fhcob2` in Chapters 4 and 5
[4] `fwdiaf` in Chapters 4 and 5

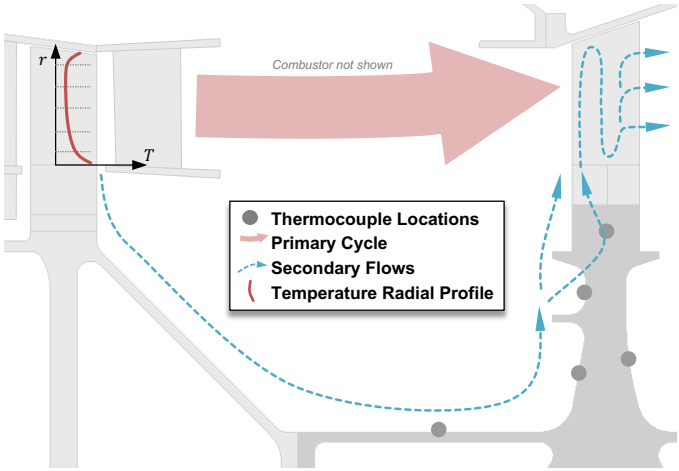


Figure 7.2: Schematic of the compressor exit air temperature profile (upper left) and its sparse measurements which result in epistemic uncertainty in the model input `fthp3`. This serves as a cooling air source for the turbine disc as it is bled off the main gaspath at the inner radius into internal cavities as indicated. Note: not all hardware is shown (i.e., combustor and the static hardware in front of turbine disc).

7.3.1.4 Measurements for calibration and validation

The test plan should be developed as part of the VVUQ requirements for the model. The objectives of testing for the purpose of model calibration and validation may be different to testing that is focused on proving the performance of an engineered product (sometimes referred to as certification testing). Calibration and validation testing involves measurements specific to the model for which the VVUQ process is applied. This usually involves more detailed measurements of the system of interest including the system outputs and inputs and performing various maneuvers that help to examine whether the behavior of the physics model is accurate for its intended use. Furthermore, since calibration adjusts the physics model parameters to gain agreement with the measurements, measurements for the purpose of calibration and validation should be separately obtained. Several possibilities were considered in Section 2.5.

Similar to previous chapters in this dissertation, the measurements for the heat transfer model were synthetically generated from the model and Gaussian noise was added (see Chapter 2.5). For this chapter, two separate data sets were produced for calibration $\boldsymbol{y}_d^C$ and validation[5] $\boldsymbol{y}_d^V$ as follows: (i) Input and parameter 'true' values $\boldsymbol{x}^*$ and $\boldsymbol{\theta}^*$ were defined for calibration and validation data (shown in Table 7.2) , except in the validation data, parameter `fwdrout` is decreased by 10% to represent an engine-to-engine difference relative to the calibration data. (ii) The heat transfer model was run using these two parameter sets and model outputs were extracted at the $n_\ell = 9$ thermocouple positions that are indicated in Figure 7.3. (iii) As noted in Section 2.4.1 and further discussed in Section 7.3.3.1, the output dimensionality was reduced by manually extracting $n_c = 14$ 'characteristic' time instants out of the more than $n_{ts} = 169$ time steps from the transient response of the model (see Figure 7.5).

[5]The nomenclature $\boldsymbol{y}_d^V$ means the replicate samples taken during the validation experiment, whereas $\boldsymbol{D}$ used in Chapter 6 and later in this chapter represents the expanded validation measurements used in the validation metric computation.

(iv) After the outputs are post-processed from the model, zero-mean Gaussian noise with $\sigma_d^* = 2$K on all 126 values is used to draw $N_d = 2$ samples (replicates). These samples represent thermocouples placed at the same location but offset circumferentially by 180°. A sample of the two sets of $n_\ell = 9$ measurements are shown in Figure 7.4 for two time instants in the square cycle (times $t = 2010$ and 4000 seconds). The 18 labels in the figure include the time and location, e.g., tk2010p1 represents $t = 2010$ seconds and location 1. The temperature values are plotted as circles corresponding to the left-hand axis, whereas the difference between the two measurements is plotted as a triangle corresponding to the right-hand axis.

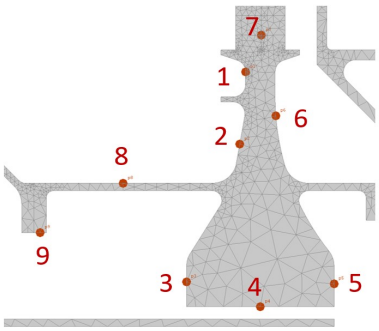


Figure 7.3: Turbine disc component model showing thermocouple positions[6]

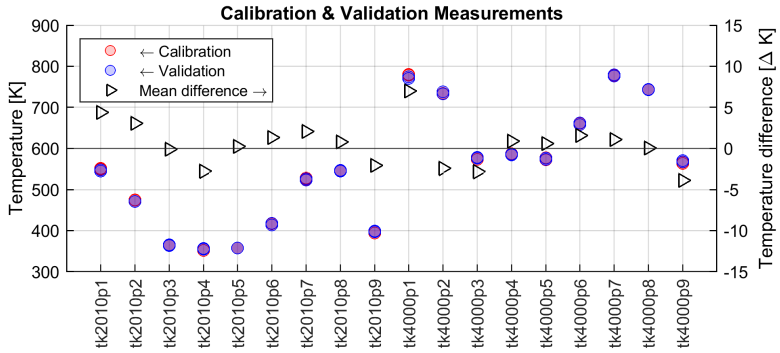


Figure 7.4: Temperature measurements at the $n_\ell = 9$ thermocouple locations (Figure 7.3) for time instants $t = 2010$ and 4000 seconds. The left-hand axis indicates measurement temperatures; the right-hand axis corresponds to the triangles, which are the average difference between the calibration and validation measurements.

7.3.2 Verification

This section reviews the main points of Chapter 3 and shows how the discretization error estimation method is incorporated into the uncertainty aggregation framework. It was stated in Section 1.2 that verification includes two parts: code verification and solution verification. Code verification is assumed to be completed, since the present study uses an established FE tool. For the solution verification of the present application, discretization error is the focus since it is assumed that round-off and truncation errors are relatively low, as was done in [3] (for existing

[6]These positions were inadvertently numbered in a different order relative to the examples in Chapter 5 and 6. Since there is no intention of cross-referencing results, the figures in this chapter were not revised.

codes, checking these may be difficult and is therefore beyond the scope of the present work). Discretization error is defined as the difference between the exact solution f^* and the approximate solution, $f^* - f_k$ at mesh level k. In some previous works, the discretization error is treated deterministically and the estimate of $\hat{f} \approx f^*$ is used to correct the model outputs before performing subsequent steps of VVUQ (e.g., calibration) [172]. Others suggest that, due to limitations in available discretization error estimation methods, the discretization error should be treated as a source of epistemic uncertainty [3, 52]. In this chapter, the latter view is taken.

The prevailing method in the VVUQ literature for estimating discretization error is Richardson extrapolation (RE) [2, 42, 48, 49] (Section 3.1.1). However, RE-based methods rely on the assumption of *systematic mesh refinement*, which requires uniform refinement (it does not require an uniform mesh) and "consistent quality" across the mesh [2]. However, since the application in this study's FE tool uses adaptive meshing and time-stepping, traditional RE is not possible.

The FE tool's adaptive mesh routine is based on iteratively reducing local solution error through a recovery-based error estimator [42, 43, 45]. However, it has been pointed out [16, 46] that the estimator's magnitude may not be useful as an indicator of errors for the purpose of UQ.

Therefore, Chapter 3 extended the GP approach proposed in [54] to the case of adaptively refined mesh and time step. GP models were fit to the solutions of five meshes and an error estimator $\boldsymbol{\epsilon}_{hg}$ called GPDE (GP discretization estimate) was developed that includes (i) the GP uncertainty, (ii) the bias between the chosen solution k and the exact solution estimate, and (iii) the additional uncertainty in the discretization error estimate due to input/parameter variability. The end result is an estimate for discretization error, $\boldsymbol{\epsilon}_h$, which is incorporated into the uncertainty aggregation process in the next subsection.

7.3.2.1 Incorporating discretization errors in the VVUQ framework

The results shown in Chapter 3.3 (Table 3.4) provide an estimate for discretization error $\boldsymbol{\epsilon}_h$ at time $t = 2010$ seconds and for each of the measurement locations. Results could be obtained at all time points, if required, by repeating the process for additional model outputs. However, to simplify the application of this error and apply it in a conservative manner, results at $t = 2010$ seconds are applied to all times through the square cycle. The discretization error estimate is included in this chapter during the subsequent VVUQ framework steps of calibration and validation by randomly sampling the normal distribution defined in Eq. 3.9.

An additional step is required to include discretization errors during prediction, which involves a computed quantity $T_{rm} = T_7 - \bar{T}$, where T_7 is temperature at the rim location (output at P7) and $\bar{T}$ is a volume-weighted-mean temperature of the disc. To estimate an approximate discretization error for $\bar{T}$, the discretization error at the n_ℓ locations is averaged, i.e., $\bar{\sigma}_h = \frac{1}{n_\ell} \sum_i \sigma_{h,i}$. Then, since the quantity represents the difference between two random variables, a further approximation is to set their combined uncertainty to $\sqrt{2}\bar{\sigma}_h$ (this assumes the rim

error is similar to the average).

Finally, the discretization errors defined above will be combined with the model reduction errors in the next section (Section 7.3.3.4) for evaluation against Requirement 5 in Table 7.1.

7.3.3 Model Reduction

Complex engineering models are often slow running, have many input/parameter sources of uncertainty, and produce a large number of outputs. To make VVUQ analysis feasible, model reduction methods are typically required. Those employed in this chapter include a two-step output dimension reduction: (1) manual 'feature selection' to reduce the size of the model output time series and (2) PCA for further output dimension reduction (Section 7.3.3.1). Then, global sensitivity analysis (GSA) is used for input/parameter dimension reduction (Section 7.3.3.2), and a surrogate model is used to speed up the MCMC solution during Bayesian inference (Section 7.3.3.3). Since these are used within the VVUQ framework, the additional sources of uncertainty should be included in the aggregation process, which is demonstrated below.

7.3.3.1 Model output dimension reduction

The number of outputs from the physics model may be large, considering that results from each node of the FE solution may be of interest. In this application, the primary interest is in calibration and validation at the $n_\ell = 9$ thermocouple locations. It was previously shown in Section 2.4.1 how the time dependent outputs from the model were reduced through feature selection from n_{ts} time steps to n_c 'characteristic' quantities. This takes advantage of the autocorrelation between successive times in the physical output [70]. Two alternative approaches for characterizing the transient response were considered in Section 2.4.1: either specific time instants or derived time constants. The first was used in the applications in Chapters 4 and 5.

In this chapter, the manual time instants feature selection approach is used, in which case $n_c = 14$ time instants were chosen based on inspection of the time traces of temperature error in Figure 7.5 (the time instants are shown as vertical lines through the acceleration and deceleration portions of the time traces, which are shown magnified in the lower part of the figure). The time instants include the two stabilized temperature time points ($t = 2000$ and $t = 4000$ seconds for idle and MTO, respectively) to characterize the steady-state outputs and an additional 12 time instants are selected to characterize the transient response of the outputs (heating and cooling rates). Thus, the multivariate model output now has size $n = n_\ell \times n_c = 126$. After performing a DOE with $N = 450$ runs, the model output dataset size is $\boldsymbol{y} \in R^{N \times n}$.

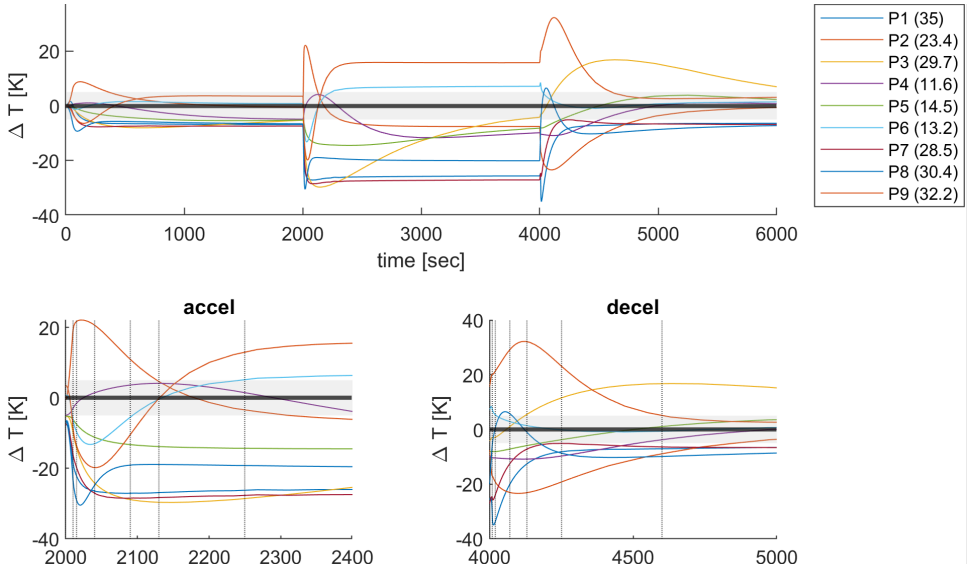


Figure 7.5: Model output to measurement difference (transient) for each of the 9 thermocouple locations. Top: full square cycle. Bottom: zoom in to the transient acceleration and deceleration, including the 14 time instants used for surrogate modeling, calibration, and validation.

Despite the initial reduction of the model output dataset to size $n = 126$, there is motivation for further reduction: (i) there is (still) redundant information in these outputs, since they are correlated; (ii) surrogate models will be constructed later and it is desirable to minimize the number of surrogate models that must be fit; (iii) sensitivity analysis will be performed to reduce the input dimension and the results may be uninterpretable for a large number of outputs. Principal components analysis (PCA), introduced in Section 4.3.1, is therefore applied to the $\boldsymbol{y} \in \mathbb{R}^{N \times n}$ sized dataset.

In Section 4.3.1 it was shown that errors in the reconstructed outputs $\hat{\boldsymbol{y}}_0 = \hat{\boldsymbol{y}}^{pc} U_1^T$ (where the retained and truncated eigenvectors are represented by $U_1 \in \mathbb{R}^{n_{pc} \times n_{pc}}$ and U_2, respectively) are calculated as

$$\boldsymbol{y}_0 - \hat{\boldsymbol{y}}_0 \odot S_y \tag{7.1}$$

where $S_y = [\boldsymbol{\sigma}'_{y1}, \boldsymbol{\sigma}'_{y2}, \ldots, \boldsymbol{\sigma}'_{yn}]$ is a matrix with N rows that each contain a copy of the standard deviations of the original outputs (to 'undo' the standardization), and $\odot$ is element-wise multiplication. These errors are plotted for the present analysis in Figure 7.6, in which case the maximum error was found to be < 2 K if 30 PCs were retained and 7-8 K if 15 PCs were retained (see further discussion on this truncation error when fitting the surrogate model, Section 7.3.3.3). In this study, $n_{pc} = 15$ was chosen, since it was found to be a good trade-off between PCA truncation error and surrogate model error (which increases for smaller PCs), i.e., the outputs for $n_{pc} > 15$ results in poor surrogate model fits due to the magnitude of the smaller PCs.

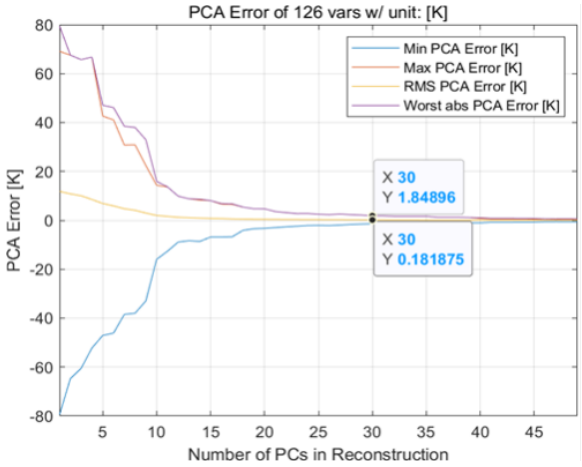


Figure 7.6: PCA reconstruction errors as a function of the number of retained PCs (n_{pc}) for the first 50 PCs. The callouts indicate the maximum (1.84 K) and root-mean-square (0.18 K) errors for $n_{pc} = 30$.

As pointed out in [109], Eq. 7.1 above does not reflect the entire truncation error since any combination of the truncated PCs may be included during reconstruction (setting them equal to zero as in Eq. 7.1 was only a convenient choice). A more complete error estimate therefore includes randomly sampled truncated PCs [109]

$$\epsilon_p = \left[\boldsymbol{y}_0 - (\hat{\boldsymbol{y}}^{pc} U_1^T + \Phi U_2^T)\right] \odot S_y \tag{7.2}$$

where $\Phi \in \mathbb{R}^{N \times (n - n_{pc})}$ is a zero-mean i.i.d. multivariate Gaussian with variances corresponding to the inactive PCs. In the VVUQ framework pursued in this chapter, this computed PCA truncation error is included when making model predictions for the validation and prediction steps which use the PC-based surrogate models. However, since the same transformation is applied to the measurements during calibration in this research, the PCA truncation errors are not required at this step of the framework (Section 7.3.4; this approach was also taken for calibration in the PC space in [35, 36]).

7.3.3.2 Model inputs and parameters dimension reduction

Physics models often contain dozens if not hundreds of model inputs and parameters. It is typical that the SME will use a combination of engineering judgment and one-factor-at-a-time studies to make a 'first cut' at the set of inputs/parameters that will be considered in the VVUQ analysis. Further reduction may be possible through applying methods such as global sensitivity analysis (GSA), which assesses the importance of each input on the output. A specific example of GSA is the Sobol' index [173]). Other approaches to handling input dimension reduction include PCA (if there are many inputs with correlation), or active subspace [65], which derives a reduced set of transformed parameters from the eigenspace of the model gradients in which the new parameters are linear combinations of the original parameters. In the present application, the list of selected inputs and parameters

(variables) were shown in Table 7.2, representing the SME's first cut set of variables. The efficient first-order Sobol' index method [87] that was used in previous chapters was again applied to this chapter in order to reduce the number of variables (and rank the importance of the variables). As previously shown, the method is efficient since it is possible to compute the Sobol' indices directly from the set of DOE results.

In this chapter, the indices were computed for each output, then the maximum index for each parameter across all outputs is selected. These maximum indices are then sorted into a Pareto plot, as shown in Figure 7.7. This result then indicates the relative importance across all model outputs. Similar to the indices in Section 5.4.2, the error due to computing the Sobol' indices using a limited number of DOE samples[7] was estimated with bootstrapping[125]. The one standard deviation errors are included as whiskers on the bars in Figure 7.7.

In the present example, the last three variables highlighted in Figure 7.7 (`fcond`, `ft42`, and `fhrimf`) have small Sobol' indices and are therefore deemed non-influential to the model outputs. These three variables are set to their original nominal values in the remainder of the VVUQ framework steps.

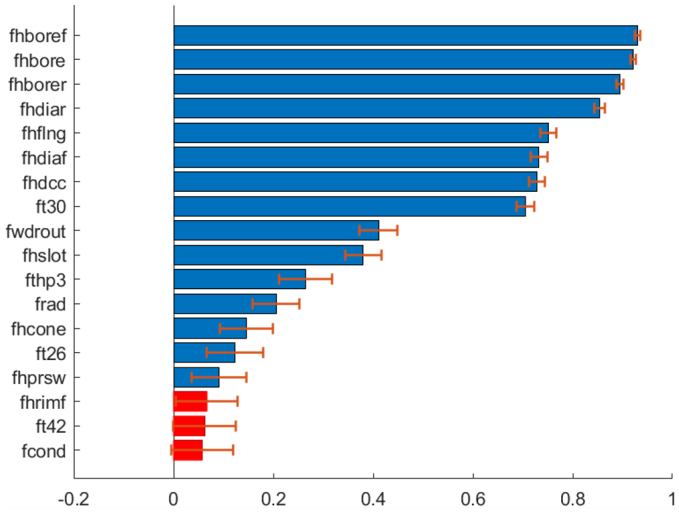


Figure 7.7: Pareto ordering of maximum Sobol' indices across all outputs. The three red bars indicate variables that are removed from further analysis (model calibration).

7.3.3.3 Surrogate model errors

As mentioned in Chapters 1, 2, and 4, it is common in engineering to replace slow-running physics models with a fast surrogate model [32] in order to speed up optimization or sampling-based UQ methods. Many types of surrogate models are possible, including classic linear regression and response surface models (RSM) [39], or machine learning models such as Gaussian process models [55], neural networks [175], support vector regression [90], or the PC-AS surrogate method (developed as part of the research in Chapter 4). For the present application,

[7]There is also error in the estimate due to model accuracy [174], which is not addressed in this research.

the RSM was selected for convenience[8] and satisfactory accuracy. It was implemented using the MATLAB® function `stepwiselm` with term selection based on Bayesian Information Criterion (BIC) [177]. These are well-established techniques that are covered in the above references and many others.

In this research, the dataset to which the surrogate will be fit represents rows corresponding to model runs from the DOE containing the input/parameter samples and corresponding outputs, where these outputs have been first transformed using PCA (Section 7.3.3.1). Since the surrogate models are typically univariate (in the output), a separate surrogate model is fit separately to each uncorrelated output (PC) using the same set of inputs/parameters. A surrogate model type is selected, fit to a portion of the dataset (i.e., trained on a subset of the model runs), and then cross-validated using the remainder of the dataset [39].

A common error estimator used with cross-validation is the mean-square error (MSE). Considering the present application, the MSE is computed for the error between $\hat{y}_{ij}^{pc}$ and the surrogate prediction $\hat{\mathcal{Y}}_{ij}^{pc}$,

$$MSE_i = \frac{1}{N'} \sum_{j=1}^{N'} (\hat{y}_{ij}^{pc} - \hat{\mathcal{Y}}_{ij}^{pc}) \tag{7.3}$$

where this is computed for each PC and each sum is over the subset of N' model runs used for testing the surrogate model (i.e., not used in training). In cross-validation, the MSE is computed several (e.g., 10) times and the average result is reported, which is denoted here as Σ_s. In the VVUQ framework, this epistemic uncertainty is treated as a zero-mean i.i.d. multivariate Gaussian,

$$\epsilon_s \sim \mathcal{N}(\mathbf{0}, \Sigma_s) \tag{7.4}$$

This source of uncertainty is then sampled during model calibration, model validation, and predictions made using the surrogate models (along with but independent to PCA truncation errors as noted in Section 7.3.3.1).

7.3.3.4 Aggregated numerical errors versus requirements

It is also possible to address the correlation between the discretization error, dimension reduction, and surrogate modeling errors derived above through an extensive UQ exercise. For example, the influence of model inputs/-parameters on the discretization error results could be incorporated by running each of the mesh levels through the DOE and computing the uncertainty in ϵ_h for the entire DOE set. Then, the influence of discretization error on PCA and surrogate modeling could be derived by repeating the model reduction steps on multiple levels of mesh refinement. Further, the impact of dimension reduction truncation errors could be related to the surrogate model error by re-fitting the surrogate models for different values of n_{pc} (number of PCs selected). The use of a Bayesian network could also be used to further aggregate these errors [5] and could be considered in future work.

[8] Other methods would work even better, but an in-house implementation along with PyMC [176] was available to the author, which afforded significant time-savings for the Bayesian inference with input sampling (Section 7.3.4.4).

In practical VVUQ applications, these additional steps add significant computational overhead and may not be possible (or worthwhile, unless these represent substantial sources of uncertainty). Therefore, in the present work these sources are treated independently and combined as indicated in Requirement 5 of Table 7.1, which states that the numerical errors $\sqrt{\sigma_h^2 + \sigma_p^2 + \sigma_s^2} \leq 5$ Kelvin. Across all outputs and time points assessed, the maximum numerical error was observed to be 2.8 K. Therefore, this numerical error requirement has been satisfactorily achieved. If errors were too high, a finer mesh could be used and a higher accuracy surrogate model method could be used (with added computational cost).

7.3.4 Calibration

Bayesian inference is used in this framework for parameter estimation (calibration) as it provides a natural way to aggregate the various sources of uncertainty identified in previous steps of the framework [5, 60] and account for parameter uncertainty. Bayesian inference and its solution by MCMC methods was introduced in Section 4.4. The same approach is applied in this chapter (Section 7.3.4.1). Several cases are defined to demonstrate the use of Bayesian calibration within the framework (Sections 7.3.4.2 to 7.3.4.4). To conclude, a summary of the relative magnitude of aggregated uncertainty sources is shown in the final subsection (Section 7.3.4.5), which is included when performing the validation step (Section 7.3.5). In this chapter, MCMC is implemented with the No-U-Turn Sampler (NUTS) available in PyMC [176], which is substantially faster than the Metropolis-Hastings method that was used in earlier chapters due to the former taking advantage of gradient information from the likelihood [176] and self-tuning [176].

7.3.4.1 Aggregation through Bayesian inference

For the purpose of uncertainty aggregation, the sources identified up to this step in the process are included during Bayesian calibration so the parameter posterior distributions reflect the uncertainty. It was stated that the goal of calibration is to update the model's parameters to improve the agreement between the model outputs and the measurements, which was expressed

$$\boldsymbol{g}(\boldsymbol{x}, \boldsymbol{\theta}) + \boldsymbol{\epsilon}(\boldsymbol{x}, \boldsymbol{\theta}) = \boldsymbol{g}_d(\boldsymbol{x}) + \boldsymbol{\epsilon}_d(\boldsymbol{x}) \tag{7.5}$$

As in previous chapters, the measurement uncertainty $\boldsymbol{\epsilon}_d(\boldsymbol{x})$, assumed to be zero-mean Gaussian with covariance matrix Σ_d. Rather than the model discrepancy approach $\boldsymbol{\epsilon}_\delta$ presented in Chapter 5, this model error is instead quantified during validation using the model reliability metric from Chapter 6 and following the approach in [5]. The remaining sources of model uncertainty addressed in this chapter include the numerical errors quantified in the previous section, that is $\boldsymbol{\epsilon}(\boldsymbol{x}, \boldsymbol{\theta}) = \boldsymbol{\epsilon}_h + \boldsymbol{\epsilon}_s + \boldsymbol{\epsilon}_p$. The discretization errors are modeled as Gaussian processes.

The linear regression-based surrogate model errors are Gaussian by definition (during the fitting process and by use of MSE). The PCA truncation errors are not included during the calibration step, but will be included during the validation and prediction steps using the re-sampling in Eq. 7.2.

Thus, $\boldsymbol{\epsilon}_h$ and $\boldsymbol{\epsilon}_s$ are included in the definition of the likelihood function (Eq. 4.13) along with measurement errors for a combined covariance matrix $\Sigma = \Sigma_d + \Sigma_h + \Sigma_s$. For calibration in the PC space, it was shown that this matrix is transformed according to $\Sigma^{pc} = U_1^T \left[\text{diag}(\boldsymbol{\sigma}_y^{-2})\right] U_1$, where the diagonal matrix $\text{diag}(\boldsymbol{\sigma}_y^{-2})$ standardizes the variance as before. If there is a strong reason to suspect any of these sources of uncertainty are non-Gaussian, the likelihood function could be modified accordingly, e.g., [60]. Finally, input uncertainty is included during evaluation of the likelihood function, which includes an evaluation of the model $\boldsymbol{y}^{pc}$. Treatment of input uncertainty is discussed further in Section 7.3.4.4.

7.3.4.2 Model cases for calibration and validation

To highlight different aspects of model calibration and validation of the heat transfer model within the VVUQ framework, a series of model cases are defined in Table 7.3. First, Case M4 is compared to M1 to demonstrate the impact of verification (discretization errors) on calibration results (Section 7.3.4.3). Next, calibration in the presence of input uncertainty is demonstrated by comparing M2 to M3 (Section 7.3.4.4). In M2, the inputs are fixed during calibration and then the input uncertainty is sampled independently from the calibrated parameters during propagation (M2). By contrast, M3 performs model calibration while simultaneously sampling from the input distributions, resulting in correlation between parameters and inputs. The M3 results are then used as the primary case for further analysis in the framework.

Table 7.3: Model configurations used to demonstrate calibration and validation of the heat transfer model.

Case	Input uncertainty	Other modification	Comparison
M1	fixed	–	M4
M2	random, independent sampling	–	M3
M3*	random, correlated sampling	–	M2 (Sec. 7.3.4.4)
M4	fixed	no discretization error ($\epsilon_h = 0$)	M1 (Sec. 7.3.4.3)

*M3 is the primary case used in subsequent analyses.

7.3.4.3 The impact of discretization errors on Bayesian inference

In this subsection, results are shown for Bayesian calibration with and without the discretization errors from Section 7.3.2. The calibration is performed for the 12 parameters that were down-selected using the Sobol' indices (Figure 7.7), while the 3 inputs were fixed. Figure 7.8 and Table 7.4 show results for the marginal posterior parameter distributions and the change in their standard deviations, respectively. Table 7.5 shows the change in marginal posterior prediction standard deviations for a subset of the model outputs, i.e., all 9 locations at three

representative times. The posterior predictions were generated by propagating the parameter posteriors through the surrogate model. The impact to parameter standard deviation was up to 24% (`fhdiar`), while the impact on model output standard deviation was even more significant. This highlights the importance of systematically tracking the uncertainty aggregation from step to step in the VVUQ framework.

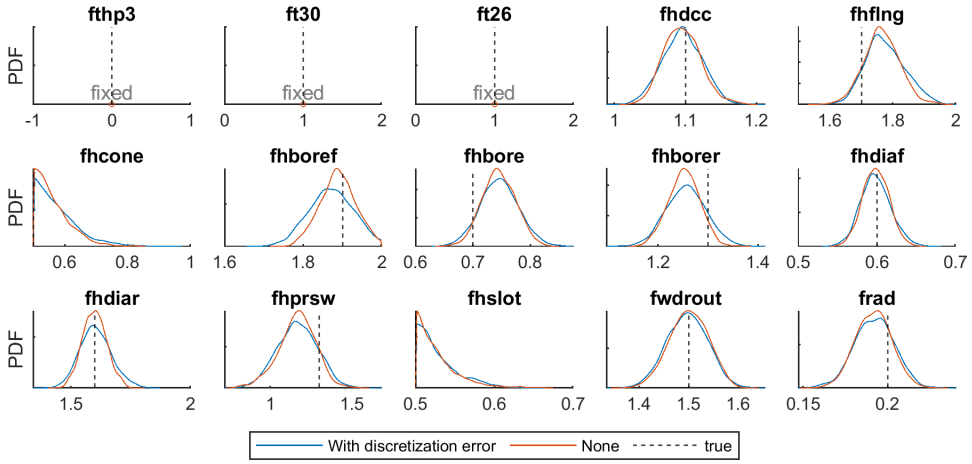


Figure 7.8: Parameter posterior marginal distributions with (M1) and without (M4) discretization errors. Inputs `fthp3`, `ft30`, and `ft26` are fixed. Corresponding standard deviations are listed in Table 7.4.

Table 7.4: Parameter posterior marginal distribution standard deviations with (M1, $\epsilon_h \neq 0$) and without (M4, $\epsilon_h = 0$) discretization errors.

θ_i	2	3	4	5	6	7	8	9	10	12	13	14
Std. Dev.												
M1, $\epsilon_h \neq 0$	0.027	0.062	0.065	0.055	0.033	0.039	0.019	0.065	0.121	0.026	0.04	0.012
M4, $\epsilon_h = 0$	0.026	0.056	0.053	0.044	0.029	0.031	0.016	0.05	0.113	0.026	0.038	0.011
$\Delta\%$	-6.1	-9.3	-18.6	-19.2	-13.4	-21.4	-13.1	-23.7	-6.7	2.2	-5	-9.3

2:`fhdcc` 3:`fhflng` 4:`fhcone` 5:`fhboref` 6:`fhbore` 7:`fhborer` 8:`fhdiaf` 9:`fhdiar` 10:`fhprsw` 12:`fhslot` 13:`fwdrout` 14:`frad`

Table 7.5: Posterior prediction marginal distribution standard deviations with (M1, $\epsilon_h \neq 0$) and without (M4, $\epsilon_h = 0$) discretization errors. Results are given at three key time points: transient acceleration ($t = 2010$ sec), stabilized high power ($t = 4000$ sec), and transient deceleration ($t = 4250$ sec).

		Output Location								
Std. Dev.		1	2	3	4	5	6	7	8	9
$t = 2010s$	M1, $\epsilon_h \neq 0$	3.02	2.49	2.65	1.22	2.22	2.51	2.99	2.74	1.13
	M4, $\epsilon_h = 0$	2.74	2.45	0.41	0.35	0.4	0.77	2.39	2.36	0.95
	$\Delta\%$	-9.3	-1.7	-84.3	-71.6	-81.8	-69.4	-20.2	-13.9	-15.6
$t = 4000s$	M1, $\epsilon_h \neq 0$	1.62	1.85	2.75	1.31	2.4	3.02	2.37	2.13	1.48
	M4, $\epsilon_h = 0$	1.27	1.71	0.64	0.55	0.71	1.65	1.5	1.67	1.33
	$\Delta\%$	-21.3	-7.8	-76.9	-58.2	-70.7	-45.3	-36.8	-21.7	-9.6
$t = 4250s$	M1, $\epsilon_h \neq 0$	3.37	4.1	4.59	3.94	4.33	4.53	3.46	3.51	3.53
	M4, $\epsilon_h = 0$	3.26	4.08	3.78	3.75	3.68	3.79	3.2	3.41	3.47
	$\Delta\%$	-3.1	-0.5	-17.6	-4.8	-14.9	-16.4	-7.5	-3	-1.5

7.3.4.4 The impact of input uncertainty on Bayesian inference

In Section 2.3.1 it was mentioned that some inputs may be only partially characterized and that this epistemic uncertainty should be addressed during model calibration. It is important to incorporate this input uncertainty during model calibration to ensure that any correlation between inputs and parameters is properly accounted for. For example, if a function has two inputs, say $Y = f(X_1, X_2) = aX_1 + bX_2$ and X_1 and X_2 are uncorrelated, then $\text{Var}[Y] = a^2\sigma_{X1}^2 + b^2\sigma_{X2}^2$. However, if the inputs are correlated, then $\text{Var}[Y] = a^2\sigma_{X1}^2 + b^2\sigma_{X2}^2 + 2ab\text{Cov}(X_1, X_2)$ [74] and there is the possibility that the variance is reduced by the additional covariance term if either a, b, or $\text{Cov}(X_1, X_2) < 0$ (or increased for positive correlation). If this correlation exists in a physics model's inputs but is neglected, the model prediction uncertainty may be either more pessimistic or more optimistic than in reality. Therefore, this section demonstrates an approach to address this dependence during calibration.

The input distributions $\pi(\boldsymbol{x})$ and priors $P(\boldsymbol{\theta})$ for the heat transfer model were defined in Table 7.2. The sensitivity analysis (Figure 7.7) resulted in selecting 3 inputs and 12 parameters. Heat transfer model cases M2 and M3 were defined in Table 7.3 to compare uncorrelated and correlated sampling approaches, respectively. For model M2, the 12 parameters were calibrated with the inputs fixed at their nominal values (using 2000 MCMC samples); then, the parameter posteriors and input distributions were both propagated through the model, with inputs sampled independently from the parameters. For M3, 120 input samples where drawn from the joint distribution $\pi(\boldsymbol{x})$ of inputs `fthp3`, `ft30`, `ft26`. Then, for each of these input samples, the model parameters were calibrated with 1000 MCMC samples. This 'double-loop calibration' estimates the correlation between inputs and parameters, as can be seen in Figure 7.9, where the scatterplots show the correlation between the input `ft30` and parameter `frad`. Another interesting observation is that the correlation between the parameters is also affected, e.g., the correlation between `fwdrout` and `frad` changes from negative to positive. These two particular parameters also showed the largest increase in the marginal standard deviations, which are listed in Table 7.6.

The posteriors and inputs of both M2 and M3 were propagated to obtain posterior predictions at each thermocouple location. The standard deviations of these results are compared in Table 7.7, showing the expected reduction (up to 50% in some locations) in posterior prediction uncertainty when the correlation between inputs and parameters is correctly accounted for. The aggregated sources of uncertainty for M3 are summarized in the next section (Section 7.3.4.5) and will be used next for model validation (Section 7.3.5).

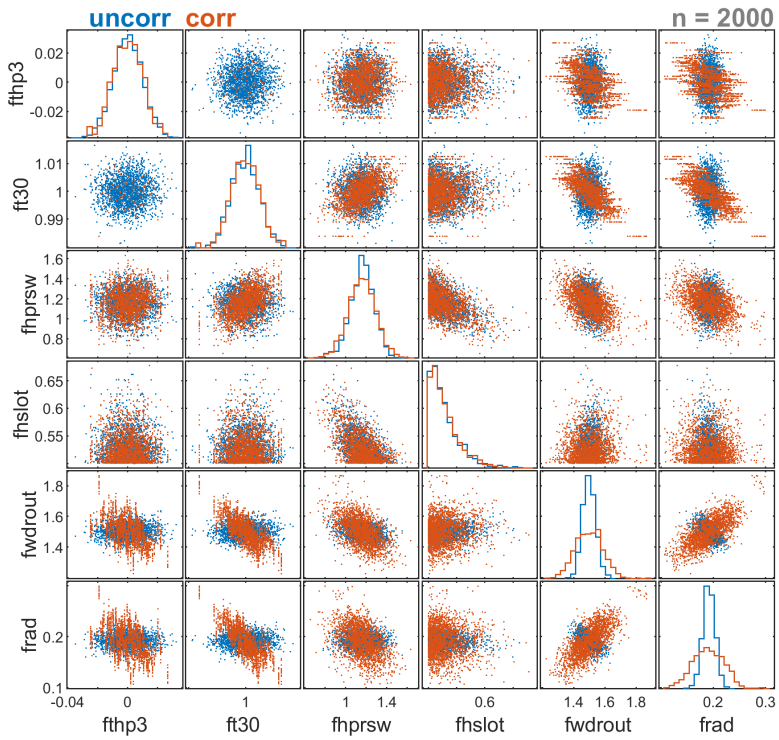


Figure 7.9: Scatterplots comparing calibration results with uncorrelated (M2) and correlated (M3) sampling. Only variable pairs that showed significant correlation are shown. Marginal distributions are shown on the diagonal and their standard deviations are listed in Table 7.6. A subset of M3 samples (2000 out of 120,000) are shown.

Table 7.6: Parameter posterior marginal distribution mean and standard deviations for calibration with uncorrelated (M2) and correlated (M3) inputs. Both results have very similar means, so only the M3 result is shown.

θ_i	2	3	4	5	6	7	8	9	10	12	13	14
Mean	1.1	1.78	0.58	1.87	0.75	1.26	0.6	1.61	1.16	0.53	1.5	0.19
Std. Dev.												
M2, uncorr.	0.026	0.063	0.064	0.06	0.033	0.04	0.018	0.063	0.118	0.026	0.042	0.012
M3, corr.	0.029	0.062	0.068	0.058	0.033	0.04	0.018	0.067	0.138	0.027	0.091	0.031
$\Delta\%$	7.7	-0.9	6.6	-3.5	1	0.6	0.9	7.4	16.9	1.4	113.3	159.2

2:`fhdcc` 3:`fhflng` 4:`fhcone` 5:`fhboref` 6:`fhbore` 7:`fhborer` 8:`fhdiaf` 9:`fhdiar` 10:`fhprsw` 12:`fhslot` 13:`fwdrout` 14:`frad`

Table 7.7: Posterior prediction marginal distribution standard deviations for calibration with uncorrelated (M2) and correlated (M3) inputs.

		Output Location								
Std. Dev.		1	2	3	4	5	6	7	8	9
$t = 2010s$	M2, uncorrelated	3.66	3.15	2.74	1.32	2.33	2.62	3.86	3.7	1.26
	M2, correlated	2.91	2.52	2.68	1.23	2.28	2.55	3.02	2.85	1.14
	$\Delta\%$	-20.4	-19.9	-2.1	-6.6	-2.4	-2.9	-21.7	-22.9	-10.1
$t = 4000s$	M2, uncorrelated	3.93	3.75	2.9	1.69	2.48	3.13	4.23	4.2	1.8
	M3, correlated	1.68	1.86	2.74	1.32	2.36	2.99	2.41	2.26	1.49
	$\Delta\%$	-57.3	-50.5	-5.5	-21.8	-4.8	-4.4	-43	-46.3	-17
$t = 4250s$	M2, uncorrelated	2.49	2.83	3.02	1.86	2.74	3.19	2.65	2.78	1.9
	M3, correlated	1.52	2.02	2.91	1.42	2.52	2.83	2.01	2.21	1.71
	$\Delta\%$	-38.7	-28.4	-3.7	-23.4	-8.2	-11.2	-24.1	-20.4	-10.1

7.3.4.5 A summary of aggregated uncertainty sources

At this stage in the VVUQ study, many sources of uncertainty have been aggregated and the impact on the model output is shown in Figure 7.10 for model M3. The figure shows the impact of a given uncertainty source (groups) on each of the nine outputs (bars) at time $t = 2010$ seconds. The sources of uncertainty include the inputs sampled during calibration, parameters estimated through Bayesian inference, and numerical errors estimated during verification (discretization) and model reduction (PC truncation and surrogate model errors). Measurement uncertainty influences parameter uncertainty through the likelihood during calibration and it is included during the validation assessment later (in Section 7.3.5.1).

The results in Figure 7.10 also provide insight for model improvements. The most significant uncertainty is around the drive cone cavity and rim region of the turbine disc model (locations 1, 2, 7, and 8). This observation may lead the analyst to consider ways to reduce the sources of epistemic uncertainty in this region either through more detailed modeling (CFD), through refinement of the surrogate model, a reduction of discretization error using a finer mesh (potentially resulting in higher computational cost), or obtaining additional sensor information (which may reduce model calibration parameter uncertainty).

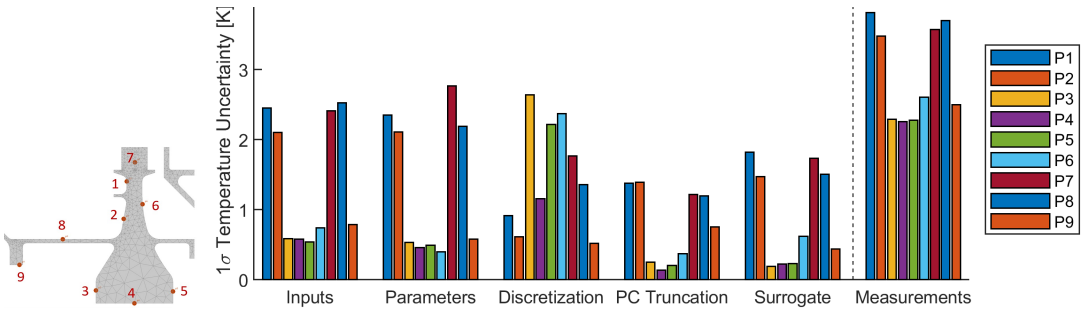


Figure 7.10: Sources of uncertainty aggregated through the VVUQ framework for the nine thermocouple locations at $t = 2010$ seconds (model M3).

7.3.5 Validation

The process of model validation was considered in detail in the previous chapter (Chapter 6), and more specifically, the *validation assessment* compared model predictions to measurement distributions using mathematical *validation metrics*. In this chapter, it has been shown how the sources of uncertainty are quantified through the first four steps of the VVUQ process (model definition, verification, model reduction, and calibration) for the model prediction, $\boldsymbol{y}$. These predictions are at the validation test measurement locations (Figure 7.3) and are based on the tested conditions (e.g., the square cycle and associated inputs, Figure 2.3). In the next section, the predictions discussed are for the primary QoI, which is different from the measured locations in the test specimin. Additionally, uncertainty propagation may be based on the surrogate model or the physics model. For the validation step in this chapter, the surrogate model is used to speed up the sampling process for the comparison of model

output and measurement distributions (the additional surrogate model uncertainty is included during uncertainty propagation). To verify this approach, the propagation by the surrogate model is compared to propagation using the physics model in Section 7.3.6 which showed good agreement between the two approaches.

This section briefly summarizes the approach outlined in Chapter 6 and applies it to the heat transfer model outputs in this chapter (model M3 defined in Table 7.3), which include propagation of the uncertainty from previous steps of the VVUQ framework. Then, the validation metric result is used to expand the posterior parameter uncertainty (Section 7.3.5.3) for prediction of the QoI.

7.3.5.1 Model predictions and measurements for validation

After model calibration is performed, the sources of uncertainty discussed in Section 7.3.4.5 are propagated through the model at measurement locations producing model output distribution $\boldsymbol{y}$. The uncertainty propagation is performed by drawing samples from the input and parameter joint distributions (obtained from MCMC samples), obtaining an output from the surrogate model, then further sampling from the numerical errors ($\boldsymbol{\epsilon}_h$, $\boldsymbol{\epsilon}_p$, and $\boldsymbol{\epsilon}_s$) which are treated as zero-mean Gaussian sources of uncertainty. Finally, as described in Section 6.3, for the purpose of the validation assessment, $\boldsymbol{y}$ is expanded to obtain the *posterior predictive distribution*, i.e., $\boldsymbol{z} \sim \boldsymbol{y} + \mathcal{N}(\boldsymbol{0}, \Sigma_d)$, where $\Sigma_d = diag(\sigma_d^2)$. Including the measurement uncertainty in this way addresses numerical issues (due to strong correlation between outputs during inversion of the covariance matrix) and addresses potential inconsistency in the comparison of distributions for limited measurement samples (Section 6.3.1). Figure 7.11 compares the distributions of $\boldsymbol{y}$ and $\boldsymbol{z}$ for selected model outputs.

Also shown in Figure 7.11 are the corresponding calibration and validation measurements. Although there are only $N_d = 2$ replicate validation measurements $\boldsymbol{y}_d^V$ shown in this figure, an estimated measurement uncertainty σ_d is also available (see Section 2.5). This information is used to construct a measurement distribution for the validation assessment by computing the mean $\bar{\boldsymbol{y}}_d^V$ and assuming zero-mean Gaussian measurement error and sampling error of the mean, resulting in $\boldsymbol{D} \sim \mathcal{N}(\bar{\boldsymbol{y}}_d^V, \Sigma_D)$, where $\Sigma_D = diag(\sigma_d^2/N_d + \sigma_d^2)$. The nomenclature $\boldsymbol{D}$ is therefore used for the validation assessments in the next section for the measurements rather than $\boldsymbol{y}_d^V$.

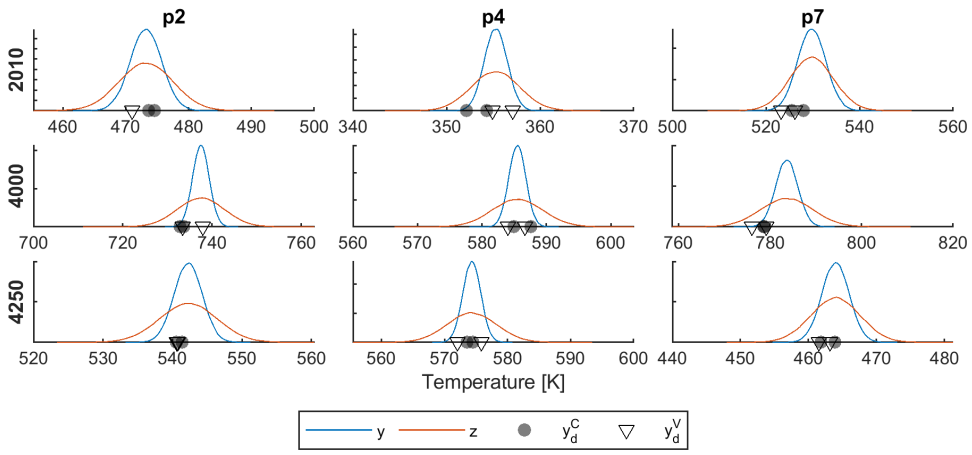


Figure 7.11: Comparison of posterior prediction distributions of $\boldsymbol{y}$ and $\boldsymbol{z}$ which are based on propagating posteriors $P(\boldsymbol{\theta}|\boldsymbol{y}_d^C)$ from model M3. This subset of results includes three model output locations: 2 (front diaphragm), 4 (bore), and 7 (rim); and three time instants: 2010 sec (transient acceleration), 4000 sec (stabilized high power), and 4250 sec (transient deceleration). The calibration y_d^C and validation y_d^V measurements are shown (see Section 7.3.5).

7.3.5.2 Model validation assessment

The *validation assessment* is a quantitative comparison between the model outputs and measurements using a validation metric. Several metric types were reviewed in Section 6.2.1 and the model reliability metric was selected since it has both univariate [141, 150] and multivariate [17, 141] versions (the model reliability metric and area metric were both selected for extension and application, but only the model reliability metric is considered in this chapter).

Recall from Section 6.2.2.2 that the univariate model reliability metric evaluates the probability that the difference between the model outputs and measurements $|z - D|$ is less than a specified accuracy requirement λ, i.e., $R = P(|z - D| < \lambda)$. The univariate (single model output) assessment allows for model diagnosis by showing which parts of the model have the largest error (See Figure 7.12).

To complement this, the multivariate version of the metric condenses the outputs into a single probability that accounts for correlation between the outputs. Its result is used in this chapter for uncertainty aggregation. As shown in Section 6.2.3, the metric is computed by comparing the multivariate Mahalanobis distance $M_{ij} = \sqrt{(\boldsymbol{z}_i - \boldsymbol{D}_j)^T \Sigma^{-1} (\boldsymbol{z}_i - \boldsymbol{D}_j)}$ to a multivariate accuracy requirement $\lambda_M = \sqrt{\boldsymbol{\lambda}^T \Sigma^{-1} \boldsymbol{\lambda}}$, where $\boldsymbol{\lambda} = [\lambda_1, \ldots, \lambda_n]$ combines accuracy requirements for each of the individual outputs. Since the aggregation process in this chapter is sample-based, the Mahalanobis distance is computed for the i^{th} model sample and j^{th} measurement replicate (sample) of the distributions $\boldsymbol{z}$ and $\boldsymbol{D}$, respectively. Thus, for each measurement sample j there is a distribution $\boldsymbol{M}_j$, implying that a separate multivariate model reliability metric is computed for each sample, $R_{Mj} = P(\boldsymbol{M}_j \leq \lambda_M)$.

Figure 7.12 shows a histogram of R_{Mj} plotted for all of the measurement samples. The distribution of

R_{Mj} represents the impact of both model bias/uncertainty (by affecting its central tendency) and measurement uncertainty (by affecting its variance) on the validation metric. If $R_{Mj} = 1$, then the model is in good agreement with the measurements with respect to the accuracy requirement. In this study, the distribution is skewed toward probability 1 with a mean of $\bar{R}_M = 0.75$. For the purpose of uncertainty aggregation, this additional source of uncertainty may be used for the prediction of the QoI through modification of the parameter posteriors or model outputs. The first of these (modification of the parameter posteriors) will be shown in the next section.

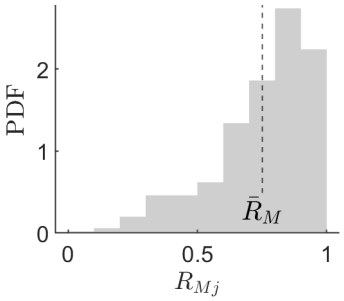


Figure 7.12: Computed multivariate model reliability metric distribution for model M3. The mean is $\bar{R}_M = 0.75$.

Finally, the mean of the above distribution $\bar{R}_M$ for the multivariate metric and the mean results for the univariate metrics of individual output locations are summarized in Figure 7.13 (in this example, the metrics were computed for four time instants). This plot provides additional detail regarding individual output metrics that contribute to the overall result in $\bar{R}_M$. As noted above, this information is useful for model diagnosis within the VVUQ framework. However, unlike $\bar{R}_M$, the univariate metric does not account for output correlation and any conclusions drawn from univariate results should be interpreted with care.

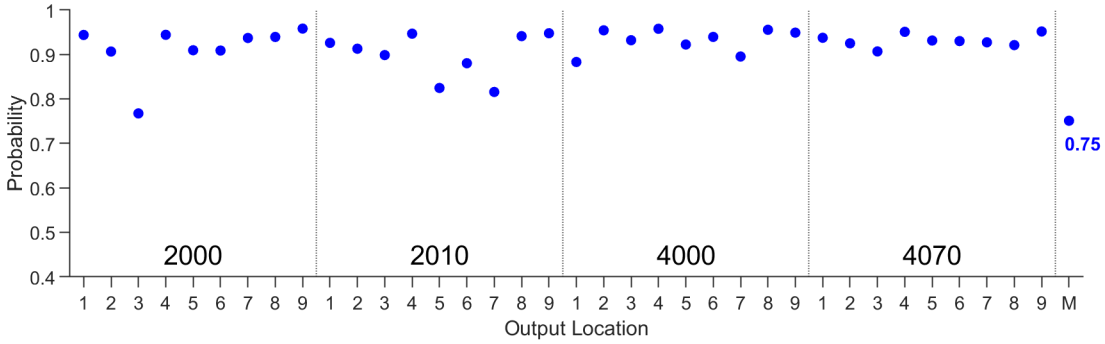


Figure 7.13: Computed mean univariate and multivariate model reliability metrics for model M3. The multivariate metric mean result is $\bar{R}_M = 0.75$, as shown in Figure 7.12.

7.3.5.3 Incorporating the validation assessment into final model predictions

An advantage of the Bayesian VVUQ framework is that the results from model validation may be included during the prediction of the QoI. Sankararaman et al [5] and Mullins [41] proposed weighting the posterior results (either

parameters or outputs) using a probability-based validation metric. Sankararaman et al used Bayesian hypothesis testing for the validation metric, which also results in the probability that the model prediction agrees with the observation. Mullins used the univariate model reliability metric. In this research, we adopt the parameter-weighting approach using the multivariate model reliability metric [141] since (i) modifying the parameters rather than outputs allows for making model predictions of QoI other than the outputs used in validation, and (ii) as previously noted, the multivariate model reliability metric allows for assessing multivariate and correlated model outputs.

Mullins computed the univariate model reliability metric for multiple samples of the inputs, which formed a distribution of the metric that was used in the weighting scheme. In this chapter, the multivariate model reliability metric is used to provide an overall result across measurement locations and temporal conditions (which may be thought of as non-stochastic inputs). It was shown above that, due to the many measurement samples, a collection of metric results is produced R_{Mj} that results in a distribution shown in Figure 7.12. This distribution of probability values R_{Mj} range from 0 to 1 and are used to weight the parameter prior $P(\boldsymbol{\theta})$ and posterior $P(\boldsymbol{\theta}|y_d^C)$ distributions. This approach incorporates both calibration and validation measurements and treats the certainty in the parameters based on the accuracy achieved in the validation step. This weighting process is expressed as [5]

$$P^j(\boldsymbol{\theta}|y_d^C, y_d^V) = R_{Mj} \cdot P(\boldsymbol{\theta}|y_d^C) + (1 - R_{Mj}) \cdot P(\boldsymbol{\theta}) \tag{7.6}$$

The distributions $P^j(\boldsymbol{\theta}|y_d^C, y_d^V)$ are combined for all j to obtain the overall weighted posterior distribution $P(\boldsymbol{\theta}|y_d^C, y_d^V)$, which is shown Figure 7.14. The weighted distribution (dotted) falls between the prior and posterior distributions, but is more similar to the posterior distribution due to the skew in the distribution of R_M towards higher probability. In general, a validation metric skewed towards high probability will favor the parameter posterior distribution and a validation metric skewed toward low probability will favor the prior distribution. The mean and standard deviation of the parameter distributions in Figure 7.14 are compared in Table 7.8.

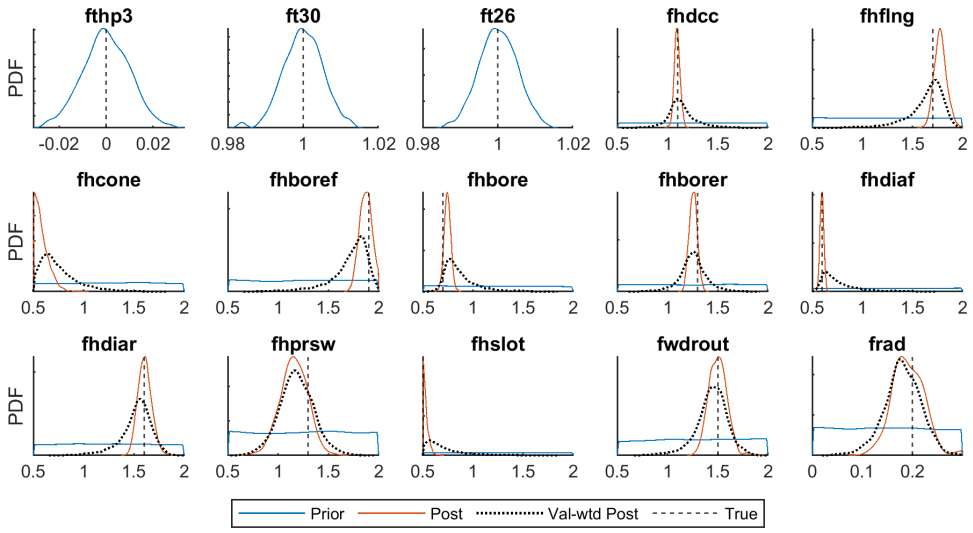


Figure 7.14: Model M3 prior $P(\boldsymbol{\theta})$, posterior $P(\boldsymbol{\theta}|y_d^C)$, and validation-metric-weighted posterior $P(\boldsymbol{\theta}|y_d^C, y_d^V)$.

Table 7.8: Parameter marginal distribution mean E[·] and standard deviations S[·] for calibration posteriors and validation-metric-weighted posteriors.

θ_i	2	3	4	5	6	7	8	9	10	12	13	14
$E[P(\boldsymbol{\theta}\|y_d^C)]$	1.10	1.78	0.58	1.87	0.75	1.26	0.60	1.61	1.16	0.53	1.5	0.19
$E[P(\boldsymbol{\theta}\|y_d^C, y_d^V)]$	1.13	1.65	0.75	1.71	0.87	1.25	0.76	1.52	1.18	0.70	1.44	0.18
$S[P(\boldsymbol{\theta}\|y_d^C)]$	0.029	0.062	0.068	0.058	0.033	0.04	0.018	0.067	0.138	0.027	0.091	0.031
$S[P(\boldsymbol{\theta}\|y_d^C, y_d^V)]$	0.14	0.18	0.19	0.19	0.16	0.14	0.18	0.16	0.17	0.19	0.16	0.036

2:`fhdcc` 3:`fhflng` 4:`fhcone` 5:`fhboref` 6:`fhbore` 7:`fhborer` 8:`fhdiaf` 9:`fhdiar` 10:`fhprsw` 12:`fhslot` 13:`fwdrout` 14:`frad`

The above procedure incorporates model and measurement uncertainty through model parameters. However, if the model exhibits significant model form error through output bias, the above approach may result in biased model predictions. Alternative approaches in the case of model form error include correcting the model output bias by calibrating a *discrepancy model* [13, 98] or correcting the model form (*model form error estimation*) using Bayesian state estimation of additional inputs to the model's governing equations [102]. This is an intrusive method as it requires access to the governing equations, however, a non-intrusive approach to model form error estimation has recently been developed [103]. In the present study, to allow for prediction throughout the entire FE model domain (temporal and spatial), we have chosen the parameter-weighting approach. Future research should consider these alternate approaches where model form error is significant.

7.3.6 Prediction

The final step in the VVUQ framework is the prediction of the QoI, which incorporates the aggregated effects of all the dominant sources of uncertainty. In the prediction step it is assumed that there are no direct measurements available for comparison to the model-based QoI. The validated model may be used either to directly predict the

QoI or may be used indirectly as an input to a downstream model (that is in turn used to predict the QoI). In this section, the former is briefly demonstrated.

For the heat transfer model example, the selected QoI is the rim-to-volume-weighted mean disc temperature, T_{rm}. A requirement was given in Table 7.1 that $T_{rm} < T^*_{rm}$. It is further specified here that $T^*_{rm} = -105$ Kelvin. A negative result for the temperature difference suggests that the bulk of the disc is hotter than the rim, which occurs during deceleration of the engine (cooling of the disc). This situation can result in higher thermal stress in the rim which may set a limit on the useful life of the turbine disc. Thus, T_{rm} may be thought of as an indicator for thermal stress in the turbine disc, and a larger negative result is more adverse. For the purpose of this demonstration, it is further assumed that the failure mechanism is dependent on the maximum that occurs over a given cycle of the engine. Thus, the instantaneous probability of exceeding T^*_{rm} is to be evaluated.

To propagate the aggregated sources of uncertainty that were identified through the previous sections, two options are compared. The first is based on constructing a new surrogate model to predict T_{rm} (a low-fidelity approach). The same DOE and outputs (the 14 time instants) are used to train a response surface model, similar to the model described in Section 7.3.3. However, for the QoI, PCA is not used since the output is a single time-dependent quantity that is comprised of combinations of the outputs (i.e., the difference between the rim output and the disc average temperature). Since the surrogate model runs quickly, many samples of the results may be produced (3×10^4 in this case), which improves the computational efficiency of the probability of failure calculations. This approach may be more practical than many runs of a full-fidelity model when probability of failure calculation is needed.

The second option is to propagate the aggregated uncertainty through the higher-fidelity FE physics model. The number of runs is 857 (there were a few failed runs), which used randomly selected sets of the validation-weighted posterior samples derived in the previous section. Using this method improves the fidelity of the outputs and enables computing all time-steps rather than the 14 steps selected for surrogate modeling (which may miss the worst time point). However, since only a few runs are affordable in the high fidelity case, this introduces additional statistical error for the computed failure probabilities.

Uncertainty is propagated through the surrogate model and high-fidelity FE model and Figure 7.15 compares results for T_{rm} from the two approaches at the same time instant $t = 4250$ seconds (this was identified as is the worst time instant for the surrogate model among the 14 time instants with which it was defined). At $t = 4250$ seconds, the surrogate and FE model estimates for probability of failure were $p_f = 0.0066$ and $p_f = 0.082$, respectively[9]. Then, the probability of failure estimate was determined for the worst time point observed in the FE model results (Figure 7.16), which occurs at $t = 4275.7$ seconds, slightly after the worst time instant that

[9]The difference is thought due to differences in the FE tool's calculation of ϵ_{hr} (Section 3.3), since the surrogate model and discretization errors were computed with the prior parameters and the FE model predictions were based on posterior parameters.

was identified using the surrogate model ($t = 4250$ seconds). This results in $p_f = 0.0187$, which still meets the requirement $p_f < 0.02$ stated in Table 7.1.

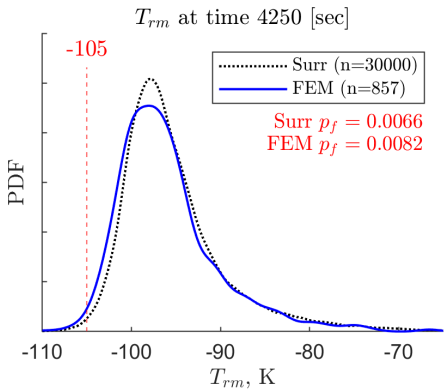


Figure 7.15: Distribution of T_{rm} for the surrogate model and FE model.

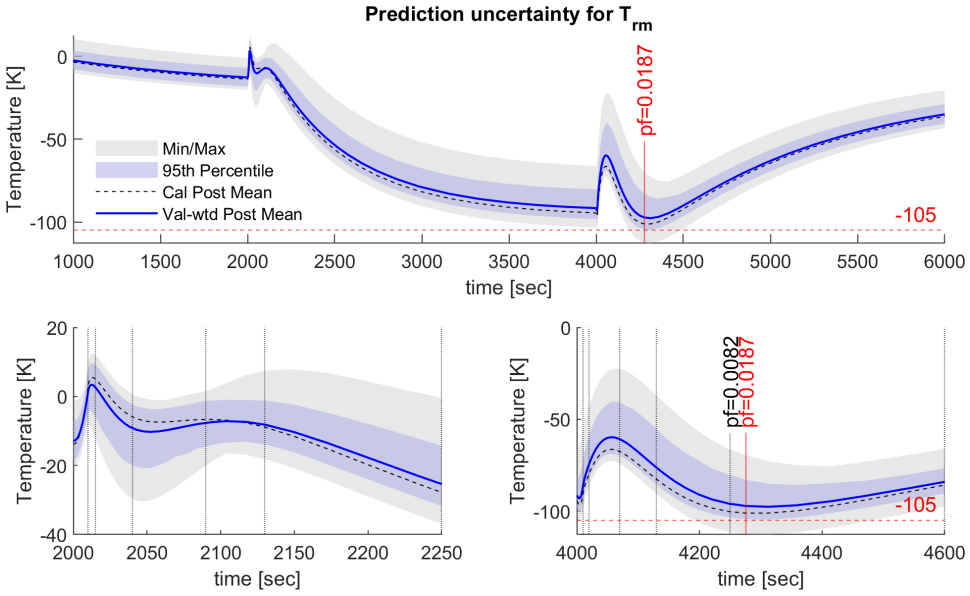


Figure 7.16: Prediction of T_{rm} with the full-fidelity heat transfer model

Finally, the surrogate model and FE model predictions are compared at one of the locations used in validation, P2 (forward diaphragm, see Figure 7.3). This check verifies that the surrogate used for the earlier parts of the VVUQ framework is in agreement with the higher-fidelity FE model, except for the additional sources of uncertainty due to surrogate errors and PCA errors. The distributions of surrogate model and FE model at P2 are plotted for two different time instants in Figure 7.17, while Figure 7.18 plots the FE model result over time at P2. The results show good agreement between the two propagation methods. Note that the surrogate for T_{rm} was developed using the physical outputs directly so there are no PCA errors, i.e., $\epsilon_p = 0$.

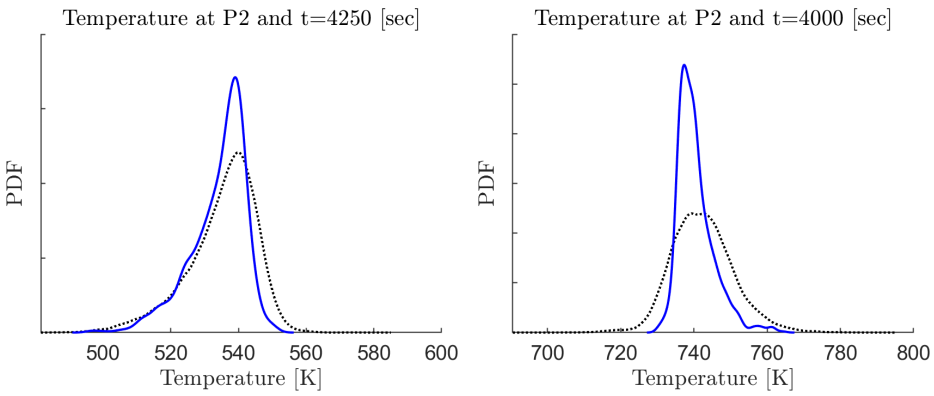


Figure 7.17: Distribution of temperature at P2 for the surrogate model and FE model. The dashed line represents the surrogate model and solid is the FE model.

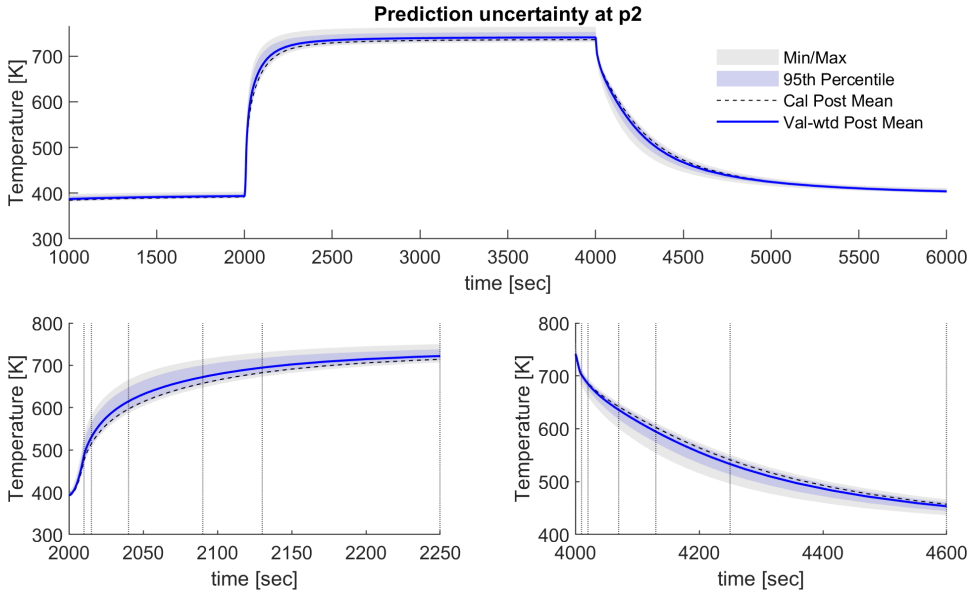


Figure 7.18: Prediction of temperature and uncertainty bounds at P2 with the full-fidelity heat transfer model. The upper panel shows the full square cycle, while the lower two show the acceleration (left) and deceleration (right) with the time instants used for surrogate modeling shown as vertical lines.

7.4 Conclusion

The framework shown in Figure 7.1 provides a structured approach to model development and uncertainty quantification. The Bayesian approach adopted from [5] enables an aggregation process that can address correlation between each source of uncertainty as appropriate. Examples demonstrated include the impact of discretization errors on model parameter calibration, how uncertain inputs affect the correlation between model parameters, and the validation metric results are incorporated to impact the final model predictions. The sources of uncertainty were propagated through both low-fidelity and high-fidelity models to predict the QoI.

The main contribution of this chapter is the development and application of an end-to-end uncertainty aggre-

gation framework for a practical engineering physics model. There is a significant body of excellent work on VVUQ methodologies, but very few examples that take a specific application through the steps proposed in this chapter. Other contributions in this chapter are as follows: the addition of a 'model reduction' step for the uncertainty aggregation framework explicitly highlights and incorporates uncertainty sources from dimension reduction and surrogate modeling; the impact of input uncertainty on model parameter correlation was demonstrated; and, a multivariate model reliability metric was applied for the purpose of model validation and parameter weighting, which also offers a metric for use in model selection.

With such a broad range of steps and methods, there are many areas for improvement and future work. Three important areas are mentioned here:

- A better approach to estimate and apply discretization error results for the transient errors and combined quantities such as T_{rm} (a spatially averaged quantity) would be beneficial.

- The numerical errors were combined independently, but the interactions between these could be incorporated within this framework by deriving appropriate correlation matrices and including them in the likelihood function (Eq. 5.2). However, further work is needed to reduce the computational expense of addressing each of these correlations. The use of a Bayesian network [5] could be used to further improve the integration of these errors.

- The prediction step is an area of active research and methods to address extrapolation issues are needed. In Section 7.3.5.3 a method of Bayesian parameter-weighting was shown as a way to incorporate uncertainty from the validation results. It was noted that this approach may not be appropriate when there is significant model form error. Recently developed model form error estimation methodology, which aims at correcting the model form to improve the model prediction at new untested outputs [102, 103] may be investigated in this regard.

CHAPTER 8

Conclusion

The value of models and measured data to advancements in modern engineering cannot be overstated. However, uncertainties in model predictions and measurements inhibit the exploitation of physics-based models to their fullest extent. Yet, with the democratization of physics-based models and pressure on test budgets, the risk is that no one questions the model or compares it to measurements'. The methods of verification, validation, and uncertainty quantification (VVUQ) aim to temper the excitement and ensure the proper use of models and data in three ways. First, VVUQ seeks to ensure that the right model is developed for the application – that they are sufficiently "accurate for their intended use". Second, it provides evaluation tools and quantitative evidence to establish model credibility. Third, it fuses models and data (and their sources of uncertainty) to make the best of both and improve decision-making.

8.1 Summary of contributions

This research has focused on bridging the gap between the wealth of VVUQ literature, methods, and advancements in recent years, and their application to an industrial-level problem. Several methodologies have been adopted and extended for the estimation and aggregation of uncertainty for multivariate physics models.

- An end-to-end VVUQ framework (Figure 7.1), based on the Bayesian approach in [5], was extended as a road map for the quantification and aggregation of several sources of uncertainty. The application of an end-to-end VVUQ framework to the multivariate output heat transfer model of Chapter 2 serves as one of first published examples to address several sources of model and measurement uncertainty.
- The multi-metric, multi-output validation approach in Chapter 6 improves insights into the model validation process through (i) simultaneous univariate and multivariate assessment, (ii) addressing correlated model outputs using the multivariate model reliability metric, and (iii) improving interpretability of the metric by better accuracy requirement setting and reduced confounding between model bias/variance relative to the measurements.
- The Gaussian process discretization error estimator [54] was extended to an adaptively refined mesh and time step for a transient finite element model. The results were then included in the aggregation of uncertainty within the developed VVUQ framework of Chapter 7.
- The estimation of parameter uncertainty is crucial to models such as the one studied in this research (Chapter 2). Doing this efficiently was achieved in Chapter 4 by (i) developing the dimension-reduced PC-AS

surrogate model method and (ii) performing the calibration within the same dimension reduced space.

- In Chapter 5, the first exploration of model calibration with additive discrepancy in principal component space was performed. This provided insights into such issues as parameter non-identifiability, feasible functional forms for discrepancy models within this subspace, and the application of discrepancy models for diagnostic purposes.

8.2 Future work

VVUQ is a rapidly growing field and there are unending possibilities for improvements in all aspects of the research in this dissertation. There are also many other physics-based modeling applications to which these approaches may be applied and refined. Several of these opportunities for improvement were identified in earlier chapters, and key items are summarized here.

- The use of verification methods is often limited or non-existent in many industrial settings. The GP-based estimation approach deserves further development towards a practical estimation tool for discretization error. The extension of the research in Chapter 3 should focus on rigorous comparison of the GP approach to the well-established Richardson extrapolation methodology with meshes that are both within and extended beyond the asymptotic range. In addition, error estimates are univariate by nature, and focus on a single model solution. Approaches to address multivariate outputs (the entire domain of space and time) and the relationship to other sources of model uncertainty need to be established. The Bayesian network approach to aggregation should be more fully exploited.

- This research addressed time dependence through feature selection and PCA. The proposed approach seems to cover many other similar situations that arise in the industrial setting for model and development testing. However, other approaches for surrogate modeling and measurement processing (such as LSTM, functional PCA, or other time-series methods) may make the methods more generally applicable (e.g., more complex test maneuvers).

- A multivariate metric for the purpose of model validation was demonstrated in Chapter 6. This metric should be exploited for related tasks such as model selection and sensor placement optimization problems. This extension should consider the use of the PCA-based sensitivity analysis approach (demonstrated in Sections 4.3.4 and 5.4.2) when assessing sensor placement. Further work on the metric could look into approaches to test and handle highly-non-normal distributions and guidance on setting metric thresholds for various applications (these are application specific issues).

- Model calibration and validation require separate measurement data sets to ensure the calibration process

is not over-fitting the model to the available data. There are well-established methods to check this in the statistical/machine learning context. Addressing this issue when only a single test and sparse data are available need further investigation. The multivariate validation metric could be used in this context as well.

- Model calibration of multivariate outputs was improved through the formation of the PC-AS surrogate model. However, there were limitations for less significant PCs when using the simplest active subspace definition (linear gradient approximation, single active variable). Further work on this should consider improving these elements to increase accuracy of the method.

- In the discussion of calibration with discrepancy, the formulation of discrepancy functions in the PC space was shown to offer (and require) simplified construction, set up and solution (simultaneous). It was also shown how calibration with and without these discrepancy functions could offer diagnostic insights into sources of model form error and an indication of model uncertainty in the outputs. This diagnostic use of model discrepancy should be further explored with other types of model form error and with increased model complexity (only five parameters were considered in the study).

- Finally, Subramanian and Mahadevan demonstrate an alternative to the model output discrepancy approach with an intrusive [133] and non-intrusive [103] model form error estimation approach. Rather than modifying the outputs, it seeks to adjust the governing equations, making the correction more extensible to the prediction regime (e.g., for locations in the model other than the sensor positions, or for running the model at untested input settings). Improving the accuracy of predictions is a necessary and challenging topic, since it is often not possible to perfectly model the system. This subject requires significant future research investment.

References

[1] ASME. *V&V10-2019 Standard for Verification and Validation in Computational Solid Mechanics*. American Society of Mechanical Engineers, 2020.

[2] William L Oberkampf and Christopher J Roy. *Verification and validation in scientific computing*. Cambridge University Press, 2010.

[3] Christopher J Roy and William L Oberkampf. A comprehensive framework for verification, validation, and uncertainty quantification in scientific computing. *Computer methods in applied mechanics and engineering*, 200(25-28):2131–2144, 2011. ISSN 00457825. doi: 10.1016/j.cma.2011.03.016. URL http://dx.doi.org/10.1016/j.cma.2011.03.016.

[4] National Research Council et al. *Assessing the reliability of complex models: mathematical and statistical foundations of verification, validation, and uncertainty quantification*. National Academies Press, 2012.

[5] Shankar Sankararaman and Sankaran Mahadevan. Integration of model verification, validation, and calibration for uncertainty quantification in engineering systems. *Reliability Engineering and System Safety*, 138:194–209, 2015. ISSN 09518320. doi: 10.1016/j.ress.2015.01.023.

[6] NASA. Standard for Models and Simulations. *NASA-STD-7009A*, 2019.

[7] Stefan Riedmaier, Benedikt Danquah, Bernhard Schick, and Frank Diermeyer. Unified framework and survey for model verification, validation and uncertainty quantification. *Archives of Computational Methods in Engineering*, pages 1–34, 2020.

[8] George Casella and Roger L Berger. *Statistical inference*. Cengage Learning, 2002.

[9] Luc Jaulin, Michel Kieffer, Olivier Didrit, and Eric Walter. Interval analysis. In *Applied interval analysis*, pages 11–43. Springer, 2001.

[10] Luis G. Crespo, Sean P. Kenny, and Daniel P. Giesy. A comparison of metamodeling techniques via numerical experiments. In *18th AIAA Non-Deterministic Approaches Conference*, page 0432, 2016.

[11] V. Novák, I. Perfilijeva, and J. Mockor. *Mathematical principles of fuzzy logic*. Kluwer Academic Publishers Boston, Dordrecht, London, 1999.

[12] Lotfi Asker Zadeh. Fuzzy sets as a basis for a theory of possibility. *Fuzzy sets and systems*, 1(1):pp. 3–28, 1978.

[13] Marc C Kennedy and Anthony O'Hagan. Bayesian calibration of computer models. *J. R. Statist. Soc. B*, 63, Part 3:425–464, 2001.

[14] ASME. *V&V40-2018 Assessing Credibility of Computational Modeling Through Verification and Validation: Application to Medical Devices*. American Society of Mechanical Engineers, 2018.

[15] Marc P. Mignolet, Adam Przekop, Stephen A. Rizzi, and S. Michael Spottswood. A review of indirect/non-intrusive reduced order modeling of nonlinear geometric structures. *Journal of Sound and Vibration*, 332 (10):2437–2460, 2013.

[16] ASME. *V&V20-2009 (R2021) Standard for Verification and Validation in Computational Fluid Dynamics and Heat Transfer*. American Society of Mechanical Engineers, 2009.

[17] Andrew White, Sankaran Mahadevan, Jason Schmucker, and Alexander Karl. Multi-metric validation under uncertainty for multivariate model outputs and limited measurements (under review). *Journal of Verification, Validation, and Uncertainty Quantification*, 2022.

[18] Ogierd Cecil Zienkiewicz. *The finite element method*. London: McGraw-Hill, 1977.

[19] J. A. Dixon, J. A. Verdicchio, D. Benito, A. Karl, and K. M. Tham. Recent developments in gas turbine component temperature prediction methods, using computational fluid dynamics and optimization tools, in conjunction with more conventional finite element analysis techniques. *Proceedings of the Institution of Mechanical Engineers, Part A: Journal of Power and Energy*, 218(4):241–255, 2004. URL https://doi.org/10.1243/0957650041200641.

[20] David T. Williams, Peter Smout, Matteo Bianchi, and Martin B. Joinson. The use of probabilistic methods in determining turbine disc cyclic life uncertainty. In *Turbo Expo: Power for Land, Sea, and Air*, volume 55164, page V03CT18A004. American Society of Mechanical Engineers, 2013.

[21] Godwin Ita Ekong, Christopher A. Long, and Peter Childs. The Effect Of Heat Transfer Coefficient Increase On Tip Clearance Control In Hp Compressors In Gas Turbine Engine. In *ASME 2013 International Mechanical Engineering Congress and Exposition*. American Society of Mechanical Engineers Digital Collection, 2013.

[22] Rolls-Royce. *The Jet Engine*. Rolls-Royce plc, London, U.K., 2005.

[23] Jakob T. Hartl. *Informing Industry End-Users on the Credibility of Model Predictions for Design Decisions*. PhD thesis, Purdue University Graduate School, 2022.

[24] W. J. Minkowycz, Ephraim M. Sparrow, and J. Y. Murthy. *Handbook of Numerical Heat Transfer, Second Edition*. John Wiley & Sons, Hoboken, NJ, 2006.

[25] A. Northrop and J. M. Owen. Heat transfer measurements in rotating-disc systems part 1: The free disc. *International journal of heat and fluid flow*, 9(1):19–26, 1988.

[26] James V. Beck and Kenneth Arnold. *Parameter estimation in engineering and science*. Wiley series in probability and mathematical statistics. John Wiley & Sons. Inc., New York, 1977. ISBN 0471061182.

[27] Helcio R. B. Orlande. Inverse problems in heat transfer: new trends on solution methodologies and applications. *Journal of Heat Transfer*, 134(3):1–13, 2012. ISSN 00221481. doi: 10.1115/1.4005131.

[28] Ralph C. Smith. *Uncertainty quantification: theory, implementation, and applications*. SIAM, 2014.

[29] A Gelman, JB Carlin, HS Stern, DB Dunson, A Vehtari, and BD Rubin. Bayesian data analysis, 3rd edition, 2014.

[30] A. Keane and P. Nair. *Computational approaches for aerospace design*. Wiley, Hoboken, NJ, Hoboken, NJ, 2005.

[31] Felipe AC Viana. A tutorial on latin hypercube design of experiments. *Quality and reliability engineering international*, 32(5):1975–1985, 2016.

[32] Alexander Forrester, Andras Sobester, and Andy Keane. *Engineering design via surrogate modelling: a practical guide*. American Institute of Aeronautics and Astronautics, Inc. John Wiley & Sons, 2008.

[33] Felipe A. C. Viana, Christian Gogu, and Tushar Goel. Surrogate modeling: tricks that endured the test of time and some recent developments. *Structural and Multidisciplinary Optimization*, 64(5):2881–2908, 2021.

[34] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[35] Xu Wu, Tomasz Kozlowski, and Hadi Meidani. Kriging-based inverse uncertainty quantification of nuclear fuel performance code BISON fission gas release model using time series measurement data. *Reliability Engineering and System Safety*, 169:422–436, 2017. doi: 10.1016/j.ress.2017.09.029.

[36] K. Neal, B. Schroeder, J. Mullins, A. Subramanian, and S. Mahadevan. Robust Importance Sampling for Bayesian Model Calibration with Spatio-Temporal Data. *International Journal for Uncertainty Quantification*, 11(4):59–80, 2021. doi: 10.1615/Int.J.UncertaintyQuantification.2021033499.

[37] Berkcan Kapusuzoglu, Sankaran Mahadevan, Shunsaku Matsumoto, Yoshitomo Miyagi, and Daigo Watanabe. Adaptive surrogate modeling for high-dimensional spatio-temporal output. *Structural and Multidisciplinary Optimization*, 65(10):1–20, 2022.

[38] ASME. PTC 19.1-2018 (Revision of ASME PTC 19.1-2013). *American Society of Mechanical Engineers*, 2018.

[39] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An Introduction to Statistical Learning with Applications in R*. Springer, New York, 2013. ISBN 9780387781884. doi: 10.1016/j.peva.2007.06.006. URL http://books.google.com/books?id=9tv0taI8l6YC.

[40] J Smith. *NAFEMS Engineering Simulation Quality Management Guidelines*. NAFEMS, 2020.

[41] Joshua Mullins, Chenzhao Li, Sankaran Mahadevan, and Angel Urbina. Optimal selection of calibration and validation test samples under uncertainty. In *Model Validation and Uncertainty Quantification, Volume 3*, pages 391–401. Springer, 2014.

[42] Christopher J. Roy. Review of code and solution verification procedures for computational simulation. *Journal of Computational Physics*, 205(1):131–156, 2005.

[43] Mark Ainsworth and J. Tinsley Oden. A posteriori error estimation in finite element analysis. *Computer methods in applied mechanics and engineering*, 142(1-2):1–88, 1997.

[44] Thomas Grätsch and Klaus-Jürgen Bathe. A posteriori error estimation techniques in practical finite element analysis. *Computers & Structures*, 83(4-5):235–265, 2005.

[45] T. Edmunds. Practical three dimensional adaptive analysis. In *Proceedings of 4th International Conference on Quality Assurance and Standards, NAFEMS*, 1993.

[46] L. Eça and M. Hoekstra. An evaluation of verification procedures for CFD applications. In *24ˆ¡ th¿ Symposium on Naval Hydrodynamics, Fukuoka, Japan*, pages 568–587, 2002.

[47] Luıs Eça and Martin Hoekstra. An uncertainty estimation exercise with the finite-difference and finite-volume versions of parnassos. In *Workshop on CFD Uncertainty Analysis (L. Eça and M. Hoekstra, eds.), Lisbon, Portugal*, volume 6, 2004.

[48] Lewis Fry Richardson. The approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stresses in a masonry dam. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 210(459-470):307–357, 1911.

[49] Patrick J. Roache. Perspective: a method for uniform reporting of grid refinement studies. *Journal of Fluids Engineering*, 116:405–413, 1994.

[50] R. Difonzo, E. Gajetti, L. Savoldi, and N. Fathi. Assessment of different RANS turbulence models in mini-channels for the cooling of MW-class gyrotron resonators. *International Journal of Heat and Mass Transfer*, 193:122922, 2022.

[51] Daniel C. Kammer, Kenneth F. Alvin, and David S. Malkus. Combining metamodels with rational function representations of discretization error for uncertainty quantification. *Computer methods in applied mechanics and engineering*, 191(13-14):1367–1379, 2002.

[52] Roger W. Logan and Cynthia K. Nitta. Comparing 10 methods for solution verification, and linking to model validation. *Journal of Aerospace Computing, Information, and Communication*, 3(7):354–373, 2006.

[53] James L. Thomas, Boris Diskin, and Christopher L. Rumsey. Towards verification of unstructured-grid solvers. *AIAA journal*, 46(12):3070–3079, 2008.

[54] Sirisha Rangavajhala, Venkata S. Sura, Vadiraj K. Hombal, and Sankaran Mahadevan. Discretization error estimation in multidisciplinary simulations. *AIAA journal*, 49(12):2673–2683, 2011.

[55] Rasmussen CE and Williams CKI. *Gaussian Processes for Machine Learning*. MIT press Cambridge, MA, 2006. ISBN 026218253X.

[56] MathWorks. Gaussian process regression models, 2022. URL https://www.mathworks.com/help/stats/gaussian-process-regression-models.html.

[57] Shane A. Richards. Completed Richardson Extrapolation in Space and Time. *Communications in numerical methods in engineering*, 13(7):573–582, 1997.

[58] A. White, S. Mahadevan, Z. Grey, J. Schmucker, and A. Karl. Efficient calibration of a turbine disc heat transfer model under uncertainty. *Journal of Thermophysics and Heat Transfer*, pages 1–11, 2020. doi: https://doi.org/10.2514/1.T6047.

[59] Paromita Nath, Zhen Hu, and Sankaran Mahadevan. Sensor placement for calibration of spatially varying model parameters. *Journal of Computational Physics*, 343:150–169, 2017. ISSN 10902716. doi: 10.1016/j.jcp.2017.04.033. URL http://dx.doi.org/10.1016/j.jcp.2017.04.033.

[60] Saideep Nannapaneni and Sankaran Mahadevan. Reliability analysis under epistemic uncertainty. *Reliability Engineering & System Safety*, 155:9–20, 2016. ISSN 09518320. doi: 10.1016/j.ress.2016.06.005. URL http://dx.doi.org/10.1016/j.ress.2016.06.005.

[61] Dave Higdon, Charles Nakhleh, James Gattiker, and Brian Williams. A Bayesian calibration approach to the thermal problem. *Computer Methods in Applied Mechanics and Engineering*, 197(29-32), 2008. ISSN 00457825. doi: 10.1016/j.cma.2007.05.031.

[62] Srikanth Akkaram, Don Beeson, Harish Agarwal, and Gene Wiggs. Inverse modeling techniques for parameter estimation in engineering simulation models. In *Turbo Expo: Power for Land, Sea, and Air*, volume 42371, pages 63–72, 2006.

[63] Ghina N. Absi and Sankaran Mahadevan. Multi-fidelity approach to dynamics model calibration. *Mechanical Systems and Signal Processing*, 2016. ISSN 10961216. doi: 10.1016/j.ymssp.2015.07.019.

[64] Paul Constantine, Qiqi Wang, Alireza Doostan, and Gianluca Iaccarino. A Surrogate Accelerated Bayesian Inverse Analysis of the HyShot II Flight Data. In *52nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, 2011. ISBN 978-1-60086-951-8. doi: 10.2514/6.2011-2037.

[65] Paul G Constantine. *Active subspaces: Emerging ideas for dimension reduction in parameter studies*. SIAM, 2015.

[66] A. O'Hagan. Bayesian analysis of computer code outputs: A tutorial. *Reliability Engineering and System Safety*, 91(10-11):1290–1300, 2006. ISSN 09518320. doi: 10.1016/j.ress.2005.11.025.

[67] R. Rebba and S. Mahadevan. Markov Chain Monte Carlo Methods: A Tutorial. Vanderbilt University, Nashville, TN, 2002.

[68] Charles J Geyer. Introduction to markov chain monte carlo. *Handbook of markov chain monte carlo*, 20116022:45, 2011.

[69] C. Davidson-Pilon. Probabilistic Programming & Bayesian Methods for Hackers. *On-line*, 2016. URL http://camdavidsonpilon.github.io/Probabilistic-Programming-and-Bayesian-Methods-for-Hackers.

[70] I.T. Jolliffe. *Principal Component Analysis*. Springer, New York, 2002. ISBN 0-387-95442-2. URL http://www.springer.com/us/book/9780387954424.

[71] Ian T Jolliffe and Jorge Cadima. Principal component analysis : a review and recent developments. *Philosophical Transactions of the Royal Society*, 374:16, 2016. doi: http://dx.doi.org/10.1098/rsta.2015.0202.

[72] Jonathon Shlens. A Tutorial on Principal Component Analysis. *arXiv preprint arXiv:1404.1100*, 2014. doi: 10.1.1.115.3503. URL http://arxiv.org/abs/1404.1100.

[73] G. Strang. *Introduction to Linear Algebra, 3rd ed.* Wessley-Cambridge Press, 2003.

[74] A. Haldar and Sankaran Mahadevan. *Probability, reliability and statistical method in engineering design*. John Wiley & Sons, New York, 2000.

[75] Olivier Zahm, Paul G. Constantine, Clementine Prieur, and Youssef M. Marzouk. Gradient-based dimension reduction of multivariate vector-valued functions. *SIAM Journal on Scientific Computing*, 42(1): A534–A558, 2020. URL https://doi.org/10.1137/18M1221837.

[76] Zachary J. Grey and Paul G. Constantine. Active subspaces of airfoil shape parameterizations. *AIAA Journal*, 56(5):2003–2017, 2017.

[77] Trent W Lukaczyk, Paul Constantine, Francisco Palacios, and Juan J Alonso. Active subspaces for shape optimization. In *10th AIAA multidisciplinary design optimization conference*, page 1171, 2014.

[78] Steven H. Berguin, David Rancourt, and Dimitri N. Mavris. Method to facilitate high-dimensional design space exploration using computationally expensive analyses. *AIAA Journal*, 53(12):3752–3765, 2015.

[79] Alexander Karl, Andrew White, Zach Grey, and Paul Constantine. Application of stochastic and probabilistic methods to the design of advanced aero engines. In *2017 NAFEMS World Congress*. NAFEMS Stockholm, Sweden, 2017.

[80] J. Jacod and P. Protter. *Probability Essentials*. Springer Science & Business Media, 2012.

[81] Andrew Glaws and Paul G. Constantine. Gauss-Christoffel quadrature for inverse regression: applications to computer experiments. *Statistics and Computing*, pages 1–19, 2018. ISSN 15731375. doi: 10.1007/s11222-018-9816-4.

[82] Juan J Alonso, Michael S Eldred, Paul Constantine, Karthikeyan Duraisamy, Charbel Farhat, Gianluca Iaccarino, and John Jakeman. Scalable environment for quantification of uncertainty and optimization in industrial applications (sequoia). In *19th AIAA Non-Deterministic Approaches Conference*, page 1327, 2017.

[83] Paul G Constantine, Michael Emory, Johan Larsson, and Gianluca Iaccarino. Exploiting active subspaces to quantify uncertainty in the numerical simulation of the hyshot ii scramjet. *Journal of Computational Physics*, 302:1–20, 2015.

[84] Paul G Constantine, Armin Eftekhari, Jeffrey Hokanson, and Rachel A Ward. A near-stationary subspace for ridge approximation. *Computer Methods in Applied Mechanics and Engineering*, 326:402–421, 2017. ISSN 00457825. doi: 10.1016/j.cma.2017.07.038. URL http://dx.doi.org/10.1016/j.cma.2017.07.038.

[85] Zach Grey, Paul Constantine, and Andrew White. Enabling aero-engine thermal model calibration using active subspaces. In *AIAA Propulsion and Energy 2019 Forum*, page 4329, 2019.

[86] Paul G. Constantine and Paul Diaz. Global sensitivity metrics from active subspaces. *Reliability Engineering & System Safety*, 162:1–13, 2017.

[87] Chenzhao Li and Sankaran Mahadevan. An efficient modularized sample-based method to estimate the first-order Sobol index. *Reliability Engineering & System Safety*, 153:110–121, 2016. ISSN 09518320. doi: 10.1016/j.ress.2016.04.012.

[88] Matieyendou Lamboni. Multivariate sensitivity analysis: Minimum variance unbiased estimators of the first-order and total-effect covariance matrices. *Reliability Engineering & System Safety*, 187:67–92, 2019.

[89] Sinan Xiao, Zhenzhou Lu, and Liyang Xu. Multivariate sensitivity analysis based on the direction of eigen space through principal component analysis. *Reliability Engineering & System Safety*, 165:1–10, 2017.

[90] Kevin P. Murphy. *Machine Learning: A Probabilistic Perspective*. MIT Press, 2012. ISBN 9780262018029. URL http://www.springerreference.com/index/doi/10.1007/SpringerReference{_}35834.

[91] Mark Steyvers. Computational Statistics with Matlab. Lecture, University of California Irvine. *University of California Irvine*, page 78, 2011. ISSN 19395108. doi: 10.1002/wics.138. URL http://psiexp.ss.uci.edu/research/teachingP205C/205C.pdf.

[92] You Ling, Joshua Mullins, and Sankaran Mahadevan. Selection of model discrepancy priors in Bayesian calibration. *Journal of Computational Physics*, 276:665–680, 2014. ISSN 10902716. doi: 10.1016/j.jcp.2014.08.005. URL http://dx.doi.org/10.1016/j.jcp.2014.08.005.

[93] Jenni Brynjarsdottir and Anthony O'Hagan. Learning about physical parameters: The importance of model discrepancy. *Inverse Problems*, 30(11):24, 2014. doi: https://doi.org/10.1088/0266-5611/30/11/114007.

[94] Henry Stark and John Woods. *Probability, Statistics, and Random Processes for Engineers, 4th ed.* Pearson, 2011.

[95] Yulin Guo, Sankaran Mahadevan, Shunsaku Matsumoto, Shunsuke Taba, and Daigo Watanabe. Surrogate modeling with high-dimensional input and output. In *AIAA Scitech 2021 Forum*, 2021.

[96] Izabel Pirimai Aguiar. *Dynamic active subspaces: a data-driven approach to computing time-dependent active subspaces in dynamical systems*. PhD thesis, University of Colorado at Boulder, 2018.

[97] Min Li, Ruo-Qian Wang, and Gaofeng Jia. Efficient dimension reduction and surrogate-based sensitivity analysis for expensive models with high-dimensional outputs. *Reliability Engineering & System Safety*, 195:106725, 2020.

[98] Andrew White and Sankaran Mahadevan. Discrepancy modeling for model calibration with multivariate output (under review). *International Journal for Uncertainty Quantification*, 2023.

[99] V. Roshan Joseph and Huan Yan. Engineering-driven statistical adjustment and calibration. *Technometrics*, 57(2):257–267, apr 2015. ISSN 15372723. doi: 10.1080/00401706.2014.902773.

[100] Dave Higdon, Marc Kennedy, James C Cavendish, John A Cafeo, and Robert D Ryne. Combining Field Data and Computer Simulations for Calibration and Prediction. *SIAM J. Scientific Computing*, 26(2):448–466, 2004. doi: 10.1137/S1064827503426693. URL http://www.siam.org/journals/sisc/26-2/42669.html.

[101] K. Sargsyan, H. N. Najm, and R. Ghanem. On the statistical calibration of physical models. *International Journal of Chemical Kinetics*, 47(4):246–276, 2015. ISSN 10974601. doi: 10.1002/kin.20906.

[102] Abhinav Subramanian and Sankaran Mahadevan. Model error propagation from experimental to prediction configuration. *Journal of Computational Physics*, 443(110529), 2021. doi: https://doi.org/10.1016/j.jcp.2021.110529.

[103] Abhinav Subramanian and Sankaran Mahadevan. Nonintrusive estimation of model error and discrepancy in dynamics models. *Journal of Computational Physics*, 471(111617), 2022. doi: https://doi.org/10.1016/j.jcp.2022.111617.

[104] Na Qiu, Chanyoung Park, Yunkai Gao, Jianguang Fang, Guangyong Sun, and Nam H Kim. Sensitivity-based parameter calibration and model validation under model error. *Journal of Mechanical Design*, 140 (1):011403, 2018.

[105] PR Wagner, J Nagel, S Marelli, and B Sudret. Uqlab user manual: Bayesian inversion for model calibration and validation. *Chair of Risk, Safety and Uncertainty Quantification: ETH Zurich, Switzerland*, 2019.

[106] Kathryn A Maupin and Laura P Swiler. Model discrepancy calibration across experimental settings. *Reliability Engineering & System Safety*, 200:106818, 2020.

[107] J. Sacks, W. J. Welch, T. J. Mitchell, and H. P. Wynn. Design and analysis of computer experiments. *Statistical Science*, 4(4):409–423, 1989. doi: 10.1214/ss/1177012413.

[108] Maria J. Bayarri, Rui Paulo, James O. Berger, Jerry Sacks, John A. Cafeo, James Cavendish, Chin Hsu Lin, and Jian Tu. A framework for validation of computer models. *Technometrics*, 49(2):138–154, 2007. ISSN 00401706. doi: 10.1198/004017007000000092.

[109] R. D. Wilkinson. Bayesian calibration of expensive multivariate computer experiments. *Large-Scale Inverse Problems and Quantification of Uncertainty*, pages 195–215, 2010. doi: 10.1002/9780470685853.ch10.

[110] Paul D. Arendt, Daniel W. Apley, and Wei Chen. Quantification of model uncertainty: Calibration, model discrepancy, and identifiability. *Journal of Mechanical Design, Transactions of the ASME*, 134(10):1–12, 2012. ISSN 10500472. doi: 10.1115/1.4007390.

[111] Paul D. Arendt, Daniel W. Apley, Wei Chen, David Lamb, and David Gorsich. Improving identifiability in model calibration using multiple responses. *Journal of Mechanical Design, Transactions of the ASME*, 134(10):1–9, 2012. ISSN 10500472. doi: 10.1115/1.4007573.

[112] Paul D. Arendt, Daniel W. Apley, and Wei Chen. A preposterior analysis to predict identifiability in the experimental calibration of computer models. *IIE Transactions (Institute of Industrial Engineers)*, 48(1):75–88, jan 2016. ISSN 15458830. doi: 10.1080/0740817X.2015.1064554.

[113] F. Liu, M. J. Bayarri, and J. O. Berger. Modularization in Bayesian analysis, with emphasis on analysis of computer models. *Bayesian Analysis*, 4(1):119–150, 2009. ISSN 19360975. doi: 10.1214/09-BA404.

[114] Xu Wu, Koroush Shirvan, and Tomasz Kozlowski. Demonstration of the relationship between sensitivity and identifiability for inverse uncertainty quantification. *Journal of Computational Physics*, 396:12–30, 2019. doi: 10.1016/j.jcp.2019.06.032. URL www.elsevier.com/locate/jcp.

[115] Joseph H. A. Guillaume, John D. Jakeman, Stefano Marsili-Libelli, Michael Asher, Philip Brunner, Barry Croke, Mary C. Hill, Anthony J. Jakeman, Karel J. Keesman, Saman Razavi, et al. Introductory overview of identifiability analysis: A guide to evaluating whether you have the right type of data for your modeling purpose. *Environmental Modelling & Software*, 119:418–432, 2019.

[116] Andreas Raue, C. Kreutz, T. Maiwald, J. Bachmann, M. Schilling, U. Klingmüller, and J. Timmer. Structural and practical identifiability analysis of partially observed dynamical models by exploiting the profile likelihood. *Bioinformatics*, 25(15):1923–1929, 2009. ISSN 13674803. doi: 10.1093/bioinformatics/btp358.

[117] Sujit K Sahu and Alan E Gelfand. Identifiability, Improper Priors, and Gibbs Sampling for Generalized Linear Models. *Journal of the American Statistical Association*, 94(445):247–253, 1999. URL file:///Files/8C/8C497F80-C991-4E10-BD0C-E2A718C26DFC.pdf{%}0Apapers3://publication/doi/10.2307/2669699?ref=search-gateway:7f973ce29797f0723af4172a1a582990.

[118] John McFarland and Sankaran Mahadevan. Multivariate significance testing and model calibration under uncertainty. *Computer Methods in Applied Mechanics and Engineering*, 197(29-32):2467–2479, may 2008. ISSN 00457825. doi: 10.1016/j.cma.2007.05.030.

[119] John McFarland and Sankaran Mahadevan. Error and variability characterization in structural dynamics modeling. *Computer Methods in Applied Mechanics and Engineering*, 197(29-32):2621–2631, 2008. doi: https://doi.org/10.1016/j.cma.2007.07.029.

[120] Xu Wu, Tomasz Kozlowski, Hadi Meidani, and Koroush Shirvan. Inverse uncertainty quantification using the modular Bayesian approach based on Gaussian process, Part 1: Theory. *Nuclear Engineering and Design*, 2018. ISSN 00295493. doi: 10.1016/j.nucengdes.2018.06.004.

[121] Xu Wu, Tomasz Kozlowski, Hadi Meidani, and Koroush Shirvan. Inverse uncertainty quantification using the modular Bayesian approach based on Gaussian Process, Part 2: Application to TRACE. *Nuclear Engineering and Design*, 335(May):417–431, 2018. ISSN 00295493. doi: 10.1016/j.nucengdes.2018.06.003.

[122] Ghina N Absi and Sankaran Mahadevan. Simulation and sensor optimization for multifidelity dynamics model calibration. *AIAA Journal*, 58(2):879–888, 2020. URL https://doi.org/10.2514/1.J058485.

[123] Kyle Daniel Neal, Benjamin Schroeder, Joshua Grady Mullins, Sankaran Mahadevan, and Abhinav Subramanian. Bayesian calibration of the thermal battery. Technical report, Sandia National Lab.(SNL-NM), Albuquerque, NM (United States), 2019.

[124] Clemens Kreutz. An easy and efficient approach for testing identifiability. *Bioinformatics*, 34(11):1913–1921, 2018.

[125] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning Data Mining, Inference, and Prediction*. Springer Series in Statistics, 2009.

[126] Matthew Plumlee. Bayesian Calibration of Inexact Computer Models. *Journal of the American Statistical Association*, 112:1274–1285, 2017. ISSN 0162-1459. doi: 10.1080/01621459.2016.1211016. URL https://www.tandfonline.com/action/journalInformation?journalCode=uasa20.

[127] Zachary J. Grey. Vector-valued Taylor Series Expansions (personal communication). Personal Communication, 2020.

[128] Luyi Li and Zhenzhou Lu. A new method for model validation with multivariate output. *Reliability Engineering and System Safety*, 169(July 2016):579–592, 2018. ISSN 09518320. doi: https://doi.org/10.1016/j.ress.2017.10.005.

[129] Scott Ferson, William L. Oberkampf, and Lev Ginzburg. Model validation and predictive capability for the thermal challenge problem. *Computer Methods in Applied Mechanics and Engineering*, 2008. ISSN 00457825. doi: 10.1016/j.cma.2007.07.030.

[130] S. Kullback and R.A. Leibler. On Information and Sufficiency. *Annals of Mathematical statistics*, 22: 79–86, 1951. doi: https://doi.org/10.1214/aoms/1177729694.

[131] John Duchi. Derivations for linear algebra and optimization. *Stanford University*, pages 2325–5870, 2007. URL http://web.stanford.edu/~jduchi/projects/general_notes.pdf.

[132] Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092, 1953.

[133] Abhinav Subramanian and Sankaran Mahadevan. Error estimation in coupled multi-physics models. *Journal of Computational Physics*, 395:19–37, 2019. doi: https://doi.org/10.1016/j.jcp.2019.06.013.

[134] AIAA. *Guide for the verification and validation of computational fluid dynamics simulations*. American Institute of Aeronautics and Astronautics, 1998.

[135] CEN16799. Validation of computational solid mechanics models, 2014.

[136] BIPM, IEC, IFCC, ILAC, ISO, IUPAC, IUPAP, and OIML. Evaluation of measurement data — Guide to the expression of uncertainty in measurement. Joint Committee for Guides in Metrology, JCGM 100:2008, 2008. URL https://www.bipm.org/documents/20126/2071204/JCGM_100_2008_E.pdf/cb0ef43f-baa5-11cf-3f85-4dcd86f77bd6.

[137] Joshua Mullins. *Resource allocation for uncertainty quantification and reduction*. PhD thesis, Vanderbilt University, Nashville, TN, 2014.

[138] You Ling and Sankaran Mahadevan. Quantitative model validation techniques: New insights. *Reliability Engineering & System Safety*, 111:217–231, 2013. ISSN 09518320. doi: 10.1016/j.ress.2012.11.011. URL http://dx.doi.org/10.1016/j.ress.2012.11.011.

[139] William L Oberkampf and Matthew F Barone. Measures of agreement between computation and experiment: validation metrics. *Journal of Computational Physics*, 217(1):5–36, 2006.

[140] Vicente Romero. Comparison of several model validation conceptions against a "real space" end-to-end approach. *SAE International Journal of Materials and Manufacturing*, 4(1):396–420, 2011.

[141] Chenzhao Li and Sankaran Mahadevan. Role of calibration, validation, and relevance in multi-level uncertainty integration. *Reliability Engineering & System Safety*, 148:32–43, 2016. ISSN 09518320. doi: 10.1016/j.ress.2015.11.013. URL http://dx.doi.org/10.1016/j.ress.2015.11.013.

[142] Joshua Mullins, You Ling, Sankaran Mahadevan, Lin Sun, and Alejandro Strachan. Separation of aleatory and epistemic uncertainty in probabilistic model validation. *Reliability Engineering & System Safety*, 147: 49–59, 2016. ISSN 09518320. doi: 10.1016/j.ress.2015.10.003. URL http://dx.doi.org/10.1016/j.ress.2015.10.003.

[143] Yu Liu, Wei Chen, Paul Arendt, and Hong-Zhong Huang. Toward a better understanding of model validation metrics. *Journal of Mechanical Design*, 133(7), 2011. ISSN 10500472. doi: 10.1115/1.4004223.

[144] Kathryn A Maupin, Laura P Swiler, and Nathan W Porter. Validation metrics for deterministic and probabilistic data. *Journal of Verification, Validation and Uncertainty Quantification*, 3(3), 2018.

[145] Paul Gardner, Charles Lord, and Robert J Barthorpe. An evaluation of validation metrics for probabilistic model outputs. In *Verification and Validation*, volume 40795, page V001T06A001. American Society of Mechanical Engineers, 2018.

[146] Ksenija Dvurecenska. *Developing quantitative validation metrics to assess quality of computational mechanics models relative to reality*. The University of Liverpool (United Kingdom), 2019.

[147] Paul Gardner, Charles Lord, and Robert J. Barthorpe. A Unifying Framework for Probabilistic Validation Metrics. *Journal of Verification, Validation and Uncertainty Quantification*, 4(3), sep 2019. ISSN 2377-2158. doi: 10.1115/1.4045296.

[148] Richard Guy Hills and Timothy G. Trucano. Statistical validation of engineering and scientific models: a maximum likelihood based metric. Technical report, Sandia National Laboratories, 2002.

[149] Ramesh Rebba and Sankaran Mahadevan. Model predictive capability assessment under uncertainty. *AIAA journal*, 44(10):2376–2384, 2006. URL https://doi.org/10.2514/1.19103.

[150] Ramesh Rebba and Sankaran Mahadevan. Computational methods for model reliability assessment. *Reliability Engineering & System Safety*, 93(8):1197–1207, 2008. ISSN 09518320. URL https://doi.org/10.1016/j.ress.2007.08.001.

[151] Nolan Wagner Whiting. *Assessment of model validation, calibration, and prediction approaches in the presence of uncertainty*. PhD thesis, Virginia Tech, 2019.

[152] Mirko Nentwig, Maximilian Miegler, and Marc Stamminger. Concerning the applicability of computer graphics for the evaluation of image processing algorithms. In *2012 IEEE International Conference on Vehicular Electronics and Safety (ICVES 2012)*, pages 205–210. IEEE, 2012.

[153] Wei Li, Wei Chen, Zhen Jiang, Zhenzhou Lu, and Yu Liu. New validation metrics for models with multiple correlated responses. *Reliability Engineering and System Safety*, 127:1–11, 2014. ISSN 09518320. doi: 10.1016/j.ress.2014.02.002. URL http://dx.doi.org/10.1016/j.ress.2014.02.002.

[154] Roy De Maesschalck, Delphine Jouan-Rimbaud, and Désiré L Massart. The mahalanobis distance. *Chemometrics and intelligent laboratory systems*, 50(1):1–18, 2000.

[155] Bin Wang, Wenzhong Shi, and Zelang Miao. Confidence analysis of standard deviational ellipse and its extension into higher dimensional euclidean space. *PloS one*, 10(3):e0118537, 2015.

[156] Dan Ao, Zhen Hu, and Sankaran Mahadevan. Dynamics model validation using time-domain metrics. *Journal of Verification, Validation and Uncertainty Quantification*, 2(1):011005, 2017. ISSN 2377-2158. doi: 10.1115/1.4036182. URL http://verification.asmedigitalcollection.asme.org/article.aspx?doi=10.1115/1.4036182.

[157] Jurgen A. Doornik and Henrik Hansen. An omnibus test for univariate and multivariate normality. *Oxford bulletin of economics and statistics*, 70:927–939, 2008.

[158] Amirhossein Amiri, Mahdi Bashiri, Hamed Mogouie, and Mohammad Hadi Doroudyan. Non-normal multi-response optimization by multivariate process capability index. *Scientia Iranica*, 19(6):1894–1905, 2012.

[159] V. K. Hombal and S. Mahadevan. Model selection among physics-based models. *Journal of mechanical design*, 135(2):021003, 2013. URL https://doi.org/10.1115/1.4023155.

[160] Rimple Sandhu, Mohammad Khalil, Abhijit Sarkar, and Dominique Poirel. Bayesian model selection for nonlinear aeroelastic systems using wind-tunnel data. *Computer Methods in Applied Mechanics and Engineering*, 282:161–183, 2014.

[161] Anis Ben Abdessalem, Nikolaos Dervilis, David Wagg, and Keith Worden. Model selection and parameter estimation in structural dynamics using approximate bayesian computation. *Mechanical Systems and Signal Processing*, 99:306–325, 2018.

[162] Laura Mainini and Karen E Willcox. Sensor placement strategy to inform decisions. In *18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, page 3820, 2017.

[163] William Sisson, Sankaran Mahadevan, and Benjamin P. Smarslok. Optimization of information gain in multifidelity high-speed pressure predictions. *AIAA Journal*, 59(8):3096–3105, 2021. URL https://doi.org/10.2514/1.J059507.

[164] DoD. DoD Modeling and Simulation (M&S) Verification, Validation, and Accreditation (VV&A). *DoD Instruction 5000.61*, 2018.

[165] ASME. *V&V10-2006 Guide for Verification and Validation in Computational Solid Mechanics*. American Society of Mechanical Engineers, 2006.

[166] Ian T. Voyles and Christopher J. Roy. Evaluation of model validation techniques in the presence of aleatory and epistemic input uncertainties. In *17th AIAA Non-Deterministic Approaches Conference*, page 1374, 2015.

[167] M. Tanaka. Application of area validation methods for uncertainty quantification in validation process of thermalhydraulic code for thermal fatigue issue in sodium-cooled fast reactors. In *ASME 2016 verification & validation symposium*, 2016.

[168] Ning Wang, Wen Yao, Yong Zhao, Xiaoqian Chen, Xiang Zhang, and Lanmin Li. A new interval area metric for model validation with limited experimental data. *Journal of Mechanical Design*, 140(6), 2018.

[169] Magnus Eek. *On Credibility Assessment in Aircraft System Simulation*. Thesis, Linkoping University, 2016.

[170] Nitya Kamdar, Fangyuan Lou, and Nicole L. Key. Details of shrouded stator hub cavity flow in a multi-stage axial compressor part 1: Interactions with the primary flow, June 2021. URL https://doi.org/10.1115/GT2021-60103.

[171] Pranay Seshadri, A Duncan, George Thorne, Geoffrey Parks, R Vazquez, and Mark Girolami. Bayesian assessments of aeroengine performance with transfer learning. *Preprint*, 2020.

[172] Shankar Sankararaman, You Ling, Chris Shantz, and Sankaran Mahadevan. Uncertainty Quantification in Fatigue Damage Prognosis. *International Conference on Prognostics and Health Management (PHM)*, pages 1–13, 2009.

[173] Andrea Saltelli, Marco Ratto, Terry Andres, Francesca Campolongo, Jessica Cariboni, Debora Gatelli, Michaela Saisana, and Stefano Tarantola. *Global sensitivity analysis: the primer*. John Wiley & Sons, 2008.

[174] Berkcan Kapusuzoglu and Sankaran Mahadevan. Information fusion and machine learning for sensitivity analysis using physics knowledge and experimental data. *Reliability Engineering & System Safety*, 214: 107712, 2021. URL https://doi.org/10.1016/j.ress.2021.107712.

[175] Christopher M. Bishop. *Pattern recognition and machine learning*. Springer, 2006.

[176] John Salvatier, Thomas V. Wiecki, and Christopher Fonnesbeck. Probabilistic programming in Python using PyMC3. *PeerJ Computer Science*, 2016. doi: 10.7717/peerj-cs.55. URL https://github.com/pymc-devs/pymc3.

[177] Mathworks. Stepwise linear regression, 2022. URL https://www.mathworks.com/help/stats/stepwiselm.html.