In mathematical modeling, global sensitivity analysis (GSA) has emerged as a powerful tool for quantifying the importance of uncertain parameters in computationally-expensive, high-dimensional models. A primary tool for GSA is the Sobol’ index, which defines parameter importance in terms of their relative contribution to the model variance. Computing Sobol’ indices is a difficult task and there are a variety of methods available, each with their own strengths and weaknesses. In this thesis, we show that the feasibility of performing GSA can be extended in certain situations where the model in question belongs to a family of related models. In such situations, GSA information from lower-fidelity and lower-cost models, can be used to approximate the sensitivity of the high-fidelity and high-cost model. Accelerating the GSA process can then enable further model analysis, dimension reduction, and the design of future experiments. In this dissertation, we will examine several scenarios in which GSA poses significant computational challenges and the use of multiple model fidelities can be shown to overcome these challenges. In the context of chemical kinetics, we will show that the Sobol’ indices of an expensive, stochastic model can be approximated by those of a cheaper, deterministic model, with provable convergence properties. In the context of multilevel models, we will show that polynomial surrogate models can be used to create a goal-oriented method for efficient GSA, while providing probabilistic information about the accuracy of the computed Sobol’ indices. Finally, in the challenging context of computing rare event probabilities, we will show that performing GSA with respect to model hyperparameters can be accelerated by combining inexpensive estimations of the probability and sparse polynomial surrogate models.
Multifidelity Global Sensitivity Analysis for Complex Problems

by
Michael B. Merritt

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Applied Mathematics

Raleigh, North Carolina
2022

APPROVED BY:

_________________________ Pierre Gremaud
Co-chair of Advisory Committee

_________________________ Alen Alexanderian
Co-chair of Advisory Committee

_________________________ Mohammad Farazmand

_________________________ Gianluca Geraci

_________________________ James Kneller
DEDICATION

To Amanda
BIOGRAPHY

Michael Merritt was born in Sheboygan, Wisconsin and grew up in the nearby town of Cedar Grove. He attended Carthage College in Kenosha, Wisconsin, earning degrees in Physics and Mathematics in 2017. During his time at Carthage, he was first introduced to topics in numerical analysis and statistical methods.

In the fall of 2017, he moved to Raleigh, North Carolina and began pursuing a Doctor of Philosophy degree in Applied Mathematics at North Carolina State University. In 2019, while at NC State, he earned a Master of Science degree in Applied Mathematics and will earn a Ph.D. in the spring of 2022. During his time at NC State, Michael was supported by an NSF graduate fellowship through the Research Training Group in Randomized Numerical Analysis. Michael has also served as the Industry Outreach Coordinator for the NC State chapter of the Society of Industrial and Applied Mathematics (SIAM) and mentored undergraduate students through Undergrads Union Grads. Starting in the summer of 2020, Michael began an internship in the Department of Optimization and Uncertainty Quantification at Sandia National Laboratories, which continued through the spring of 2022.
ACKNOWLEDGEMENTS

First, I would like to thank my advisors, Pierre Gremaud and Alen Alexanderian. I am fortunate to have had two excellent advisors, who each have their own complementary strengths and yet a unanimous focus toward my success. I am grateful to Mike Eldred, Teresa Portone, and many others that provided me with an enriching and enjoyable internship experience at Sandia, with special thanks due to Gianluca Geraci. I would like to thank the RTG group at NC State for their helpful feedback over the years and their financial support through NSF grants DMS 1745654 and DMS 1953271. I would like to thank my friends and officemates in LAU 108 for their years of support and camaraderie.

Thank you to my parents, Stuart and Lynelle Merritt, for passing on to me what I know to be of first importance. To the friends and family that have supported me, especially during the COVID-19 pandemic, thanks for always being willing to chat over Zoom. Special thanks are due to the dear friends we have made at Midtown Community Church in Raleigh.

Finally, to my wonderful wife, Amanda, who has steadfastly been by my side since the beginning of this journey, thank you.
# TABLE OF CONTENTS

**LIST OF TABLES** ........................................... vii

**LIST OF FIGURES** ........................................ viii

**Chapter 1** INTRODUCTION .................................................. 1
  1.1 Motivation ....................................................... 1
  1.2 Contents of the thesis ........................................ 2
  1.3 Overview of the author’s publications ...................... 3

**Chapter 2** BACKGROUND MATERIAL ..................................... 5
  2.1 Global sensitivity analysis ...................................... 5
    2.1.1 Variance-based GSA ........................................ 5
    2.1.2 Estimation of Sobol’ indices ............................... 7
    2.1.3 Alternative GSA methods .................................. 8
  2.2 Polynomial chaos expansions .................................... 9
    2.2.1 PCE basis construction ................................... 9
    2.2.2 Non-intrusive spectral projection ......................... 11
    2.2.3 GSA via PCE .............................................. 12

**Chapter 3** MULTISCALE GLOBAL SENSITIVITY ANALYSIS FOR STOCHASTIC CHEMICAL SYSTEMS ........................................... 13
  3.1 Introduction ..................................................... 13
  3.2 Chemical kinetics models ....................................... 15
      3.2.1 The Random Time Change representation ................. 16
      3.2.2 An example reaction system .............................. 17
      3.2.3 The thermodynamic limit ................................ 18
  3.3 The Next Reaction Method ....................................... 20
  3.4 Global sensitivity analysis for stochastic models ......... 22
    3.4.1 Theoretical setup ........................................ 23
    3.4.2 Convergence of stochastic Sobol’ indices ................. 23
    3.4.3 Application to stochastic chemical kinetics ............. 25
  3.5 Numerical results ............................................... 27
      3.5.1 Application to the Michaelis–Menten system ........... 28
      3.5.2 Application to the genetic oscillator system .......... 33
  3.6 Conclusion ..................................................... 37

**Chapter 4** A HYBRID MULTILEVEL MONTE CARLO - POLYNOMIAL CHAOS METHOD FOR SENSITIVITY ANALYSIS ........................................... 39
  4.1 Introduction ..................................................... 39
  4.2 Review of GSA and PCE background ............................ 41
      4.2.1 Monte Carlo and multilevel Monte Carlo sampling ...... 42
      4.2.2 MLMC estimation of PCE coefficients ................... 44
  4.3 Theoretical development ........................................ 45
      4.3.1 Derivation of variance and covariance terms .......... 47
  4.4 Optimal sample allocation ...................................... 49
  4.5 Numerical verification ......................................... 51
Chapter 5 PRACTICAL ASPECTS OF THE HYBRID MLMC-PCE METHOD 56

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1 Introduction</td>
<td>56</td>
</tr>
<tr>
<td>5.2 Challenges of the hybrid method</td>
<td>57</td>
</tr>
<tr>
<td>5.3 Derivation of unbiased estimators</td>
<td>57</td>
</tr>
<tr>
<td>5.4 Optimal sample allocation</td>
<td>59</td>
</tr>
<tr>
<td>5.5 Implementation of MLMC-PCE</td>
<td>60</td>
</tr>
<tr>
<td>5.6 Numerical results</td>
<td>62</td>
</tr>
<tr>
<td>5.6.1 Comparison with competing methods for the Ishigami example</td>
<td>62</td>
</tr>
<tr>
<td>5.6.2 Application to chemical reaction network results</td>
<td>65</td>
</tr>
<tr>
<td>5.7 Conclusions and future work</td>
<td>69</td>
</tr>
<tr>
<td>5.8 Acknowledgements</td>
<td>70</td>
</tr>
</tbody>
</table>

Chapter 6 GLOBAL SENSITIVITY ANALYSIS OF RARE EVENT PROBABILITIES 71

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1 Introduction</td>
<td>71</td>
</tr>
<tr>
<td>6.2 A motivating example</td>
<td>73</td>
</tr>
<tr>
<td>6.3 Rare event simulation</td>
<td>74</td>
</tr>
<tr>
<td>6.3.1 The subset simulation method</td>
<td>75</td>
</tr>
<tr>
<td>6.3.2 The modified Metropolis-Hastings algorithm</td>
<td>76</td>
</tr>
<tr>
<td>6.3.3 Implementation of subset simulation</td>
<td>77</td>
</tr>
<tr>
<td>6.3.4 Computational cost of subset simulation</td>
<td>78</td>
</tr>
<tr>
<td>6.4 Surrogates for GSA of rare event probabilities</td>
<td>79</td>
</tr>
<tr>
<td>6.5 PCE surrogate for rare event probability</td>
<td>80</td>
</tr>
<tr>
<td>6.5.1 GSA of $P_\tau$ using the PCE surrogate</td>
<td>81</td>
</tr>
<tr>
<td>6.6 Numerical results</td>
<td>82</td>
</tr>
<tr>
<td>6.6.1 Results for the analytic test problem</td>
<td>82</td>
</tr>
<tr>
<td>6.6.2 Subsurface flow application</td>
<td>84</td>
</tr>
<tr>
<td>6.6.3 The statistical model for the permeability field</td>
<td>85</td>
</tr>
<tr>
<td>6.6.4 Definition of the QoI and rare event problem</td>
<td>86</td>
</tr>
<tr>
<td>6.6.5 Rare event probabilities and GSA</td>
<td>88</td>
</tr>
<tr>
<td>6.7 Conclusion</td>
<td>89</td>
</tr>
</tbody>
</table>

Chapter 7 SUMMARY OF CONTRIBUTIONS 91

BIBLIOGRAPHY 93

APPENDIX 101

Appendix A DERIVATION OF ESTIMATORS 102

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>A.1 Single level covariance $C \left[ \left( \hat{\beta}_k \right)^2, \left( \hat{\beta}_z \right)^2 \right]$</td>
<td>102</td>
</tr>
<tr>
<td>A.2 Multilevel MC covariance $C \left[ \left( \hat{\beta}_k \right)^2, \left( \hat{\beta}_z \right)^2 \right]$</td>
<td>104</td>
</tr>
</tbody>
</table>
LIST OF TABLES

Table 3.1  Genetic oscillator reactions, propensity functions, and nominal parameter values, see [66]. .......................................................... 35

Table 4.1  True first order and total Sobol’ indices for the high-fidelity QoI, $q_2$, for $a = 5$ and $b = 0.1$. .......................................................... 53

Table 5.1  GSA method comparison using the Ishigami function and analytical Sobol’ values from [70]. For each method, the mean, variance, and MSE are reported over 1000 realizations. Each method uses a total cost of 2000 and computes all first order and total Sobol’ indices. ............................................ 63

Table 5.2  Levels based on ODE tolerance and final time, $T$, where $\rho_{HF}$ gives the correlation with the highest-fidelity QoI. The cost is represented by the runtime required to evaluate 1000 QoI samples. ............................................ 65
**LIST OF FIGURES**

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 3.1</td>
<td>Schematic representation of the question under consideration: for what type of limiting process is the diagram commutative? The model $q$ is expensive-to-evaluate and stochastic while the surrogate model $\tilde{q}$ is deterministic and cheap. The GSA process results in $d$ sensitivity indices.</td>
</tr>
<tr>
<td>Figure 3.2</td>
<td>25 realizations of Michaelis–Menten trajectories computed via NRM with nominal parameters, varying $\omega$.</td>
</tr>
<tr>
<td>Figure 3.3</td>
<td>Convergence of the product $P^V(t,\omega)$ the corresponding RRE solution at the nominal parameter values plotted as system size grows.</td>
</tr>
<tr>
<td>Figure 3.4</td>
<td>Estimated PDFs of $q_V$ sampled over $\Omega$ and $\Theta \times \Omega$ and $q$ sampled over $\Theta$, respectively.</td>
</tr>
<tr>
<td>Figure 3.5</td>
<td>Histogram and PDF estimates for the total Sobol’ indices for $k_1$, $k_2$, and $k_3$, respectively. Black dashed lines indicate the deterministic value of the RRE total indices.</td>
</tr>
<tr>
<td>Figure 3.6</td>
<td>Convergence of the mean total Sobol’ index as a function of $V$ for parameters $k_1$, $k_2$, and $k_3$, respectively. Note the vertical axes of each figure are not over the same range. The lower and upper bounds of the error bars indicate the 5th and 95th percentiles, respectively.</td>
</tr>
<tr>
<td>Figure 3.7</td>
<td>Histograms at discrete $V$ values of the total Sobol’ indices for $k_1$, $k_2$, and $k_3$, respectively. The vertical axes represent the relative frequency of the indices due to normalized histograms.</td>
</tr>
<tr>
<td>Figure 3.8</td>
<td>Log-log plot of the rate of convergence of the second moment for each total index.</td>
</tr>
<tr>
<td>Figure 3.9</td>
<td>Trajectories of the three dominant species at nominal parameters via the NRM.</td>
</tr>
<tr>
<td>Figure 3.10</td>
<td>Estimated total Sobol’ indices for the genetic oscillator RRE, computed with $N_{MC}$ samples.</td>
</tr>
<tr>
<td>Figure 3.11</td>
<td>PDFs of the stochastic QoI, $q_V$, sampled while fixing the following parameters: black line ($\alpha_A, \beta_A, \delta_{MA}, \alpha_a$), green line ($\gamma_C, \gamma_A, \theta_A, \gamma_R, \theta_R, \delta_A, \delta_R, \delta'_{A}$), red line ($\gamma_C, \gamma_A, \theta_A, \gamma_R, \theta_R, \delta_A$), black line without fixed parameters. Total index thresholds are provided for each PDF.</td>
</tr>
<tr>
<td>Figure 4.1</td>
<td>Estimated PCE coefficients for Ishigami, up to a total polynomial order of 6. Dots denote $E[\hat{\beta}_k]$ and error bars denote 2 standard deviations, computed with 1000 realizations.</td>
</tr>
<tr>
<td>Figure 4.2</td>
<td>Estimated $\nabla[\hat{\beta}_2^k]$ from (4.18) and diagonals of $\nabla[\hat{\beta}_2^k, \hat{\beta}_2^z]$ from (4.21). Left: used $10^5$ samples per level, ensuring convergence. Right: green dots indicate data from 1000 PCE realizations.</td>
</tr>
<tr>
<td>Figure 4.3</td>
<td>Blue bars indicate the variance of the conditional terms, obtained from the derived estimators, with $10^5$ samples at each level. Red bars indicate the variance over 1000 realizations of the conditional terms, with sample allocation $(1000, 100, 10)$.</td>
</tr>
<tr>
<td>Figure 4.4</td>
<td>Blue bars indicate analytic $S_i$ and $T_i$ values for $i = 1, 2, 3$. Red dots indicate the results of 1000 realizations of the conditional variances. Black errorbars indicate one converged estimate of each $\nabla[S_i]$ and $\nabla[T_i]$, reporting 2 standard deviations.</td>
</tr>
</tbody>
</table>
Figure 5.1 PDFs for multiple Sobol’ index realizations from various methods, in this order: single-fidelity Saltelli sampling (Chapter 2), multifidelity GSA results from [70], single-fidelity PCE, and normalized histograms from the hybrid MLMC-PCE method. The black dashed line indicates the analytical value of each index. 64

Figure 5.2 Estimated PCE coefficients for the genetic oscillator up to a total polynomial order of 3, where \((N_0, N_1, N_2) = (10^5, 10^4, 10^5)\). The mean, \(\beta_0\), has been omitted as it is nearly 25000. 66

Figure 5.3 Top: first order ANOVA terms, \(\hat{S}_i\), bottom: total order ANOVA terms, \(\hat{T}_i\), \(i = 1, \ldots, 16\), both in red. Errorbars represent 1 standard deviation, estimated via (4.16). Reference values in blue. 67

Figure 5.4 First order Sobol’ index results, comparing the single-fidelity PCE (blue PDF), the MF-GSA method (orange histogram), and the optimally-sampled hybrid MLMC-PCE method (green histogram). A representative sample of indices are shown, 6 out of 16. Reference values given by black lines. 68

Figure 5.5 Total Sobol’ index results, comparing the single-fidelity PCE (blue PDF), the MF-GSA method (orange histogram), and the optimally-sampled hybrid MLMC-PCE method (green histogram). Again, 6 out of 16 Sobol’ indices are shown. Reference values given by black lines. 69

Figure 6.1 Left: probability density function (PDF) of \(q\) from (6.3) with the rare event threshold \(\bar{\tau} = 3\) indicated by a vertical line; middle: PDF of \(P_3(\xi)\), note that from (6.5), \(P_3(\xi_{nom}) \approx 3.69 \times 10^{-5}\); right: coefficient of variation of \(P_\bar{\tau}(\xi)\) (ratio of standard deviation to mean) as \(\bar{\tau}\) varies. 74

Figure 6.2 Example (6.3) with \(d = 2\). Left: samples in input space generated by MMA sampling with thresholds for each level. Right: QoI output for each level with respective thresholds. 79

Figure 6.3 Total Sobol’ indices of \(P_\bar{\tau}\), with \(\bar{\tau} = 3\), from (6.5); the error bars illustrate the variability of the two sampling methods (Saltelli sampling and sparse regression PCE) around the reference values (blue bars). 83

Figure 6.4 Mean Total Sobol’ indices over 1000 realizations, varying the computational cost of SS and the PCE construction. Each plot varies \(N_{SAMP}\) and each colored bar varies \(N_{SS}\), with the final bar of each index corresponding to the analytic \(P_\bar{\tau}\). 84

Figure 6.5 Left: plots showing two realizations of the log permeability field. Right: the corresponding pressure solution and arrows indicating the resulting Darcy velocity field. 87

Figure 6.6 Left: Histogram of \(q\) for nominal hyperparameters. Vertical line indicates rare event threshold of \(\bar{\tau} = 4.5\). Right: histogram of the rare event probability, estimated via SS with uniformly distributed hyperparameters. 87

Figure 6.7 Left: total Sobol’ indices for \(P_\bar{\tau}(\xi)\) computed from recovered PCE coefficients; results are reported with regularization constant \(\alpha = 1\) and \(\alpha = 5 \times 10^{-2}\). Right: PDF of PCE surrogate compared with \(P_\bar{\tau}\) evaluation histogram. Used \(N_{SAMP} = 10^4\) for better resolution of distributions. 88
Figure 6.8  Distributions of $P_\tau$ for $N_{SS} = 500$ and varying values of $N_{SAMP}$. For each $N_{SAMP}$, we build the PCE surrogate and approximate its PDF with $10^5$ samples. The sets of $P_\tau$ samples used for differing $N_{SAMP}$ are nested within sets of larger samples. Corresponding total indices are included, computed directly from the PCE surrogates.
1.1 Motivation

Across the fields of science and engineering, mathematical modeling is the predominant tool for describing phenomena quantitatively. These models take a variety of forms; they may be algebraic, differential equations, probabilistic, and so forth. In each of these cases, it can be helpful to conceptualize this general model as a relation between inputs and outputs. The outputs can refer to the results of a physical or biological process, the probability of a given system failure, or performance of a financial asset, for example. The set of inputs can refer to physical quantities, experimental conditions, or other relevant data. The appropriate values for these model inputs are not always known a priori. In the majority of applications, obtaining values for these quantities often requires expensive and time-consuming experiments. Input parameters are thus naturally described by random variables, leading to model uncertainty and a probabilistic description of that uncertainty.

As a discipline, uncertainty quantification (UQ) is concerned with the analysis of mathematical models and the effects of uncertainty upon them. Sensitivity analysis is a subfield of UQ, which is concerned with identifying the input factors, or parameters, which are most important to the output of a mathematical model. Information concerning parameter sensitivity can then be used to reduce model complexity, to aid in the design of future experiments, or to enable further analysis, such as optimization or construction of surrogate models.

We consider the task of performing sensitivity analysis on a high-fidelity model, that is, a model which accurately represents the “ground truth” behavior of the system it describes. High-fidelity models tend to be costly to evaluate, and so performing sensitivity analysis on the
high-fidelity model can become prohibitively expensive. Additional challenges to performing sensitivity analysis include high-dimensionality, where the model has a large number of input parameters, and corruption of model evaluations by noise due to random sampling or lack of precision. In this thesis, we will examine the quite common case where the high-fidelity model exists in a family of related models. These related, lower-fidelity, models can take a variety of forms, including coarse-mesh approximations to the solution of a differential equation, simplified physics models, multiscale models with respect to time and space, and surrogate models built from function approximations. In fact, the notion of multiple model fidelities can encompass a very broad range of model families. Equipped with a model family, we will use information from the lower-fidelity models to make sensitivity analysis of the high-fidelity model more effective, and in some cases, make sensitivity analysis tractable where it was originally intractable.

1.2 Contents of the thesis

Chapter 2 will introduce background material necessary for the discussion of sensitivity analysis and the use of polynomial surrogate models. The following chapters will cover a variety of contexts in which sensitivity analysis on a complex and expensive model can be made feasible through the use of related, lower-cost models. Chapter 3 will focus on a problem from chemical kinetics, where chemical reactions can be described with high-fidelity using stochastic models. In the event that sensitivity analysis of the high-fidelity model in infeasible, a deterministic analog of the chemical reaction model can be used to approximate the stochastic dynamics. The author has made the following contributions in Chapter 3:

- A theorem is developed, which shows that the sensitivity indices of the expensive, stochastic model converge to the those of the cheaper, deterministic model.
- A description is given of the conditions under which the convergence theorem applies to a general chemical reaction system modeled by stochastic processes.
- Numerical experiments have been performed with results illustrating this convergence on the well known Michaelis–Menten reaction system.
- Further numerical experiments demonstrate the implications of this theorem when applied to the task of dimension reduction in a high-dimensional system.

In Chapter 4, we will focus on a class of model families, in which the component models can be organized into a hierarchy in terms of cost and accuracy. In this context, multilevel Monte Carlo methods will be combined with polynomial surrogates to make sensitivity analysis feasible, especially for high-dimensional problems. The author has made the following contributions in Chapter 4:

- Novel statistical estimators are derived, which allow one to assess the accuracy of their sensitivity estimates without need for additional computational work.
A hybrid method, combining Monte Carlo sampling with polynomial surrogates, is introduced, with the expressed goal of sampling across fidelities in a manner that is optimal for sensitivity analysis.

Numerical results are presented, which demonstrate the accuracy of the novel estimators and their ability to aid in sensitivity analysis.

Chapter 5 explores a series of practical issues related to the hybrid method introduced in Chapter 4. We will also present a series of numerical experiments, comparing the hybrid method with other competing approaches for sensitivity analysis. The author has made the following contributions in Chapter 5:

- A formula for correcting the estimator bias is derived, which improves estimation accuracy of the sensitivity indices.
- An algorithm implementation is presented, as well as a discussion of how to optimally balance the accuracy in sensitivity analysis and the incurred computational cost.
- Numerical experiments are presented, which demonstrate the superiority of the hybrid method over competing techniques, as well as the analysis of high-dimensional problem from chemical kinetics.

Chapter 6 focuses on the problem of computing sensitivity indices for rare event probabilities, with respect to the hyperparameters that define the distribution of the uncertain parameters. In this context, the rare event probability cannot be evaluated exactly, and so sampling-based methods can be used to estimate the probability with differing levels of fidelity. We will investigate the efficiency of Markov Chain Monte Carlo methods and surrogate models to accelerate the process of sensitivity analysis. The author has made the following contributions in Chapter 6:

- A novel method is proposed for effectively analyzing the sensitivity of the rare event probability to its underlying hyperparameters, which enables sensitivity analysis in scenarios where it has previously been considered infeasible.
- This method combines polynomial surrogates as well as Markov Chain Monte Carlo sampling. An implementation for this method is discussed as well as techniques for surrogate construction which are particularly suitable for the rare event context.
- Numerical experiments are presented, which quantify the sensitivity of the rare event probability for an expensive, PDE-based application, as well as an analytical test problem.

1.3 Overview of the author’s publications

The contents of this thesis derive from four articles which are in various stages of preparation, review, or publication, at the time this thesis was composed. The published articles corresponding
to Chapters 3 and 4 are respectively [1] and [2]. The article corresponding to Chapter 6 is available in preprint form [3]. Finally, a journal article covering Chapters 4 and 5 is in preparation [4]. The author of this thesis is the primary author on each of these articles.
In this chapter, we review background materials on global sensitivity analysis and polynomial chaos expansions.

2.1 Global sensitivity analysis

There are numerous notions of sensitivity, each of them employing different metrics for what constitutes parameter importance. We distinguish first between local and global sensitivity analysis. Local sensitivity analysis (LSA) seeks to characterize model sensitivity at a particular point in the parameter space while global sensitivity analysis (GSA) seeks to characterize sensitivity over a range or subset of the parameter space. The focus of this thesis will be on GSA, which has emerged as a particularly valuable subfield of UQ, and has found numerous applications [5]. We continue with a review of the essential mathematical foundations of GSA.

2.1.1 Variance-based GSA

When considering GSA, it is important first to define a metric by which one quantifies sensitivity. We focus on variance-based GSA, where the variance of the model output is decomposed and apportioned according to its respective components. We begin by defining the probability triple $(\Omega, \mathcal{F}, \mathbb{P})$, where $\Omega$ is a sample space, $\mathcal{F}$ is a corresponding $\sigma$-algebra, and $\mathbb{P}$ is a probability measure on $\Omega$. We then define a random vector, $\theta = (\theta_1, \theta_2, \ldots, \theta_d)$, which maps from $\Omega$ to $\Theta \subseteq \mathbb{R}^d$. Let $\mathcal{B}$ be the Borel $\sigma$-algebra on $\Theta$ and let $\mu$ denote the law of $\theta$. The corresponding probability density function (PDF) of $\theta$ will be denoted as $\pi(\theta)$. Our quantity of interest (QoI)
will then be defined as a scalar-valued function, \( q(\theta) \), which we require to be square-integrable. We assume the components of \( \theta \) are independent random variables, and thus there exists a decomposition of \( q(\theta) \), where each term depends on a unique subset of \( \theta \). This decomposition is called the ANOVA decomposition (short for analysis of variance), and it has the following form:

\[
q(\theta) = q_0 + \sum_{i=1}^{d} q_i(\theta_i) + \sum_{i<j} q_{i,j}(\theta_i, \theta_j) + \cdots + q_{1,...,d}(\theta_1, \ldots, \theta_d).
\] (2.1)

Let \( E[q(\theta)] \) denote the expectation of \( q \) and let \( E[q(\theta)|\theta_i] \) denote the conditional expectation of \( q \) with respect to the set of values that \( \theta_i \) takes. The component terms of (2.1) are defined as

\[
q_0 = E[q(\theta)],
\]

\[
q_i(\theta_i) = E[q(\theta)|\theta_i] - q_0,
\]

\[
q_{i,j}(\theta_i, \theta_j) = E[q(\theta)|\theta_i, \theta_j] - q_i - q_j - q_0,
\]

and so forth. More generally, if we define a subset of the variables, \( \theta_u \), where \( u \subseteq \{1, \ldots, d\} \), then the component function, \( q_u(\theta_u) \), is defined as

\[
q_u(\theta_u) = E[q(\theta)|\theta_u] - \sum_{v \subset u} q_v(\theta_v).
\] (2.2)

In the decomposition (2.1), the terms are pairwise-orthogonal, that is,

\[
E[q_u(\theta_u) q_v(\theta_v)] = 0, \quad \forall u \neq v,
\] (2.3)

which follows directly from the statistical independence of each component of \( \theta \). Using the orthogonality of the ANOVA terms, the variance of (2.1), denoted \( \mathbb{V}[q(\theta)] \), can be expressed as

\[
\mathbb{V}[q(\theta)] = \sum_{i=1}^{d} \mathbb{V}[q_i] + \sum_{i<j} \mathbb{V}[q_{i,j}] + \cdots + \mathbb{V}[q_{1,...,d}],
\] (2.4)

where each component variance term will be denoted as \( V_u = \mathbb{V}[q_u], u \subseteq \{1, \ldots, d\} \). The component terms in (2.4) are known as conditional or partial variances. Thus the decomposed variance of \( q(\theta) \) can be expressed compactly as

\[
\mathbb{V}[q(\theta)] = \sum_{u \subseteq \{1,...,d\}} V_u.
\] (2.5)

We finally arrive at the formal definition of the Sobol’ index [6], named after Ilya M. Sobol’. For a subset \( u \), we define the Sobol’ index with respect to \( u \) as

\[
S_u(q) = \frac{\mathbb{V}[q_u(\theta_u)]}{\mathbb{V}[q(\theta)]}
\] (2.6)
The Sobol’ index quantifies the relative contribution of $\theta_u$ to the variance of $q(\theta)$ and so can be considered a metric for the sensitivity of $q$ with respect to $\theta_u$. For $u$ where $|u| = 1$, the index $S_u$ is called the first order Sobol’ index or the main effect index. One can consider higher order sensitivity indices, where $|u| > 1$. In this context, $S_u$ denotes the contribution of $\theta_u$ to the variance of $q$, excluding the contributions of all proper subsets of $\theta_u$. One may also define the total Sobol’ index,

$$T_i(q) = \sum_{v \ni i} S_v(q). \quad (2.7)$$

The total index, $T_i$, measures the relative importance of variable $\theta_i$, as well as all interaction terms that include $\theta_i$. In practice, computing both main effect and total indices will be desirable, from which one can obtain sensitivity information about individual variables and more general information regarding higher order interactions. A few notable properties of Sobol’ indices include:

1. $0 \leq S_u, T_i \leq 1$, $\forall u \subseteq \{1, \ldots, d\}$ and $\forall i \in \{1, \ldots, d\}$

2. $\sum_{u \subseteq \{1, \ldots, d\}} S_u = 1$

3. $\sum_{i=1}^{d} T_i \geq 1$

The sum of total indices will equal one only if $q$ is an additive function, meaning the ANOVA decomposition of $q$ contains only first order terms. If, by contrast, the sum of the total indices is much larger than one, this indicates very strong higher order interactions, that is, $S_u$ is significant, for $|u| > 1$. In practice, instead of requiring the accuracy of the Sobol’ index to reach a given degree of precision, GSA results are often interpreted qualitatively, enabling one to rank parameters in terms of their importance. For example, a large Sobol’ index for a particular variable clearly indicates a high relative importance. Similarly, variables with a small Sobol’ index can be considered unimportant. However, the notion of what constitutes a “small” or “large” Sobol’ index in practice is a matter of interpretation and pertains to specific applications. Consequently, using GSA results to inform dimension reduction or design of experiments requires some specialized knowledge of a particular application, and so GSA is often part of collaborative efforts between UQ specialists and experts in other domains.

### 2.1.2 Estimation of Sobol’ indices

There are a variety of methods for computing Sobol’ indices, including sampling methods, quadrature, and surrogate-based methods. We briefly cover the widely-used “Saltelli” sampling method [7, 8, 9, 10], which estimates the variance and conditional variances using Monte Carlo (MC) integration. In many GSA studies, the Saltelli method acts as a the baseline against which other GSA methods can be compared [5].
To give some intuition, we denote $N_{MC}$ as the number of MC samples required to estimate a single conditional variance. We can express the conditional variance in the numerator of $S_i$ as

$$\text{Var}[E[q(\theta) | \theta_i]] = \int E[q(\theta) | \theta_i]^2 d\theta_i - \left( \int E[q(\theta) | \theta_i] d\theta_i \right)^2.$$  \hspace{1cm} (2.8)

The final term in (2.8) is simply $q_0^2$, which can be estimated by a sample mean. The first term can be further decomposed in order to produce a sampling scheme. First let $\theta_{\sim i}$ denote all components of $\theta$, with the exception of $\theta_i$. Then we have

$$E[q(\theta) | \theta_i]^2 = \int q(\theta_1, \ldots, \theta_i, \ldots, \theta_d) q(\theta'_1, \ldots, \theta_i, \ldots, \theta'_d) \, d\theta_{\sim i} \, d\theta'_{\sim i},$$

where $\theta'$ is used as an artificial variable to reinterpret the conditional expectation. Continuing,

$$\int E[q(\theta) | \theta_i]^2 d\theta_i = \int q(\theta_1, \ldots, \theta_i, \ldots, \theta_d) q(\theta'_1, \ldots, \theta_i, \ldots, \theta'_d) \, d\theta \, d\theta'_{\sim i},$$

from which we can derive a sampling scheme where $\theta$ and $\theta'$ denote different realizations of the same random vector. Let $A$ and $B$ each denote independent $N_{MC} \times d$ matrices of parameter samples. Then if $A_{Bi}$ denotes the matrix $A$, where the $i$th column has been interchanged with the $i$th column of $B$, we obtain the following estimator,

$$\text{Var}[E[q(\theta) | \theta_i]] \approx \frac{1}{N_{MC}} \sum_{j=1}^{N_{MC}} q(A)_j \, q(A_{Bi})_j - q_0^2,$$  \hspace{1cm} (2.9)

where $q_0^2$ can be estimated at no additional cost using evaluations of $A$ and $B$. This general approach to estimating Sobol’ indices has also come to be known as the “pick and freeze” method, as it samples the QoI by varying one element of $\theta$ at a time. There is a large literature [7, 9, 11] of alternative estimators of a similar flavor, not only for main effect Sobol’ indices, but higher order and total indices. In general, estimating the full set of first order and total Sobol’ indices of $q$ requires $N_{MC}(d + 2)$ function evaluations [7, 8]. The convergence of these estimators follows from the Central Limit Theorem, with a convergence rate that is $O(1/\sqrt{N_{MC}})$ [12]. These estimators have enjoyed widespread use due to their provable convergence properties and simple-to-implement estimators. Nonetheless, they inherit a slow rate of convergence that is characteristic of MC methods and the cost of estimation is likely to be intractable for high-dimensional problems.

### 2.1.3 Alternative GSA methods

While Sobol’ indices have been the predominant tool for GSA within the UQ community, there are a variety of other tools that have been developed. Derivative-based global sensitivity measures (DGSMS) have been introduced as an alternative to Sobol’ indices, when the underlying
QoI is differentiable. Relationships between Sobol’ indices and DGSMs have been proven [13] and there have been efforts to make their computation feasible in practical scenarios [14, 15, 16]. The Morris screening method [17], one of the earliest GSA methods, relies on computing difference approximations of the QoI, known as elementary effects. Although developments in the elementary effects context have become less common in recent years, improved versions have been proposed [18, 19]. The active subspace method [20], which aims to identify important directions in the parameter space, has also gained popularity for its uses in GSA [21, 22, 16]. A sensitivity metric, known as the activity score, with known relations to the Sobol’ index, can be computed as a part of the active subspace workflow [21]. Various other GSA metrics have been developed in recent years, such as moment-independent metrics [5], those for functions with dependent variables [23, 24], function-valued Sobol’ indices [25, 15], and QoIs with multiple sources of uncertainty [26, 1]. Lastly, we highlight the large body of work that has been done in using surrogate models to perform GSA [27, 28, 29, 30], specifically the computation of Sobol’ indices. Surrogates aim to approximate the underlying QoI, while being inexpensive to evaluate, and so can accelerate GSA. The next section will be dedicated to a review of polynomial surrogate methods for GSA.

### 2.2 Polynomial chaos expansions

In recent years, surrogate models have become a popular choice for performing GSA, as well as uncertainty propagation, and other UQ goals. The concept of a surrogate model is fairly broad, encompassing techniques from approximation theory to machine learning [31]. We limit our focus to polynomial chaos expansions (PCE), which are a class of surrogate models that create a spectral expansion of a random variable in terms of orthogonal polynomials [32]. Given the scalar random variable \( q(\theta) \) with finite second moment and statistically independent parameters, we define the PCE of \( q \) as

\[
\tilde{q}(\theta) = \sum_{k=0}^{N_{PC}} \beta_k \Psi_k(\theta),
\]

where \( \{\beta_k\}_{k=0}^{N_{PC}} \) are the set of scalar PCE coefficients, \( \{\Psi_k\}_{k=0}^{N_{PC}} \) is the orthogonal polynomial basis, and \( N_{PC} \) is the truncation level of the expansion. Creating such a surrogate model of the QoI \( q(\theta) \) relies on choosing the proper set of basis functions. In cases where the QoI is expensive, the intent is to construct a surrogate model which is simpler and cheaper to evaluate than the original model.

#### 2.2.1 PCE basis construction

Creating the orthogonal polynomial basis for the PCE relies on information about the random variable itself. The goal is to build a polynomial basis that guarantees orthogonality with respect to the PDF of the input random variable. This property will be especially valuable when we return to the issue of using polynomial surrogates for GSA.
In the univariate case, if \( \theta \) is a continuous random variable, with the PDF \( \pi(\theta) \), then we are able to define a corresponding family of basis polynomials, \( \{ \psi_0(\theta), \psi_1(\theta), \ldots \} \), such that the following holds:

\[
\int_{\Theta} \psi_i(\theta) \psi_j(\theta) \pi(\theta) \, d\theta = 0, \quad \forall i \neq j.
\] (2.11)

For example, if the PDF of \( \theta \) is described by the uniform distribution on the interval \([a, b]\), then the set of Legendre polynomials will satisfy (2.11). Similarly, the set of Hermite polynomials guarantees orthogonality with respect to the PDF of the Normal distribution. For a more detailed list of orthogonal polynomials and their corresponding probability distributions, see the Weiner-Askey scheme in Chapter 2 of [32] or [33].

For a multivariate random variable, \( \theta \), taking values in \( \mathbb{R}^d \), we need to build a multivariate orthogonal polynomial basis. We follow a tensorization approach to creating the basis, which involves taking products of the univariate orthogonal polynomials. If we let \( \{ \psi_0, \psi_1, \psi_2 \ldots \} \) denote the set of univariate orthogonal polynomials, corresponding to a given probability distribution, then the \( k \)th multivariate basis polynomial is defined as

\[
\Psi_k(\theta_1, \ldots, \theta_d) = \prod_{i=1}^{d} \psi_{m^k_i}(\theta_i), \quad m^k = (m^k_1, \ldots, m^k_d),
\] (2.12)

where \( m^k \) is a multi-index, with \( m^k_i \) denoting the degree of the \( i \)th 1D polynomial for the \( k \)th multivariate polynomial. Thus, for a given set of univariate polynomials, the multivariate basis can be completely determined by prescribing the multi-indices.

In practice, the polynomial basis must be truncated and there are a variety of approaches to choosing the finite set of multi-indices, see [34, 35] for more details. For a total order basis construction, we require that the PCE contain all polynomial terms up to a given total polynomial order, \( r \), or formally, the truncated polynomial basis, \( \Psi \), is defined as

\[
\Psi = \left\{ \Psi_k(\theta) : \sum_{i=1}^{d} m^k_i \leq r \right\}.
\]

Using the orthogonality of the univariate polynomials, given by (2.11), the set of multivariate polynomials \( \Psi \), is also orthogonal. Under the total order basis construction scheme, the number of included basis functions is given by the relation

\[
N_{PC} + 1 = \frac{(d + r)!}{d!r!}.
\] (2.13)

The factorial growth of PCE terms is one of the main challenges of this approach as it requires the estimation of \( N_{PC} + 1 \) scalar PCE coefficients. We address one particular method of estimating PCE coefficients in the next section.
2.2.2 Non-intrusive spectral projection

For a PCE defined as (2.10) with an associated polynomial basis, one is left with the challenge of computing \(N_{PC} + 1\) scalar PCE coefficients. Under a non-intrusive framework [32], one only has access to evaluations of the QoI and cannot modify the function itself. Projecting \(q\) onto \(\Psi_k\), the PCE coefficients are defined as

\[
\beta_k = \frac{\mathbb{E}[q(\theta) \Psi_k(\theta)]}{\mathbb{E}[\Psi_k^2(\theta)]}, \quad k = 0, 1, \ldots, N_{PC}.
\]  

(2.14)

In this thesis, we will let \(\Psi_0 = 1\) and \(\beta_0 = \mathbb{E}[q(\theta)]\), as is the usual convention for the PCE [32]. In (2.14), the denominator is the \(L^2\)-norm of the squared polynomial basis element. For most standard families of orthogonal polynomials, the polynomial norms are known analytically [27], and so the main challenge in building a PCE is in approximating the spectral projection in the numerator of (2.14). A variety of methods exist for accomplishing this task including full-tensor quadrature, sparse-grid quadrature, and sampling-based methods [29, 32]. While many quadrature methods can be shown to converge quickly when evaluating low-dimensional integrals of smooth functions, these methods become prohibitively expensive when dealing with high-dimensional and noisy functions [36]. Sparse-quadrature approaches have been developed to alleviate the computational drawbacks of full-tensor quadrature [37], although their performance also degrades as the dimension increases. Despite their slow convergence rate, MC sampling methods have proven to be attractive for evaluating high-dimensional integrals, due to the fact that their convergence rate is independent of the integrand dimension [29]. A main portion of this thesis (including Chapters 4, 5, and 6) will focus on augmenting the MC approach for the purpose of estimating PCE coefficients and downstream GSA. Finally, regression methods deserve mention for their efficiency in computing PCE coefficients, although they do so without evaluating the spectral projection in (2.14). In Chapter 6, we will sketch out the regression approach in more detail, where it will prove to be useful.

A standard MC estimator for \(\beta_k\) would be expressed as

\[
\hat{\beta}_k = \frac{1}{\mathbb{E}[\Psi_k^2]} \frac{1}{N_{MC}} \sum_{i=1}^{N_{MC}} q(\theta^{(i)}) \Psi_k(\theta^{(i)}),
\]  

(2.15)

where we have \(N_{MC}\) independent and identically distributed realizations of \(\theta\). Under this approach, the same \(N_{MC}\) evaluations of \(q\) can be used to estimate the full spectrum of PCE coefficients, mitigating the effects of high-dimensionality. As with other MC approaches, the estimator \(\hat{\beta}_k\) converges at a rate that is \(O(1/\sqrt{N_{MC}})\). Variants of MC have been proposed to accelerate the convergence of (2.15), such as Latin hypercube sampling and, more broadly, Quasi-Monte Carlo sampling [36, 32]. In Chapter 4, we will introduce the multilevel Monte Carlo method [38], which aims to accelerate the convergence of MC by using information from multiple models in a hierarchy. This multilevel framework has been generalized to the multi-fidelity
setting [39], which allows for more flexibility in generating the model hierarchy and building the MC estimators involved.

### 2.2.3 GSA via PCE

We return to the original task of performing GSA, now with the PCE as a tool. Given a PCE surrogate, \( \tilde{q} \), we will leverage the properties of orthogonal polynomials to derive expressions for Sobol' indices (2.6). We begin by using the orthogonality of the PCE basis to compute \( \mathbb{V}[\tilde{q}] \),

\[
\mathbb{V}[\tilde{q}] = \mathbb{E}[(\tilde{q})^2] - \mathbb{E}[\tilde{q}]^2 = \mathbb{E} \left[ \left( \sum_{k=0}^{N_{PC}} \beta_k \Psi_k \right)^2 \right] - \mathbb{E} \left[ \sum_{k=0}^{N_{PC}} \beta_k \Psi_k \right]^2
\]

\[
= \mathbb{E} \left[ \sum_{k=0}^{N_{PC}} \beta_k^2 \Psi_k^2 + 2 \sum_{j < k} \beta_j \beta_k \Psi_j \Psi_k \right] - \beta_0^2 \mathbb{E}[\Psi_0^2] \tag{2.16}
\]

\[
= \sum_{k=0}^{N_{PC}} \beta_k^2 \mathbb{E}[\Psi_k^2] - \beta_0^2 \mathbb{E}[\Psi_0^2] = \sum_{k=1}^{N_{PC}} \beta_k^2 \mathbb{E}[\Psi_k^2].
\]

Next, we compute the conditional variance in (2.6), using the ANOVA decomposition of \( \tilde{q} \) into components for each respective subset of \( \theta \). In order to determine which ANOVA terms are present in the conditional expectation in (2.6), we define the following set for a variable subset, \( u \):

\[
K_u = \{ k : \text{m}_k^i > 0 \ \forall \ i \in u, \ \text{and} \ \text{m}_k^i = 0 \ \forall \ i \notin u \}. \tag{2.17}
\]

For the subset, \( \theta_u \), the set \( K_u \) denotes all PCE terms which only depend on \( \theta_u \) [27]. Similarly, this denotes all terms of the PCE which correspond to the ANOVA term \( \tilde{q}_u(\theta_u) \). As a result, the conditional variance can be computed as

\[
\mathbb{V}[\tilde{q}_u(\theta_u)] = \mathbb{E}[(\tilde{q}_u)^2] - \mathbb{E}[\tilde{q}_u]^2 = \mathbb{E} \left[ \left( \sum_{k \in K_u} \beta_k \Psi_k \right)^2 \right] = \sum_{k \in K_u} \beta_k^2 \mathbb{E}[\Psi_k^2].
\]

Thus we have the expression for the Sobol' index \( S_u \) of \( \tilde{q} \) as

\[
S_u(\tilde{q}) = \frac{\sum_{k \in K_u} \beta_k^2 \mathbb{E}[\Psi_k^2]}{\sum_{k=1}^{N_{PC}} \beta_k^2 \mathbb{E}[\Psi_k^2]} \tag{2.18}
\]

For a more detailed derivation of Sobol' indices via PCE, including total index calculations and a discussion of alternative methods for PCE coefficient computation, see [27]. Throughout the thesis, the PCE will be used as an efficient method for approximating Sobol' indices, with an emphasis on efficient methods for evaluating (2.14), particularly in high-dimensional cases.
3.1 Introduction

In this chapter, we address the task of performing GSA in the context of chemical reaction networks, where the highest fidelity models exhibit parametric uncertainty and additional internal stochasticity. In this context, GSA tends to be quite expensive and so we consider the use of deterministic surrogates, derived from physical principles, to render GSA feasible. We develop herein the notion of stochastic Sobol’ indices and, by examining their relationship to the corresponding deterministic indices, prove a convergence result establishing the usefulness of these surrogates for GSA. We then illustrate these theoretical insights through numerical experiments and, by means of this GSA framework, we perform dimension reduction for a high-dimensional application. The resulting article from this study, written in collaboration with Alen Alexanderian and Pierre Gremaud, was published in the SIAM Journal for Multiscale Modeling and Simulation [1].

The central motivation of this chapter is striking a balance between cost and accuracy. A high fidelity, high cost model, \( q \), is thus often replaced by a lower cost model, \( \tilde{q} \), usually of lower fidelity, to enable the analysis of the problem under study. The techniques used to develop and construct surrogate models are many and range from approximation theory to
simplified-physics [40]. The analysis of the original model, \( q \), is then replaced by the analysis of a surrogate, \( \tilde{q} \), with the implicit assumption that

\[
\text{if } q \approx \tilde{q} \text{ then } I(q) \approx I(\tilde{q}),
\]

where \( I \) represents some operation on \( q \). The extent to which (3.1) is satisfied clearly depends on \( I \) and on the relationship between \( q \) and \( \tilde{q} \). Here, we consider the case where \( I \) stands for the sensitivity of the model to its input parameters. We restrict our attention to an important family of physically-based surrogates, corresponding to \( \tilde{q} \) being obtained through a limiting process of \( q \) and take chemical reaction networks as a motivating application. This approach stands in contrast to the PCE approach of surrogate construction outlined in Section 2.2.

Consider the evolution of a system of chemically reacting molecules; molecular dynamics simulation is the most faithful way of modeling such a system. There, each individual molecule and corresponding species population are tracked and chemical reactions are modeled as distinct events. Due to quantum effects, molecular populations are integer variables which evolve stochastically [41]. In spite of this, chemical kinetics is often analyzed using real, as opposed to integer variables, which evolve deterministically. In this context, such simplified, low-cost models have proven to be very appealing. Stochastic chemical kinetics is, however, necessary to the study of many cellular systems in biology where the relatively small molecular populations may preclude the use of simplified models obtained through the thermodynamic limit (i.e. in the limit of large volumes) and may require a stochastic rather than deterministic model.

Assume we have both a high cost stochastic model, \( q(k, \omega) \), and a low cost deterministic surrogate, \( \tilde{q}(k) \), such that \( q \approx \tilde{q} \), in some sense. In this context, \( \omega \) corresponds to the intrinsic stochasticity of the model, \( q \), and \( k = (k_1, \ldots, k_d) \) is a list of \( d \) uncertain parameters, shared by both models. As shown in Section 3.2, the field of chemical kinetics falls under this framework. A fundamental assumption made in this chapter is that the intrinsic model stochasticity is independent of the randomness in the uncertain parameters. This assumption, which also appears in related works [42, 43, 44], is a natural one from the point of view of modeling under uncertainty. In the present setting, parametric uncertainty, sometimes referred to as epistemic uncertainty, is due to lack of knowledge, whereas model stochasticity, sometimes called aleatoric uncertainty, is inherent to the system and cannot be eliminated by a greater knowledge of the system.

We analyze whether global sensitivity analysis (GSA) can be performed on the surrogate, \( \tilde{q} \), rather than \( q \) and still yield valuable information about the original model, \( q \). In other words, we are asking whether the diagram in Figure 3.1 is commutative.

In Figure 3.1, \( I \) and \( \tilde{I} \) refer to importance indices from some GSA method; presumably, when applied to stochastic models, the GSA approach yields indices which themselves are random variables. This is the case for variance-based GSA, resulting in Sobol’ indices, as discussed in Chapter 2. For a broader discussion of stochastic Sobol’ indices, see [45]. In the case of chemical kinetics, the limiting process in the above diagram is the thermodynamic limit, which we will cover in Section 3.2. The above diagram does not, in general, commute; see [45] for simple
analytical examples of non-commutativity when the limiting process linking the stochastic model to its surrogate is the expectation or some other $\omega$-moment. Similarly, it has been shown that simply taking the average of the stochastic chemical model, $q(k, \omega)$, does not result in the associated deterministic model [46]. A main contribution of this chapter will be to state the circumstances under which Figure 3.1 commutes and to show how this can be advantageous for the GSA process.

### 3.2 Chemical kinetics models

There are a variety of paradigms for modeling chemical systems. Traditional molecular dynamics requires a full characterization of the position and velocity of each chemical molecule in time. This approach, while ideal for fine-grained simulation of reaction dynamics, is ultimately impractical for a system with even a modest number of chemical components [46]. A common simplifying assumption is to consider a well-mixed system, which is spatially homogeneous. One then models the evolution of the chemical system in time only. In this context, the most accurate description of the chemical reaction is provided by the Chemical Master Equations (CME), a system of ODEs that treat the evolution of the chemical system probabilistically. Each ODE in the CME describes the probability of the chemical system being in a particular state for any time $t$. This implies that the CME may easily include hundreds, thousands, or even millions of equations, depending on the number of molecules present. Thus, for most standard chemical systems, the CME are too high dimensional to solve numerically. However, one is able to compute realizations of the chemical system state that are consistent with the probabilistic description given by the CME. In this context, models that produce realizations of the true statistics of the CME are called *exact* [47, 48]. Exact simulation algorithms are the most common choice for reaction modeling, resulting in an extensive literature on the subject [46, 41, 49, 50]. After starting from the theoretical foundations of these algorithms, we will provide a brief overview of these various approaches.
3.2.1 The Random Time Change representation

We consider chemical systems with \( N \) reacting species. Let \( X(t) \) denote the state vector of the chemical system with \( N \) components, where the \( i \)th component of \( X(t) \), \( X_i(t) \), corresponds to the number of molecules of the \( i \)th species at time \( t \). If we let \( M \) denote the number of possible reactions, the evolution of the chemical system can be modeled by the following relation,

\[
X(t) = X(0) + \nu R(t),
\]

where \( \nu \in \mathbb{Z}^{N \times M} \) is known as the stoichiometric matrix, whose \( j \)th column, \( \nu_j \), quantifies the change in \( X \) due to reaction \( j \). The vector, \( R(t) \in \mathbb{Z}^M \), represents the number of times each reaction takes place between time 0 and \( t \). Traditionally, each component of \( R \) is modeled as a stochastic process with the following form [51]

\[
R_j(t) = Y_j \left( \int_0^t a_j(X(s)) \, ds \right).
\]

The stochastic process, \( Y_j \), is a Poisson process, meaning that the number of occurrences or events in a given time interval follows a Poisson distribution [52]. The rate at which each reaction occurs is governed by its respective propensity function \( a_j \), \( j = 1, \ldots, M \), where \( a_j(X(t)) \, dt \) represents the probability that the \( j \)th reaction occurs during the time interval \([t, t + dt]\) [46].

Each Poisson process is a function of an independent variable, which is called its index set [53, Chapter 1]. This index set, often interpreted as time, may take either discrete or, in this case, continuous values. Since the index set of \( Y_j \) is governed by a propensity function, often called the rate of the Poisson process, which varies in time, \( Y_j \) is known as an inhomogeneous Poisson process [52]. Naturally, the number of times a reaction fires, beginning from time \( t = 0 \), is dependent upon the cumulative propensity up to the current time, and so the time-integrated propensity is a suitable index set for each respective Poisson process.

The resulting evolution equation of the chemical system, often referred to as the random time change representation (RTC) [54, 55, 56, 57], is then

\[
X(t) = X(0) + \sum_{j=1}^{M} \nu_j Y_j(\tau_j(t)),
\]

\[
\tau_j(t) = \int_0^t a_j(X(s)) \, ds, \quad j = 1, \ldots, M.
\]

We note that each \( Y_j \) is an independent Poisson process. In (3.5), we follow [50] and define an internal time, denoted \( \tau_j \), for each reaction. Each internal time acts as the index set for its respective Poisson process, rather than the standard interpretation of the index set representing time. This fact will be useful when developing an algorithm to simulate reaction dynamics. In Section 3.3, we will need to distinguish between the internal time for each reaction and the relationship to physical or “global” time. It is conceptually helpful to think of each reaction as
having its own timer or clock, which is related to, but distinct from, the global time.

The Law of Mass Action [54] provides the intuition for creating the propensity functions for each reaction. A natural interpretation of the propensity function is that they are proportional to the number of possible ways a given reaction can occur. Therefore, they can be derived using the appropriate combinatorial formula. Below, letting $S_n$ and $S_m$ denote two generic chemical species, we provide the propensity functions for three of the most common types of reactions [46]:

$$S_m \rightarrow \text{something} \quad \Rightarrow \quad a_j(X(t)) = k_j X_m(t), \quad (3.6)$$

$$S_m + S_n \rightarrow \text{something} \quad \Rightarrow \quad a_j(X(t)) = k_j X_m(t)X_n(t) \quad \text{if } m \neq n, \quad (3.7)$$

$$S_m + S_m \rightarrow \text{something} \quad \Rightarrow \quad a_j(X(t)) = k_j \frac{1}{2} X_m(t)(X_m(t) - 1). \quad (3.8)$$

The reactions (3.6), (3.7), and (3.8) are known as first order, second order, and dimerization reactions, respectively. In order to obtain a general formula for the propensity function of any reaction, we first partition each stoichiometric vector as follows:

$$\nu_j = \nu_j' - \nu_j'', \quad j = 1, \ldots, M, \quad (3.9)$$

where the entries of $\nu_j'$ and $\nu_j''$ are the number of molecules of system species that are created and consumed in the $j$th reaction, respectively. The general formula for obtaining the propensity function is

$$a_j(X(t)) = k_j \left( \prod_{i=1}^{N} (\nu_j''_i)! \right)^{X_i} \left( \frac{X_i}{\nu_j'_i} \right)^{X_i} = k_j \prod_{i=1}^{N} \frac{X_i!}{(X_i - \nu_j''_i)!}. \quad (3.10)$$

A more detailed treatment of mass-action kinetics and propensity functions for other common reaction types can be found, for example, in [49]. Notice that in (3.10), each reaction includes a constant, $k_j$, which is known as a reaction rate constant. These rate constants are often determined experimentally and are therefore subject to *epistemic* uncertainty. This stands in contrast to the irreducible or *aleatoric* uncertainty associated with each Poisson process. We also note that it is not a requirement that a system with $M$ reactions also have $M$ rate constants. A given propensity function can have multiple rate constants or a single rate constant can be shared between multiple propensity functions. As a result, when we address the task of performing GSA in Section 3.4.1, we must distinguish between the number of reactions, denoted by $M$, and the number of uncertain parameters, $d$.

### 3.2.2 An example reaction system

Consider, as an example, the Michaelis–Menten reaction for enzymatic catalysis, a fundamental reaction in the field of chemical kinetics [46, 54, 47], wherein, an enzyme, $E$ binds to a substrate, $S$, to form a complex, $C$. The complex can then either dissociate back into the enzyme and substrate, or it can dissociate into the enzyme and a product, $P$. We have the following reaction
For clarity, we abuse notation, denoting the components of the state vector with the notation of the chemical species in the reaction diagram, \( X = (E, S, C, P) \). From the reaction diagram, the entire RTC equation system can be derived, starting with the stoichiometric matrix and the propensity functions:

\[
\nu = \begin{bmatrix}
-1 & 1 & 1 \\
-1 & 1 & 0 \\
1 & -1 & -1 \\
0 & 0 & 1
\end{bmatrix}
\]

\[
a_1(X(t)) = k_1 E(t) S(t) \\
a_2(X(t)) = k_2 C(t) \\
a_3(X(t)) = k_3 C(t)
\]

Given the stoichiometric matrix and propensity functions, one has all the information necessary to characterize the RTC as in (3.4):

\[
X(t) = X(0) + \begin{bmatrix}
-1 \\
-1 \\
1 \\
0
\end{bmatrix} Y_1 \left( \int_0^t k_1 E(s) S(s) \, ds \right) + \begin{bmatrix}
1 \\
1 \\
-1 \\
0
\end{bmatrix} Y_2 \left( \int_0^t k_2 C(s) \, ds \right) + \begin{bmatrix}
1 \\
0 \\
-1 \\
1
\end{bmatrix} Y_3 \left( \int_0^t k_3 C(s) \, ds \right)
\]

Now that we have reviewed some of the basic theoretical aspects of chemical reaction modeling, we will discuss the thermodynamic limiting process, which acts as the link between the RTC and its deterministic analogue, known as the reaction rate equations.

### 3.2.3 The thermodynamic limit

We consider the limiting behavior of chemical systems as the system size approaches infinity. For example, as the system size increases, the likelihood of a particular reaction occurring may change, in the event that certain molecules must interact. To this end, we aim to update the propensity functions by introducing a system size parameter, \( V = V \cdot n_A \), given by the product of the system volume, \( V \), and the Avogadro number, \( n_A \). We again begin by partitioning the stoichiometric vectors according to (3.9). Following the notation of [58], we define the \( V \)-dependent propensity functions as follows:

\[
a^V_j(x) = \frac{k_j}{V^{\|\nu_j\|}} \prod_{i=1}^N \left( \frac{x_i}{\nu_{ij}} \right), \quad j = 1, \ldots, M,
\]
where the state vector is now denoted by the lowercase $x$, signifying the transition from discrete state variables to continuous values. The $V$-dependent system trajectory is described by the updated RTC representation,

$$X^V(t) = Vx_0 + \sum_{j=1}^{M} \nu_j Y_j \left( \int_0^t a_j^V(X^V(s)) \, ds \right). \quad (3.11)$$

Here we have let $X^V(0) = VX_0$ where $X_0 \in \mathbb{R}^N_{\geq 0}$ is a fixed vector, independent of the system size. In the following discussion, we will work with a sequence of $V$ values such that $VX_0$ is in $\mathbb{Z}^N_{\geq 0}$.

Ensuring existence of such a sequence requires some assumptions on $x_0$ and the nominal (initial) system volume. Specifically, in our study of limiting behavior of systems, we may assume that the system’s nominal volume, $V_{\text{nom}}$, and $x_0$ are such that $V_{\text{nom}}x_0 = V_{\text{nom}}n_Ax_0$ is a vector in $\mathbb{Z}^N_{\geq 0}$. We then consider a sequence of system sizes given by $V_m = mV_{\text{nom}}$, $m = 1, 2, \ldots$. Thus the sequence of system sizes can be parameterized by $m$, which we will informally refer to as the “system size parameter.”

Notice that the RTC formulation (3.11) is a restatement of (3.4), except with the dependence on system size made precise. For instance, considering the system at its nominal volume, $V_{\text{nom}}$, then $X(0)$ in (3.4) is given by

$$X(0) = X^{V_{\text{nom}}}(0) = V_{\text{nom}}x_0 = V_{\text{nom}}n_Ax_0.$$ 

Next, we consider the limit as the system size goes to infinity, also known as the thermodynamic limit, in order to understand the behavior of chemical systems as the volume and number of particles becomes arbitrarily large. We define the limiting propensity functions as in [58], as

$$\bar{a}_j(x) = \lim_{V \to \infty} a_j^V(Vx)/V, \quad j = 1, \ldots, M.$$ 

For example, if the $j$th reaction is as in (3.7),

$$a_j^V(x) = \frac{k_j}{V}x_m x_n \quad \text{and} \quad \bar{a}_j(x) = k_j x_m x_n.$$ 

On the other hand, if the $j$th reaction is of the form (3.8),

$$a_j^V(x) = \frac{k_j}{2V}x_m(x_m - 1) \quad \text{and} \quad \bar{a}_j(x) = \frac{1}{2}k_j x_m^2.$$ 

To describe the thermodynamic limit, we consider the concentration-based state vector $Z^V(t) = X^V(t)/V$. Updating the RTC representation to model the concentration vector, $Z^V$, following [58], we have

$$Z^V(t) = x_0 + \sum_{j=1}^{M} \nu_j V^{-1} Y_j \left( \int_0^t a_j^V(VZ^V(s)) \, ds \right). \quad (3.12)$$
In the limit as $V \to \infty$, the state vector $Z^V$ approaches, almost surely, to a deterministic function, $Z(t)$, that is obtained by solving a system of ODEs known as the system of reaction rate equations (RREs). The corresponding system of RREs is described by

\[
\frac{dZ}{dt} = F(Z(t)) \quad t \in [0,T], \\
Z(0) = x_0,
\]

where $F(z) = \sum_{j=1}^{M} \nu_j \bar{a}_j(z)$ and $[0,T]$ is the maximal interval of existence of solution for (3.13). The following convergence result, originally proven in [56, Chapter 11, Theorem 2.1], provides the necessary foundation for the theoretical developments in Section 3.4.2. Although we state Theorem 1 as it applies to the RTC representation, we note that it can be applied to a more general class of Markov processes. We follow the form of this result as presented in [58]. We also point the reader to [59, Chapter 2], for a more detailed proof of this result.

**Theorem 1.** Let $Z^V$ be a continuous-time Markov process described by the RTC representation in (3.12). If for all compact $K \subset \mathbb{R}^N$, the following conditions hold:

\[
\sum_{j=1}^{M} \|\nu_j\| \sup_{z \in K} \bar{a}_j(z) < \infty, \quad \text{and} \\
F \text{ is Lipschitz on } K,
\]

then, in the limit as $V \to \infty$, $Z^V$ converges, almost surely, to the solution of the corresponding RREs in (3.13). Or formally, we have

\[
\lim_{V \to \infty} \sup_{s \leq T} \|Z^V(s) - Z(s)\| = 0 \quad a.s. - \omega.
\]

Note that here $\| \cdot \|$ denotes the Euclidean norm. Therefore, we know that in the limit, as $V \to \infty$, the stochastic solutions obtained from the RTC model (3.12) will converge to the solution of the ODE system (3.13), with that convergence holding on a set of full measure in $\Omega$. Note also that both of the conditions in (3.14) hold for the chemical systems under study, because the limiting propensity functions, $\bar{a}_j$, $j = 1, \ldots, M$, are polynomials. Theorem 1 establishes the theoretical connection between the stochastic RTC formulation and the RREs. This result is foundational to the analysis in Section 3.4.2, related to the convergence of stochastic Sobol’ indices. Before we turn to the issue of stochastic Sobol’ indices, we briefly address algorithms for simulating chemical systems modeled by the RTC.

### 3.3 The Next Reaction Method

While the simulation of a chemical system from the RREs is straightforward, efficiently simulating the RTC dynamics has been a subject of a great amount of research in recent decades. Several
algorithms have been developed for simulating the dynamics of a stochastic chemical reaction network; these include Gillespie’s stochastic simulation algorithm (SSA) [46, 41] as well as the Next Reaction Method (NRM) of Gibson and Bruck [50] and its variants [60, 47, 42]. The NRM approach has a number of advantages over the SSA, see [60, Section 1] and [61, Section III.B], among others: (i) it is cheaper to simulate than the SSA in terms of random numbers generated per iteration; and (ii) it has the ability to handle time-dependent propensity functions and reactions that exhibit delays between initiation and completion. The variant of the NRM that we use was developed by Anderson in [60], where it is referred to as the modified next reaction method. An outline of the full NRM algorithm for a general reaction network is given in Algorithm 1.

**Algorithm 1** Modified Next Reaction Method [60].

**Input:** Initial state $X_0$, final simulation time $T$, stoichiometric matrix $\nu$, and propensity functions, $\{a_j(\cdot)\}_{j=1}^M$.

**Output:** A realization of $X(t, \omega)$.

1: % initialization %
2: for $j = 1, \ldots, M$ do
3:   Generate random number $r_j \sim U(0,1)$
4:   $\tau_j = 0$, $\tau_j^+ = -\ln(r_j)$
5: end for
6: $t = 0$, $X(0) = X_0$
7: % simulation loop %
8: while $t < T$ do
9:   for $j = 1, \ldots, M$ do
10:      Evaluate $a_j(X(t))$ and $\Delta t_j = \frac{\tau_j^+ - \tau_j}{a_j(X(t))}$
11:   end for
12:   Set $l = \text{argmin}\{\Delta t_j\}_{j=1}^M$
13:   $X(t + \Delta t_l) \leftarrow X(t) + \nu_l$ \{Update state vector\}
14:   $t \leftarrow t + \Delta t_l$ \{Update global time\}
15:   for $j = 1, \ldots, M$ do
16:      $\tau_j \leftarrow \tau_j + a_j \Delta t_l$ \{Update internal times of each reaction\}
17:   end for
18:   Generate random number $r_l \sim U(0,1)$
19:   $\tau_l^+ \leftarrow \tau_l^+ - \ln(r_l)$ \{Update next reaction time for reaction $l$\}
20: end while

The NRM simulates RTC dynamics by treating each reaction as an independent stochastic process: indeed, (3.4), (3.5) correspond to a linear combination of Poisson processes with different internal times $\tau_j$, $j = 1, \ldots, M$. The approach is then to track the firing of each reaction in terms of these internal times. Given the “current” internal time $\tau_j$, $j = 1, \ldots, M$, we denote by $\tau_j^+$ the internal time at which reaction $j$ fires next. By definition, the interval between events
in a Poisson process is distributed as an exponential random variable \([52]\). At each iteration, the vectors \(\tau_1 \tau_2 \cdots \tau_M^+\) and \(\tau_1^+ \tau_2^+ \cdots \tau_M^+\) store the current internal time and the next internal time for each reaction. Given these two vectors, one can determine how much physical or global time will elapse before reaction \(j\) fires again. If one denotes the global time interval between subsequent firings of \(Y_j\) as \([t, t + dt]\), then we consider the difference

\[
\tau_j^+ - \tau_j = \int_0^{t+dt} a_j(X(s)) \, ds - \int_0^{t} a_j(X(s)) \, ds = \int_t^{t+dt} a_j(X(s)) \, ds.
\] (3.16)

We then assume that the interval \([t, t + dt]\) is small enough, such that \(a_j\) is nearly constant over this interval and (3.16) is well-approximated by the left-hand Riemann sum, \(a_j(X(t)) \, dt\). If this assumption holds, then the global time between firings of \(Y_j\) can be computed in practice as

\[
\Delta t_j = \frac{\tau_j^+ - \tau_j}{a_j(X(t))} = \frac{\int_t^{t+dt} a_j(X(s)) \, ds}{a_j(X(t))} \approx \frac{a_j(X(t)) \, dt}{a_j(X(t))}.
\] (3.17)

Thus the global time between subsequent firings of any particular reaction can be computed using only the internal time vectors and evaluations of the propensities. As the size of the system grows and the time interval between reactions becomes smaller, the assumption that \(a_j\) is constant in \([t, t + dt]\) will become better. Given any current global time \(t\), one must determine which reaction will be the next to occur. The index of the next reaction to fire is simply \(l = \text{argmin}(\Delta t_j)\), from which the system state and propensities may be updated and the global time incremented by \(\Delta t_l\). One then updates \(\tau_l^+\) using \(\tau_l^+ = \tau_l^+ + \xi\), where \(\xi\) is exponentially distributed. In Algorithm 1, we note that \(\xi = -\ln(r_l)\) is exponentially distributed, given \(r_l\) is uniformly distributed in the interval \([0, 1]\). After updating \(\tau_l\), each \(\tau_j\) where \(j \neq l\), corresponding to an internal time that has not reached firing, is given the approximate update, \(\tau_j = \tau_j + a_j \Delta t_l\), which makes use of the Riemann sum approximation used in (3.17). This framework has the advantage that the sequence of internal firing times for each reaction can be prescribed without any knowledge of how the state will evolve. In this way, the randomness of each Poisson process can be decoupled from the evolution of the state vector.

### 3.4 Global sensitivity analysis for stochastic models

In this section, we study the convergence of the sensitivity indices corresponding to stochastic models to their deterministic counterparts. In Section 3.4.1, we describe the underlying probabilistic setup, which is necessary for working with stochastic Sobol’ indices. In Section 3.4.2, we present a generic result regarding convergence of the Sobol’ indices of a family of random processes. Then, in Section 3.4.3, we show how the generic convergence result can be applied to stochastic chemical systems.
3.4.1 Theoretical setup

Stochastic models with uncertain parameters present two sources of uncertainties: intrinsic uncertainty due to stochasticity of the system and uncertainty in model parameters. We denote the probability space carrying intrinsic stochasticity of the system by \((\Omega, \mathcal{F}, \nu)\), where \(\Omega\) is the sample space equipped with a sigma-algebra, \(\mathcal{F}\), and a probability measure, \(\nu\). In stochastic chemical systems, the uncertain model parameters of interest are the reaction rates constants, \(k_1, \ldots, k_d\). We model these as independent, uniformly distributed random variables.

Following common practice, we parameterize the uncertainty in the \(k_i\)'s using a random vector, \(\theta = [\theta_1, \ldots, \theta_M]^\top\), whose entries are independent random variables, distributed according to \(U(-1,1)\). For example, if \(k_i \sim U(a_i, b_i)\), then \(k_i(\theta_i) = \frac{1}{2}(a_i + b_i) + \frac{1}{2}(b_i - a_i)\theta_i\).

The uncertain parameter vector, \(\theta\), takes values in \(\Theta = [-1, 1]^d\). It is convenient to work with the probability space, \((\Theta, \mathcal{E}, \lambda)\), for the uncertain parameters, where \(\mathcal{E}\) is the Borel sigma-algebra on \(\Theta\) and \(\lambda\) is the law of \(\theta\), \(\lambda(d\theta) = 2^{-d}d\theta\). The present setup can be easily extended to cases where the \(\theta_i\)'s are independent random variables belonging to other suitably chosen distributions. It is important also to note that while we have \(M\) distinct chemical reactions in our modeling framework, the number of uncertain parameters will be denoted by \(d\).

Recalling Section 2.1.1, we use Sobol’ indices to characterize the sensitivity of a quantity of interest (QoI) to input parameter uncertainties. For example, let \(q(\theta)\) be a scalar-valued QoI defined in terms of the solution of the RREs corresponding to a chemical system. The first order Sobol’ indices corresponding to \(q(\theta)\) are

\[
S_j(q) := \frac{\nabla [E[q(\theta) \mid \theta_j]]}{\nabla [q(\theta)]}, \quad j = 1, \ldots, d. \tag{3.18}
\]

In the context of the RRE model for chemical systems, \(S_j\) quantifies the proportion of the QoI variance due to the \(j\)th reaction rate constant. Higher order Sobol’ indices and total indices may be defined as outlined in Section 2.1.1.

3.4.2 Convergence of stochastic Sobol’ indices

Motivated by the application to stochastic chemical systems, we consider a QoI that incorporates both sources of uncertainty. Consider a family of stochastic processes \(\{q_V(\theta, \omega)\}_{V>0}\) with

\[
q_V(\theta, \omega) : \Theta \times \Omega \to \mathbb{R},
\]

which, as discussed below, are assumed to admit a deterministic limit as \(V \to \infty\). The first order stochastic Sobol’ indices corresponding to \(q_V(\theta, \omega)\) are

\[
S_j(q_V(\cdot, \omega)) := \frac{\nabla [E[q_V(\theta, \omega) \mid \theta_j]]}{\nabla [q_V(\theta, \omega)]}, \quad j = 1, \ldots, d. \tag{3.19}
\]

The following result concerns the convergence of these indices in the limit as \(V \to \infty\).
Theorem 2. Assume the following conditions hold:

1. (Existence of the limiting function). There exists \( q \in L^2(\Theta, \mathcal{E}, \lambda) \) such that, for almost all \( \omega \in \Omega \),
\[
q(V(\theta, \omega)) \to q(\theta), \quad \text{as } V \to \infty, \quad \text{for all } \theta \in \Theta. \tag{3.20}
\]

2. (Boundedness of the stochastic process). For almost all \( \omega \in \Omega \), \( q(V(\theta, \cdot)) \) is \( \mathcal{E} \)-measurable and there exists \( \varphi_\omega(\theta) \in L^2(\Theta, \mathcal{E}, \lambda) \) such that for all \( \theta \in \Theta \),
\[
|q(V(\theta, \omega))| \leq \varphi_\omega(\theta), \quad \text{for all } V > 0. \tag{3.21}
\]

Then the stochastic Sobol’ indices satisfy,
\[
S_j(q(V(\cdot, \omega))) \to S_j(q), \quad \text{as } V \to \infty, \quad \nu\text{-almost surely.}
\]

Proof. By the assumptions of the theorem, there exists a set \( F \in \mathcal{F} \) with \( \nu(F) = 1 \) such that the conditions (3.20) and (3.21) hold for every \( \omega \in F \). By (3.21), we observe that \( q(V(\theta, \omega)) \in L^2(\Theta, \mathcal{E}, \lambda) \), for every \( \omega \in F \) and \( V > 0 \). Thus, we can define the stochastic Sobol’ indices (3.19) for \( \{q(V(\cdot, \omega))\}_{V>0} \), for every \( \omega \in F \).

To show that \( q(V(\theta, \omega)) \to q(\theta) \) in \( L^2(\Theta, \mathcal{E}, \lambda) \), we note that for every \( \omega \in F \)
\[
|q(V(\theta, \omega)) - q(\theta)|^2 \to 0 \text{ pointwise in } \Theta
\]
and
\[
|q(\theta)|^2 = 4 \varphi_\omega(\theta)^2 \in L^1(\Theta, \mathcal{E}, \lambda).
\]
Therefore, invoking the Lebesgue Dominated Convergence Theorem, we have that for all \( \omega \in F \),
\[
\int_\Theta |q(V(\theta, \omega)) - q(\theta)|^2 \lambda(d\theta) \to 0
\]
and thus for every \( \omega \in F \)
\[
\lim_{V \to \infty} \int_\Theta [q(V(\theta, \omega))]^r \lambda(d\theta) = \int_\Theta [q(\theta)]^r \lambda(d\theta), \quad r = 1, 2.
\]
The convergence of the first and second moments of \( q(V(\cdot, \omega)) \) clearly implies
\[
\lim_{V \to \infty} \mathbb{V}[q(V(\cdot, \omega))] = \mathbb{V}[q(\cdot)], \quad \text{for all } \omega \in F.
\]
To finish the proof of the theorem, we need to show
\[
\lim_{V \to \infty} \mathbb{V}[\mathbb{E}[q(V(\cdot, \omega))|\theta_j]] = \mathbb{V}[\mathbb{E}[q(\cdot)|\theta_j]], \quad \text{for all } \omega \in F, j = 1, \ldots, d.
\]
Using the reverse triangle inequality and Jensen’s inequality we observe
\[
\left| \left| \mathbb{E}[q(V(\cdot, \omega))|\theta_j] \right|_{L^2(\Theta)} - \left| \mathbb{E}[q(\cdot)|\theta_j] \right|_{L^2(\Theta)} \right| \leq \left| \mathbb{E}[q(V(\cdot, \omega))|\theta_j] - \mathbb{E}[q(\cdot)|\theta_j] \right|_{L^2(\Theta)} \leq \left| q(V(\cdot, \omega)) - q(\cdot) \right|_{L^2(\Theta)} \leq \pi,
\]

24
and thus, for all \( \omega \in F \),

\[
\lim_{V \to \infty} \| E[q_V(\cdot, \omega) | \theta_j] \|_{L^2(\Theta)} = \| E[q(\cdot) | \theta_j] \|_{L^2(\Theta)}.
\]

Since

\[
\nabla [E[q_V(\cdot, \omega) | \theta_j]] = E [E[q_V(\cdot, \omega) | \theta_j]^2] - E [E[q_V(\cdot, \omega) | \theta_j]]^2
\]

we have, for all \( \omega \in F \),

\[
\lim_{V \to \infty} \nabla [E[q_V(\cdot, \omega) | \theta_j]] = \| E[q(\cdot) | \theta_j] \|_{L^2(\Theta)} - \nabla [E[q(\cdot) | \theta_j]]. \tag{3.22}
\]

This, along with the convergence of the (unconditional) variance implies

\[
\lim_{V \to \infty} V [E[q_V(\cdot, \omega) | \theta_j]] = V [E[q(\cdot) | \theta_j]],
\]

for all \( \omega \in F, j = 1, \ldots, d \). \( \square \)

**Remark 1.** A slight modification of the proof of Theorem 2 leads to a more general result: namely, we can obtain almost sure convergence of the indices,

\[
S_u(q_V(\cdot, \omega)) := \frac{\nabla [E[q_V(\theta, \omega) | \theta_u]]}{\nabla [q_V(\theta, \omega)]}, \tag{3.23}
\]

where \( u = \{j_1, j_2, \ldots, j_s\} \subseteq \{1, 2, \ldots, d\} \) and \( \theta_u = \begin{bmatrix} \theta_{j_1} & \theta_{j_2} & \cdots & \theta_{j_s} \end{bmatrix}^\top \), to \( S_u(q(\cdot)) \).

We recall the total Sobol’ indices from Section 2.1.1 are defined as

\[
T_j(q_V(\cdot, \omega)) := \sum_{u \not\supseteq j} S_u(q_V(\cdot, \omega)), \quad j = 1, \ldots, d. \tag{3.24}
\]

These indices quantify the relative contribution of \( \theta_j \) by itself, and through its interactions with the other coordinates of \( \theta \), to the variance of \( q_V(\cdot, \omega) \). In view of Remark 1, under the conditions of Theorem 2, the total indices show the same convergence properties,

\[
\lim_{V \to \infty} T_j(q_V(\cdot, \omega)) = T_j(q(\cdot)), \quad \text{for almost all } \omega \in \Omega, j = 1, \ldots, d.
\]

### 3.4.3 Application to stochastic chemical kinetics

Consider the (concentration-based) state vector, \( Z_V(t, \theta, \omega) \), of a stochastic chemical system and its deterministic counterpart, \( Z(t, \theta) \), resulting from the thermodynamic limit. Recall that \( \theta \in \Theta \) parameterizes the uncertainty in reaction rate constants. In order to perform GSA, we consider a scalar, time-independent QoI, \( G(Z_V(t, \theta, \omega)) \), and its deterministic counterpart, \( G(Z(t, \theta)) \).
Specifically, $G$ takes a vector function, $z(t)$, and returns a scalar QoI. Examples include
\begin{align}
G(z(t)) &= z_i(t^*) \quad \text{for fixed } t^* \in [0,T] \text{ and } i \in \{1, \ldots, N\}, \quad \text{or} \quad (3.25a) \\
G(z(t)) &= \frac{1}{T} \int_0^T z_i(t) \, dt \quad \text{for a fixed } i \in \{1, \ldots, N\}. \quad (3.25b)
\end{align}

In general, we assume $G: L^\infty([0,T]; \mathbb{R}^N) \to \mathbb{R}$ to be a continuous function. Note that $L^\infty([0,T]; \mathbb{R}^N)$ is equipped with norm $\| \cdot \|_\infty$ given by $\| z \|_\infty = \sup_{t \in [0,T]} \| z(t) \|$, where, as before, $\| \cdot \|$ denotes the Euclidean vector norm.

To adopt the notation of the previous subsection, we consider
\begin{align}
q_V(\theta, \omega) = G(Z^V(t, \theta, \omega)), \quad \theta \in \Theta, \omega \in \Omega,
\end{align}
and the corresponding limiting (deterministic) quantity, $q(\theta) = G(Z(t, \theta))$. Note that by (3.15), for fixed $\theta \in \Theta$, as $V \to \infty$
\begin{align}
\| Z^V(\cdot, \theta, \omega) - Z(\cdot, \theta) \|_\infty \to 0, \quad \text{for almost all } \omega \in \Omega.
\end{align}

Therefore, by the Continuous Mapping Theorem (see e.g. [62]), for each $\theta \in \Theta$,
\begin{align}
q_V(\theta, \omega) \to q(\theta), \quad \text{almost surely } \omega, \quad (3.26)
\end{align}
as $V \to \infty$. We consider the convergence of the stochastic Sobol’ indices, $S_j(q_V(\cdot, \omega))$, to their deterministic counterparts, $S_j(q(\cdot))$, $j = 1, \ldots, d$, as $V \to \infty$ (i.e. in the thermodynamic limit). Here we discuss the conditions necessary for applying Theorem 2 to stochastic chemical systems, which would then imply almost sure convergence of the stochastic Sobol’ indices to their deterministic counterparts.

Theorem 2 requires the existence of a set of full measure in $\Omega$ such that the convergence in (3.26) holds. To ensure this, we consider a modification of $q_V(\theta, \omega)$ as follows. We know that for each $\theta \in \Theta$, there exists a set of full measure $F_\theta \subseteq \Omega$ for which the convergence (3.26) holds. Define
\begin{align}
\tilde{q}_V(\theta, \omega) = \begin{cases} 
q_V(\theta, \omega) & \text{if } \omega \in F_\theta, \\
q(\theta) & \text{otherwise.}
\end{cases}
\end{align}
Note that, we have $\nu(\{\omega \in \Omega : \tilde{q}_V(\theta, \cdot) = q_V(\theta, \omega)\}) = 1$, for every $\theta \in \Theta$. That is, $\tilde{q}_V(\theta, \cdot)$ is a modification of $q_V(\theta, \cdot)$. Note that this modification satisfies the following: for every $\omega \in \Omega$, $\tilde{q}_V(\theta, \omega) \to q(\theta)$ for all $\theta \in \Theta$. With a slight abuse of notation, we will denote this modification by $q_V(\theta, \omega)$ from this point on. To ensure that Theorem 2 applies, we need also the boundedness assumption (3.21).

To discuss the boundedness assumption (3.21), we take a step back and first discuss conditions ensuring boundedness of the stochastic system trajectory, $\{Z^V(t, \theta, \omega)\}_{V>0}$. Consider the state
vector $X^V(t)$. Non-negativity of this state vector requires the propensity functions to be proper \cite{63}: for $j = 1, \ldots, M$, we assume for all $x \in \mathbb{Z}_+^N$, if $x + \nu_j \notin \mathbb{Z}_+^N$, then $a_j^V(x) = 0$. Boundedness of components of $X^V(t)$ requires further (mild) assumptions, as formalized in \cite[Theorem 2.8 and 2.11]{63}. Interestingly, the only requirements concern the stoichiometric matrix, $\nu$. Namely, assuming the existence of a vector $\alpha \in \mathbb{Z}_+^N$ such that $\alpha^\top \nu \leq 0$ and $\alpha_i > 0$ is necessary and sufficient for boundedness of $X^V_i(t)$. Specifically, if such $\alpha$ exists, $\alpha^\top X^V(t) = \alpha^\top (X^V(0) + \nu R(t)) \leq \alpha^\top X^V(0)$. Therefore,

$$X^V_i(t) \leq \frac{1}{\alpha_i} \alpha^\top X^V(0) = \frac{V}{\alpha_i} \alpha^\top x_0.$$  

Thus, in terms of concentrations,

$$Z^V_i(t) = \frac{X^V_i}{V} \leq \frac{1}{\alpha_i} \alpha^\top x_0.$$  

Therefore, we have that the $i$th component of $Z^V$ remains uniformly bounded by $(1/\alpha_i)\alpha^\top x_0$. Moreover, this bound is independent of the reaction rate constants (i.e. independent of $\theta$). Thus, if a vector, $\alpha$, satisfying the aforementioned properties exists for all the components of the state vector, then the concentration based state vector, $Z^V$, remains uniformly bounded by a constant. In fact, we only need to ensure boundedness of the components of $Z^V$ that appear in definition of $G$. Given the function $G$, which defines the QoI, is sufficiently well-behaved, one may argue that $q_V$ inherits the boundedness necessary to satisfy (3.21). For example, if $G$ is defined as in (3.25), then establishing boundedness of $\{Z^V_i(t, \theta, \omega)\}_{V>0}$ is sufficient to satisfy (3.21) for the QoI, $q_V$.

To summarize, with an appropriately defined QoI (including those in (3.25)) and a stochastic chemical system satisfying the aforementioned boundedness properties, the results of Theorem 2 follow, namely, the stochastic Sobol’ indices will converge to the corresponding deterministic Sobol’ indices, and this convergence holds almost surely.

### 3.5 Numerical results

In light of the convergence properties exhibited by stochastic chemical reaction systems, we aim to demonstrate numerically the results of Theorem 2. Convergence results will be presented first for the Michaelis–Menten reaction system, followed by an application of Theorem 2 to the task of dimension reduction, considering a higher-dimensional example arising from the study of genetics. Attention will also be devoted to the computation of Sobol’ indices and the random sampling necessary to compute the stochastic Sobol’ indices introduced in Section 3.4.
3.5.1 Application to the Michaelis–Menten system

We return to the Michaelis–Menten reaction from Section 3.2.2:

\[
S + E \overset{k_1}{\rightarrow} C
\]

\[
C \overset{k_2}{\rightarrow} S + E
\]

\[
C \overset{k_3}{\rightarrow} P + E
\]  

(3.27)

Figure 3.2 depicts 25 realizations of the reaction dynamics using the NRM algorithm with a final time of \( T = 50 \). The parameters, corresponding to the rate constants in the propensity functions, are fixed to the nominal values \( \bar{k}_1 = 10^6 \), \( \bar{k}_2 = 10^{-4} \), and \( \bar{k}_3 = 0.1 \) provided in [64]. Figure 3.2 depicts concentrations of each species for a system size of \( V_{\text{nom}} = n_A V_{\text{nom}} \), where the nominal volume of the reaction system is \( V_{\text{nom}} = 10^{-15} m^3 \).

Figure 3.2 25 realizations of Michaelis–Menten trajectories computed via NRM with nominal parameters, varying \( \omega \).

In Figure 3.3 we illustrate convergence of the RTC trajectories to the RRE trajectories as the system size increases. This plot illustrates the convergence behavior described by Theorem 1. We hold the parameters fixed to their nominal values and plot 25 realizations of the product, \( P(t, \omega) = Z_Y(t, \omega) \), along with the corresponding RRE trajectory. As the system size increases, the ensemble of RTC trajectories converge to the RRE trajectory. In Figure 3.3, the quantity \( m \)
denotes the system size parameter introduced in Section 3.2.3. For the purpose of the simulation, \( m \) is related to the system size by the relation \( V = m V_{\text{nom}} = mn_A V_{\text{nom}} \).

![Figure 3.3](image)

**Figure 3.3** Convergence of the product \( P^V(t, \omega) \) the corresponding RRE solution at the nominal parameter values plotted as system size grows.

3.5.1.1 The QoI

In order to perform GSA, we define the following the stochastic QoI,

\[
q_V(\theta, \omega) = \frac{1}{T} \int_0^T Z^V_4(t, \theta, \omega) \, dt,
\]

where \( Z^V \) is the solution of the RTC. The corresponding deterministic QoI is

\[
q(\theta) = \frac{1}{T} \int_0^T Z_4(t; \theta) \, dt,
\]

where \( Z \) is computed by solving the accompanying RRE. To get a sense of the statistical properties of the QoI, we sample \( q_V \) and \( q \) over the uncertain parameter domain, following a uniform distribution on \( \Theta = [-1, 1]^3 \), and with the uncertain rate constants defined as

\[
k_i(\theta_i) = \bar{k}_i + (0.1\bar{k}_i)\theta_i, \quad i = 1, 2, 3,
\]
where $\bar{k}_i$’s are the nominal reaction rate constants as defined in Section 3.5.1. Figure 3.4 shows probability density functions (PDFs) of $q$ sampled in $\Theta$, $q_V$ sampled in $\Theta \times \Omega$, and $q_V$ sampled in $\Omega$ while using nominal parameters. All samples of $q_V$ used in Figure 3.4 use $V = V_{\text{nom}}$.

![Figure 3.4 Estimated PDFs of $q_V$ sampled over $\Omega$ and $\Theta \times \Omega$ and $q$ sampled over $\Theta$, respectively.](image)

3.5.1.2 Global sensitivity analysis

In this section, we turn to estimating Sobol’ indices in both the stochastic and deterministic settings. We choose to focus specifically on illustrating the convergence of the total Sobol’ indices, although the following results apply generally to the full set of Sobol’ indices.

Sobol’ indices measure the relative contribution of a subset of uncertain parameters to the variance of some QoI. Consequently, it is natural to consider QoIs that are deterministic functions of these uncertain parameters, without any additional variance contributed by a secondary source. When modeling chemical systems using stochastic processes, such as the RTC, the model parameters and internal stochasticity both provide sources of uncertainty, which must be accounted for separately. We summarize the process of estimating Sobol’ indices in the deterministic and stochastic cases in the Algorithm 2, where the number of uncertain parameters is denoted $d$.

In the stochastic setting, fixing a particular $\omega_i$ turns $q_V$ into a function of only the uncertain parameters. From that point, the process of estimating Sobol’ indices is identical to the deterministic case. In the following experiments, we estimate Sobol’ indices using the Saltelli sampling
Algorithm 2 Sobol’ indices for a chemical system with fixed system size.

**Input:** Method of evaluating $q_V(\theta, \omega)$ and $q(\theta)$, $N_{MC}$: number of parameter samples, set of $M_{MC}$ random seeds $\{\xi_i\}_{i=1}^{M_{MC}}$, system size $V$.

**Output:** Total Sobol’ indices: $\{T_1^V(\omega_i), \ldots, T_d^V(\omega_i)\}_{i=1}^{M_{MC}}$ and $\{T_1, \ldots, T_d\}$.

1: Draw $N_{MC}(d+2)$ samples uniformly in $\Theta$ {see [8] for details}
2: % stochastic indices %
3: for $i = 1, \ldots, M_{MC}$ do
4: Seed random number generator with $\xi_i$, corresponding to realization $\omega_i$
5: for $j = 1, \ldots, N_{MC}(d+2)$ do
6: Evaluate and store $q_V(\theta_j, \omega_i)$ samples
7: end for
8: Using $q_V$ samples, estimate Sobol’ indices: $\{T_1^V(\omega_i), \ldots, T_d^V(\omega_i)\}$
9: end for
10: % deterministic indices %
11: for $j = 1, \ldots, N_{MC}(d+2)$ do
12: Evaluate and store $q(\theta_j)$ samples
13: end for
14: Using $q$ samples, estimate Sobol’ indices: $\{T_1, \ldots, T_d\}$

The approach described in Section 2.1.2 (see also [8, Section 4.6] for further details). In Algorithm 2, the cost of estimating all first order and total indices, for each fixed $\omega_i$, is $N_{MC}(d+2)$ evaluations of the QoI, where $N_{MC}$ is user-defined.

The realizations of the stochastic indices correspond to $\omega_i \in \Omega, i = 1, \ldots, M_{MC}$, where one is able to control $\omega$ by specifying the random seed used in the NRM algorithm. We note that the stochastic indices are functions of the system size and so we adopt the compact notation, $T_i^V = T_i(q_V)$ and $S_i^V = S_i(q_V)$. The corresponding deterministic indices do not depend on $V$, as they result from the thermodynamic limit. In Figure 3.5, we compare the distribution of each $T_i^V$ with its respective deterministic limit, $T_i$. In Figure 3.5, the system size corresponds to the nominal $V$, where we have $m = 1$. The deterministic indices, estimated with $N_{MC} = 10^7$

![Figure 3.5](image-url)

**Figure 3.5** Histogram and PDF estimates for the total Sobol’ indices for $k_1$, $k_2$, and $k_3$, respectively. Black dashed lines indicate the deterministic value of the RRE total indices.
samples, are \( T_1 \approx 1.5 \times 10^{-1}, T_2 \approx 1.2 \times 10^{-7}, \) and \( T_3 \approx 8.5 \times 10^{-1}, \) indicating that the third reaction, where the complex dissociates into the enzyme and the product, is the most important and the second reaction, where complex dissociates into the enzyme and substrate, is the least important, contributing almost no variance.

### 3.5.1.3 Convergence of Sobol’ indices

We are now in a position to demonstrate numerically the convergence of the stochastic Sobol’ indices predicted by Theorem 2. One may verify that the conditions on the QoI necessary for Theorem 2 to hold are satisfied in the present case. After we have computed multiple realizations of the stochastic indices at increasing, discrete values of \( V \), we examine the evolution of their distribution as \( V \) increases.

![Figure 3.6](image)

**Figure 3.6** Convergence of the mean total Sobol’ index as a function of \( V \) for parameters \( k_1, k_2, \) and \( k_3, \) respectively. Note the vertical axes of each figure are not over the same range. The lower and upper bounds of the error bars indicate the 5th and 95th percentiles, respectively.

Figure 3.6 demonstrates the convergence of \( \mathbb{E}[T_{i}^{V_m}(\omega)] \) for \( i = 1, 2, 3, \) for increasing values of system size \( V_m = mV_{\text{nom}}, m = 1, \ldots, 200. \) The error bars represent the 5th and 95th percentiles of the distribution of stochastic indices, at a particular system size, where \( M_{MC} = 100 \) different values of \( \omega \) are sampled to construct the distribution for each discrete value of \( V \). Figure 3.6 suggests the convergence of the PDF for each \( T_{i}^{V}(\omega) \) to a Dirac distribution centered at the deterministic value of the Sobol’ index corresponding to the RRE. This sort of convergence may also be demonstrated for lower order Sobol’ indices, as addressed in Remark 1.

Figure 3.7 gives a three-dimensional view of the convergence in Figure 3.6. We plot a series of normalized histograms at specific values of \( m \), converging to Dirac distributions centered at the RRE total indices. These histograms, even for two orders of magnitude difference in \( V \), show a clear trend towards the limiting values given by the RRE.

Figures 3.6 and 3.7 can perhaps most naturally be understood as illustrating the convergence in distribution of the RTC Sobol’ indices, an implication of the pointwise convergence of the PDF. In this case, \( T_{i}^{V}(\omega) \) is the random variable that converges in distribution for each \( i = 1, 2, 3 \) as \( V \) approaches infinity.
We briefly touch on the rate of convergence achieved for the stochastic Sobol’ indices. Figure 3.8 displays the variance of the stochastic total Sobol’ indices for increasing values of $m$. As Figures 3.6 and 3.7 indicate, the variance of the total indices approach zero as the system size approaches infinity. Figure 3.8 indicates that this convergence occurs with a rate of $\mathcal{O}(1/V)$. Here the sample variance is estimated with 100 realizations of the stochastic total Sobol’ indices. We hypothesize that the faster decay of the variance of $T_2(\omega)$ is due to its small size in the thermodynamic limit.

### 3.5.2 Application to the genetic oscillator system

Returning to the original question illustrated in Figure 3.1, we aim to use the sensitivity information from a deterministic chemical model to infer the sensitivities of its stochastic counterpart. The goal is to perform well-informed dimension reduction on the expensive stochastic model, while only requiring samples from the cheaper deterministic model. To perform meaningful dimension reduction, here we consider a higher dimensional model than previously considered. We consider the genetic oscillator system presented in [65], which models the evolution of activator and repressor proteins that govern the circadian clocks of a wide variety of organisms.
The system consists of nine species, including genes, mRNAs, and the two proteins. We have $M = 16$ reactions and $d = 16$ uncertain parameters. Following the form of the chemical system presented in [66], we provide the reaction diagrams, propensity functions, and nominal parameter values in Table 3.1.

As with the Michaelis–Menten system, the RTC models the evolution of the stochastic system and the RRE models the deterministic system, with the two models linked by the thermodynamic limiting process. Figure 3.9 shows a sample trajectory of the stochastic system, simulated via the NRM. In Figure 3.9, all parameters are set to nominal values and the only nonzero initial states are $P_a$ and $P_r$, with one molecule of each. We plot the activator protein, $A$, the repressor protein, $R$, and the complex, $C$ up to final time $T = 50$. We then will use the sensitivity information gained from the cheaper, deterministic model (RRE) to make conclusions about parameter importance in the more expensive, stochastic model (RTC).

We define the stochastic and deterministic QoIs, respectively, as

$$q_V(\theta, \omega) = \frac{1}{T} \int_0^T R_V(t; \theta, \omega) \, dt$$

and

$$q(\theta) = \frac{1}{T} \int_0^T R(t; \theta) \, dt,$$

where $R_V$ is the concentration of the repressor computed via the NRM, $R$ is the concentration of the repressor computed as the solution to the accompanying RRE, and $\theta$ is a random vector that parameterizes the uncertainty in the reaction rate constants. As with the Michaelis–Menten example in the previous section, the parameters will be uniformly distributed 10% about the nominal parameters. Using the Saltelli sampling method, we then estimate the total Sobol’ indices for the deterministic model. Figure 3.10 shows the total Sobol’ indices, computed with $N_{MC} = 10^5$ samples for each total index (recall the cost depends on $d$ for Saltelli sampling). We note that computing a similar number of stochastic QoI samples (on the order of $10^6$) would be
Table 3.1 Genetic oscillator reactions, propensity functions, and nominal parameter values, see [66].

<table>
<thead>
<tr>
<th>Reaction</th>
<th>Propensity Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_a \rightarrow P_a + mRNA_a$</td>
<td>$\alpha_A P_a$</td>
</tr>
<tr>
<td>$P_a - A \rightarrow P_a - A + mRNA_a$</td>
<td>$\alpha_A \alpha_A P_a - A$</td>
</tr>
<tr>
<td>$P_r \rightarrow P_r + mRNA_r$</td>
<td>$\alpha_R P_r$</td>
</tr>
<tr>
<td>$P_r - A \rightarrow P_r - A + mRNA_r$</td>
<td>$\alpha_R \alpha_R P_r - A$</td>
</tr>
<tr>
<td>$mRNA_a \rightarrow mRNA_a + A$</td>
<td>$\beta_A mRNA_a$</td>
</tr>
<tr>
<td>$mRNA_r \rightarrow mRNA_r + R$</td>
<td>$\beta_R mRNA_r$</td>
</tr>
<tr>
<td>$A + R \rightarrow C$</td>
<td>$\gamma_C AR$</td>
</tr>
<tr>
<td>$P_a + A \rightarrow P_a - A$</td>
<td>$\gamma_A P_a A$</td>
</tr>
<tr>
<td>$P_a - A \rightarrow P_a + A$</td>
<td>$\theta_A P_a - A$</td>
</tr>
<tr>
<td>$P_r + A \rightarrow P_r - A$</td>
<td>$\gamma_R P_r A$</td>
</tr>
<tr>
<td>$P_r - A \rightarrow P_r + A$</td>
<td>$\theta_R P_r - A$</td>
</tr>
<tr>
<td>$A \rightarrow \emptyset$</td>
<td>$\delta_A A$</td>
</tr>
<tr>
<td>$R \rightarrow \emptyset$</td>
<td>$\delta_R R$</td>
</tr>
<tr>
<td>$mRNA_a \rightarrow \emptyset$</td>
<td>$\delta_{MA} mRNA_a$</td>
</tr>
<tr>
<td>$mRNA_r \rightarrow \emptyset$</td>
<td>$\delta_{MR} mRNA_r$</td>
</tr>
<tr>
<td>$C \rightarrow R$</td>
<td>$\delta'_A C$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha_A$</td>
<td>50.0</td>
</tr>
<tr>
<td>$\alpha_R$</td>
<td>0.01</td>
</tr>
<tr>
<td>$\beta_A$</td>
<td>50.0</td>
</tr>
<tr>
<td>$\beta_R$</td>
<td>5.0</td>
</tr>
<tr>
<td>$\gamma_C$</td>
<td>20.0</td>
</tr>
<tr>
<td>$\gamma_A$</td>
<td>1.0</td>
</tr>
<tr>
<td>$\theta_A$</td>
<td>50.0</td>
</tr>
<tr>
<td>$\gamma_R$</td>
<td>1.0</td>
</tr>
<tr>
<td>$\theta_R$</td>
<td>1.0</td>
</tr>
<tr>
<td>$\delta_A$</td>
<td>1.0</td>
</tr>
<tr>
<td>$\delta_R$</td>
<td>0.2</td>
</tr>
<tr>
<td>$\delta_{MA}$</td>
<td>10.0</td>
</tr>
<tr>
<td>$\delta_{MR}$</td>
<td>0.5</td>
</tr>
<tr>
<td>$\delta'_A$</td>
<td>1.0</td>
</tr>
<tr>
<td>$\alpha_a$</td>
<td>10.0</td>
</tr>
<tr>
<td>$\alpha_r$</td>
<td>5000</td>
</tr>
</tbody>
</table>

Figure 3.9 Trajectories of the three dominant species at nominal parameters via the NRM.
computationally infeasible for even a modest system size.

![Figure 3.10](image)

**Figure 3.10** Estimated total Sobol' indices for the genetic oscillator RRE, computed with $N_{MC}$ samples.

It is clear that $\alpha_A, \beta_A, \delta_{MA}$, and $\alpha_a$ are the four most important parameters, capturing over 50\% of the variance of the deterministic QoI. We can determine unimportant inputs by putting an importance threshold on the total Sobol' indices; parameters whose Sobol' index falls below the threshold will be considered unimportant. For instance, using 0.02 as a threshold, we identify $\gamma_C, \gamma_A, \theta_A, \gamma_R, \theta_R$, and $\delta_A$ as the six least important parameters, capturing less than 5\% of the variance of the deterministic QoI. We then propose a reduced-dimensional model, where the six least important parameters are fixed at their nominal values, reducing the dimensionality from sixteen to ten. To verify that this lower-dimensional model remains an accurate representation of the full model, we sample the stochastic QoI and plot its PDF, while fixing and varying the unimportant parameters; see Figure 3.11. The red dashed line, corresponding to the reduced model with the six least important parameters fixed has a negligible difference with the PDF of the full model. Increasing the threshold from 0.02 to 0.05 adds $\delta_R$ and $\delta'_A$ to the unimportant category. However, as seen in Figure 3.11, the PDF of the resulting reduced model (dashed green line), obtained by fixing now eight parameters, shows a notable difference with the PDF of the full model. This illustrates the balance one must strike between fixing unimportant parameters to reduce parameter dimension and the loss of information that may result from removing sources of uncertainty. Finally, we illustrate the impact of fixing the four most important parameters (black dashed line in Figure 3.11). This approach fixes every parameter with a total Sobol' index greater than 0.15 ($\alpha_A, \beta_A, \delta_{MA}$, and $\alpha_a$). This results in a substantial underestimation of the variance and a potential loss of valuable model information.

In this case, setting the threshold to 0.02 seems appropriate, resulting in a dimension reduction from 16 to 10, with a negligible change in the QoI output.
3.6 Conclusion

This chapter has focused on the theoretical aspects of performing GSA in the context of two families of related models with differing sources of uncertainty. GSA is often performed on surrogate models with the assumption that (3.1) holds; i.e., that the results from the analysis of a surrogate model will hold for the original model. We have presented here a partial result in that direction, showing this assumption to be true for a specific specific class of problems (chemical systems), a specific type of surrogate (obtained from the thermodynamic limit) and a specific GSA approach (Sobol’ indices). This study not only shows and justifies, in an arguably restricted framework, that GSA can sometimes be done on a reduced computational budget, we argue that it reflects important properties of the GSA methods themselves.

While this chapter focused on a specific type of modeling hierarchy, defined by varying chemical system sizes, these ideas do fit into a broader context of using multiple related models to inform the GSA process. In subsequent chapters of this thesis, we consider other notions of model hierarchy, including those derived from different resolution levels of a numerical method and from estimated probabilities with varying levels of fidelity.
Acknowledgements

We acknowledge the computing resources provided on Henry2, a high-performance computing cluster operated by North Carolina State University. We also thank Andrew Petersen for his assistance with distributed and parallel computations, which was provided through the Office of Information Technology HPC services at NC State.
4.1 Introduction

In this chapter, we develop a hybrid GSA framework, pairing the polynomial chaos expansion, introduced in Chapter 2, with a sampling-based approach, known as multilevel Monte Carlo. This method leverages the advantages of surrogate models for GSA and the robustness of MC sampling, especially in the case of high-dimensional functions and functions lacking regularity. We also seek to create a method that allows one to strategically allocate computational resources for the purpose of Sobol’ index computation. This chapter is primarily concerned with the theoretical foundations of the hybrid approach, although verification results will be presented to illustrate key aspects of the method. Aside from these illustrative tests, we reserve a full discussion of the practical aspects of this method, including implementation and detailed numerical tests, for Chapter 5. The contents of this chapter were developed in collaboration with Gianluca Geraci, Mike Eldred, and Teresa Portone of Sandia National Laboratories. The results of this work include an article published as a contribution in the 2020 CSRI Summer Proceedings [2] as well as a forthcoming journal article [4], which is currently in preparation.

Researchers in computational science continue to expand the state of the art in high-fidelity modeling and simulation, including complex multiphysics and multiscale simulations. In the
context of GSA, this often leads to the dual challenge of high computational expense and high
dimensionality, as driven by this increasing model complexity. As discussed in Chapter 2.1.1,
traditional techniques for GSA, specifically Saltelli sampling, rely on evaluating potentially high-
dimensional integrals through MC integration. These MC methods are simple to implement and
their asymptotic convergence properties are independent of the dimensionality of the problem.
However, the convergence rate of these MC methods remains slow, requiring a large number of
evaluations of the QoI to obtain reliable results [7, 9].

When faced with a model that is computationally expensive to simulate, it is often advanta-
geous to consider a hierarchy of related models with differing levels of fidelity and associated
computational cost. This approach has become popular, specifically through the use of multilevel
Monte Carlo (MLMC) [38], which aims to accelerate MC convergence by leveraging information
from multiple models across the hierarchy. These ML models are often found in the context of
using a hierarchy of mesh refinement levels when solving a differential equation [67], although
they encompass a broader class of models, such as financial models and risk management,
biochemical reaction networks, and groundwater flow applications [38]. The concept of leveraging
multiple model fidelities has been generalized to more heterogeneous model hierarchies than
those treated in the MLMC framework, giving rise to the notion of multi-fidelity UQ methods
such as multi-fidelity Monte Carlo (MFMC) and approximate control variate methods [39, 68].

The use of a hierarchy of models to accelerate Sobol’ index computation is an active area
of research, with previous efforts being made in both the MLMC context [69] and the more-
general MFMC context [70]. While these previous efforts have been successful in reducing the
estimation error well below that of single-fidelity MC, they use independent sets of QoI samples
for estimating each Sobol’ index. As a result, the cost of these methods scales with the parameter
dimension, limiting their utility in high-dimensional applications.

The polynomial chaos expansion (PCE), covered in Section 2.2, can be advantageous for
GSA in high-dimensional settings. Using a PCE allows one to link each term in the ANOVA
decomposition directly to the relevant PCE coefficients. In this case, although the number of
coefficients is expected to increase with input dimension, the model evaluations required to
estimate each PCE coefficient can be shared. The only factor that differentiates each respective
coefficient is that the polynomial basis differs, although for high-fidelity models, the compared
cost of evaluating the basis is typically negligible.

The focus of this chapter is the fusion of MLMC methods with the PCE, which will combine
the dimension-independent convergence properties of MC with the dimension-independent cost of
performing GSA via the PCE. In the resulting hybrid MLMC-PCE method, the model hierarchy
can be leveraged to improve the estimation of PCE coefficients, and by extension, Sobol’ indices.
Furthermore, the hybrid method proposed in this chapter will take into account how the MC
samples allocated to each level of the hierarchy affect the estimation of each PCE coefficient
and the downstream effect on the estimation of the Sobol’ indices. This analysis will allow for
goal-oriented computation of Sobol’ indices, where the ML sample allocation will be optimized
in order to balance the error in a chosen set of GSA targets and the associated computational cost.

4.2 Review of GSA and PCE background

Building from the introduction to GSA and Sobol’ indices provided in Chapter 2, we define the specific GSA features which will be relevant to our discussion. Given a model \( q(\theta) : \Theta \rightarrow \mathbb{R} \) and a set of uncertain parameters, \( \theta = [\theta_1, \ldots, \theta_d]^\top \in \Theta \subseteq \mathbb{R}^d \), we provide the a slightly modified form of the ANOVA decomposition of variance covered in Section 2.1.1,

\[
\mathbb{V}[q(\theta)] = \sum_{u \subseteq \{1,2,\ldots,d\}} S_u, \tag{4.1}
\]

where each conditional variance is defined as in (2.6).

Using the notation of (4.1), we then define the Sobol’ index, \( S_u \), as

\[
S_u = \frac{S_u}{\mathbb{V}[q]}, \tag{4.2}
\]

Distinguishing between the numerator and denominator in (4.2) will be necessary in the following discussion, as each term requires different statistical estimators. The total Sobol’ index can be defined in a similar manner as (4.2),

\[
T_i = \frac{T_i}{\mathbb{V}[q]}, \quad \text{where} \quad T_i = \sum_{u \subseteq \{1,\ldots,d\}} S_u \tag{4.3}
\]

As covered in Chapter 2, Sobol’ indices are often computed either using MC sampling methods to estimate the various conditional variance terms (i.e. Saltelli sampling) or by use of a surrogate model, such as the PCE. We focus specifically on PCE-based Sobol’ index computation, as introduced in Section 2.2. Recall, the PCE of \( q \) is defined as

\[
\tilde{q}(\theta) = \sum_{k=0}^{N_{PC}} \beta_k \Psi_k(\theta), \quad \text{where} \quad \beta_k = \frac{\mathbb{E}[q(\theta) \Psi_k(\theta)]}{\mathbb{E}[\Psi_k^2(\theta)]}. \tag{4.4}
\]

Again, \( \{\Psi_k\}_{k=1}^{N_{PC}} \) is a family of orthogonal polynomials and \( \{\beta_k\}_{k=0}^{N_{PC}} \) is the corresponding set of PCE coefficients. Computing the set of PCE coefficients tends to be expensive when the input dimension of \( q \) is large, because the number of terms included in the expansion, denoted by \( N_{PC} \), grows factorially with the dimension (see (2.13)). As a result, methods for PCE construction based on regression could require the solution of large linear systems with associated issues related to memory requirements and numerical precision [27]. Quadrature-based methods will be the method of choice for low-dimensional QoIs that have sufficient regularity [36]. However, for high-dimensional QoIs, even sparse quadrature will become infeasibly expensive. For example,
see [37, Section 6.4], where a level 5 sparse-grid approach in 10 dimensions requires over 77,000 function evaluations. For applications where the QoI is both high-dimensional and expensive, requiring hours or even days for a single function evaluation, this approach infeasible. When dealing with these computational challenges, which are commonplace in large-scale applications [5, Section 3.3], Monte Carlo is often the only viable method for computing a large number of high-dimensional integrals and thus performing GSA.

As previously shown in Chapter 2, given a PCE for \( q \), one is able to compute Sobol’ indices analytically as in (2.18). The question then remains whether the Sobol’ indices of the PCE surrogate are an accurate representation of the Sobol’ indices of the underlying QoI. Suppose \( S_u \) represents the Sobol index of \( q \) with respect to \( u \subseteq \{1, \ldots, d\} \) and \( \tilde{S}_u \) represents the Sobol index obtained from a PCE surrogate. The mean-squared error (MSE) can be decomposed as

$$
\mathbb{E} \left[ (\tilde{S}_u - S_u)^2 \right] = \mathbb{E} \left[ (\tilde{S}_u)^2 \right] - \mathbb{E} \left[ \tilde{S}_u \right]^2 + \left( S_u - \mathbb{E} [\tilde{S}_u] \right)^2.
$$

(4.5)

The variance and bias are both properties of a statistical estimator, however, minimizing one of these quantities does not necessarily minimize the other. We will seek to construct unbiased estimators for each Sobol’ index (i.e. zero bias) and to derive sampling methods that achieve an optimal level of variance reduction. We will next introduce the multilevel Monte Carlo method, specifically for the purpose of variance reduction in GSA computations. We will return to the problem of constructing unbiased estimators in Section 5.3.

4.2.1 Monte Carlo and multilevel Monte Carlo sampling

We begin with the idea of different fidelities or resolution levels for the QoI. Let \( q_L \) denote an approximation of \( q \) at the highest resolution level available. A natural context for this notation is when \( q \) derives from the solution of a differential equation and the subscript \( L \) represents the finest mesh one is able to use to solve said differential equation. If one wishes to estimate the expectation \( \mathbb{E}[q_L(\theta)] \) using MC sampling, they may use the sample average estimator,

$$
\hat{q}_L = \frac{1}{N_{MC}} \sum_{i=1}^{N_{MC}} q_L(\theta^{(i)}),
$$

(4.6)

where \( N_{MC} \) is the number of samples drawn from the joint probability distribution of \( \theta \). Let \( \theta^{(i)}, i = 1, \ldots, N_{MC} \) denote \( N_{MC} \) realizations of \( \theta \), which are independent and identically distributed.

Notice the estimator, \( \hat{q}_L \), computed using a finite number of samples, is itself a random variable with its own mean, variance, and higher moments. Furthermore, the finite resolution associated with \( q_L \) introduces an error with respect to the “true” QoI, \( q \), and thus the MSE of
the estimator, \( \hat{q}_L \), accounts for this bias with the same decomposition as in (4.5),

\[
E[(\hat{q}_L - E[q])^2] = E[(\hat{q}_L)^2] - (E[\hat{q}_L])^2 + (E[q])^2 - (E[q])^2.
\]

Thus, to improve the quality of this particular MC estimate there are two terms to consider: the variance and the bias. In the context of using an approximation, \( q_L \), the variance term measures the error due to sampling, while the bias measures the error due to approximation of the QoI. Due to the fact that the samples of \( \theta \) are i.i.d., one is able to further decompose \( \text{Var}[\hat{q}_L] \) as

\[
\text{Var}[\hat{q}_L] = \text{Var} \left[ \frac{1}{N_{MC}} \sum_{i=1}^{N_{MC}} q_L(\theta^{(i)}) \right] = \frac{1}{N_{MC}^2} \sum_{i=1}^{N_{MC}} \text{Var} \left[ q_L(\theta^{(i)}) \right] = \frac{\text{Var}[q_L]}{N_{MC}}.
\]

(4.7)

In this case, because (4.6) is an unbiased estimator, the MSE of the estimator is equal to the variance, which decays at a rate of \( O(1/N_{MC}) \). This slow rate of convergence is a common shortcoming of standard MC methods. Several approaches have been proposed to reduce the variance of \( \hat{q}_L \); see for instance [38, 68, 39]. Among the various variance reduction strategies, multilevel Monte Carlo (MLMC) represents the prototypical example. While the work presented in this thesis will be primarily focused on MLMC, we emphasize that extensions to multi-fidelity and control variate strategies have allowed one to generalize this approach [39, 68].

Let \( q_0, q_1, \ldots, q_L \) be a hierarchy of models indexed by \( \ell \), where an increasing \( \ell \) corresponds to an increasing fidelity or accuracy. This model hierarchy is analogous to a quantity derived from the solution of a differential equation, where an increasing \( \ell \) corresponds to an increasing number of mesh points. Here \( q_L \) is the highest-fidelity model and the goal is to efficiently estimate \( E[q_L] \) by making use of the lower-level model evaluations. Typically, this approach is advantageous, due to the fact that the cost of evaluating \( q_\ell \), denoted \( c_\ell \), follows the relation \( c_0 \leq c_1 \leq \cdots \leq c_L \). Under these conditions, leveraging cheaper model evaluations can greatly reduce the cost of estimating \( E[q_L] \). Using the linearity of the expectation operator, we observe

\[
E[q_L] = E[q_0] + E[q_1] - E[q_0] + \cdots + E[q_L] - E[q_{L-1}] = \sum_{\ell=0}^{L} E[q_\ell - q_{\ell-1}],
\]

(4.8)

where \( q_{-1} = 0 \), by convention. Thus, we are able to estimate the mean of the difference between adjacent levels, and combine these to form an estimate of the mean at the highest level of accuracy. We define \( Y_\ell = q_\ell - q_{\ell-1} \) and, by forming a MC estimate of each term in (4.8), we obtain the expression for the MLMC estimator,

\[
\hat{q}_L^{ML} = \sum_{\ell=0}^{L} \hat{Y}_\ell = \sum_{\ell=0}^{L} \frac{1}{N_\ell} \sum_{i=1}^{N_\ell} (q_\ell^{(i)} - q_{\ell-1}^{(i)}).
\]

(4.9)

In the MLMC framework, one samples each level independently, meaning a newly generated
set of inputs is used at each level. The result is zero covariance between the levels in (4.9) (i.e. \( \mathbb{C}[Y_\ell, Y_{\ell'}] = 0, \ \ell' \neq \ell \)). Thus the variance of \( \hat{q}_L^{ML} \) can be expressed as

\[
\mathbb{V}[\hat{q}_L^{ML}] = \sum_{\ell=0}^{L} \mathbb{V}[\hat{Y}_\ell] = \sum_{\ell=0}^{L} \frac{\mathbb{V}[Y_\ell]}{N_\ell}
\]

(4.10)

and by the linearity of the expectation, \( \mathbb{E}[\hat{q}_L^{ML}] = \mathbb{E}[q_L] \), thus (4.9) is an unbiased estimator of \( \mathbb{E}[q_L] \).

The goal of this approach is to derive an estimator for \( \mathbb{E}[q_L] \) with lower variance than the standard MC estimator in (4.6), without increasing the cost. In the standard convergence analysis for MLMC, we require that \( \mathbb{V}[Y_\ell] \) decreases monotonically as \( \ell \to L \) (see [38, Theorem 1]). As a result, fewer samples can be allocated to each successive level of the model, which become increasingly expensive to evaluate. We thus distribute the computational cost across the model hierarchy, with an emphasis on leveraging the cheaper, lower-fidelity models.

The MSE of \( \hat{q}_L^{ML} \) can be decomposed, as before, in terms of the variance and bias

\[
\mathbb{E}[(\hat{q}_L^{ML} - \mathbb{E}[q])^2] = \mathbb{V}[\hat{q}_L^{ML}] + (\mathbb{E}[q_L] - \mathbb{E}[q])^2.
\]

(4.11)

Notice that because (4.9) is an unbiased estimator of \( \mathbb{E}[q_L] \), the bias in (4.11) results from the fact that \( q_L \) is an approximation of \( q \).

The optimal sample allocation is defined in order to minimize the total computational cost across levels, while achieving a balance between the bias and variance in (4.11). Defined in this manner, the optimal sample allocation can be derived in closed form [38]. We will elaborate on the specifics of the optimal sample allocation in the following section, which concerns using MLMC to estimate PCE coefficients.

4.2.2 MLMC estimation of PCE coefficients

We will now describe how MLMC can be used to estimate a particular PCE coefficient. Recall, that in (4.4), the polynomial norm in the denominator is known analytically and so the main cost is in estimating the spectral projection in the numerator. Thus, the MLMC estimator for \( \beta_k \) is

\[
\hat{\beta}_k = \frac{q_L \Psi_k}{\mathbb{E}[\Psi_k^2]} = \frac{1}{\mathbb{E}[\Psi_k^2]} \sum_{\ell=0}^{L} \hat{Y}_\ell \Psi_k = \frac{1}{\mathbb{E}[\Psi_k^2]} \sum_{\ell=0}^{L} \frac{1}{N_\ell} \sum_{i=1}^{N_\ell} (q_\ell^{(i)} - q_{\ell-1}^{(i)}) \Psi_k^{(i)}.
\]

(4.12)

For notational simplicity, we will denote \( P_{\ell,k} = (q_\ell - q_{\ell-1}) \Psi_k \) and \( b_k = \mathbb{E}[\Psi_k^2] \). We also let \( C_\ell \) denote the cost of evaluating the difference of \( q_\ell \) and \( q_{\ell-1} \), instead of using \( c_\ell + c_{\ell-1} \). In keeping with the PCE literature, we assume that the cost of evaluating the orthogonal polynomials is negligible and thus \( C_\ell \) also denotes the cost of evaluating \( P_{\ell,k} \) for any \( k \). Therefore, the total
cost of estimating $\beta_k$ as in (4.12) is given by

$$C_{\text{tot}} = \sum_{\ell=0}^{L} N_\ell C_\ell.$$ 

To formulate the optimization problem for sample allocation, we consider the variance of $\hat{\beta}_k$, where again we enforce independent sampling on each level:

$$\mathbb{V}[\hat{\beta}_k] = \mathbb{V} \left[ \frac{1}{b_k} \sum_{\ell=0}^{L} \frac{N_\ell}{N_\ell} \sum_{i=1}^{N_\ell} P_{\ell,k}^{(i)} \right] = \frac{1}{b_k^2} \sum_{\ell=0}^{L} \mathbb{V}[P_{\ell,k}] N_\ell.$$ 

The optimal sample allocation can be derived by solving the minimization problem

$$\arg \min_{N_0, \ldots, N_L} \sum_{\ell=0}^{L} N_\ell C_\ell + \lambda^2 \left( \mathbb{V}[\hat{\beta}_k] - \varepsilon^2 \right),$$

where $\lambda^2$ is a Lagrange multiplier and $\varepsilon^2$ is the desired accuracy of the estimator. The optimization problem (4.13) is simply an application of the canonical MLMC optimization problem to the task of estimating PCE coefficients. The solution of the canonical MLMC problem is derived in the work of Giles [38]. For a target estimator variance of $\varepsilon^2$, the optimal sampling conditions can be expressed in closed form as

$$N_\ell = \lambda \sqrt{\frac{\mathbb{V}[P_{\ell,k}]}{b_k^2 C_\ell}} \quad \text{where} \quad \lambda = \varepsilon^{-2} \sum_{\ell=0}^{L} \sqrt{\mathbb{V}[P_{\ell,k}]} \frac{C_\ell}{b_k}.$$ 

If the variance is minimized subject to an upper bound on the cost, the resulting optimal sample profile has the same proportion of samples per level as in (4.13) (see [67]). From a practical standpoint, it is worth noting that the statistics of $P_{\ell,k}$, which are needed to compute the optimal sample profile, are not known a priori. As a result, it is standard practice to proceed iteratively, alternating between sampling the QoI and computing the optimal allocation until the relevant statistics have converged [38].

While this example illustrates the foundational ideas of MLMC and optimal sample allocation, we emphasize that our goal is to use MLMC for GSA. Therefore in the next section we extend the optimal sample allocation problem to target an ensemble of PCE coefficients, and by extension, the Sobol’ indices themselves.

### 4.3 Theoretical development

To extend the MLMC optimization to target Sobol’ indices, we first consider how the uncertainty in the PCE propagates through to the Sobol’ indices. Just as with individual PCE coefficients, the goal is to achieve a reduction in the variance of a particular target. In this case, we consider
the variance of $\hat{S}_u$, which denotes the PCE estimator for $S_u$, rather than the traditional Sobol’ index, which is a ratio of variances. In general, there is no closed-form solution for the variance of a ratio of random variables (see [71, Chapter 6]). Instead, we decompose the variance of $\hat{S}_u$, as obtained from the PCE. It is important to note that each estimated PCE coefficient, $\hat{\beta}_k$, is a random variable and therefore the variance of the PCE surrogate can be considered a random variable as well. We will first expand the PCE variance, $V[\hat{q}]$, in terms of its constituent random variables. After this, we derive an estimator for the variance of $V[\hat{q}]$. Decomposing the variance of $V[\hat{q}]$ will allow one to quantify the uncertainty in each Sobol’ term, as it relates to the uncertainty in the corresponding PCE terms. Finally, a sample allocation scheme will be formulated in order to achieve a minimal variance estimate of the desired Sobol’ indices.

In the following discussion, let $\hat{\beta}_k^2$ denote the square of the estimated $k$th PCE coefficient, rather than an estimator for the squared coefficient itself. We begin with the PCE variance, given in (2.16), as

$$V[\hat{q}] = \sum_{k=1}^{N_{PC}} \hat{\beta}_k^2 b_k.$$  \hspace{1cm} (4.14)

Since each $\hat{\beta}_k^2$ is a random variable, (4.14) can be viewed as an estimator for the PCE variance, having its own statistical properties. Thus we may decompose the variance of (4.14) as

$$V\left[ \sum_{k=1}^{N_{PC}} \hat{\beta}_k^2 b_k \right] = \sum_{k=1}^{N_{PC}} b_k^2 V[\hat{\beta}_k^2] + \sum_{k=1}^{N_{PC}} \sum_{z=1, z \neq k} b_k b_z C[\hat{\beta}_k^2, \hat{\beta}_z^2].$$  \hspace{1cm} (4.15)

Equation (4.15) not only contains the variance of each squared PCE coefficient, it also incorporates the interaction terms between coefficients, expressed as covariances. Similar to the ANOVA decomposition, it is now possible to decompose the variance of a PCE-computed Sobol index. Recall that computing the Sobol’ indices from a PCE (see (2.18)) involves summing over a particular set of PCE coefficients for each $S_u$, denoted $K_u$ (see (2.17)). Therefore, we are now able to decompose the variance of $\hat{S}_u$ as

$$V[\hat{S}_u] = V\left[ \sum_{k \in K_u} \hat{\beta}_k^2 b_k \right] = \sum_{k \in K_u} b_k^2 V[\hat{\beta}_k^2] + \sum_{k \in K_u} \sum_{z \in K_u, z \neq k} b_k b_z C[\hat{\beta}_k^2, \hat{\beta}_z^2].$$  \hspace{1cm} (4.16)

Thus one is able to target the variance of a particular Sobol’ index and express that variance in terms of its dependence on each PCE coefficient. However, more information is required to accurately estimate the variances and covariances in (4.16). One must be able to characterize (4.16) in terms of information available to the user, namely evaluations of the multilevel QoI and evaluations of the polynomial bases. Thus the goal is to derive an alternate expression for $V[\hat{S}_u]$ in terms of the statistics of the $P_{l,k}$ terms, from which a sample allocation scheme can be derived, as in Section 4.2.2.
4.3.1 Derivation of variance and covariance terms

We begin by deriving an expression for $\hat{\beta}_k^2$ from (4.16) in terms of raw moments (i.e. moments centered around zero) of $q_\ell$, $\ell = 0, \ldots, L$ and $\Psi_k$, $k = 0, \ldots, P$. For notational convenience, we will seek to use the raw moments of the product term, $P_{\ell,k} = (q_\ell - q_{\ell-1})\Psi_k$. We know from the Central Limit Theorem that each $\hat{\beta}_k$, estimated via MLMC sampling, will be normally distributed with mean given by $\beta_k$ [62] and variance given by

$$V[\hat{\beta}_k] = \frac{1}{b_k^2} \sum_{\ell=0}^L \frac{V[ P_{\ell,k} ]}{N_\ell}.$$  

(4.17)

For a generic, normally-distributed random variable, $X \sim N(\mu, \sigma^2)$, we have

$$V[X^2] = E[X^4] - E[X^2]^2 = (\mu^4 + 6\mu^2\sigma^2 + 3\sigma^4) - (\mu^2 + \sigma^2)^2 = 4\mu^2\sigma^2 + 2\sigma^4.$$ 

Using this fact, the variance of our estimator can be expressed as

$$V[\hat{\beta}_k^2] = 4E[\hat{\beta}_k]^2V[\hat{\beta}_k] + 2V[\hat{\beta}_k]^2$$

$$= \frac{4\beta_k^2}{b_k^2} \left[ \sum_{\ell=0}^L \frac{V[ P_{\ell,k} ]}{N_\ell} + \frac{2}{b_k^2} \left( \sum_{\ell=0}^L \frac{V[ P_{\ell,k} ]}{N_\ell} \right)^2 \right].$$  

(4.18)

The above expression can be further expanded into raw moments of $P_{\ell,k}$ terms by using the MLMC definition of $\beta_k$. In order to evaluate (4.18), in practice, one will need to estimate the necessary statistics using a set of available samples.

Deriving an expression for the covariance terms in (4.16) will require a different approach. There are no analogous properties of normal random variables that can be used to easily obtain a new estimator for $C[\hat{\beta}_k^2, \hat{\beta}_z^2]$. Instead, using the bilinearity of the covariance and matching correlated samples of the QoI, one can derive the an estimator for the covariance directly. To illustrate the results of this technique, we will start by considering a single-level estimator and then move on to the multilevel case. We present the following proposition:

**Proposition 1.** Single-level covariance. Let $q$ be a single-level QoI. The PCE coefficients $\hat{\beta}_k$ and $\hat{\beta}_z$ are each computed via Monte Carlo with $N$ samples. Then the covariance $C[\hat{\beta}_k^2, \hat{\beta}_z^2]$ is
decomposed as

\[
C \left[ \hat{\beta}_k^2, \hat{\beta}_z^2 \right] = C \left[ \left( \frac{1}{b_k N} \sum_{i=1}^{N} q^{(i)} \Psi^{(i)}_k \right)^2, \left( \frac{1}{b_z N} \sum_{i=1}^{N} q^{(i)} \Psi^{(i)}_z \right)^2 \right] \\
= \frac{1}{b_k^2 b_z^2} \left[ \frac{\mathbb{E}[q^4 \Psi^2_k \Psi^2_z] - \mathbb{E}[q^2 \Psi^2_k] \mathbb{E}[q^2 \Psi^2_z]}{N^3} \right.
+ \frac{(2N - 2) \left( \mathbb{E}[q^3 \Psi^2_k \Psi_z] \mathbb{E}[q \Psi_z] - \mathbb{E}[q^2 \Psi^2_z] \mathbb{E}[q \Psi_z] \right)}{N^3} \\
+ \frac{(2N - 2) \left( \mathbb{E}[q^3 \Psi^2_k \Psi_z] \mathbb{E}[q \Psi_z] - \mathbb{E}[q \Psi_z]^2 \mathbb{E}[q^2 \Psi^2_z] \right)}{N^3} \\
+ \frac{(2N - 2) \left( \mathbb{E}[q^2 \Psi^2_k \Psi_z]^2 \right)}{N^3} \\
+ \frac{4(N - 1)(N - 2) \left( \mathbb{E}[q^2 \Psi_k \Psi_z] \mathbb{E}[q \Psi_z] \mathbb{E}[q \Psi_z] \right)}{N^3} \\
\left. - \frac{(4N^2 - 10N + 6) \left( \mathbb{E}[q \Psi_k]^2 \mathbb{E}[q \Psi_z]^2 \right)}{N^3} \right].
\] (4.19)

**Proof.** See Appendix A.1 for a detailed proof of Proposition 1. \[ \square \]

The multilevel estimator for the covariance with \( L \) levels is built upon the expression derived for the single-level estimator. First we define

\[
\hat{P}_{\ell,k} = \frac{1}{N_{\ell}} \sum_{i=1}^{N_{\ell}} P_{\ell,k}^{(i)} = \frac{1}{N_{\ell}} \sum_{i=1}^{N_{\ell}} \left( q_{\ell}^{(i)} - q_{\ell-1}^{(i)} \right) \Psi_k^{(i)}
\] (4.20)
as the multilevel estimator for \( \mathbb{E}[P_{\ell,k}] \) at level \( \ell \) with respect to the PCE coefficient \( k \).

**Proposition 2.** Multilevel covariance. Let \( q_0, q_1, \ldots, q_L \) be an \( L \)-level QoI. The PCE coefficients \( \hat{\beta}_k \) and \( \hat{\beta}_z \) are each computed via MLMC sampling according to (4.12). Then the covariance...
\[ \mathbb{C} \left[ \beta_k^2, \beta_s^2 \right] \text{ is decomposed as} \]
\[
\begin{align*}
\mathbb{C} \left[ \beta_k^2, \beta_s^2 \right] &= \frac{1}{b_k^2 b_s^2} \sum_{r=0}^{L} \left[ \frac{\mathbb{E} \left[ P_{r,k}^2 P_{r,s}^2 \right]}{N_r^2} - \mathbb{E} \left[ P_{r,k}^2 \right] \mathbb{E} \left[ P_{r,s}^2 \right] \right] + \frac{(2N_f - 2) \left( \mathbb{E} \left[ P_{r,k}^2 P_{r,s} \right] \mathbb{E} \left[ P_{r,s} \right] - \mathbb{E} \left[ P_{r,k}^2 \right] \mathbb{E} \left[ P_{r,s} \right]^2 \right)}{N_f^2}
+ \frac{2(2N_f - 2) \left( \mathbb{E} \left[ P_{r,k}^2 P_{r,s} \right] \mathbb{E} \left[ P_{r,k} \right] - \mathbb{E} \left[ P_{r,k}^2 \right] \mathbb{E} \left[ P_{r,k} \right]^2 \right)}{N_f^2}
+ \frac{2(N_f - 1)(2N_f - 2) \left( \mathbb{E} \left[ P_{r,k}^2 P_{r,s} \right] \mathbb{E} \left[ P_{r,k} \right] - \mathbb{E} \left[ P_{r,k}^2 \right] \mathbb{E} \left[ P_{r,k} \right]^2 \right)}{N_f^2}
+ \frac{4(N_f - 1)(2N_f - 2) \left( \mathbb{E} \left[ P_{r,k}^2 P_{r,s} \right] \mathbb{E} \left[ P_{r,k} \right] - \mathbb{E} \left[ P_{r,k}^2 \right] \mathbb{E} \left[ P_{r,k} \right]^2 \right)}{N_f^2}
+ \frac{2}{N_f} \sum_{r=0}^{L} \sum_{\ell \neq r}^{L} \left( \mathbb{E} \left[ P_{r,k}^2 P_{\ell,k} \right] \mathbb{E} \left[ P_{r,k} \right] - \mathbb{E} \left[ P_{r,k}^2 \right] \mathbb{E} \left[ P_{r,k} \right] \right)
+ \frac{1}{N_f} \sum_{r=0}^{L} \sum_{\ell \neq r}^{L} \left( \mathbb{E} \left[ P_{r,k}^2 P_{\ell,k} \right] \mathbb{E} \left[ P_{r,k} P_{\ell,k} \right] - \mathbb{E} \left[ P_{r,k}^2 P_{\ell,k} \right] \mathbb{E} \left[ P_{r,k} P_{\ell,k} \right] \right)
+ \frac{2}{N_f} \sum_{r=0}^{L} \sum_{\ell \neq r}^{L} \left( \mathbb{E} \left[ P_{r,k} P_{\ell,k} \right] \mathbb{E} \left[ P_{r,k} P_{\ell,k} \right] - \mathbb{E} \left[ P_{r,k} \right] \mathbb{E} \left[ P_{r,k} P_{\ell,k} \right] \right)
+ \frac{4}{N_f} \sum_{r=0}^{L} \sum_{\ell \neq r}^{L} \left( \mathbb{E} \left[ P_{r,k} P_{\ell,k} \right] \mathbb{E} \left[ P_{r,k} \right] - \mathbb{E} \left[ P_{r,k} \right] \mathbb{E} \left[ P_{r,k} \right] \right).
\end{align*}
\]

\[(4.21)\]

**Proof.** See Appendix A.2 for a detailed proof of Proposition 2. \(\square\)

### 4.4 Optimal sample allocation

The variance and covariance expressions derived in Section 4.3.1 can be used to evaluate (4.16) and build an estimator for \(\mathbb{V}[\hat{S}_u]\), using the raw moments of \(P_{\ell,k}\), which are available to the user. Additionally, (4.16) allows one to characterize the dependence of \(\mathbb{V}[\hat{S}_u]\) upon the multilevel sample allocation, \((N_0, \ldots, N_L)\). Just as with the traditional MLMC approach, this variance can then be used to derive an optimal sample allocation for the multilevel QoI.

The optimization problem for sample allocation may be formulated in a variety of ways. We will present one option for formulating the optimization problem and discuss the relevant distinguishing factors. In Chapter 5, we will present additional options for formulating the optimization problem, which each correspond to various practical scenarios. The following formulation minimizes the error in a particular Sobol’ index subject to a prescribed computational budget:

\[
\begin{align*}
\arg \min_{N_0, \ldots, N_L} \mathbb{V}[\hat{S}_u] \quad \text{subject to} \quad \sum_{\ell=0}^{L} N_{\ell}C_{\ell} \leq \bar{C}, \quad 0 \leq N_0, \ldots, N_L, \quad (4.22)
\end{align*}
\]

where \(\bar{C}\) is the upper limit on the total cost of the ML estimator. Equation (4.22) concisely...
expresses the goal of the hybrid MLMC-PCE method: to perform efficient GSA by using MLMC and PCE where samples are optimally allocated across levels to improve GSA accuracy. The problem (4.22) is equivalent to the standard optimization problem presented in the literature on MLMC sample allocations [38, 67, 69].

In (4.22), notice a single conditional variance term is targeted and so, while all Sobol’ indices can be obtained from the resulting sample profile, it will only be optimally targeting the accuracy of \( S_u \). We briefly describe three additional options for GSA targets: 1) all first order Sobol’ indices, 2) all total Sobol’ indices, 3) an arbitrary set of Sobol’ indices. Defining the objective function for each of the options must be done with some care so as to include only the relevant terms for each set of GSA targets. To illustrate this point, we provide the following bivariate example, where the variance of the PCE estimate of the variance (see (4.1)) is decomposed as follows:

\[
V[\hat{S}_1 + \hat{S}_2 + \hat{S}_{1,2}] = V[\hat{S}_1] + V[\hat{S}_2] + V[\hat{S}_{1,2}] + 2C[\hat{S}_1, \hat{S}_2] + 2C[\hat{S}_1, \hat{S}_{1,2}] + 2C[\hat{S}_2, \hat{S}_{1,2}]
\]

Each of the above variances and covariances will be further decomposed in terms of the relevant PCE coefficients according to (2.18). Given a set of GSA targets, several of the terms in (4.23) will not directly influence the quality of the estimated Sobol’ indices and should be removed. For example, if one wishes to target all \( d \) first order indices, one might be tempted to minimize \( V[S_1 + S_2] \), the decomposition of which will include the covariance, \( C[S_1, S_2] \), which is not relevant to the estimation of either first order index. Instead, only \( V[\hat{S}_1] \) and \( V[\hat{S}_2] \) should be retained. Thus, for a \( d \)-dimensional QoI, one should solve the optimization problem,

\[
\arg\min_{N_0, \ldots, N_L} \sum_{i=1}^d V[\hat{S}_i] \quad \text{subject to} \quad \sum_{\ell=0}^L N_\ell C_\ell \leq \bar{C}, \quad 0 \leq N_0, \ldots, N_L.
\] (4.23)

Similarly, if one wishes to target all \( d \) total Sobol’ indices (see (4.3)), instead of minimizing \( V[T_1 + T_2] \), which would include several unnecessary terms, one would minimize \( V[\hat{T}_1] + V[\hat{T}_2] \). For a \( d \)-dimensional QoI, one should solve the optimization problem,

\[
\arg\min_{N_0, \ldots, N_L} \sum_{i=1}^d V[\hat{T}_i] \quad \text{subject to} \quad \sum_{\ell=0}^L N_\ell C_\ell \leq \bar{C}, \quad 0 \leq N_0, \ldots, N_L.
\] (4.24)

If one were to target an arbitrary set of Sobol’ indices, \( \{S_{u_1}, \ldots, S_{u_n}\} \) for \( u_i \in \{1, \ldots, d\}, i = 1, \ldots, n \), then the optimization problem would be formulated as

\[
\arg\min_{N_0, \ldots, N_L} \sum_{i=1}^n V[\hat{S}_{u_i}] \quad \text{subject to} \quad \sum_{\ell=0}^L N_\ell C_\ell \leq \bar{C}, \quad 0 \leq N_0, \ldots, N_L.
\] (4.25)

As we have discussed, the objective function for sample allocation may very well include a large number of GSA terms (i.e. variances and covariances of ANOVA terms), especially
for high-dimensional problems. Each of these GSA terms must then be decomposed according to (4.16) into their relevant PCE components. For a PCE with many terms, each with their own estimation error, this may result in an accumulation of uncertainty that dominates the objective of the optimization problem. In order to reduce the estimation error entering the optimization problem, one should consider strategies for truncating the PCE basis. The goal of truncating the PCE basis would be to remove terms that contribute more noise than information to both the optimization problem and the computation of Sobol’ indices. Given that the total order basis truncation strategy was discussed in Section 2.2.1, we reserve a discussion of more-advanced basis truncation strategies for the forthcoming journal article [4].

This section has served as a discussion of the theoretical aspects of defining the optimization problem for sample allocation. Practical elements of solving the optimization problem, as well as alternative formulations, will be discussed in Chapter 5.

4.5 Numerical verification

We next present a set of verification experiments illustrating the theoretical developments in Section 4.3. The following results are obtained using the Ishigami function, a standard test problem in the GSA literature [27, 29, 8], following the three level structure presented in [70].

4.5.1 Single-level results

We begin by considering the single-level Ishigami function,

\[ q(\theta) = \sin(\theta_1) + a \sin^2(\theta_2) + b \theta_3^4 \sin(\theta_1), \tag{4.26} \]

where \(\theta_1, \theta_2,\) and \(\theta_3\) are uncertain parameters following a uniform distribution on \([-\pi, \pi]\), with \(a\) and \(b\) being constants.

Using multivariate Legendre polynomials, we will compute a single-level PCE of (4.26) where we let \(a = 5\) and \(b = 0.1\). We first consider the variability of the estimators for \(\hat{\beta}_k\) by computing 1000 realizations of the PCE spectrum, up to a total order of 6. We then plot the mean and two standard deviations for each coefficient below.

Figure 4.1 shows the increasing variance of the PCE coefficients with the polynomial order. As the order of the Legendre polynomial increases, the variance of the PCE terms will also increase. Notice also that the Ishigami function has a sparse PCE, where the majority of the coefficients are zero. This property of the Ishigami function makes it challenging for PCE surrogate construction and so alternative basis constructions have been explored in this context [72]. In the following discussion, we will use the total order construction, with the knowledge that the PCE basis can be tailored as a post-processing step. In order to capture the appropriate number of PCE terms, we will use a total polynomial order of 7 in the following experiments, resulting in 120 coefficients.
Figure 4.1 Estimated PCE coefficients for Ishigami, up to a total polynomial order of 6. Dots denote $E[\hat{\beta}_k]$ and error bars denote 2 standard deviations, computed with 1000 realizations.

4.5.2 Multilevel results

Continuing to the multilevel form of the Ishigami function from [70], we define a three-level model hierarchy, with associated costs, as follows:

\[ q_0(\theta) = \sin(\theta_1) + (0.6) \ a \sin^2(\theta_2) + (9)b\theta_3^2 \sin(\theta_1), \quad c_0 = 0.001 \]  \hspace{1cm} (4.27)
\[ q_1(\theta) = \sin(\theta_1) + (0.95) \ a \sin^2(\theta_2) + b\theta_3^2 \sin(\theta_1), \quad c_1 = 0.05 \]  \hspace{1cm} (4.28)
\[ q_2(\theta) = \sin(\theta_1) + a \sin^2(\theta_2) + b\theta_3^2 \sin(\theta_1), \quad c_2 = 1.0 \]  \hspace{1cm} (4.29)

The costs associated with the multilevel Ishigami function, given in [70], are assigned artificially and are simply used for demonstration. The costs were intended to approximate the level-by-level cost structure that is common in many practical applications of MLMC. Note that $q_0$ is the low fidelity and $q_2$ is the high fidelity, in this case the original function. We also note the cost difference is a factor 50 from $q_0$ to $q_1$ and a factor 20 from $q_1$ to $q_2$. The goal will be to perform MLMC, entailing a majority of the samples be allocated to the lower-fidelity models.

A benefit of using the Ishigami function is the ability to compute all relevant quantities analytically. In this case, the true variance of the high fidelity, is $\mathbb{V}[q_2] = \frac{1}{2} + \frac{a^2}{8} + \frac{\pi^4b}{5} + \frac{\pi^8b^2}{18} \approx 10.845$. The mean of each function is: $E[q_0] = 1.5$, $E[q_1] = 2.375$, and $E[q_2] = 2.5$. Finally, we present the true first order and total Sobol’ indices for the high fidelity function:

We can then compute the MLMC estimate of the PCE coefficients, using the method described in Section 4.2.2, from the estimator

\[ \hat{\beta}_k = \frac{1}{b_k} \sum_{\ell=0}^{L} \frac{1}{N_\ell} \sum_{i=1}^{N_\ell} P^{(i)}_{\ell,k} = \frac{1}{b_k} \sum_{\ell=0}^{L} \frac{1}{N_\ell} \sum_{i=1}^{N_\ell} (q_\ell^{(i)} - q_{\ell-1}^{(i)}) \Psi_k^{(i)}. \]  \hspace{1cm} (4.30)
Table 4.1 True first order and total Sobol’ indices for the high-fidelity QoI, $q_2$, for $a = 5$ and $b = 0.1$.

<table>
<thead>
<tr>
<th></th>
<th>$i = 1$</th>
<th>$i = 2$</th>
<th>$i = 3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>First order index, $S_i(q_2)$</td>
<td>0.401</td>
<td>0.288</td>
<td>0</td>
</tr>
<tr>
<td>Total index, $T_i(q_2)$</td>
<td>0.712</td>
<td>0.288</td>
<td>0.311</td>
</tr>
</tbody>
</table>

After the set of PCE coefficients are computed, we want to show that our estimators for $\hat{\beta}_k^2$ and $C[\hat{\beta}_k^2, \hat{\beta}_z^2]$ from Section 4.3.1 are consistent with empirical PCE data. We also compare the variance estimators with the covariance diagonal to guarantee that they are consistent.

Figure 4.2 Estimated $\text{var}(\hat{\beta}_k^2)$ from (4.18) and diagonals of $C[\hat{\beta}_k^2, \hat{\beta}_z^2]$ from (4.21). Left: used $10^5$ samples per level, ensuring convergence. Right: green dots indicate data from 1000 PCE realizations.

As we have shown in Section 4.3.1, we are able to propagate the uncertainty in the estimated PCE coefficients through to the conditional variances in the ANOVA decomposition, resulting in the estimate for $\text{var}(\hat{S}_u)$, for any $u \subseteq \{1, \ldots, d\}$. To further illustrate the results of the derivation in Section 4.3.1, we define a modest sample allocation, $(N_0, N_1, N_2) = (1000, 100, 10)$. We then compute 1000 independent realizations of the set of PCE coefficients, followed by computing the conditional variances corresponding to the first order and total indices, $(S_1, S_2, S_3, T_1, T_2, T_3)$. Using this set of realizations, we are able to form an empirical estimate of each $\text{var}(\hat{S}_i)$ and $\text{var}(\hat{T}_i)$, $i = 1, 2, 3$, from the PCE data. We then compare these empirical variance estimates with our derived estimators for each $\text{var}(\hat{S}_i)$ and $\text{var}(\hat{T}_i)$, obtained in Section 4.3.1. In order to mitigate estimation noise in the statistics used in the derived estimators, we use $10^5$ samples per level. In Figure 4.3, we compare the empirical data for $\text{var}(\hat{S}_u)$ with our fully-converged novel estimators.

To demonstrate the power of these novel estimators for $\text{var}(\hat{S}_i)$ and $\text{var}(\hat{T}_i)$, Figure 4.4 shows 1000 realizations of the first order and total indices, computed with the allocation: $(N_0, N_1, N_2) = (1000, 100, 10)$. We compare this with the true values of the conditional variances (see Table 4.1) and the confidence intervals obtained from our new estimators. We report errorbars showing 2 standard deviations about the mean, which encompasses approximately 95% of the replicates.
Figure 4.3 Blue bars indicate the variance of the conditional terms, obtained from the derived estimators, with $10^5$ samples at each level. Red bars indicate the variance over 1000 realizations of the conditional terms, with sample allocation $(1000, 100, 10)$.

Figure 4.4 Blue bars indicate analytic $S_i$ and $T_i$ values for $i = 1, 2, 3$. Red dots indicate the results of 1000 realizations of the conditional variances. Black errorbars indicate one converged estimate of each $V[S_i]$ and $V[T_i]$, reporting 2 standard deviations.

Figure 4.4 illustrates the practical applicability of the hybrid MLMC-PCE method. Not only can the novel estimators derived in Section 4.3.1 quantify the accuracy of a particular set of Sobol’ indices, but they express the relationship between each $V[S_i]$ term and the underlying sample profile, $(N_i)_L=0$. We have so far illustrated numerically that these estimators do converge to the empirical variance of any desired conditional term in the ANOVA decomposition.

In the next chapter, we will incorporate the optimal sample allocation into the numerical results, showing that this method can be competitive with other current multifidelity methods for GSA. Further elaboration will also be given on a number of practical aspects of the hybrid method.
4.6 Conclusion

In this chapter, we have explored a hybrid MLMC-PCE approach for GSA that leverages the ANOVA decomposition traditionally used with the PCE, but for which the polynomial coefficients are evaluated by means of MLMC. This hybrid approach improves the efficiency of Sobol’ index computation by fusing information from multiple model fidelities. This chapter focused primarily on developing the theoretical components of the MLMC-PCE framework for GSA and presenting preliminary numerical results that demonstrate the potential of this approach. In the next chapter, we address a series of practical issues related to implementation of the algorithm, practical aspects the optimization problem, and we present a series of numerical results which test the efficiency and accuracy of the hybrid method.

A natural application of this framework is presented in Chapter 3, that is, to use the hybrid approach to accelerate GSA of chemical reaction networks. For instance, one may define multiple fidelities for a chemical reaction model and then use the hybrid method to optimally apportion the limited computational budget between model evaluations. In Chapter 5, we will present a series of numerical results on this topic, as well as a comparison with other current multifidelity GSA methods.
5.1 Introduction

In Chapter 4, the hybrid multilevel Monte Carlo Polynomial Chaos Expansion (MLMC-PCE) method was introduced, which is intended for goal-oriented computation of Sobol’ indices. This method aims to leverage the information from multiple model fidelities by distributing one’s computational resources across the model hierarchy in a manner that is optimal for GSA. This method also aims to be robust with respect to high-dimensionality by taking advantage of the convergence properties of MC methods, which are not affected by dimension. Traditionally, MLMC methods have been used to allocate samples for estimation of the mean or variance of a QoI [38, 57, 67], although efforts have been made to extend this approach to the computation of Sobol’ indices [69, 70]. To that end, we combine MLMC with the PCE, which allows one to use a shared set of model evaluations for estimating each PCE term. From the PCE, Sobol’ indices can be computed essentially for free. In Chapter 4, novel estimators were derived, which allow one to propagate the uncertainty incurred in MLMC sampling of the PCE coefficients through the ANOVA decomposition. As a result, the accuracy of an estimated Sobol’ index can be described as a function of the multilevel sample allocation and this sample allocation can be optimized. One is thus able to characterize the uncertainty in each estimated ANOVA term and obtain confidence intervals for each GSA target without requiring additional model evaluations.

In this chapter, with the foundational aspects of this hybrid MLMC-PCE method in place, we address some challenges and practical considerations associated with the hybrid method,
including construction of unbiased estimators, alternative formulations for optimal sample allocation, and an algorithm implementation. We conclude with a series of numerical results. First, using the Ishigami function as a test problem, we compare the accuracy of the hybrid method with that of other popular approaches to computing Sobol’ indices. This is followed by the application of the hybrid method to a problem from chemical kinetics. The contents of this chapter, a direct continuation of Chapter 4, is the result of a collaboration of the author with Gianluca Geraci, Mike Eldred, and Teresa Portone of Sandia National Laboratories. At the time of writing this dissertation, a journal article on the material covered in this chapter is in preparation [4].

5.2 Challenges of the hybrid method

We begin by summarizing some challenges associated with the hybrid method, all of which will be addressed in this chapter:

1. While the MLMC estimators for the PCE coefficients are unbiased, the resulting estimators for the squared coefficients incur a bias, which, in turn, results in a biased estimate of the Sobol’ indices (see (2.18)). We derive a bias-correction formula in Section 5.3.

2. The optimization problem for sample allocation cannot be solved in closed form and thus requires numerical strategies, which must be investigated. There are also alternative formulations for the sample allocation problem, which are discussed in Section 5.4.

3. An implementation of the hybrid method involves multiple steps. If one takes an iterative approach to QoI sampling and optimizing the sample profile, this can be advantageous. We provide an algorithm implementation of the hybrid method in Section 5.5 and discuss best practices for structuring the algorithm.

In addition to addressing these practical challenges, we present a comparison of the hybrid method with other competing methods in Section 5.6.1. Finally, we apply the hybrid method to a problem from chemical kinetics in Section 5.6.2.

5.3 Derivation of unbiased estimators

Recall, in Section 2.2.3, we denote $q(\theta)$ as the QoI, $\Psi_k(\theta)$ denotes the $k$th polynomial basis function in the PCE, and, in Section 4.2.2, we let $b_k$ denote the associated basis norm. The $k$th PCE coefficient, $\beta_k$, is defined as in (4.4). The formula for computing Sobol’ indices via the PCE (2.18) requires the squared PCE coefficients. While the MLMC estimator for $\beta_k$ is unbiased, meaning that

$$E \left[ \hat{\beta}_k \right] = E \left[ \frac{1}{b_k} \sum_{\ell=0}^{L} \sum_{i=1}^{N_\ell} P_{\ell,k}^{(i)} \right] = \frac{1}{b_k} \sum_{\ell=0}^{L} N_\ell E \left[ P_{\ell,k} \right] = \beta_k,$$
it is not true, in general, that $\mathbb{E}[(\hat{\beta}_k)^2] = \beta_k^2$. Thus, we will present an unbiased version of the estimator for $\beta_k^2$, in the case of a multilevel QoI.

**Proposition 3.** Unbiased estimator of $\beta_k^2$. The estimator,

$$
\hat{\beta}_k^2 = (\hat{\beta}_k)^2 - V[\hat{\beta}_k] = (\hat{\beta}_k)^2 - \frac{1}{b_k^2} \sum_{\ell=0}^{L} \frac{V[P_{\ell,k}]}{N_\ell},
$$

(5.1)

is an unbiased estimator for $\beta_k^2$, that is, $\mathbb{E}[\hat{\beta}_k^2] = \beta_k^2$.

**Proof.** Let the random variable $\frac{q_L \Psi_k b_k}{b_k}$, whose expected value is $\beta_k$, be denoted as $X$. Let $\hat{X}$ denote the MLMC estimator for $\mathbb{E}[X]$. The goal is to produce an unbiased estimator for $\beta_k^2$.

First, for convenience, we define the mean-zero random variable, $Z = X - \beta_k$. Just as $\beta_k$ can be decomposed over levels as in (4.12), we will decompose $Z$ over its levels as

$$
Z = Z_0 + Z_1 + \cdots + Z_L,
$$

where, for a fixed $k$, we define each $Z_\ell$ in the notation of Chapter 4,

$$
Z_\ell = \frac{(q_{\ell} - q_{\ell-1}) \Psi_k}{b_k} - \frac{\mathbb{E}[(q_{\ell} - q_{\ell-1}) \Psi_k]}{b_k} = \frac{P_{\ell,k}}{b_k} - \frac{\mathbb{E}[P_{\ell,k}]}{b_k}.
$$

(5.2)

Thus each $Z_\ell$ has mean zero. Then, to determine the bias in an estimate of $\beta_k^2$ we compute

$$
(\hat{X})^2 = \left( \hat{Z} + \beta_k \right)^2
$$

$$
= \left( \sum_{\ell=0}^{L} \frac{1}{N_\ell} \sum_{i=1}^{N_\ell} Z^{(i)}_{\ell} + \beta_k \right)^2
$$

$$
= \left( \sum_{\ell=0}^{L} \frac{1}{N_\ell} \sum_{i=1}^{N_\ell} Z^{(i)}_{\ell} \right)^2 + 2\beta_k \left( \sum_{\ell=0}^{L} \frac{1}{N_\ell} \sum_{i=1}^{N_\ell} Z^{(i)}_{\ell} \right) + \beta_k^2
$$

$$
= \left( \sum_{\ell=0}^{L} \frac{1}{N_\ell^2} \left( \sum_{i=1}^{N_\ell} Z^{(i)}_{\ell} \right)^2 \right) + \sum_{\ell \neq z} \frac{1}{N_\ell N_z} \sum_{i=1}^{N_\ell} Z^{(i)}_{\ell} \sum_{j=1}^{N_z} Z^{(j)}_{z} + 2\beta_k \left( \sum_{\ell=0}^{L} \frac{1}{N_\ell} \sum_{i=1}^{N_\ell} Z^{(i)}_{\ell} \right) + \beta_k^2
$$

Then taking the expectation of $(\hat{X})^2$, we use the fact that $\mathbb{E}[Z_\ell] = 0, \ell = 0, \ldots, L$, and
\[E[Z_{\ell}Z_z] = 0, z \neq \ell, \text{ then we have} \]

\[
E\left[\hat{X}^2\right] = E\left[\sum_{\ell=0}^{L} \frac{1}{N_{\ell}^2} \left(\sum_{i=1}^{N_{\ell}} Z_{\ell}^{(i)}\right)^2 + \sum_{\ell \neq z} \frac{1}{N_{\ell}N_z} \sum_{i=1}^{N_{\ell}} \sum_{j=1}^{N_z} Z_{\ell}^{(i)}Z_z^{(j)} + 2\beta_k \left(\sum_{\ell=0}^{L} \frac{1}{N_{\ell}^2} \sum_{i=1}^{N_{\ell}} Z_{\ell}^{(i)}\right) + \beta_k^2\right]
\]

\[
= E\left[\sum_{\ell=0}^{L} \frac{1}{N_{\ell}^2} \left(\sum_{i=1}^{N_{\ell}} Z_{\ell}^{(i)}\right)^2 + \beta_k^2\right]
\]

\[
= \sum_{\ell=0}^{L} \frac{1}{N_{\ell}^2} E\left[\sum_{i=1}^{N_{\ell}} \left(Z_{\ell}^{(i)}\right)^2 + \sum_{i \neq j} Z_{\ell}^{(i)}Z_{\ell}^{(j)}\right] + \beta_k^2.
\]

Then using the fact that the distinct samples of \(Z_{\ell}\) are uncorrelated, we have

\[
E\left[\hat{X}^2\right] = \sum_{\ell=0}^{L} \frac{1}{N_{\ell}^2} E\left[\sum_{i=1}^{N_{\ell}} \left(Z_{\ell}^{(i)}\right)^2\right] + \beta_k^2.
\]

We further know that because each \(Z_{\ell}\) has mean zero, we have \(\mathbb{V}[Z_{\ell}] = E[Z_{\ell}^2]\) and so

\[
E\left[\hat{X}^2\right] = \sum_{\ell=0}^{L} \frac{1}{N_{\ell}^2} \sum_{i=1}^{N_{\ell}} \mathbb{V}[Z_{\ell}^{(i)}] + \beta_k^2 = \sum_{\ell=0}^{L} \frac{N_{\ell}}{N_{\ell}^2} \mathbb{V}[Z_{\ell}] + \beta_k^2.
\]

From this, we know that obtaining an unbiased estimate of \(\beta_k^2\) requires one to subtract the estimator variance from the biased estimator. Finally, by the definition of \(Z_{\ell}\) (5.2), we know \(\mathbb{V}[Z_{\ell}] = \mathbb{V}[P_{t,k}]/b_k^2\). Thus we have the following formula for the unbiased estimator of \(\beta_k^2\),

\[
\hat{\beta}_k^2 = (\hat{\beta}_k)^2 - \sum_{\ell=0}^{L} \mathbb{V}[Z_{\ell}] = (\hat{\beta}_k)^2 - \frac{1}{b_k^2} \sum_{\ell=0}^{L} \mathbb{V}[P_{t,k}] = (\hat{\beta}_k)^2 - \mathbb{V}[\hat{\beta}_k].
\] (5.3)

With this bias correction, we can go about obtaining unbiased estimators for each of the squared PCE coefficients and, by extension, for any conditional variance term in the ANOVA decomposition. Although, we are able to correct the bias in the numerator and denominator of the Sobol’ index formula (see (2.18)), there is no guarantee that the ratio of two unbiased estimators will be unbiased. Since it is sufficient to use the conditional variance in the numerator of (2.18) for our sample allocation scheme, we will not pursue this issue further.

### 5.4 Optimal sample allocation

A defining feature of the hybrid MLMC-PCE method is the ability to obtain a ML sample allocation, which is optimized for GSA. Section 4.4 dealt with one possible strategy for the optimization problem (4.22), where the variance of an estimated GSA target is minimized,
constrained by an upper bound on the total computational cost, $\bar{C}$. We then examined the case where multiple GSA quantities are being targeted. While (4.22) is a viable option for sample allocation, one may elect to choose a different optimization strategy based on their particular goals. This is akin to the discussion in [69], in which multiple optimization strategies were considered. We present two alternative formulations for the optimization problem:

$$\arg \min_{N_0, \ldots, N_L} C_{tot} = \sum_{\ell=0}^{L} N_{\ell} C_{\ell} \quad \text{subject to} \quad \mathbb{V}[\mathcal{S}_u] \leq \varepsilon, \quad 0 \leq N_0, \ldots, N_L \quad (5.4)$$

$$\arg \min_{N_0, \ldots, N_L} C_{tot} = \sum_{\ell=0}^{L} N_{\ell} C_{\ell} \quad \text{subject to} \quad \mathbb{V}[\mathcal{S}_u] \leq \eta(\mathbb{V}[\mathcal{S}_u])_0, \quad 0 \leq N_0, \ldots, N_L \quad (5.5)$$

In (5.4), one minimizes the total cost of MLMC estimation, while requiring that the variance of the Sobol’ estimator be below some absolute threshold, $\varepsilon$, chosen by the user. In (5.5), one minimizes $C_{tot}$, while requiring $\mathbb{V}[\mathcal{S}_u]$ to be some factor, $\eta$, below the pilot variance, denoted $(\mathbb{V}[\mathcal{S}_u])_0$, which is the variance computed with a baseline (or pilot) sample profile. This implies an iterative implementation of the hybrid method, where an initial round of QoI sampling results in an estimate of the pilot variance and a subsequent round of sampling is required to meet the variance constraint in (5.5).

Recall that in Section 4.4 we considered a common scenario, where multiple GSA targets are involved, for example, the first order Sobol’ indices or the total indices [7, 8, 5]. In each case, a scalar objective function was discussed, with the intention of removing unnecessary variance and covariance terms from the objective function. When considering multiple GSA targets in either (5.4) or (5.5), the discussion is more straightforward because the variance estimator is now involved in a constraint. In either (5.4) or (5.5), multiple variance constraints can be imposed, whether they are absolute or relative to the pilot variance.

Now that we have defined the optimization problem for sample allocation and discussed alternative formulations, we address how the optimization problem is solved in practice. As a result of the complexity and number of terms present in each $\mathbb{V}[\mathcal{S}_u]$ estimator (see discussion in Section 4.3.1), solving the sample allocation problem in closed form is far more challenging than for the canonical MLMC problem [38]. In practice, the optimal sample allocation will be determined numerically. We use SciPy’s sequential least squares programming algorithm (SLSQP), which is designed to perform constrained minimization of a scalar objective function with any combination of equality, inequality, and bound constraints. For further details on this optimization method, we refer to [73, 68], in which SLSQP is used in the context of multifidelity MC.

### 5.5 Implementation of MLMC-PCE

We proceed to a discussion of the implementation of the hybrid MLMC-PCE method. While this implementation will not cover all possible optimization strategies or GSA targets, it will
demonstrate a particular instance of the hybrid method and be useful for introducing other practical considerations.

For simplicity, we let \( S_u \) denote a single GSA target. This algorithm can be extended to multiple GSA targets, as covered in Section 4.4. We will use optimization strategy (5.5), which aims to reduce \( V[\hat{S}_u] \) a factor, \( \eta \), below the pilot variance.

Algorithm 3 Estimate Sobol’ indices via hybrid MLMC-PCE

\textbf{Input:} Multilevel model, \( \{ q_{\ell}(\Theta) \}_{\ell=0}^L \), number of pilot samples, \( \{ \hat{N}_{\ell} \}_{\ell=0}^L \), GSA target index, \( u \), total polynomial order, \( r \), variance reduction factor, \( \eta \), max iterations

\textbf{Output:} Sobol index, \( S_u \), error in index, \( V[S_u] \), optimal sample allocation, \( (N^*_\ell)_{\ell=0}^L \), estimated PCE coefficients, \( \{ \hat{\beta}_k \}_{k=0}^{N_{PC}} \)

1: Initialize PCE variables: truncation level, multi-indices, polynomial norms
2: Draw \( (N_{\ell})_{\ell=0}^L \) pilot samples: \( (\hat{\Theta})_{\ell=0}^L \) \{Distribution included with the model\}
3: Evaluate polynomials, \( (V_k)_{k=0}^{N_{PC}} \) and multilevel QoI, \( q_{\ell} - q_{\ell-1} \) for \( \ell = 0, \ldots, L \) at \( (\hat{\Theta})_{\ell=0}^L \)
4: Evaluate raw moments: \( E[P_{\ell,k}], E[P_{\ell,k}^2], E[P_{\ell,k}^2 P_{\ell,z}], E[P_{\ell,k}^2 P_{\ell,z}^2], \ell = 0, \ldots, L \) and \( k, z = 0, \ldots, N_{PC} \)
5: Evaluate \( V[\hat{\beta}_k^2] \) and \( C[\hat{\beta}_k^2, \hat{\beta}_z^2] \) expressions for \( k, z = 0, \ldots, N_{PC} \) \{see (4.18) and (4.21)\}
6: Evaluate necessary terms for pilot variance: \( (V[S_u])_0 \) \{see (4.16)\}
7: Set iteration = 0
8: while \( V[S_u] > \eta \cdot (V[S_u])_0 \) and iteration < max iterations do
9: Obtain optimal sample allocation: \( (N_0, \ldots, N_L) \) \{see Section 5.4\}
10: Determine the number of additional samples to be taken per level: \( (N^*_0, \ldots, N^*_L) \)
11: Draw \( (N^*_\ell)_{\ell=0}^L \) additional samples: \( (\hat{\Theta})_{\ell=0}^L \)
12: Evaluate polynomials, \( (V_k)_{k=0}^{N_{PC}} \) and multilevel QoI, \( q_{\ell} - q_{\ell-1} \) for \( \ell = 0, \ldots, L \) at \( (\hat{\Theta})_{\ell=0}^L \)
13: Evaluate raw moments: \( E[P_{\ell,k}], E[P_{\ell,k}^2], E[P_{\ell,k}^2 P_{\ell,z}], E[P_{\ell,k}^2 P_{\ell,z}^2], \ell = 0, \ldots, L \) and \( k, z = 0, \ldots, N_{PC} \)
14: Evaluate \( V[\hat{\beta}_k^2] \) and \( C[\hat{\beta}_k^2, \hat{\beta}_z^2] \) expressions for \( k, z = 0, \ldots, N_{PC} \) \{see (4.18) and (4.21)\}
15: Evaluate updated \( V[S_u] \) at optimal sample profile, \( (N_{\ell})_{\ell=0}^L \) \{see (4.16)\}
16: iteration = iteration + 1
17: end while
18: Compute MLMC estimate of PCE coefficients: \( \hat{\beta}_0, \ldots, \hat{\beta}_{N_{PC}} \) \{see (4.12)\}
19: Compute relevant Sobol’ indices: \( S_u \) \{see (2.18)\}
20: Obtain final sample allocation: \( (N_0^*, \ldots, N_L^*) \) \{see Section 5.4\}

Algorithm 3 presents an iterative sample allocation strategy, where, on the basis of the pilot samples, a new sample profile will be obtained and additional samples computed. The algorithm will iterate between QoI sampling and optimizing the sample profile until convergence is reached or the maximum number of iterations is exceeded. Alternative iterative approaches will be discussed as well. In practice, if the pilot sample profile is small, then the initial estimate the statistics and target variance may be far from their converged values. This may result in a sample profile that is much larger than the optimal profile with the statistics fully converged, and thus over-sampling of the QoI. To avoid this, we suggest either an optimization strategy
that enforces an upper bound on the computational budget (as in Section 4.4) or using a relaxed sample allocation strategy. The relaxed strategy entails taking a subset of the recommended QoI samples at each iteration, after which one reruns the optimization and continues sampling the QoI, converging to the optimal sample profile from below. In situations where a costly, high-fidelity QoI is considered, the additional calls to the optimization routine will add a negligible computational cost to the algorithm.

As mentioned briefly in Chapter 4, truncating the PCE basis is a useful strategy for reducing the number of sources of uncertainty and can improve the performance of the optimization routine. While a total order basis truncation scheme is discussed in Chapter 2, other basis truncation schemes have been examined for the hybrid MLMC-PCE method. We reserve a discussion of these basis truncation strategies for the forthcoming journal article [4].

5.6 Numerical results

We present the following numerical results for two applications. The first being a comparison of the hybrid MLMC-PCE method with other current methods for variance-based GSA in the literature. The test problem for this experiment will be the multilevel Ishigami function, as defined in Section 4.5.2. The second set of results will apply the hybrid method to the 16-dimensional genetic oscillator system discussed in Chapter 3.

5.6.1 Comparison with competing methods for the Ishigami example

We return to the three-level Ishigami function [70], covered in Section 4.5.2. This test problem will allow us to compare the hybrid method with three other common GSA methods. We include the standard Saltelli sampling method, which is covered briefly in Chapter 2 and used to perform GSA on chemical reaction networks in Chapter 3. This is a single-fidelity method whose cost scales linearly with the dimension of the problem. We next include the multifidelity GSA method developed by Qian and Willcox in [70]. This method is a multifidelity generalization of the Saltelli method, which takes a multifidelity model hierarchy and combines samples across levels to augment the Saltelli framework. Again, this method has a cost that scales linearly with the problem dimension. Finally, we present the single-fidelity PCE method, which computes PCE coefficients via MC sampling. This method is rarely used in practical GSA scenarios and will act solely as a baseline for comparison. We choose to include the results of the single-fidelity PCE, in order to show the significant computational advantages of using a multifidelity versus a single-fidelity PCE approach to GSA, as we have described with our hybrid method.

The three-level Ishigami function, with associated costs, is defined in [70] as

\[
q_0(\theta) = \sin(\theta_1) + (0.6) \ a \sin^2(\theta_2) + (9) b \theta_2^2 \sin(\theta_1), \quad c_0 = 0.001 \\
q_1(\theta) = \sin(\theta_1) + (0.95) \ a \sin^2(\theta_2) + b \theta_2^4 \sin(\theta_1), \quad c_1 = 0.05 \\
q_2(\theta) = \sin(\theta_1) + (a \sin^2(\theta_2) + b \theta_2^4 \sin(\theta_1), \quad c_2 = 1.0
\]
The costs assigned with the three-level Ishigami function are artificial, chosen in [70] to emulate the cost separation between levels in a practical scenario. The Ishigami function has three variables and so we will provide results for the 3 first order indices and the 3 total indices, for each GSA method. We will also compare these methods with the analytical values for the Sobol’ indices (see Table 4.1). The hybrid method will have a sample profile that is either optimized to target all first order indices (4.23) or all total indices (4.24). The multifidelity GSA method [70] will target the variance of the estimated QoI variance. Figure 5.1 provides distributions for the computed Sobol’ indices, using an equivalent computational cost of 2000 for each method. The reference values for the true Sobol’ indices are plotted with vertical lines.

We also present in Table 5.1 results for the mean, variance, and mean squared error (MSE) of each GSA method.

<table>
<thead>
<tr>
<th>Method</th>
<th>( S_1 ) Mean</th>
<th>( S_1 ) Variance</th>
<th>( S_1 ) MSE</th>
<th>( S_2 ) Mean</th>
<th>( S_2 ) Variance</th>
<th>( S_2 ) MSE</th>
<th>( S_3 ) Mean</th>
<th>( S_3 ) Variance</th>
<th>( S_3 ) MSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>SF-Saltelli</td>
<td>4.017</td>
<td>2.00 \times 10^{-3}</td>
<td>2.0 \times 10^{-3}</td>
<td>2.894</td>
<td>2.5 \times 10^{-3}</td>
<td>2.5 \times 10^{-3}</td>
<td>-0.0010</td>
<td>5.8 \times 10^{-4}</td>
<td>5.8 \times 10^{-4}</td>
</tr>
<tr>
<td>SF-PCE</td>
<td>3.683</td>
<td>4.66 \times 10^{-4}</td>
<td>1.5 \times 10^{-4}</td>
<td>2.611</td>
<td>6.4 \times 10^{-4}</td>
<td>1.36 \times 10^{-3}</td>
<td>0.0081</td>
<td>2.5 \times 10^{-5}</td>
<td>9.1 \times 10^{-4}</td>
</tr>
<tr>
<td>MF-GSA</td>
<td>3.997</td>
<td>7.76 \times 10^{-5}</td>
<td>7.9 \times 10^{-5}</td>
<td>2.883</td>
<td>7.7 \times 10^{-5}</td>
<td>7.7 \times 10^{-5}</td>
<td>0.00014</td>
<td>1.5 \times 10^{-4}</td>
<td>1.5 \times 10^{-4}</td>
</tr>
<tr>
<td>MLMC-PCE</td>
<td>4.025</td>
<td>1.50 \times 10^{-3}</td>
<td>1.7 \times 10^{-3}</td>
<td>2.839</td>
<td>1.2 \times 10^{-3}</td>
<td>3.18 \times 10^{-3}</td>
<td>0.00013</td>
<td>5.6 \times 10^{-4}</td>
<td>2.3 \times 10^{-3}</td>
</tr>
<tr>
<td>True</td>
<td>0.401</td>
<td>-</td>
<td>0.288</td>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

As Table 5.1 indicates, the hybrid MLMC-PCE method incurs the lowest MSE for every Sobol’ index reported. The MF-GSA method, although it too allows for the use of multiple fidelities to estimate Sobol’ indices, has a dimension-dependent cost. In this case, the total cost of the MF-GSA method is divided into separate budgets for estimating each conditional variance. This is also true for the Saltelli method in which the total cost is divided by \( d + 2 \), for each conditional term. An advantage of the MLMC-PCE method is the ability to use the full set of QoI samples for each PCE term and therefore each Sobol’ index. We can see that the hybrid method outperforms the dimension-dependent methods even for this low-dimensional problem.

It should also be noted that the PCE based methods perform quite well when estimating small Sobol’ indices. This is, in part, due to the fact that Sobol’ indices computed from a PCE (as in (2.18)) are guaranteed to be non-negative, unlike the results of the Saltelli or MF-GSA approaches. The hybrid method, although it is shown to be competitive in this application, has
Figure 5.1 PDFs for multiple Sobol’ index realizations from various methods, in this order: single-fidelity Saltelli sampling (Chapter 2), multifidelity GSA results from [70], single-fidelity PCE, and normalized histograms from the hybrid MLMC-PCE method. The black dashed line indicates the analytical value of each index.

additional advantages when applied to higher-dimensional problems.
5.6.2 Application to chemical reaction network results

We return to the genetic oscillator application from the field of chemical kinetics [66, 65], which was covered in both the stochastic and deterministic modeling contexts in Chapter 3. Here, we are interested in using a deterministic model described by the reaction rates equations (RREs), a system of ODEs which model the time evolving concentration of each chemical species. The genetic oscillator system consists of 9 chemical species and 16 reactions, with each reaction including one uncertain reaction rate parameter. Table 3.1 lists the chemical reactions, propensity functions, and nominal reaction rate parameters for the genetic oscillator problem.

We will derive a multilevel model hierarchy with three levels of fidelity and use the hybrid method to obtain Sobol’ index estimates for the full set of 16 first order indices and 16 total indices. As in Chapter 3, the QoI will be the time-integrated value of the repressor protein,

\[ q(\theta) = \int_0^T R(t; \theta) \, dt. \] (5.9)

Often, when MLMC is applied to a differential equation system, each level of fidelity is defined by adjusting the mesh used to solve the differential equation [38, 67, 74]. Due to the stiff nature of the RREs for the genetic oscillator, a nested mesh refinement scheme in time will not be possible for defining multiple fidelities. Instead, we define different model fidelities by controlling the final time and the required convergence tolerance for the stiff ODE solver. We use SciPy’s stiff ODE solver tools, which acts as a wrapper for the LSODA method [75]. One can specify the tolerance for the ODE solver, which for this experiment will be the same for the relative and absolute. Our three-level model hierarchy is defined in Table 5.2.

Table 5.2 Levels based on ODE tolerance and final time, \( T \), where \( \rho_{HF} \) gives the correlation with the highest-fidelity QoI. The cost is represented by the runtime required to evaluate 1000 QoI samples.

<table>
<thead>
<tr>
<th>tol.</th>
<th>1 x 10^{-1}</th>
<th>1 x 10^{-4}</th>
<th>1 x 10^{-9}</th>
</tr>
</thead>
<tbody>
<tr>
<td>( T )</td>
<td>15</td>
<td>47.5</td>
<td>50</td>
</tr>
<tr>
<td>( \rho_{HF} )</td>
<td>0.884</td>
<td>0.954</td>
<td>1.0</td>
</tr>
<tr>
<td>time</td>
<td>9.8 s</td>
<td>51.5 s</td>
<td>375.5 s</td>
</tr>
</tbody>
</table>

Using the data in Table 5.2, we are able to define a three-level model hierarchy. After the genetic oscillator RREs are solved, the QoI defined by (5.9) is evaluated using the composite trapezoid rule. We next provide a scaled version of the cost. If, as with the Ishigami example, we let the highest-fidelity have a cost of one, then we have the following level-by-level cost, \( (c_0, c_1, c_2) = (0.026, 0.137, 1) \). The costs are proportional to the required runtime in Table 5.2. The correlation between each level and the highest-fidelity is also an important factor in designing a model hierarchy; this fact is discussed in [38]. The correlation structure presented in Table 5.2...
is meant to resemble the structure of the Ishigami example found in [70].

In the following experiments, we distribute the uncertain parameters, denoted by \( \theta \), uniformly \( \pm 10\% \) about the nominal parameters, which is consistent with the distributions used for the genetic oscillator system in Section 3.5.2. We use Legendre polynomials to build the PCE and we use a total polynomial order of 3. For a 16-dimensional function, this results in 969 total PCE terms. In this situation, truncating the PCE basis as a post-processing step could improve the performance of the hybrid method.

As a benchmark, we plot an estimate of the PCE coefficients for the genetic oscillator, computed using MLMC with a sample profile of \( (10^5, 10^4, 10^3) \):

![Figure 5.2 Estimated PCE coefficients for the genetic oscillator up to a total polynomial order of 3, where \((N_0, N_1, N_2) = (10^5, 10^4, 10^3)\). The mean, \( \beta_0 \), has been omitted as it is nearly 25000.](image)

Figure 5.2 indicates some structure in the low-order PCE coefficients, with an increasing level of estimation noise as the order of the PCE basis increases. This is consistent with the variance information provided in Figure 4.1, where the higher-order PCE modes have an increased estimator variance. In this case, a total order truncation of order 1 or 2 is recommended.

In the results that follow, we compute estimates of the conditional variances corresponding to the first order indices and the total indices, which are respectively, \( \hat{S}_i \) and \( \hat{T}_i \), for \( i = 1, \ldots, 16 \). Using our novel estimators for the variances, \( \mathbb{V}[\hat{S}_i], \mathbb{V}[\hat{T}_i] \), for \( i = 1, \ldots, 16 \), we can then place confidence intervals around the estimated ANOVA terms. The confidence intervals are then compared with reference values for the ANOVA terms. We use a total order 2 truncation.

Figure 5.3 illustrates the value of the hybrid method for generating confidence intervals around the Sobol’ index results. For the majority of the estimated GSA terms in Figure 5.3, they are within two standard deviations of the reference values. In practice, if the confidence interval

66
indicates that a certain Sobol' index is unlikely to exceed a certain threshold, it can be considered unimportant, and it can be removed from the optimization target. This sort of iterative scheme, where the confidence intervals inform the targets entering the optimization problem is now made straightforward, given the alternative formulations we have suggested in Sections 4.4 and 5.4. The reference values reported in Figure 5.3 were computed using an adaptive PCE built in the Sandia National Laboratories UQ tool, Dakota, by using an anisotropic expansion refinement and the generalized sparse-grid algorithm [76].

We conclude this section of numerical experiments by comparing the hybrid method with the MF-GSA method of Qian and Willcox [70]. We fix the computational budget at $\bar{C} = 2000$, which is equivalent to 2000 evaluations of the highest-fidelity model. We then compute optimal sample profiles for each method and compare their results. For the hybrid method, if one targets the accuracy of all first order indices with a budget constraint of $\bar{C} = 2000$ (see (4.23)), then the
optimal sample profile is \((C_0, C_1, C_2) = (13279, 7508, 378)\). Similarly, if one targets the accuracy of all total indices with a budget constraint of \(\bar{C} = 2000\) (see (4.24)), then the optimal sample profile is \((C_0, C_1, C_2) = (13306, 7531, 375)\). Similarly, the MF-GSA method does come with the ability to optimize the sample profile, however it is only able to target the mean or the variance of the QoI as an optimization target. The optimal sample profile, targeting the variance of the mean estimator is \((C_0, C_1, C_2) = (974, 172, 50)\) for each conditional term. Recall that the MF-GSA method has a cost which is dimension-dependent and so the total computational budget of 2000 must be divided among each of the Sobol’ indices that are being estimated. We reiterate that as the dimension increases, the hybrid method will become increasing more accurate and cost effective, due in part to the fact that its cost does not depend on the problem dimension. We plot in Figures 5.4 and 5.5 a comparison of the Sobol’ results over multiple realizations of each method. We also include the single-fidelity PCE as a baseline for comparison.

![Graphs showing comparison of methods](image)

**Figure 5.4** First order Sobol’ index results, comparing the single-fidelity PCE (blue PDF), the MF-GSA method (orange histogram), and the optimally-sampled hybrid MLMC-PCE method (green histogram). A representative sample of indices are shown, 6 out of 16. Reference values given by black lines.

Figure 5.4 illustrates that for all first order Sobol’ indices, the hybrid method has a lower variance than the MF-GSA method, while both have an equivalent cost. The MF-GSA method shows improved performance when estimating the total Sobol’ indices, as shown in Figure 5.5, in some cases resulting in a lower variance than the hybrid method. In reality, the hybrid method maintains a lower for variance for the majority of the total Sobol’ indices, especially \(T_5\) through \(T_{10}\), which are nearly zero. The ability to truncate the hybrid method basis, as a post-process, makes it very flexible as a GSA tool. In Figures 5.4 and 5.5, a total order of 1 was used. Additionally, the MF-GSA method is prone to estimate negative Sobol’ index values, while
the hybrid method maintains the non-negativity property of all Sobol’ indices. The single-fidelity PCE, while it maintains non-negativity, does not perform as well as the hybrid method, showing skewed Sobol’ index results, likely due to poor resolution of the PCE coefficients.

Figures 5.4 and 5.5 demonstrate the strong performance of the hybrid method for Sobol’ index computations, as well as its adaptability to the structure of a particular problem. As the dimension increases, the advantages of the hybrid method over other methods, whose cost depends on the problem dimension, will be even more apparent. We also reiterate the unique ability of the hybrid method to compute an optimal sample profile for any desired GSA target and that this optimization can be carried out without need for additional QoI evaluations.

5.7 Conclusions and future work

In this chapter, we have considered practical issues related to the hybrid MLMC-PCE method, introduced in Chapter 4. These included theoretical improvements through the derivation of unbiased estimators for the purposes of GSA, a discussion of the complexities involved with multiple GSA targets, and a discussion of alternative optimization strategies for sample allocation. Further considerations of the hybrid algorithm included an implementation of one possible instance of the hybrid method as well as a discussion of other implementation options. A series of numerical results were presented on the Ishigami function and, as a high-dimensional test case, the genetic oscillator system.

We reiterate the strengths of this method for GSA computations, namely that it uses information from models with multiple fidelities to compute Sobol’ indices using a goal-oriented
approach. This hybrid method is especially useful for high-dimensional QoIs, where other methods for computing Sobol’ indices perform poorly, due to their dimension-dependent cost. In contrast, the PCE allows one to use all model evaluations for each of the relevant coefficients. Furthermore, the hybrid method deals with the major pitfalls of quadrature and sparse-grid methods (i.e. the curse of dimensionality). In these scenarios, high-dimensionality and the number of PCE basis terms can cause performance issues that the hybrid method avoids.

In the future, generalizations of this hybrid framework to the multifidelity MC [39, 70] and approximate control variate setting [68] would be valuable. In this setting, a more diverse range of model forms could be incorporated, including those from machine learning, data assimilation, and simplified-physics models. It is also of interest to the author to explore the limits of this method, turning to even higher-dimensional problems from a variety of domains.

5.8 Acknowledgements

The author would like to acknowledge the computational resources provided by the Dakota toolkit, developed at Sandia National Laboratories.

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia LLC, a wholly owned subsidiary of Honeywell International Inc. for the U.S. Department of Energy’s National Nuclear Security Administration under contract DE-NA0003525. This dissertation describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the dissertation do not necessarily represent the views of the U.S. Department of Energy or the United States Government.
CHAPTER 6

GLOBAL SENSITIVITY ANALYSIS OF RARE EVENT PROBABILITIES

6.1 Introduction

In this chapter, we investigate the challenges of performing GSA for an especially expensive QoI, rare event probabilities. Unlike the previous chapters, the QoI in this section is a moment-based quantity and must be approximated through multiple evaluations of an underlying QoI. The fidelity with which the rare event probability is estimated is thus able to vary and so this notion of multiple estimation fidelities will be applied to the task of GSA.

Rare or extreme events are commonly associated with system failures or anomalies which pose a significant risk. Take, for example, the structural failure of a bridge with unknown material quantities, the damage caused by a tsunami where the ocean floor topography is unknown, or the collateral damage caused by a missile with unknown aerial position and velocity [74, 77, 78, 79]. In such scenarios, it is imperative that rare event probabilities be computed reliably. As we have mentioned in previous chapters, the nominal distributional parameters used in GSA are often assumed and are themselves subject to uncertainty. Thus, in this chapter, we undertake the task of determining the sensitivity of the rare event probability to these distributional hyperparameters. At the time this chapter was written, the resulting article from this study was under review. A preprint of the manuscript, written in collaboration with Alen Alexanderian and Pierre Gremaud, is available online [3].

Let $q(\theta)$ to be a scalar-valued QoI whose inputs are drawn from the sample space, $\Theta \subseteq \mathbb{R}^d$, with associated sigma algebra, $\mathcal{F}$, and probability measure, $P$. For a given threshold value, $\bar{\tau}$,
the corresponding rare event probability is defined as

$$P_\tau = \mathbb{P}(q(\theta) > \tau),$$

(6.1)

where \(\theta \in \Theta\) is a random vector whose entries represent uncertain model parameters. Rare event probabilities (or failure probabilities) are notoriously challenging to compute [78, 80]; indeed, Monte Carlo simulations of (6.1) are extremely expensive for the simple reason that few samples actually hit the rare event domain. Several methods have been proposed to compute \(P_\tau\) more efficiently [78, 80], ranging from importance sampling and Taylor series approximations to subset simulation, the latter of which we explore more in this chapter (see Section 6.3).

The evaluation of the rare event probability (6.1) requires knowledge of the distribution law governing the model parameters, \(\theta\). In practice, such a law is typically assumed or determined experimentally at a high cost. In either scenario, the uncertainty involved with these distributions propagates through to \(q\). Additionally, \(P_\tau\) depends on these assumptions; should they be misguided, the resulting rare event probability is likely be misleading. Our goal is to understand the sensitivity of \(P_\tau\) with respect to distributional assumptions which underly it. To that end, we develop an efficient method to quantify, through GSA, the robustness of \(P_\tau\) to the choice of hyperparameters characterizing the distribution law of the model parameters.

We let \(\xi\) denote a set of hyperparameters characterizing the distribution law of \(\theta\). To account for the uncertainty in \(\xi\), we model the corresponding hyperparameters as random variables. The rare event probability takes the form

$$P_\tau(\xi) = \mathbb{P}(\{q(\theta) > \tau\} | \xi).$$

(6.2)

A number of recent studies have considered how to assess the sensitivity of rare event estimation procedures to uncertain inputs and/or to the distributions of these inputs. There is a general consensus that the double-loop approach — whereby for each realization of \(\xi\), multiple samples of \(\theta\) are used to estimate \(P_\tau\) — is computationally infeasible but for the simplest of problems. An early work [79] combines rare event estimation techniques with the traditional Monte Carlo approach for GSA of the hyperparameters (see Section 2.1.2 for details on Saltelli sampling). Several studies introduce new sensitivity measures [81, 82, 83, 84] which are tailored to make the rare event SA process more tractable. Others perform sensitivity analysis in the joint space of both input parameters and hyperparameters [81, 84, 85, 86]. These methods increase computational efficiency through use of local SA methods [81], surrogate models [84], kernel density estimates [85], and Kriging [86]. A thorough overview of current methods at the intersection of SA and rare event simulation can be found in [87].

The main contribution of this chapter is to show that a double-loop approach can in fact be not only feasible, but computationally efficient in order to perform GSA of \(P_\tau(\xi)\) with respect to \(\xi\). This may seem counterintuitive since, while informative, this type of second-level sensitivity analysis is expensive. However, our approach is structurally simpler than most of the previously
cited work and achieves computational efficiency through a combination of fast methods for rare event simulation together with the use of surrogate models. Specifically, we rely on subset simulation [78] to estimate rare event probabilities and approximate \( P_{\tau}(\xi) \) using a polynomial chaos expansion (PCE); see respectively in Section 6.3 and Section 6.4. The Sobol’ indices for appropriate approximations to \( P_{\tau}(\xi) \) can then be obtained analytically from the PCE. To demonstrate the efficiency gains of the proposed method, we present an illustrative example in Section 6.2 and deploy our approach on it in Section 6.6.1. In Section 6.6.2, we apply the method to a Darcy flow problem, requiring multiple estimates of the rare event probability, to show feasibility in a more computationally demanding framework. We discuss additional challenges, perspectives, and future work in Section 6.7.

6.2 A motivating example

We consider the following illustrative example [88, 78, 89] throughout the chapter,

\[
q(\theta) = -\frac{1}{\sqrt{d}} \sum_{i=1}^{d} \theta_i, \quad \text{(6.3)}
\]

where \( q \) is the QoI in (6.1) and \( \theta = [\theta_1 \ldots \theta_d]^\top \) with independent normally distributed entries, \( \theta_i \sim N(\mu_i, \sigma_i^2), i = 1, \ldots, d \). It is elementary to show that, for any values of the hyperparameters \( \xi = [\mu_1 \ldots \mu_d \sigma_1^2 \ldots \sigma_d^2]^\top \)

\[
q \sim N(\bar{\mu}, \bar{\sigma}^2) \quad \text{with} \quad \begin{cases} 
\bar{\mu} = -\frac{1}{\sqrt{d}} \sum_{i=1}^{d} \mu_i \\
\bar{\sigma}^2 = \frac{1}{d} \sum_{i=1}^{d} \sigma_i^2.
\end{cases} \quad \text{(6.4)}
\]

For a given \( \xi \), the rare event probability is simply

\[
P_{\tau}(\xi) = \frac{1}{2} - \frac{1}{2} \text{erf} \left( \frac{\bar{\tau} - \bar{\mu}}{\sqrt{2} \bar{\sigma}} \right). \quad \text{(6.5)}
\]

We model the uncertainty in the hyperparameters by considering them as independent uniformly distributed random variables with a 10 percent perturbation around their respective nominal values. Figure 6.1 illustrates the case \( d = 5 \) with \( \xi_{\text{nom}} = [1 \ 2 \ 3 \ 4 \ 5 \ 10 \ 8 \ 6 \ 4 \ 2]^\top \) as the nominal value for \( \xi \). In particular, Figure 6.1, right, shows how the uncertainty in \( P_{\tau} \) changes as \( \bar{\tau} \) varies. As \( \bar{\tau} \) increases, i.e., as the event becomes rarer, the uncertainty in \( P_{\tau} \), measured through its coefficient of variation, increases. We contend that this latter behavior is generic for rare event simulations, illustrating the need for methods allowing the quantification of the effects of hyperparameter choices on the uncertainty in \( P_{\tau} \).

To provide qualitative insight, we present a rough estimate for the decrease in the coefficient of variation of \( P_{\tau} \), as the event becomes less rare. We consider a generic \( P_{\tau}(\xi) \) as defined in (6.2) and assume \( P_{\tau} \) is a random variable. Let \( \mu = \mathbb{E}[P_{\tau}] \) and \( \sigma^2 = \mathbb{V}[P_{\tau}] \) be the mean and variance
of \( P_\tau \). Recall that the coefficient of variation of \( P_\tau \) is given by \( \delta(P_\tau) = \sigma/\mu \). Note that for every \( \xi \), we have \( 0 \leq P_\tau(\xi) \leq 1 \); thus, \( P_\tau(\xi) \geq P_\tau(\xi)^2 \) and

\[
\sigma^2 = \mathbb{E}[P_\tau^2] - \mu^2 \leq \mu - \mu^2 = \mu(1 - \mu). \tag{6.6}
\]

Therefore, \( \delta^2(P_\tau) = \sigma^2/\mu^2 \leq \mu(1 - \mu)/\mu^2 = (1 - \mu)/\mu \). Note that as the event becomes less rare, \( \mu \) will grow, resulting in the diminishing of the bound on the coefficient of variation. We point out that the inequality \( (6.6) \) can be obtained directly from the more general Bhatia–Davis inequality [90]. Practically, this means that as the event becomes more rare, its relative uncertainty, measured by the coefficient of variation, will increase.

### 6.3 Rare event simulation

Monte Carlo simulation is a straightforward way of approximating the rare event probability \( P_\tau \) defined in (6.1). Observe that

\[
P_\tau = \mathbb{E}[\chi_\tau] = \int_{\Theta} \chi_\tau(\theta) \pi(\theta) \, d\theta, \tag{6.7}
\]

where \( \chi_\tau \) denotes the indicator function of the set \( \{ \theta \in \Theta : q(\theta) > \bar{\tau} \} \) and \( \pi(\theta) \) is the PDF of \( \theta \). This leads to the following Monte Carlo (MC) estimator,

\[
\hat{P}^{MC}_\tau = \frac{1}{N_{MC}} \sum_{i=1}^{N_{MC}} \chi_\tau(\theta^{(i)}), \tag{6.8}
\]

where \( \theta^{(i)} \) are independent and identically distributed realizations of \( \theta \).

In the case of rare events (i.e., of small probabilities, \( P_\tau \)) the basic MC estimator \( (6.8) \) becomes computationally inefficient. Indeed, consider the coefficient of variation \( \delta(\hat{P}^{MC}_\tau) \) of the
above estimator and observe
\[ \delta^2 \left( \hat{P}_{MC} \right) = \frac{\text{Var} \left[ \hat{P}_{MC} \right]}{\text{E} \left[ \hat{P}_{MC}^2 \right]} = \frac{1 - P_\tau}{NP_\tau} \approx \frac{1}{NP_\tau} \text{ if } 0 < P_\tau \ll 1. \] (6.9)

In other words, ensuring a given accuracy requires \( N_{MC} \approx \frac{1}{P_\tau \delta^2} \). For increasingly rare events, with decreasing \( P_\tau \), the error in (6.8) will increase accordingly. Standard MC methods are thus poor candidates for rare event estimation due to their slow convergence rate, which is compounded by the challenge of estimating a very small quantity.

6.3.1 The subset simulation method

We rely on the subset simulation (SS) method [91, 80] to accelerate rare event computation. This approach decomposes the rare event estimation problem into a series of “frequent event” estimation problems that are more tractable; it has been observed that this may reduce the coefficient of variation by more than an order of magnitude over standard MC [91, 78, 80]. This corresponds to a substantially lower computational burden for estimating rare event probabilities.

Consider the rare event domain, \( F = \{ \theta \in \Theta \mid q(\theta) > \bar{\tau} \} \), and a sequence of nested subsets of \( F \),
\[ F = F_L \subset \cdots \subset F_2 \subset F_1, \]
where \( F_i = \{ \theta \in \Theta \mid q(\theta) > \tau_i \}, i = 1, \ldots, L \) with \( \tau_1 < \tau_2 < \cdots < \tau_L = \bar{\tau} \). The rare event probability \( P_\tau \) can thus be decomposed into a product of conditional probabilities,
\[ P_\tau = \mathbb{P}(F) = \mathbb{P} \left( \bigcap_{i=1}^{L} F_i \right) = \prod_{i=1}^{L} \mathbb{P}(F_i \mid F_{i-1}), \] (6.10)
with, by convention, \( F_0 = \Theta \). Computing \( P_\tau \) according to (6.10) requires an efficient and accurate method for estimating the \( L \) conditional probabilities. We use a modification of the Metropolis-Hastings algorithm to accomplish this [91], which we describe more in Section 6.3.2.

Choosing a proper sequence of thresholds, \( \{ \tau_i \}_{i=1}^{L} \), is a major challenge of the SS method. Since one has little prior knowledge of the PDF of \( q(\theta) \), it is often not feasible to prescribe the sequence of thresholds a priori. Instead, one may require that \( \mathbb{P}(F_i \mid F_{i-1}) = p_0 \), \( i = 1, \ldots, L - 1 \), for some chosen quantile probability, \( p_0 \) [91]. We can then iteratively estimate the proper threshold at each “level” of the algorithm. The SS estimator of (6.10) then takes the form
\[ P_\tau \approx \hat{P}_{\tau}^{SS} = p_0^{L-1} \mathbb{P}(F_L \mid F_{L-1}), \] (6.11)
where the final conditional probability \( \mathbb{P}(F_L \mid F_{L-1}) \) may be estimated using the modified Metropolis procedure. Although \( p_0 = 0.1 \) is a standard choice in engineering applications [74], there has been significant work done to determine optimal values for \( p_0 \) [92]; this, in general,
depends on the QoI under consideration. It has been shown that, for practical purposes, the optimal $p_0$ lies in the interval $[0.1, 0.3]$ and that, within this interval, the efficiency of SS is insensitive to the particular choice of $p_0$ [92]. With the approach for computing the sequence of thresholds in (6.10), each $\tau_i$ is a random variable, estimated via a finite number of conditional samples. Consequently the number of levels, or iterations necessary to terminate SS, is also random. For a sufficiently large number of samples, the number of levels is given in [88] as

$$L - 1 = \left\lceil \frac{\log P_x}{\log p_0} \right\rceil.$$  \hspace{1cm} (6.12)

Before we present a full algorithm outline for SS, we briefly discuss the modified Metropolis-Hastings algorithm for drawing conditional samples.

### 6.3.2 The modified Metropolis-Hastings algorithm

The goal of Markov Chain Monte Carlo (MCMC) sampling is to create a Markov Chain whose stationary distribution is that of the targeted quantity. MCMC has proven to be useful for efficiently sampling from distributions with an unknown PDF. In the context of SS, given a set of samples, $\{\theta^{(1)}, \theta^{(2)}, \ldots \} \in F_{i-1}$, which are elements of a given rare event domain, we want to draw conditional samples allowing for the computation of the conditional probability, $P(F_i \mid F_{i-1})$. The Metropolis-Hastings algorithm is perhaps the best-known method for MCMC sampling; several variants of Metropolis-Hastings are covered in the SS literature [74, 92, 88, 91]. We choose to employ the modified Metropolis algorithm (MMA) described in [74, 89] for generating the conditional samples required for SS.

We begin by assuming that the input parameters of $q$ are statistically independent and normally distributed, although we will discuss the extension to non-Gaussian inputs later. If one begins with a set of parameters which belong to some rare event domain, $F_i$, one seeks to draw samples with the following conditional PDF,

$$\pi(\theta \mid F_i) = \frac{\pi(\theta) \cdot \chi_{F_i}(\theta)}{P(F_i)},$$  \hspace{1cm} (6.13)

where $\chi_{F_i}$ is an indicator function for the set $F_i$. The MMA allows one to construct a Markov Chain where (6.13) is its stationary distribution [52]. Given an initial seed belonging to $F_i$, we have the following algorithm for constructing the Markov Chain:
Algorithm 4 Modified Metropolis algorithm for conditional sampling [74]

Input: Markov Chain seed, $\theta^{(1)} \in F_i$, correlation parameter, $\gamma \in [0, 1]$, length of chain, $n_c$

Output: Markov Chain, $\theta^{(1)}, \ldots, \theta^{(n_c)}$, with stationary distribution given by (6.13)

1: for $i = 1, \ldots, n_c - 1$ do
2: for $j = 1, \ldots, d$ do
3: $\theta'_j = \gamma \theta^{(i)}_j + \sqrt{1 - \gamma^2} Z$, $Z \sim N(0, 1)$
4: end for
5: Accept or reject $\theta'$ according to

$$
\theta^{(i+1)} = \begin{cases} 
\theta', & \text{if } \theta' \in F_i \\
\theta^{(i)}, & \text{if } \theta' \notin F_i 
\end{cases}
$$
6: end for

In Algorithm 4, a correlation parameter, $\gamma$, must be chosen by the user. It is reasonable to expect that if a sample belongs to a rare event domain $F_i$, then samples for the next domain should be in its near vicinity. Thus the user is tasked with balancing how closely to draw conditional samples in the Markov Chain. We will use 0.8 as the default value for $\gamma$, just as in [74]. Notice, for a seed, $\theta^{(1)}$, that follows a standard normal distribution, each generated candidate, $\theta'$, will have mean zero and variance controlled directly by the correlation parameter and so normality is preserved. We refer the interested reader to [78], which provides a high level discussion of MMA as well as other variants of SS. A thorough analysis of MMA and other MCMC algorithms for SS can be found in [89].

Given a set of samples generated by the Markov Chain, one is able to estimate $P(F_i \mid F_{i-1})$ by evaluating (6.8) with the MCMC-generated samples. In this case, the samples are not statistically independent and the estimator will be biased. As we have stated before, although conditional samples will be drawn using the MMA, instead of evaluating the conditional probability directly, we will use the $p_0$ quantile to estimate $\tau_i$ at each level. Next, we discuss the full implementation of SS, as we have described it, and issues related to its computational complexity.

6.3.3 Implementation of subset simulation

We provide an algorithm outline for the SS method in Algorithm 5. As stated before, although we assume Gaussian inputs in the examples considered in this chapter, the SS method can however be applied to non-Gaussian input distributions, see [93, Appendix B] for details. Additional information on the implementation of the SS algorithm, including several variants and their convergence analysis, is available in [91, 88, 74, 89]. As this MMA implementation reuses the input parameters from each previous level to estimate the threshold for the next level, this method does not require any burn-in samples to draw from the conditional distribution; it begins by sampling from the previous rare event domain [74]. This is an attractive feature of the method and which is not standard among MCMC methods. On the theoretical side, the SS algorithm is
asymptotically unbiased and \( \hat{P}^{SS}_\tau \) converges almost surely to the true rare event probability, \( P_\tau \). For a detailed convergence analysis of SS and derivation of its statistical properties, see [91].

\begin{algorithm}
\textbf{Algorithm 5} Subset Simulation
\begin{algorithmic}
\State \textbf{Input:} Rare event threshold, \( \bar{\tau} \), MCMC samples per level, \( N_{SS} \), quantile probability, \( p_0 \), routine that evaluates QoI, \( q(\theta) \)
\State \textbf{Output:} Estimate of rare event probability: \( \hat{P}^{SS}_\tau \)
\State \hspace{1cm} 1: \( i = 1 \) \hspace{1cm} \{i indicates the current level\}
\State \hspace{1cm} 2: Draw \( N_{SS} \) samples of \( \theta \) from the appropriate distribution
\State \hspace{1cm} 3: Evaluate \( N_{SS} \) samples of \( q(\theta) \) and compute \( \tau_1 \) as the \( p_0 \) quantile
\State \hspace{1cm} 4: Save the \( \lfloor N_{SS} \cdot p_0 \rfloor \) inputs such that \( q(\theta) > \tau_1 \) as seeds for the next level
\State \hspace{1cm} 5: \textbf{while} \( \tau_i < \bar{\tau} \) \textbf{do}
\State \hspace{1cm} \hspace{1cm} 6: \( i = i + 1 \)
\State \hspace{1cm} \hspace{1cm} 7: Sample \( \theta \) by creating \( \lfloor N_{SS} \cdot p_0 \rfloor \) Markov Chains, each with length \( \lfloor p_0^{-1} \rfloor \) \hspace{1cm} \{See Algorithm 4\}
\State \hspace{1cm} \hspace{1cm} 8: Using MCMC samples of \( \theta \), evaluate \( q(\theta) \) and compute \( \tau_i \) as the \( p_0 \) quantile
\State \hspace{1cm} \hspace{1cm} 9: Save the \( \lfloor N_{SS} \cdot p_0 \rfloor \) inputs such that \( q(\theta) > \tau_i \) as seeds for the next level
\State \hspace{1cm} 10: \textbf{end while}
\State 11: \( L = i + 1 \)
\State 12: Using \( \theta \) samples from \( F_{L-1} \), sample \( q(\theta) \) and estimate \( P(F_L \mid F_{L-1}) \) using MC
\State 13: Evaluate \( \hat{P}^{SS}_\tau = p_0^L P(F_L \mid F_{L-1}) \)
\end{algorithmic}
\end{algorithm}

Before discussing practical issues related to the computational complexity of running SS, we illustrate the results of Algorithm 5, specifically the MMA, using a two variable example. Consider the motivating example from Section 6.2, where the dimension is \( d = 2 \). Let both \( [\theta_1, \theta_2] \sim \mathcal{N}(0, I) \) and \( \bar{\tau} = 4 \). We then illustrate the iterative nature of the sampling algorithm outlined in Algorithm 4.

In Figure 6.2, the plot on the left shows the input samples in the \( (\theta_1, \theta_2) \)-space for each level of the SS algorithm. The plot on the right shows the corresponding output of the QoI for each level. In each plot, the dashed lines indicate the rare event threshold for each level of the iteration. The SS algorithm can be considered a search algorithm, as it iteratively explores the input space and uses conditional sampling to approach the rare event domain, \( F \). By updating the definition of a rare event at each level, one can guarantee that more samples will hit your specified rare event domain, and thus your estimation will be more efficient.

\subsection{6.3.4 Computational cost of subset simulation}

We turn now to the computational cost of estimating \( P_\tau \) using SS. The computational cost is measured in terms of the number of QoI evaluations required to run the algorithm. As the number of levels, \( L \), is random, so is the computational cost associated with SS. For simplicity, we assume for our cost analysis that a sufficient number of samples has been used so that \( L \)
Figure 6.2 Example (6.3) with \( d = 2 \). Left: samples in input space generated by MMA sampling with thresholds for each level. Right: QoI output for each level with respective thresholds.

is deterministic. The total number of QoI evaluations required by SS is \( L \cdot N_{SS} \), where \( N_{SS} \) is a user-defined parameter that determines the number of samples per intermediate level of the iteration. As one can see in Algorithm 5, the \( \lfloor N_{SS} \cdot p_0 \rfloor \) Markov Chain samples from one level will acts as the seeds for the next level of Markov Chains, each with length \( \lfloor p_0^{-1} \rfloor \).

It is important to emphasize the computational advantages of this method over standard MC. Say, for example, the true rare event probability is \( 10^{-6} \) and we wish to estimate \( P_\tau \) with a coefficient of variation within \( \delta = 0.1 \). For standard MC sampling, we would need \( N_{MC} \geq 1/(\delta^2 \cdot P_\tau) = 10^8 \) evaluations of the QoI. Take the SS method with a quantile probability of \( p_0 = 0.1 \). Then, according to (6.12), we would expect to have \( L = 7 \), corresponding to 7 iterations of conditional sampling. The coefficient of variation for each of the conditional probabilities is more difficult to quantify, however, as in the case of the standard MC estimator, they are inversely proportional to the probability itself [91], in this case, \( p_0 \). Roughly speaking, for 7 iterations of SS, with each iteration of conditional sampling requiring \( N_{MC} \geq 1/(\delta^2 \cdot p_0) = 10^3 \) samples, one would expect to achieve the desired accuracy with approximately \( 10^5 \) evaluations of the QoI. This significant reduction in the cost of estimating \( P_\tau \) with SS makes it a powerful method for rare event estimation.

We lastly emphasizes the advantages of SS for estimating rare event probabilities in the context of QoIs with high-dimensional inputs. Not only does SS improve upon the slow convergence rates of standard MC by a wide margin, it also inherits the property of having a convergence rate which is independent of the input dimension [91, 78].

6.4 Surrogates for GSA of rare event probabilities

We seek to apply variance-based GSA to \( P_\tau(\xi) \), defined in (6.2), with respect to components of \( \xi \). To mitigate the computational expense of performing such analysis, we combine the SS
algorithm and surrogate models, in the form of polynomial chaos expansions (PCEs). We assume \( \xi \) to be an \( d \)-dimensional vector with independent entries. Notice that \( d \) need not be smaller or larger than the size of \( \theta \); we will cover a case later where the number of hyperparameters is far smaller than the number of parameters. The procedure, which amounts to a double-loop sampling approach, is outlined below:

- Generate hyperparameter samples: \( \{ \xi^{(j)} \}_{j=1}^{N_{\text{samp}}} \)
- For each \( j \in \{1, \ldots, N_{\text{samp}}\} \), estimate \( P_{\bar{\tau}}(\xi^{(j)}) \) using SS; denote these estimates by \( \tilde{P}_{\bar{\tau}}^{(j)} = \text{SS}(P_{\bar{\tau}}(\xi^{(j)})) \)
- Use the (noisy) function evaluations \( \{ \tilde{P}_{\bar{\tau}}^{(j)} \}_{j=1}^{N_{\text{samp}}} \) to compute a surrogate model: \( \tilde{P}_{\bar{\tau}}(\xi) \approx P_{\bar{\tau}}(\xi) \)
- Compute the Sobol’ indices of \( \tilde{P}_{\bar{\tau}}(\xi) \)

Instead of using SS for computing \( P_{\bar{\tau}}(\xi^{(j)}) \), one may be tempted to apply a surrogate further “upstream” by computing a surrogate model \( \tilde{q}_{\xi^{(j)}}(\theta) \) for \( q(\theta) \) from samples \( \{ q(\theta^{(k)}) \}_{k=1}^{n} \) drawn from law of \( \theta \) as determined by \( \xi^{(j)} \). This surrogate model of \( q \) can then be used for fast approximation of the rare event probability, \( P_{\bar{\tau}}(\xi^{(j)}) \). This procedure, however, has two major pitfalls: (i) an expensive surrogate modeling procedure must be carried out for each \( j \in \{1, \ldots, N_{\text{samp}}\} \) and, more importantly, (ii) surrogate models are typically poorly suited to the task of rare event estimation. Indeed, surrogates typically fail to capture the tail behavior of the distribution of the QoI \( q \), making them unsuitable for rare event simulations. This shortcoming is well-documented in the uncertainty quantification literature [94, 95] although efforts are being made to tailor the surrogate model construction process for the efficient estimation of rare event probabilities [96, 95].

### 6.5 PCE surrogate for rare event probability

Our approach leverages the properties of PCE surrogates for fast estimation of Sobol’ indices, see Chapter 2 or [32, 29] for further information. This approach takes advantage of the regularity of the mapping \( \xi \mapsto P_{\bar{\tau}}(\xi) \). Specifically, assuming the PDF of \( \xi \) satisfies certain (mild) differentiability and integrability conditions, one can show that \( P_{\bar{\tau}}(\xi) \) is a differentiable function of \( \xi \); see [97, Proposition 3.5].

Now applying the definition of the PCE (2.10) to the rare event probability, we define the PCE of \( P_{\bar{\tau}} \) as

\[
\tilde{P}_{\bar{\tau}}(\xi) = \sum_{k=0}^{N_{\text{PC}}} \beta_k \Psi_k(\xi),
\]

(6.14)

where \( \{ \Psi_k \}_{k=0}^{N_{\text{PC}}} \) belong to a family of orthogonal polynomials and \( \{ \beta_k \}_{k=0}^{N_{\text{PC}}} \) are the (scalar) PCE coefficients. In this chapter, we use a total order truncation scheme for the PCE. As mentioned
in Chapter 2, for a total polynomial order of $r$, the truncation level, $N_{PC}$, satisfies

$$N_{PC} + 1 = \frac{(d + r)!}{d! r!},$$

and so one will need to carefully construct the polynomial basis, in order to build an accurate surrogate of $P_\tau$ with a minimal set of basis elements.

The PCE coefficients can be computed in a number of ways, including non-intrusive spectral projection or regression [32, 28, 29]. A regression-based approach is preferred in this context because the evaluations of $P_\tau$ are noisy due to sampling errors incurred in the SS procedure. The noise is balanced in recovering the PCE coefficients by using a regularization term to promote sparsity. We refer the interested reader to [35] for a more general description of regression for PCE and so-called compressive sampling techniques. We estimate the vector $\beta = [\beta_0, \beta_1, \ldots, \beta_{N_{PC}}]$ from function evaluations $\hat{P}_\tau^{(j)} = \text{SS}(P_\tau(\xi^{(j)})), j = 1, \ldots, N_{\text{SAMP}}$, by solving the penalized least squares problem,

$$\min_{\beta} \sum_{j=1}^{N_{\text{SAMP}}} \left( \sum_{k=0}^{N_{PC}} \beta_k \Psi_k(\xi^{(j)}) - \hat{P}_\tau^{(j)} \right)^2 \text{ s.t. } ||\beta||_1 \leq \alpha. \quad (6.15)$$

This particular formulation of the regularized least squares problem is known as LASSO, although other formulations have been used in the literature [35]. In (6.15), the penalty parameter, $\alpha$, acts as a sparsity control on the recovered PCE coefficients. We generate the realizations $\{\xi^{(j)}\}_{j=1}^{N_{\text{SAMP}}}$ of the hyperparameter vector through Latin hypercube sampling of $\xi^{(j)}$ [98]. Latin hypercube is a popular choice for PCE due to its space-filling properties, although Quasi-Monte Carlo or other sampling approaches may be useful. For further details on the implementation of sparse regression for PCE, see [99, 35]. The numerical results in Section 6.6 are obtained using the SPGL1 solver [100]. Finally, since the main cost in this process is using SS to estimate the rare event probabilities, the post-processing work needed to choose the appropriate PCE basis and tune $\alpha$ can be done at a negligible cost. This freedom is an additional benefit of choosing the sparse regression approach over quadrature-based techniques.

**6.5.1 GSA of $P_\tau$ using the PCE surrogate**

As covered in Chapter 2, the Sobol’ indices of a PCE surrogate can be computed analytically. For example, the first order Sobol’ indices, $S_i(P_\tau), i = 1, \ldots, d$, of $P_\tau$ can be approximated as follows:

$$S_i(P_\tau) \approx S_i(\hat{P}_\tau) = \frac{\sum_{k \in K_i} \beta_k^2 E[\Psi_k^2]}{\sum_k^{N_{PC}} \beta_k^2 E[\Psi_k^2]}, \quad (6.16)$$

where $K_i$ denotes the set of all PCE terms that depend only on $\xi_i$ (see (2.17)). Sobol’ indices for arbitrary subsets of variables, as well as total indices, can be computed in an analogous manner [27]. In practice, PCE surrogates with modest accuracy are often sufficient to obtain reliable estimates of Sobol’ indices, a point which will be revisited in the next section.

While the above approach for GSA of $P_\tau$ does require repeated simulations of the QoI, $q$,
during the calls to the SS algorithm, it can provide a dramatic computational speedup over the standard Saltelli method (see Section 2.1.2) for computing the Sobol’ indices of $P_T$ [7]. Indeed, a fixed sample $\{\xi(j)\}_{j=1}^{N_{\text{SAMP}}}$ with modest $N_{\text{SAMP}}$ is sufficient to compute the PCE surrogate from which the Sobol’ indices can be computed at a negligible computational cost. Moreover, the sparse regression approach for estimating PCE coefficients is forgiving of noisy function evaluations. Therefore, large sample sizes are not needed in the calls to the SS algorithm. We demonstrate the merits of the proposed approach for GSA of rare events in Section 6.6.

6.6 Numerical results

We summarize, in Section 6.6.1, the computational results for the motivating example from Section 6.2. A more computationally demanding model problem, involving flow through porous media, is considered in Section 6.6.2.

6.6.1 Results for the analytic test problem

We consider the example from Section 6.2, where the hyperparameter dimension is 10, and study $P_T$ where $\tau = 3$. To establish a baseline for the values of the Sobol’ indices of $P_T(\xi)$, we compute the total order Sobol’ indices directly from (6.5) using Saltelli sampling. The reference Sobol’ indices are computed with $10^6$ samples for each of the conditional terms; convergence was numerically verified. We plot the reference total indices in Figure 6.3 for comparison. We now compare the reference indices with those obtained through the PCE surrogate when $P_T(\xi)$ is computed analytically using (6.5). We allocate $10^3$ samples of $P_T(\xi)$ each for the Saltelli sampling method and sparse regression PCE method. The Saltelli method requires $N_{\text{MC}}(d + 1)$ QoI evaluations to estimate the total indices, and so we divide the budget of $10^3$ evaluations equally among each of the conditional terms. Each PCE coefficient can be estimated using the full set of $10^3$ samples. For a fair comparison, we use Latin hypercube sampling to generate the $\xi$ samples for both the PCE and Saltelli method. We also use a total PCE order of 3 and the penalty parameter $\alpha = 5 \times 10^{-2}$. Given that the set of total indices is computed, in each method, using a finite number of samples, each index is a random variable with an associated distribution. We compare two standard deviations of each total index for the two GSA methods. In each case, we compute $10^3$ realizations of the full set of total indices and compare their respective standard deviations in Figure 6.3. Figure 6.3 illustrates the higher accuracy, or lower variance, of PCE with sparse regression over Saltelli sampling: the standard deviation of the largest Sobol’ index is roughly 3 times smaller with sparse regression than it is with Saltelli sampling. This gap in accuracy appears to diminish for smaller indices, although the methods do not show comparable accuracy until the indices are below 0.1. As $P_T$ can be expressed analytically, there will be additional benefits of the sparse regression method to be seen when one performs GSA on a rare event probability with noisy estimations due to sampling. We note that the total order of the PCE basis and the penalty parameter, $\alpha$, which are user-defined parameters, can
be changed without the need for additional runs of SS. These parameters can be cross validated as a post-processing step after the rare event simulation step, providing flexibility in the PCE approach without adding any significant computational burden.

When combining PCE-based GSA with SS for estimating $P_\bar{\tau}(\xi)$, there is a tradeoff between the inner loop cost of estimating $P_\bar{\tau}$ via SS and the outer loop of aggregating $P_\bar{\tau}$ samples to build the PCE. In Figure 6.4, we separately vary $N_{SS}$ and $N_{SAMP}$ and examine the resulting distribution of the total Sobol’ indices, computed via sparse regression PCE. For a fixed $N_{SAMP}$, we compute multiple realizations of the total indices for several values of $N_{SS}$. Figure 6.4 (top) displays the expected value of the total indices for $N_{SAMP} = 100$. Regardless of how accurately we estimate $P_\bar{\tau}$, the indices do not approach their true values because the PCE is built using an inadequate number of samples, resulting in a poor surrogate. By contrast, Figure 6.4 (middle) shows that for $N_{SAMP} = 10^3$, we only need a modest $N_{SS}$ to approximate the Sobol’ indices. Indeed, for $N_{SS} = 500$, we are able to resolve the total indices very well. We also examine the case of $N_{SAMP} = 10^4$ in Figure 6.4 (bottom). Again, we are able to resolve the total indices well using only $N_{SS} = 500$ and are able to achieve the correct ordering for as little as $N_{SS} = 100$.

These results indicate that (i) a modest number of samples allocated to SS is enough to get a rough estimate of $P_\bar{\tau}$ and (ii) a moderate number of evaluations of $P_\bar{\tau}(\xi)$ is then sufficient for accurate GSA. In other words, given inexpensive, low-fidelity estimations of $P_\bar{\tau}$, we are still able to extract accurate GSA results, due to the fact that the sparse regression technique is robust to noisy QoI evaluations.

**Figure 6.3** Total Sobol’ indices of $P_\bar{\tau}$, with $\bar{\tau} = 3$, from (6.5); the error bars illustrate the variability of the two sampling methods (Saltelli sampling and sparse regression PCE) around the reference values (blue bars).
Figure 6.4 Mean Total Sobol’ indices over 1000 realizations, varying the computational cost of SS and the PCE construction. Each plot varies $N_{\text{SAMP}}$ and each colored bar varies $N_{SS}$, with the final bar of each index corresponding to the analytic $P_{\bar{\tau}}$.

6.6.2 Subsurface flow application

We next consider a problem from porous media flow. This problem has been used previously in the context of rare event estimation in [74] as it pertains to the long-term reliability of nuclear waste repositories. We consider the equations for single-phase, steady state flow in a square
domain, $\mathcal{D} = [0, 1]^2$:

$$-\nabla \cdot \left( \frac{\kappa}{\mu} \nabla p \right) = 0 \text{ in } \mathcal{D},$$

$$p = 1 \text{ on } \Gamma_1,$$

$$p = 0 \text{ on } \Gamma_2,$$

$$\nabla p \cdot n = 0 \text{ on } \Gamma_3,$$

(6.17)

where $\kappa$ is the permeability of the medium, $\mu$ is the viscosity, and $p$ is the pressure. The boundaries $\Gamma_1, \Gamma_2$, and $\Gamma_3$ indicate the left boundary, the right boundary, and the top/bottom boundaries, respectively. The Darcy velocity is defined as $v = -\frac{\kappa}{\mu} \nabla p$. In this chapter, we let $\mu = 1$. In practical scenarios, the permeability of the medium would be determined experimentally using a finite number of measurements. Thus it is appropriate to consider the permeability field as an uncertain quantity, which we model as a random field. We then consider the flow of particles through the medium and focus on determining the probability that said particles do not reach the outflow boundary in a given amount of time. Our goal is to perform GSA with respect to the hyperparameters that define the distribution law of the permeability field.

### 6.6.3 The statistical model for the permeability field

Following standard practice [74, 15], we model the permeability field as a log-Gaussian random field,

$$\log \kappa(x, \omega) = a(x, \omega) = \bar{a}(x) + \sigma_a z(x, \omega),$$

(6.18)

where $x \in \mathcal{D}$ and $\omega$ belongs to sample space that carries the random process. Here, $\bar{a}$ is the mean of the random field, $\sigma_a$ is a scalar which controls the pointwise variance of the field, and $z$ is a centered (zero-mean) random process. We then specify a statistical description of $z$, which encapsulates a number of hyperparameters. We define the covariance function of $z$ to be given by

$$c_z(x, y) = \exp \left( -\frac{|x_1 - y_1|}{\ell_x} - \frac{|x_2 - y_2|}{\ell_y} \right), \quad x, y \in \mathcal{D},$$

(6.19)

where $\ell_x$ and $\ell_y$ denote the correlation lengths in the horizontal and vertical directions. The random field is represented via a truncated Karhunen-Loève expansion (KLE) [32, 74],

$$a(x, \omega) \approx \bar{a}(x) + \sum_{k=1}^{N_{KL}} \sqrt{\lambda_k} \theta_k(\omega) e_k(x).$$

(6.20)

In this representation, $\theta_1, \ldots, \theta_{N_{KL}}$ are independent, standard normal random variables and $(\lambda_k, e_k), k = 1, \ldots, N_{KL}$ are the leading eigenpairs of the covariance operator of the stochastic process [32]. Our setup for the uncertain log-permeability field follows the one in [15]: we use permeability data from the Society for Petroleum Engineers [101] to define the mean of the random field, $\bar{a}$. After a generalized eigenvalue problem is solved [16], the KLE is truncated according to the chosen metric. Following [15], we monitor the fraction of the average variance
as measured by the eigenvalue ratio,

\[ r_i = \frac{\sum_{k=1}^{i} \lambda_k}{\sum_{k=1}^{\infty} \lambda_k}. \tag{6.21} \]

Once an acceptable ratio has been met, the truncation level, denoted \( N_{KL} \), is prescribed. The resulting random vector, \( \theta = [\theta_1 \ \theta_2 \ \cdots \ \theta_{N_{KL}}]^T \) fully describes the uncertainty in the log-permeability field.

In our numerical experiments, we truncate the KLE so that at least 90% of the average variance of the field is maintained (i.e. \( r_i \geq 0.9 \) in (6.21)). Since the eigenvalue decay is slowest for small correlation lengths, we choose the largest \( N_{KL} \) necessary and fix the number of KL modes for all realizations of \( \xi \). For \( \ell_x = \ell_y = 0.4 \), which are the smallest correlation lengths considered in this chapter, we require at least \( N_{KL} = 126 \). The number of retained KL modes then determines the dimensionality of the rare event estimation problem, and is henceforth fixed at 126. In this regime, the high-dimensional capabilities of the SS algorithm are useful.

For illustration, we plot two realizations of the random field, with the corresponding pressure and velocity fields obtained by solving the governing PDE (6.17), in Figure 6.5. In our computations, we solve the PDE using piecewise linear finite elements in MATLAB’s finite element toolbox with 50 mesh points in each direction.

### 6.6.4 Definition of the QoI and rare event problem

The solution of the PDE model (6.17) allows for the computation of the Darcy velocity, \( \mathbf{v} = -\frac{k}{\mu} \nabla p \). The position \( \mathbf{x} \) of a particle moving with the flow through the medium is then determined by the following ODE,

\[ \frac{d\mathbf{x}}{dt} = \mathbf{v}, \]
\[ \mathbf{x}(0) = \mathbf{x}_0, \tag{6.22} \]

where \( \mathbf{x}_0 \) is the initial position of the particle. In our case, we focus on particles beginning from the initial position \( \mathbf{x}_0 = [0 \ 0.5]^T \). The solution of (6.22) depends not only on time but also on the uncertainty in the permeability field, described by \( \theta \), due to dependence of \( \kappa \) on \( \theta \) (i.e., \( \mathbf{x} = \mathbf{x}(t, \theta) \)). We then define the scalar QoI, \( q \), as the hitting time, that is, the time it takes a particle to travel through the medium from left to right:

\[ q(\theta) = \min\{t : x_1(t, \theta) = 1\}. \]

We aim to determine the rare event probability, \( P_r = \mathbb{P}(q > \bar{\tau}) \). The parameters \( \ell_x, \ell_y, \) and \( \sigma_a \) parametrize the uncertainty in the permeability field; we consider them as hyperparameters and set \( \xi = [\ell_x \ \ell_y \ \sigma_a]^T \). We set the nominal values of the hyperparameters \( \xi_{\text{nom}} = [0.4 \ 0.4 \ 0.8]^T \). We simulate realizations of the permeability field at these nominal hyperparameters and plot the distribution of \( q \). Each of these realizations requires one PDE solve and one ODE solve. As
Figure 6.5 Left: plots showing two realizations of the log permeability field. Right: the corresponding pressure solution and arrows indicating the resulting Darcy velocity field.

Figure 6.6 Left: Histogram of $q$ for nominal hyperparameters. Vertical line indicates rare event threshold of $\bar{\tau} = 4.5$. Right: histogram of the rare event probability, estimated via SS with uniformly distributed hyperparameters.

illustrated in Figure 6.6, the distribution for $q$ corresponds to a heavy-tailed distribution. We select as the threshold $\bar{\tau} = 4.5$ and consider quantifying the sensitivity of $P_{\bar{\tau}}(\xi)$ with respect to
the hyperparameters defining the KLE. Note, in this case, the dimension of $\xi$ is much smaller than that of $\theta$, meaning that the PCE surrogate will be relatively low-dimensional, while the rare event estimation will be high-dimensional.

6.6.5 Rare event probabilities and GSA

In our first set of experiments, we use SS with $N_{SS} = 10^3$ samples per intermediate level; each of these samples corresponds to one solution of the full subsurface flow problem, including a PDE and ODE solve. For each evaluation of SS, approximately 5 intermediate levels are used, resulting in approximately $5 \times 10^3$ function evaluations per estimation of $P_\tau$. Our hyperparameters are drawn from a uniform distributed centered at $\xi_{nom}$ with a spread of plus or minus 10% of $\xi_{nom}$. We use these SS estimations of $P_\tau(\xi)$ in order to build the corresponding PCE surrogate, where the polynomial basis is truncated at a total polynomial order of 5. Note the decision of where to truncate the PCE basis does not need to be made prior to estimating the set of $\hat{P}_\tau(\xi)$ samples.

The samples for the hyperparameters are drawn using a Latin hypercube sampling scheme. We use $10^3$ estimations of $P_\tau(\xi)$ to construct the PCE surrogate. Again, we use sparse regression to recover the PCE coefficients, while promoting sparsity in the set of PCE coefficients, and so mitigating the effects of noise induced by SS. In Figure 6.7, we use two different values of $\alpha$ when promoting sparsity in order to illustrate the effect of $\alpha$ on the results. When $\alpha$ is made smaller, the PCE coefficients decrease in magnitude, promoting a sparser PCE spectrum. For either choice of $\alpha$ in Figure 6.7, the ordering of the total Sobol’ indices remains consistent, and thus, conclusions with respect to parameter sensitivity are unaffected. For this experiment, we therefore conclude that choosing $\alpha$ by trial and error is sufficient. Should one encounter a scenario where the GSA results are more sensitive to $\alpha$, more systematic approaches are possible,

![Figure 6.7](image.png)

**Figure 6.7** Left: total Sobol’ indices for $P_\tau(\xi)$ computed from recovered PCE coefficients; results are reported with regularization constant $\alpha = 1$ and $\alpha = 5 \times 10^{-2}$. Right: PDF of PCE surrogate compared with $P_\tau$ evaluation histogram. Used $N_{SAMP} = 10^4$ for better resolution of distributions.
such as an L-curve test, cross validation approaches, etc. [28, 35].

We lastly return to the key point made in Section 6.6.1, that the proposed method is capable of producing reliable GSA results, while using a modest number of inner and outer loop samples ($N_{SS}$ and $N_{SAMP}$, respectively). Put another way, one can obtain high-fidelity GSA results by combining low-fidelity estimations of the rare event probability with sparse PCE surrogates. In Figure 6.8, we report results corresponding to $N_{SS} = 500$. In the left panel of the Figure we study the effect of $N_{SAMP}$ on the PDF of the PCE surrogate. In the right panel, we plot the Sobol’ indices corresponding to each of the computed surrogates. The results in Figure 6.8 should also be compared with those in Figure 6.7, where larger values of $N_{SS}$ and $N_{SAMP}$ were used. This experiment indicates that $P_{\tau}$ and the Sobol’ indices themselves can be well-approximated with a modest number of samples in both the inner and outer loops. In this case, using both $N_{SS}$ and $N_{SAMP}$ on the order of $10^2$ is sufficient for obtaining accurate GSA results. The combined cost of this method is thus reduced by a significant margin compared with the similar results in Figure 6.7. The efficiency gains of this method indicate the potential for deployment on problems which would otherwise be intractable.

6.7 Conclusion

We have shown that the feasibility of the double-loop approach for GSA of rare event probabilities can be significantly extended beyond simple applications. This requires appropriate acceleration methods; in our case, this is achieved through subset simulation and the choice of a polynomial surrogate model allowing for the analytical calculation of Sobol’ indices. This approach is
conceptually simple and does not require the development of new, ad hoc sensitivity concepts. While we have extended the range of applicability of the double-loop approach, we acknowledge that more research is needed to deal with computationally expensive, high-dimensional problems. Specifically, in the event that both the inner and outer loop parameters are high-dimensional, surrogate construction methods will need to be employed that are effective in high dimensions. Here, we note the applicability the hybrid MLMC-PCE approach developed in Chapter 4, where one may define multiple fidelities according to the estimation quality of the rare event probability. We also note that multifidelity methods have been successfully used for the acceleration of subset simulation [74], indicating further potential for this work.

The efficiency of our method crucially depends on working with surrogate models for which sensitivity measures—here, Sobol’ indices—can be computed cheaply. This clearly limits the type of GSA which can be carried out by our approach. More generally, if $q$ is the original QoI and if $\tilde{q}$ is the resulting QoI for a given surrogate model, more work is needed to understand the relationship between the approximation error, $q - \tilde{q}$, and the resulting GSA error, $I(q) - I(\tilde{q})$, where $I(\cdot)$ is some sensitivity measure. More explicitly, there may be room for the development of “cheap” surrogate models with moderate approximation errors and small GSA errors. Additionally, both our sensitivity analysis method as well as surrogate modeling approach rely on the assumption that the hyperparameters are independent. In some cases one might be interested in GSA of rare event probabilities to both hyperparameters and additional parameters in a model that might be uncertain and possibly correlated. Therefore, another interesting line of inquiry is to consider GSA of rare event probabilities with respect to correlated parameters [24]. Further study may also include extensions of our approach to other moment-based QoIs (e.g. CDF approximation, skewness, kurtosis) and the use of perturbation-based methods for GSA [82] as opposed to considering a discrete set of hyperparameters.
This chapter summarizes the contributions of this dissertation and also points to future work in the areas investigated.

The main theme of this dissertation is leveraging multiple model fidelities to improve the efficiency and accuracy of GSA and, in some situations, to enable GSA where it has previously been considered infeasible. In Chapter 3, a framework was discussed, by which the Sobol’ indices of stochastic chemical models can be approximated by the Sobol’ index results of a related system of deterministic models. This type of surrogate GSA, when done appropriately, can result in approximate Sobol’ index results for the stochastic model without any need to evaluate the stochastic model itself. In Chapters 4 and 5, we examined a novel hybrid method for GSA, built from the combination of multilevel Monte Carlo and polynomial chaos expansions. We have shown that the hybrid method can distribute the computational effort across the model hierarchy in a manner than is optimally-tailored for the computation of Sobol’ indices. Practical aspects of the hybrid method were discussed and the method was compared with other state of the art sampling-based GSA methods. Finally, Chapter 6 examines the challenge of performing GSA for a moment-based quantity of interest, namely a rare event probability. In this scenario, the fidelity with which one approximates the rare event probability is determined by amount of sampling performed. In this case, accurate GSA of the rare event probability is accomplished by combining sparse polynomial surrogates with subset simulation, which obtains low-cost approximations of the rare event probability.

A critical challenge that is covered in Chapter 3 is the issue of multiple sources of uncertainty, which may not be uniform across a given model hierarchy. Some work has been done in the area of multilevel methods for stochastic chemical systems [57, 38]. Future work should examine
the challenges of performing GSA in the presence of multiple sources of uncertainty, as well as developing a general framework for using models with multiple fidelities, each with their own sources of uncertainty.

Another area of future work is the generalization of the hybrid MLMC-PCE method from Chapter 4 to the multifidelity and approximate control variate settings. Such an approach would allow one to perform goal-oriented GSA using multiple fidelities, but with the increased flexibility that is offered by multifidelity and approximate control variate methods [70, 68].

Finally, we touch on the material from Chapter 6, where the QoI is a moment-based quantity of interest which, in most cases, cannot be evaluated directly. There has been some work done in this area in recent years [39]. Future work should address the relationship between model fidelity and the accuracy of the resulting sensitivity analysis. This ties into a larger open question about surrogates and the analysis of mathematical models: how accurate must a surrogate be to enable useful analysis of the original model? While this question is pertinent to the study of polynomial surrogates and GSA, it stands out as one of the fundamental questions facing practitioners across all fields of science and engineering.


APPENDIX
A.1 Single level covariance \( C \left[ (\hat{\beta}_k)^2, (\hat{\beta}_z)^2 \right] \)

Proof. Letting \( P_k^{(i)} = q^{(i)} \Psi_k^{(i)} \) and \( b_k = \mathbb{E} [\Psi_k^2] \), we have

\[
C \left[ (\hat{\beta}_k)^2, (\hat{\beta}_z)^2 \right] = \frac{1}{N^4 b_k^2 b_z^2} C \left[ \left( \sum_{i=1}^N P_k^{(i)} \right)^2, \left( \sum_{i=1}^N P_z^{(i)} \right)^2 \right]
\]

\[= \frac{1}{N^4 b_k^2 b_z^2} \left[ \sum_{i=1}^N P_k^{2,(i)} + \sum_{i=1}^N P_k^{(i)} \sum_{j=1, j \neq i}^N P_k^{(j)} + \sum_{i=1}^N P_z^{2,(i)} + \sum_{i=1}^N P_z^{(i)} \sum_{j=1, j \neq i}^N P_z^{(j)} \right]
\]

\[= \frac{1}{N^4 b_k^2 b_z^2} \left[ C \left( \sum_{i=1}^N P_k^{2,(i)}, \sum_{i=1}^N P_z^{2,(i)} \right) + C \left[ \sum_{i=1}^N P_k^{2,(i)}, \sum_{j=1, j \neq i}^N P_z^{(j)} \right] + C \left[ \sum_{i=1}^N P_z^{2,(i)}, \sum_{j=1, j \neq i}^N P_z^{(j)} \right] \right].
\]

(A.1)

We will now consider all the covariance contributions separately. The first term is
The second contribution we consider is

\[
\begin{align*}
C \left[ \sum_{i=1}^{N} P^2_k, \sum_{i=1}^{N} P^2_z \right] &= N \left( \mathbb{E} \left[ q^2 \Psi_k^2 \Psi_z^2 \right] - \mathbb{E} \left[ q^2 \Psi_k^2 \right] \mathbb{E} \left[ q^2 \Psi_z^2 \right] \right) \\
&= 2N(N-1) C \left[ P^2_k, P^2_z \right] \\
&= 2N(N-1) \left( \mathbb{E} \left[ q^2 \Psi_k^2 \Psi_z^2 \right] - \mathbb{E} \left[ q^2 \Psi_k^2 \right] \mathbb{E} \left[ q^2 \Psi_z^2 \right] \right),
\end{align*}
\]  

(A.2)

where \( P_z \) and \( P'_z \) indicate i.i.d. realizations of \( P_z \).

For symmetry, the third term is simply

\[
\begin{align*}
C \left[ \sum_{i=1}^{N} P^2_z, \sum_{i=1}^{N} P_z \right] &= 2N(N-1) \left( \mathbb{E} \left[ q^3 \Psi_k^2 \Psi_z \right] - \mathbb{E} \left[ q^3 \Psi_z \right] \mathbb{E} \left[ q^3 \Psi_k \right] \right),
\end{align*}
\]  

(A.3)
The last contribution is obtained as

\[
C \left[ \sum_{i=1}^{N} p_{ki}^{(i)} \sum_{j \neq i}^{N} p_{kj}^{(j)} \sum_{j=1}^{N} p_{k}^{(j)} \sum_{j=1}^{N} p_{k}^{(j)} \right]
\]

\[
= 2C \left[ \sum_{i=1}^{N} \sum_{j=1}^{N} p_{ki}^{(i)} p_{kj}^{(j)} p_{k}^{(i)} p_{k}^{(j)} \left( p_{z}^{(i)} + \sum_{q=1}^{N} p_{z}^{(q)} \right) + \sum_{q=1}^{N} p_{z}^{(q)} \sum_{j=1}^{N} p_{z}^{(j)} \right]
\]

\[
= 2C \left[ \sum_{i=1}^{N} \sum_{j=1}^{N} p_{ki}^{(i)} p_{kj}^{(j)} p_{k}^{(i)} p_{k}^{(j)} \left( p_{z}^{(i)} + \sum_{q=1}^{N} p_{z}^{(q)} \right) + \sum_{q=1}^{N} p_{z}^{(q)} \sum_{j=1}^{N} p_{z}^{(j)} \right]
\]

\[
= 2C \left[ \sum_{i=1}^{N} \sum_{j=1}^{N} p_{ki}^{(i)} p_{kj}^{(j)} p_{k}^{(i)} p_{k}^{(j)} \left( p_{z}^{(i)} + \sum_{q=1}^{N} p_{z}^{(q)} \right) + \sum_{q=1}^{N} p_{z}^{(q)} \sum_{j=1}^{N} p_{z}^{(j)} \right]
\]

\[
= 2C \left[ \sum_{i=1}^{N} \sum_{j=1}^{N} p_{ki}^{(i)} p_{kj}^{(j)} p_{k}^{(i)} p_{k}^{(j)} \left( p_{z}^{(i)} + \sum_{q=1}^{N} p_{z}^{(q)} \right) + \sum_{q=1}^{N} p_{z}^{(q)} \sum_{j=1}^{N} p_{z}^{(j)} \right]
\]

\[
= 2C \left[ \sum_{i=1}^{N} \sum_{j=1}^{N} p_{ki}^{(i)} p_{kj}^{(j)} p_{k}^{(i)} p_{k}^{(j)} \left( p_{z}^{(i)} + \sum_{q=1}^{N} p_{z}^{(q)} \right) + \sum_{q=1}^{N} p_{z}^{(q)} \sum_{j=1}^{N} p_{z}^{(j)} \right]
\]

\[
= 4 \frac{N(N-1)}{2} \left[ p_{k}^{(i)} + \sum_{q=1}^{N} p_{z}^{(q)} \right] + 4 \frac{N(N-1)(N-2)}{2} \sum_{i,j} \left[ p_{z}^{(i)} p_{z}^{(j)} \right]
\]

\[
= 2N(N-1) \left[ E \left[ q^{2} \Psi_{k} \Psi_{z} \right]^{2} - E \left[ q \Psi_{k} \right]^{2} E \left[ q \Psi_{z} \right]^{2} \right]
\]

\[
+ 4N(N-1)(N-2) \left[ E \left[ q^{2} \Psi_{k} \Psi_{z} \right] E \left[ q \Psi_{k} \right] E \left[ q \Psi_{z} \right] - E \left[ q \Psi_{k} \right]^{2} E \left[ q \Psi_{z} \right]^{2} \right]
\]

\[
= 2N(N-1) E \left[ q^{2} \Psi_{k} \Psi_{z} \right]^{2} + 4N(N-1)(N-2) E \left[ q^{2} \Psi_{k} \Psi_{z} \right] E \left[ q \Psi_{k} \right] E \left[ q \Psi_{z} \right]
\]

\[-2N(N-1)(1+2(N-2)) E \left[ q \Psi_{k} \right]^{2} E \left[ q \Psi_{z} \right]^{2}
\]

(A.5)

\[\square\]

### A.2 Multilevel MC covariance

\[\mathbb{C} \left[ \left( \hat{\beta}_{k} \right)^{2}, \left( \hat{\beta}_{z} \right)^{2} \right]\]

In this section, the derivation of the multilevel covariance term:
\[
\sum_{r=\ell+1}^{L} \sum_{q=0}^{L} \frac{4}{N_r} \left( \mathbb{E}[P_{\ell,k}P_{\ell,z}] \mathbb{E}[P_{r,k}P_{r,z}] - \mathbb{E}[P_{\ell,k}] \mathbb{E}[P_{r,k}] \mathbb{E}[P_{r,z}] \right) \\
+ \frac{4}{N_r} \sum_{q=0}^{L} \left( \mathbb{E}[P_{\ell,k}P_{r,z}] \mathbb{E}[P_{r,k}] \mathbb{E}[P_{q,z}] - \mathbb{E}[P_{\ell,k}] \mathbb{E}[P_{r,k}] \mathbb{E}[P_{q,z}] \right)
\]

is presented.
Proof. The multilevel MC covariance term can be written as

\[
C \left[ \left( \hat{\beta}_k \right)^2, \left( \hat{\beta}_z \right)^2 \right] = C \left\{ \left( \frac{1}{b_k} \sum_{\ell=0}^{L} \hat{P}_{\ell,k} \right)^2, \left( \frac{1}{b_z} \sum_{\ell=0}^{L} \hat{P}_{\ell,z} \right)^2 \right\} = C \left\{ \left( \sum_{\ell=0}^{L} \hat{P}_{\ell,k} \right)^2, \left( \sum_{\ell=0}^{L} \hat{P}_{\ell,z} \right)^2 \right\} = C \left\{ \sum_{\ell=0}^{L} \left( \hat{P}_{\ell,k} \right)^2 + \sum_{\ell=0}^{L} \sum_{r \neq \ell} \hat{P}_{\ell,k} \hat{P}_{r,k} \right\}, \quad (A.7)
\]

where each single level estimator \( \hat{P}_{\ell,k} \) is defined as

\[
\hat{P}_{\ell,k} = \frac{1}{N_\ell} \sum_{i=1}^{N_\ell} \left( q_{\ell}^{(i)} - q_{\ell-1}^{(i)} \right) \Psi_k^{(i)} = \frac{1}{N_\ell} \sum_{i=1}^{N_\ell} \frac{Y_{\ell}^{(i)} \Psi_k^{(i)}}{P_{\ell,k}^{(i)}}.
\]

There are 4 terms that need to be computed, however, due to symmetry, only 3 of them need to be derived explicitly.
The first term can be written as

\[
C \left[ \sum_{\ell=0}^{L} \left( \hat{P}_{\ell,k} \right)^2, \sum_{\ell=0}^{L} \left( \hat{P}_{\ell,z} \right)^2 \right] = \sum_{\ell=0}^{L} \frac{C \left[ \left( \hat{P}_{\ell,k} \right)^2, \left( \hat{P}_{\ell,z} \right)^2 \right]}{N_{\ell}^3} \\
= \sum_{\ell=0}^{L} \left[ \frac{\left( \hat{P}_{\ell,k} \right)^2 - \frac{N_{\ell}^3}{N_{\ell}^3} \left( \frac{N_{\ell}^3}{N_{\ell}^3} \right) \left( \frac{N_{\ell}^3}{N_{\ell}^3} \right)}{N_{\ell}^3} \right] \\
= \sum_{\ell=0}^{L} \left[ \frac{\left( \hat{P}_{\ell,k} \right)^2 - \frac{N_{\ell}^3}{N_{\ell}^3} \left( \frac{N_{\ell}^3}{N_{\ell}^3} \right) \left( \frac{N_{\ell}^3}{N_{\ell}^3} \right)}{N_{\ell}^3} \right] \\
= \sum_{\ell=0}^{L} \left[ \frac{\left( \hat{P}_{\ell,k} \right)^2 - \frac{N_{\ell}^3}{N_{\ell}^3} \left( \frac{N_{\ell}^3}{N_{\ell}^3} \right) \left( \frac{N_{\ell}^3}{N_{\ell}^3} \right)}{N_{\ell}^3} \right]
\]

For this term the derivation of each single-level estimator is identical to the single-level covariance term derived in the previous section.

The second (and third term, due to symmetry) term can be written as

\[
\begin{align*}
C \left[ \sum_{\ell=0}^{L} \left( \hat{P}_{\ell,k} \right)^2, \sum_{\ell=0}^{L} \sum_{r=0}^{L} \hat{P}_{\ell,z} \hat{P}_{r,z} \right] = C \left[ \sum_{\ell=0}^{L} \left( \hat{P}_{\ell,k} \right)^2, 2 \hat{P}_{\ell,z} \sum_{r=0}^{L} \hat{P}_{r,z} \right] + C \left[ \sum_{\ell=0}^{L} \hat{P}_{\ell,z} \sum_{r=0}^{L} \hat{P}_{\ell,k} \sum_{r=0}^{L} \hat{P}_{r,z} \right] \\
= 2 \sum_{\ell=0}^{L} C \left[ \left( \hat{P}_{\ell,k} \right)^2, \hat{P}_{\ell,z} \sum_{r=0}^{L} \hat{P}_{r,z} \right]
\end{align*}
\]

The term \(C \left[ \left( \hat{P}_{\ell,k} \right)^2, \hat{P}_{\ell,z} \sum_{r=0}^{L} \hat{P}_{r,z} \right]\) can be evaluated in term of moments of the QoI, \(q_\ell\),
by further manipulating the terms:

\[
C \left[ \left( \hat{P}_{\ell,k} \right)^2, \hat{P}_{\ell,z} \sum_{r=0 \atop r \neq \ell}^L \hat{P}_{r,z} \right] = \sum_{r=0 \atop r \neq \ell}^L C \left( \hat{P}_{\ell,k} \right)^2, \hat{P}_{\ell,z} \hat{P}_{r,z} \right]
\]

\[
= \sum_{r=0 \atop r \neq \ell}^L C \left[ \left( \frac{1}{N_\ell} \sum_{i=1}^{N_\ell} P_{\ell,k}^{(i)} \right)^2, \left( \frac{1}{N_\ell} \sum_{i=1}^{N_\ell} P_{\ell,z}^{(i)} \right) \left( \frac{1}{N_r} \sum_{j=1}^{N_r} P_{r,z}^{(j)} \right) \right]
\]

\[
= \frac{1}{N_\ell^3} \sum_{r=0 \atop r \neq \ell}^L \frac{1}{N_r} \left[ \sum_{i=1}^{N_\ell} \left( P_{\ell,k}^{(i)} \right)^2, \sum_{i=1}^{N_\ell} \sum_{j=1 \atop j \neq i}^{N_\ell} P_{\ell,k}^{(i)} P_{\ell,k}^{(j)}, N_r \hat{P}_{r,z} \sum_{i=1}^{N_r} P_{\ell,z}^{(i)} \right] + C \left[ N_r \hat{P}_{r,z} \sum_{i=1}^{N_r} P_{\ell,z}^{(i)}, \sum_{i=1}^{N_\ell} \sum_{j=1 \atop j \neq i}^{N_\ell} P_{\ell,k}^{(i)} P_{\ell,k}^{(j)} \right]
\]

\[
= \frac{1}{N_\ell^3} \sum_{r=0 \atop r \neq \ell}^L \frac{1}{N_r} \left[ N_\ell C \left[ P_{\ell,k}^2, N_r \hat{P}_{r,z} P_{\ell,z} \right] + C \left[ N_r \hat{P}_{r,z} \sum_{i=1}^{N_r} P_{\ell,z}^{(i)}, \sum_{i=1}^{N_\ell} \sum_{j=1 \atop j \neq i}^{N_\ell} P_{\ell,k}^{(i)} P_{\ell,k}^{(j)} \right] \right].
\]

The first term of the previous expression, Eq. A.11, can be written as

\[
C \left[ P_{\ell,k}^2, N_r \hat{P}_{r,z} P_{\ell,z} \right] = C \left[ P_{\ell,k}^2, N_r \frac{1}{N_r} \sum_{j=1}^{N_r} P_{r,z}^{(j)} P_{\ell,z} \right] = N_r C \left[ P_{\ell,k}^2, P_{\ell,z} P_{r,z} \right] \]

\[
= N_r \left( E \left[ P_{\ell,k}^2 P_{\ell,z} \right] E \left[ P_{r,z} \right] - E \left[ P_{\ell,k}^2 \right] E \left[ P_{\ell,z} \right] E \left[ P_{r,z} \right] \right)
\]

(A.12)
The second term of Eq. A.11 can be manipulated as it follows:

\[
\mathbb{C} \left[ N_r \hat{P}_{r,z} \sum_{i=1}^{N_t} P^{(i)}_{\ell,z} \sum_{j=1}^{N_t} \sum_{j \neq i} P^{(i)}_{\ell,k} P^{(j)}_{\ell,k} \right] 
\]

\[
= \mathbb{C} \left[ N_r \hat{P}_{r,z} \sum_{i=1}^{N_t} P^{(i)}_{\ell,z} \sum_{j=1}^{N_t} P^{(j)}_{\ell,k} + \sum_{q=1}^{N_t} P^{(q)}_{\ell,k} \left( P^{(i)}_{\ell,k} + \sum_{j=1}^{N_t} P^{(j)}_{\ell,k} \right) \right] 
\]

\[
= 2N_\ell (N_\ell - 1) \mathbb{C} \left[ N_r \hat{P}_{r,z} P_{\ell,z} P_{\ell,k} P_{\ell,k}' \right] 
\]

\[
= 2N_\ell (N_\ell - 1) \left( N_r \mathbb{E} \left[ \hat{P}_{r,z} P_{\ell,z} P_{\ell,k} P_{\ell,k}' \right] - N_r \mathbb{E} \left[ \hat{P}_{r,z} P_{\ell,z} \right] \mathbb{E} \left[ P_{\ell,k} \right]^2 \right) 
\]

\[
= 2N_\ell (N_\ell - 1) N_r \left( \mathbb{E} \left[ P_{\ell,z} \right] \mathbb{E} \left[ P_{\ell,z} P_{\ell,k} \right] \mathbb{E} \left[ P_{\ell,k} \right] - \mathbb{E} \left[ P_{\ell,z} \right] \mathbb{E} \left[ P_{\ell,z} \right] \mathbb{E} \left[ P_{\ell,k} \right]^2 \right) 
\]

Finally, the last term of Eq. A.7 is written as

\[
\mathbb{C} \left\{ \sum_{\ell=0}^{L} \sum_{r=0}^{L} \hat{P}_{\ell,k} \hat{P}_{r,k}, \sum_{\ell=0}^{L} \sum_{r \neq \ell} \hat{P}_{\ell,z} \hat{P}_{r,z} \right\} 
\]

\[
= 2\mathbb{C} \left[ \sum_{\ell=0}^{L} \sum_{r=\ell+1}^{L} \hat{P}_{\ell,k} \hat{P}_{r,k} + 2 \hat{P}_{\ell,z} \sum_{q=0}^{L} \hat{P}_{q,z} + 2 \hat{P}_{r,z} \sum_{q=0}^{L} \hat{P}_{q,z} + \sum_{q=0}^{L} \sum_{t=0}^{L} \hat{P}_{q,z} \hat{P}_{t,z} \right] 
\]

\[
= 4\mathbb{C} \left[ \sum_{\ell=0}^{L} \sum_{r=\ell+1}^{L} \hat{P}_{\ell,k} \hat{P}_{r,k}, \sum_{\ell=0}^{L} \sum_{r \neq \ell} \hat{P}_{\ell,z} \hat{P}_{r,z} \right] + 4\mathbb{C} \left[ \sum_{\ell=0}^{L} \sum_{r=\ell+1}^{L} \hat{P}_{\ell,k} \hat{P}_{r,k}, \sum_{\ell=0}^{L} \sum_{r \neq \ell} \hat{P}_{\ell,z} \hat{P}_{r,z} \right] 
\]

\[
+ 4\mathbb{C} \left[ \sum_{\ell=0}^{L} \sum_{r=\ell+1}^{L} \hat{P}_{\ell,k} \hat{P}_{r,k}, \sum_{q=0}^{L} \hat{P}_{q,z} \right], 
\]

(A.14)

by using the single-level derivation.
The first term of the previous expression can be simplified as

\[ C \left[ \sum_{\ell=0}^{L} \sum_{r=\ell+1}^{L} \hat{P}_{\ell,k} \hat{P}_{r,k} \hat{P}_{\ell,z} \hat{P}_{r,z} \right] \]

\[ = \sum_{\ell=0}^{L} \sum_{r=\ell+1}^{L} C \left[ \hat{P}_{\ell,k} \hat{P}_{r,k} \hat{P}_{\ell,z} \hat{P}_{r,z} \right] \]

\[ = \sum_{\ell=0}^{L} \sum_{r=\ell+1}^{L} C \left[ \left( \frac{1}{N_t} \sum_{i=1}^{N_t} P_{\ell,k}^{(i)} \right) \left( \frac{1}{N_r} \sum_{j=1}^{N_r} P_{r,k}^{(j)} \right) \right. \]

\[ \left. \left( \frac{1}{N_t} \sum_{i=1}^{N_t} P_{\ell,z}^{(i)} \right) \left( \frac{1}{N_r} \sum_{j=1}^{N_r} P_{r,z}^{(j)} \right) \right] \]

\[ = \sum_{\ell=0}^{L} \sum_{r=\ell+1}^{L} \frac{1}{N_t^2 N_r^2} C \left[ \sum_{i=1}^{N_t} \sum_{j=1}^{N_r} P_{\ell,k}^{(i)} P_{r,k}^{(j)} P_{\ell,z}^{(i)} P_{r,z}^{(j)} + P_{\ell,z}^{(i)} \sum_{q=1}^{N_r} P_{r,z}^{(q)} + P_{r,z}^{(j)} \sum_{q=1}^{N_t} P_{\ell,z}^{(q)} \right] \]

\[ = \sum_{\ell=0}^{L} \sum_{r=\ell+1}^{L} \frac{1}{N_t^2 N_r^2} \left( N_t N_r C \left[ P_{\ell,k} P_{r,k} P_{\ell,z} P_{r,z} \right] + (N_r - 1) N_t N_r C \left[ P_{\ell,k} P_{r,k} P_{\ell,z} P'_{r,z} \right] \right) \]

\[ + (N_t - 1) N_t N_r C \left[ P_{\ell,k} P_{r,k} P_{\ell,z} P'_{\ell,z} \right] \]

\[ = \sum_{\ell=0}^{L} \sum_{r=\ell+1}^{L} \frac{1}{N_t N_r} \left( \mathbb{E} \left[ P_{\ell,k} P_{\ell,z} \right] \mathbb{E} \left[ P_{r,k} P_{r,z} \right] - \mathbb{E} \left[ P_{\ell,k} \right] \mathbb{E} \left[ P_{\ell,z} \right] \mathbb{E} \left[ P_{r,k} \right] \mathbb{E} \left[ P_{r,z} \right] \right) \]

\[ + (N_r - 1) \mathbb{E} \left[ P_{\ell,k} P_{\ell,z} \right] \mathbb{E} \left[ P_{r,k} \right] \mathbb{E} \left[ P_{r,z} \right] - (N_t - 1) \mathbb{E} \left[ P_{\ell,k} \right] \mathbb{E} \left[ P_{\ell,z} \right] \mathbb{E} \left[ P_{r,k} \right] \mathbb{E} \left[ P_{r,z} \right] \]

\[ + (N_t - 1) \mathbb{E} \left[ P_{\ell,k} \right] \mathbb{E} \left[ P_{\ell,z} \right] \mathbb{E} \left[ P_{r,k} P_{r,z} \right] - (N_t - 1) \mathbb{E} \left[ P_{\ell,k} \right] \mathbb{E} \left[ P_{r,k} \right] \mathbb{E} \left[ P_{r,z} \right] \mathbb{E} \left[ P_{\ell,z} \right] \]

(A.15)

The last two terms of Eq. A.14 are similar and can be obtained as demonstrated below for the first term.
the first of them

\[ C \left[ \sum_{\ell=0}^{L} \sum_{r=\ell+1}^{L} \hat{P}_{\ell,k} \hat{P}_{r,k} \hat{P}_{\ell,z} \sum_{q=0}^{L} \hat{P}_{q,z} \right] \]

\[ = \sum_{\ell=0}^{L} \sum_{r=\ell+1}^{L} C \left[ \hat{P}_{\ell,k} \hat{P}_{r,k} \sum_{q=0}^{L} \hat{P}_{q,z} \right] \]

\[ = \sum_{\ell=0}^{L} \sum_{r=\ell+1}^{L} \frac{1}{N_\ell N_r} \left[ \sum_{i=1}^{N_\ell} P_{\ell,k}^{(i)} \left( \sum_{j=1}^{N_r} P_{r,k}^{(j)} \right) \right] \cdot \sum_{q=0}^{L} P_{q,z} \sum_{q=0}^{L} P_{q,z} \left( \sum_{s=1}^{N_q} P_{q,z}^{(s)} \right) \]

\[ = \sum_{\ell=0}^{L} \sum_{r=\ell+1}^{L} \frac{1}{N_\ell N_r} \left[ \frac{N_\ell}{N_r} \sum_{i=1}^{N_\ell} P_{\ell,k}^{(i)} \sum_{j=1}^{N_r} P_{r,k}^{(j)} + \sum_{q=0}^{L} P_{\ell,k}^{(i)} \sum_{q=0}^{L} P_{q,z}^{(i)} \sum_{s=1}^{N_q} P_{q,z}^{(s)} \right] \]

\[ = \sum_{\ell=0}^{L} \sum_{r=\ell+1}^{L} \frac{1}{N_\ell N_r} \left[ \frac{N_\ell}{N_r} \sum_{i=1}^{N_\ell} P_{\ell,k}^{(i)} \sum_{j=1}^{N_r} P_{r,k}^{(j)} + \sum_{q=0}^{L} P_{q,z} \sum_{s=1}^{N_q} P_{q,z}^{(s)} \right] \]

\[ = \sum_{\ell=0}^{L} \sum_{r=\ell+1}^{L} \frac{1}{N_\ell N_r} \left[ \frac{N_\ell}{N_r} \sum_{i=1}^{N_\ell} P_{\ell,k}^{(i)} \sum_{j=1}^{N_r} P_{r,k}^{(j)} + \sum_{q=0}^{L} P_{q,z} \sum_{s=1}^{N_q} P_{q,z}^{(s)} \right] \]

\[ = \sum_{\ell=0}^{L} \sum_{r=\ell+1}^{L} \frac{1}{N_\ell N_r} \left[ \sum_{q=0}^{L} \sum_{q=0}^{L} \left( \mathbb{E} [P_{\ell,k} P_{\ell,z}] \mathbb{E} [P_{r,k}] \mathbb{E} [P_{q,z}] - \mathbb{E} [P_{\ell,k}] \mathbb{E} [P_{\ell,z}] \mathbb{E} [P_{r,k}] \mathbb{E} [P_{q,z}] \right) \right] \]