Accelerated Bayesian Experimental Design for Chemical Kinetic Models

by

Xun Huan

B.A.Sc., Engineering Science, Aerospace Option (2008)
University of Toronto

Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of

Master of Science in Aerospace Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2010

Author ....
Department of Aeronautics and Astronautics
May 21, 2010

Certified by ....
Youssef M. Marzouk
Assistant Professor of Aeronautics and Astronautics
Thesis Supervisor

Accepted by ....
Eytan H. Modiano
Associate Professor of Aeronautics and Astronautics
Chair, Committee on Graduate Students

Accelerated Bayesian Experimental Design for Chemical Kinetic Models

by

Xun Huan

Submitted to the Department of Aeronautics and Astronautics
on May 21, 2010, in partial fulfillment of the
requirements for the degree of
Master of Science in Aerospace Engineering

Abstract

The optimal selection of experimental conditions is essential in maximizing the value of data for inference and prediction, particularly in situations where experiments are time-consuming and expensive to conduct.

A general Bayesian framework for optimal experimental design with nonlinear simulation-based models is proposed. The formulation accounts for uncertainty in model parameters, observables, and experimental conditions. Straightforward Monte Carlo evaluation of the objective function — which reflects expected information gain (Kullback-Leibler divergence) from prior to posterior — is intractable when the likelihood is computationally intensive. Instead, polynomial chaos expansions are introduced to capture the dependence of observables on model parameters and on design conditions. Under suitable regularity conditions, these expansions converge exponentially fast. Since both the parameter space and the design space can be high-dimensional, dimension-adaptive sparse quadrature is used to construct the polynomial expansions. Stochastic optimization methods will be used in the future to maximize the expected utility.

While this approach is broadly applicable, it is demonstrated on a chemical kinetic system with strong nonlinearities. In particular, the Arrhenius rate parameters in a combustion reaction mechanism are estimated from observations of autoignition. Results show multiple order-of-magnitude speedups in both experimental design and parameter inference.

Thesis Supervisor: Youssef M. Marzouk
Title: Assistant Professor of Aeronautics and Astronautics

Acknowledgments

When I first arrived at MIT as a graduate student, I had the littlest idea of what uncertainty quantification (UQ) is. I was very lucky to have been introduced to Professor Youssef Marzouk, and soon, he convinced me. He convinced me that not only is UQ of utter importance in engineering, its analysis is also filled with wondrous unknowns and endless possibilities. Professor Marzouk became my advisor, and more importantly, he also became my friend. I remember the 7 PM meetings, "emergency" talks before quals and thesis due date, and the full day "deep-dive" discussion (with lunch of course) on a Saturday — it is these constant guidance, support, and encouragement from him throughout the two years of my Masters study that made this thesis possible.

I would like to thank the entire Aerospace Computational Design Laboratory, all the students, post-docs, faculty, and staff, for their insightful discussions from class work to research, the technical support on computational resources, and the friendships that kept me from being over-stressed. In addition, I would like to thank Professor David Darmofal for co-advising me when Professor Marzouk was away, Huafei Sun for helping revise this thesis, Masa Yano for being an awesome roommate, Laslo Diosady for the enjoyable dumpling nights, JM Modisette and Julie Andren for suspending their work to allow me to use the cluster before the thesis is due, and Hement Chaurasia and Pritesh Mody for together enduring the hardships of both the quals and thesis. But these are only a mere few of many people to be thankful for, and only a tiny count of an enormous number of things to be grateful about.

Finally, I would like to thank my parents, whose constant love and support made me who I am today. I am proud to be their son, and I wish them health and happiness.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Alternative fuels, such as biofuels [75] and synthetic fuels [39], are becoming increasingly popular in the energy market over the past years. For example, world ethanol production for transport fuel tripled between 2000 and 2007, while biodiesel expanded eleven-fold [10]. These fuels are excellent sources for safeguarding the volatile petroleum price and to ensure energy security, but more importantly, they carry the flexibility in promoting new and desirable properties that traditional fossil fuels might not offer.

Current knowledge about alternative fuels is relatively new. Their thermochemical properties and combustion kinetics remain poorly characterized, and fundamental research in their properties is still ongoing (e.g., [21, 74]). The environments designed to produce, process, and utilize the fuels may be far from optimal. These suboptimal operating conditions lead to low efficiency and adverse emissions such as high levels of nitrogen oxides (NO_x) and particulate matter (PM), both of which are not only sources of environmental and health hazard [2, 3], but are also becoming the center of political debates.

With the rapid growth of computational capability worldwide, numerical modeling has become an indispensable tool for capturing, explaining, and predicting physical phenomena. Reliable predictions from models can lead to better designs and policies for utilizing the alternative fuels, which in turn can improve their efficiency and emission rates.

One way to improve the models is to perform parameter inference. All numerical models have parameters (or constants), such as the gravitational constant in Newton's universal law of gravitation. For some models, these parameters have been experimentally determined or theoretically derived with great accuracy and precision; but these cases are the exception rather than the rule, since there are substantially more models where their parameters still carry significant uncertainty. For example, in combustion, many thermodynamic and kinetic parameters still have large uncertainties today [8, 9]. In an extreme example, the rate constant of a methyl elementary reaction has an uncertainty factor of 500% [82]. The accuracy of these parameters directly affect how good the models are able to represent the physical reality. Therefore, it is important to continue to refine these parameters, and reduce their uncertainty.

In order to perform inference, experimental data need to be used, but not all data are created equal — while some data can be very helpful in reducing the parameters' uncertainties, other data may not be useful at all. Since experiments, especially in the combustion field, are expensive, time-consuming, and delicate to perform, it is thus crucial to design experiments that yield data of the highest quality. This process is called *optimal experimental design*, and shall be the centerpiece of this thesis. A Bayesian approach to the optimal experimental design shall be undertaken, which is able to account for uncertainties in model parameters, observables, and experimental conditions.

Further background information and literature review are presented in each of the chapters accordingly, as the thesis progresses.

1.2 Objective and Outline

The primary objective of this thesis is to formulate a mathematical framework of computing the optimal experimental designs for nonlinear models under a Bayesian setting. In particular, this framework is demonstrated on a challenging chemical combustion system to illustrate its effectiveness and practicality. However, this nonlinear optimal experimental design framework is very general, and can be applied to many other models in a wide range of applications.

This thesis is outlined as follows. The physical problem of interest, the combustion problem, is introduced in Chapter 2. Its understanding is essential in establishing the goals of the physical experiment. Chapter 3 then provides the tools necessary to achieve the experimental goals, and these tools also create the foundation on how to quantify the value of an experiment. With the aid of concepts from information theory, the goodness indicator is developed in Chapter 4. This subsequently enables the optimization of experimental designs, and the experimental results from the optimal design would allow the experimenter to best achieve his or her experimental goals. However, this nonlinear optimal experimental design framework is shown to be too expensive to be practical, and its acceleration through model reduction is necessary. Chapter 5 presents one such reduction method, by forming polynomial chaos expansions for random variables. Combining all the tools, the experimental design problem for the combustion system is solved in an accelerated manner, and the results are presented in Chapter 6. Finally, the thesis ends with a summary, and some conclusions and future work in Chapter 7.

Chapter 2

Combustion Problem

Each experiment is performed with some purposes or goals. These goals are important in

1. determining which experimental states should be observed; and
2. quantifying the goodness of an experiment, which subsequently allows the optimization of experimental designs.

In order to perform these tasks, it is essential to have a good understanding of the physical problem relevant to the experiment. The physics, along with the first task, are described in this chapter; the second task is discussed in Chapter 4.

2.1 Background

The hydrogen-oxygen (H_2-O_2) combustion is chosen as the physical problem of interest. Better understanding of this simple yet representative combustion paves the path to better understanding of the combustion of larger molecules. In fact, H_2-O_2 has already been studied extensively in the history of combustion [24, 98, 99, 106], and is currently one of the best-understood mechanisms, thus conveniently providing ample data for validation. The so-called "hydrogen economy" [68], or even simply blending hydrogen with traditional fuels (e.g., [53]), has also received considerable attention as the cleaner future replacement to the current "hydrocarbon economy", further motivating continued research in the fields related to H_2-O_2.

The combustion of a fuel is a very complicated process, involving the time evolution of numerous chemical species. A spectrum of mechanisms have been developed to describe the H_2-O_2 combustion. On one extreme, a one-step global mechanism resembles a "black-box" type of approach. It captures the development of the major reactants and products, but does not reflect what is really happening with the intermediate species (i.e., the detailed kinetics). Also, the global kinetic relation often relies on curve-fitting of experimental data, leading to non-integral reaction orders which can be non-intuitive. On the other extreme, a very detailed mechanism composed of elementary reactions is able to reveal comprehensive interactions among the intermediate species. The tradeoff, however, is its high computational cost and complexity. As a result, it is important to find the simplest mechanism that captures the reaction behaviour that is relevant to the experimental goals.

This study has taken an in-between approach, by choosing a mechanism of intermediate complexity — a 19-reaction mechanism proposed in [106], which is reproduced in Table 2.1 (a more extensive table containing the recommended values of kinetic parameters can be found in Appendix A). A detailed discussion of the roles of each reaction can also be found in [106].

For demonstration purposes, constant pressure, adiabatic, and no-transport (i.e., homogeneous) conditions are considered. To some extent, these conditions can imitate the conditions of shock tube experiments for a short period of time right after the passing of the reflection wave (in some cases, constant-volume and isothermal conditions may be more appropriate). This can be of significance because shock tube experiments are one of the standard experimental methods for analyzing chemical kinetic properties [18, 19] (some other alternatives include static-, stirred-, and flow-reactors, as well as rapid compression machines and even premixed flames [44]). In an engineering example, jet engines burn fuel in the combustor at near constant pressure condition [46]. The methodology to be developed in this thesis can be easily applied to various different, more complex reaction conditions, and has great potential in a wide range of applications.

Reaction No.	Elementary Reaction		
R1	$H + O_2$	$\Longleftrightarrow$	$O + OH$
R2	$O + H_2$	$\Longleftrightarrow$	$H + OH$
R3	$H_2 + OH$	$\Longleftrightarrow$	$H_2O + H$
R4	$OH + OH$	$\Longleftrightarrow$	$O + H_2O$
R5	$H_2 + M$	$\Longleftrightarrow$	$H + H + M$
R6	$O + O + M$	$\Longleftrightarrow$	$O_2 + M$
R7	$O + H + M$	$\Longleftrightarrow$	$OH + M$
R8	$H + OH + M$	$\Longleftrightarrow$	$H_2O + M$
R9	$H + O_2 + M$	$\Longleftrightarrow$	$HO_2 + M$
R10	$HO_2 + H$	$\Longleftrightarrow$	$H_2 + O_2$
R11	$HO_2 + H$	$\Longleftrightarrow$	$OH + OH$
R12	$HO_2 + O$	$\Longleftrightarrow$	$O_2 + OH$
R13	$HO_2 + OH$	$\Longleftrightarrow$	$H_2O + O_2$
R14	$HO_2 + HO_2$	$\Longleftrightarrow$	$H_2O_2 + O_2$
R15	$H_2O_2 + M$	$\Longleftrightarrow$	$OH + OH + M$
R16	$H_2O_2 + H$	$\Longleftrightarrow$	$H_2O + OH$
R17	$H_2O_2 + H$	$\Longleftrightarrow$	$HO_2 + H_2$
R18	$H_2O_2 + O$	$\Longleftrightarrow$	$OH + HO_2$
R19	$H_2O_2 + OH$	$\Longleftrightarrow$	$HO_2 + H_2O$

Table 2.1: 19-reaction hydrogen-oxygen mechanism. Reactions involving M are three-body interactions, where M is a wild-card with different efficiencies for different species.

2.2 Governing Equations

The state of the chemical system can be completely described by the species mass fractions Y_j [dimensionless], $j = 1, \cdots, n_s$ (where n_s is the total number of species), and the system temperature T [K]; they shall be referred to as the state variables. Given the assumed conditions described earlier (constant pressure, adiabatic, no transport), the system is governed by the following set of ordinary-differential equations (ODEs):

$$\begin{aligned} \frac{dY_j}{dt} &= \frac{\dot{\omega}_j W_j}{\rho}, \forall j && (2.1)\\ \frac{dT}{dt} &= -\frac{1}{\rho c_p}\sum_{n=1}^{n_s} h_n \dot{\omega}_n W_n && (2.2) \end{aligned}$$

$$\text{Initial Conditions} = \left\{ \begin{array}{rcl} Y_j|_{t=0} &=& Y_{j,0}, \ \forall j \\ T|_{t=0} &=& T_0 \end{array} \right. , \qquad (2.3)$$

where $\dot{\omega}_j$ $[\mathrm{kmol}\cdot\mathrm{m}^{-3}\cdot\mathrm{s}^{-1}]$ is the molar production rate of the jth species, W_j $\left[\mathrm{kg}\cdot\mathrm{kmol}^{-1}\right]$ is the molecular weight of the jth species, ρ $[\mathrm{kg}\cdot\mathrm{m}^{-3}]$ is the mixture density, c_p $\left[\mathrm{J}\cdot\mathrm{K}^{-1}\cdot\mathrm{kg}^{-1}\right]$ is the mixture specific heat capacity under constant pressure, and h_n $\left[\mathrm{J}\cdot\mathrm{kg}^{-1}\right]$ is the specific enthalpy of the nth species. More specifically, the molar production rate is defined as

$$\dot{\omega}_j \equiv \frac{d[X_j]}{dt} = \sum_{m=1}^{n_r}\left(\nu''_{mj} - \nu'_{mj}\right)\left(k_{f,m}\prod_{n=1}^{n_s}[X_n]^{\nu'_{mn}} - k_{r,m}\prod_{n=1}^{n_s}[X_n]^{\nu''_{mn}}\right), \forall j, \qquad (2.4)$$

where $[X_j]$ $[\mathrm{kmol}\cdot\mathrm{m}^{-3}]$ is the molar concentration of the jth species, n_r is the total number of reactions, and ν'_{mn} and ν''_{mn} [dimensionless] are the stoichiometric coefficients on the reactant and product sides of the equation, respectively, for the nth species in the mth reaction.

The forward and reverse reaction rate constants of the mth reaction, denoted by $k_{f,m}$ $\left[\left(\mathrm{m}^3\cdot\mathrm{kmol}^{-1}\right)^{-1+\sum_{n=1}^{n_s}\nu'_{mn}}\cdot\mathrm{s}^{-1}\right]$ and $k_{r,m}$ $\left[\left(\mathrm{m}^3\cdot\mathrm{kmol}^{-1}\right)^{-1+\sum_{n=1}^{n_s}\nu''_{mn}}\cdot\mathrm{s}^{-1}\right]$ respectively, are assumed to have the modified Arrhenius form:

$$\begin{aligned} k_{f,m} &= A_m T^{b_m}\exp\left(\frac{-E_{a,m}}{R_u T}\right) && (2.5)\\ k_{r,m} &= \frac{k_{f,m}}{K_{c,m}} = \frac{k_{f,m}}{\exp\left(\frac{-\Delta G^o_{T,m}}{R_u T}\right)}, && (2.6) \end{aligned}$$

where A_m $\left[\left(\mathrm{m}^3 \cdot \mathrm{kmol}^{-1}\right)^{-1+\sum_{n=1}^{n_s} \nu'_{mn}} \cdot \mathrm{s}^{-1} \cdot \mathrm{K}^{-b_m}\right]$ is the pre-exponential factor, b_m [dimensionless] is the exponent of the temperature dependence, $E_{a,m}$ $\left[\mathrm{J} \cdot \mathrm{kmol}^{-1}\right]$ is the activation energy (A_m, b_m, and $E_{a,m}$ are collectively called the kinetic parameters of reaction m), $R_u = 8314.472$ $\left[\mathrm{J} \cdot \mathrm{kmol}^{-1} \cdot \mathrm{K}^{-1}\right]$ is the universal gas constant, $K_{c,m}$ $\left[\left(\mathrm{m}^3 \cdot \mathrm{kmol}^{-1}\right)^{\left(\sum_{n=1}^{n_s} \nu'_{mn} - \sum_{n=1}^{n_s} \nu''_{mn}\right)}\right]$ is the equilibrium constant, and $\Delta G^o_{T,m}$ $\left[\mathrm{J} \cdot \mathrm{kmol}^{-1}\right]$ is the change in Gibbs free energy at standard pressure (101,325 Pa) and temperature T.

The initial conditions of the ODEs are described by Equation 2.3, where $Y_{j,0}$ and T_0 are the initial species mass fractions and temperature, respectively. In most cases, the initial mass fractions of all species are zero except for H_2 and O_2. This leads to a compact, equivalent method to express $Y_{j,0}$, using the equivalence ratio ϕ [dimensionless], which is an indication of whether the fuel-oxidizer mixture is rich, lean, or stoichiometric:

$$\phi = \frac{(Y_{O_2}/Y_{H_2})_{\text{stoic}}}{(Y_{O_2}/Y_{H_2})} = \frac{(\chi_{O_2}/\chi_{H_2})_{\text{stoic}}}{(\chi_{O_2}/\chi_{H_2})}, \tag{2.7}$$

where the subscript "stoic" refers to the stoichiometric ratios, and χ_j [dimensionless] is the molar fraction of the jth species, related to the mass fraction through

$$\chi_j = \frac{Y_j}{W_j \sum_{n=1}^{n_s} Y_n/W_n}. \tag{2.8}$$

Often, the χ_j's are used in place of the Y_j's as the state variables, and this is adopted for the rest of the thesis.

Finally, perfect gas mixture is typically assumed, closing the system with the following equation of state:

$$\rho = \frac{p}{R_u T \sum_{n=1}^{n_s} Y_n/W_n}, \tag{2.9}$$

where p [Pa] is the (assumed constant) pressure. Note that different reaction conditions would lead to different variations of the governing equations.

Example 2.2.1 demonstrates how Equation 2.4 can be formed through a concrete case.

Example 2.2.1. Two-Reaction Mechanism

Consider a hypothetical two-reaction mechanism described in Table 2.2, which is simply constructed by reactions R1 and R3 from the 19-reaction H_2-O_2 mechanism described in Table 2.1. The reactions and species of this mechanism are arbitrarily ordered according to Table 2.3. Consequently, the stoichiometric coefficient matrices are

$$\nu'_{mn} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix} \quad \nu''_{mn} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}. \tag{2.10}$$

Equation 2.1 for $j = 6$ (i.e., species OH), for example, can be formed as

$$\begin{aligned} \dot{\omega}_6 &= \frac{d[X_6]}{dt} = \frac{d[OH]}{dt} \\ &= (1-0)\left(k_{f,1}[X_1][X_5] - k_{r,1}[X_4][X_6]\right) + \\ &\quad (0-1)\left(k_{f,2}[X_2][X_6] - k_{r,2}[X_3][X_5]\right) \\ &= k_{f,1}[O_2][H] - k_{r,1}[O][OH] - k_{f,2}[H_2][OH] + k_{r,2}[H_2O][H]. \end{aligned} \tag{2.11}$$

Reaction No.	Elementary Reaction		
R1	$H + O_2$	$\Longleftrightarrow$	$O + OH$
R3	$H_2 + OH$	$\Longleftrightarrow$	$H_2O + H$

Table 2.2: Hypothetical two-reaction mechanism for Example 2.2.1. The two reactions are R1 and R3 from the 19-reaction hydrogen-oxygen mechanism.

m	1	2
Reaction	R1	R3

n	1	2	3	4	5	6
Species	O_2	H_2	H_2O	O	H	OH

Table 2.3: Index ordering of the reactions and species of the hypothetical 2-reaction mechanism for Example 2.2.1.

□

2.3 Experimental Goals

Experiments can be designed and performed with different goals. For example, while one experiment may be used to infer certain parameters in the model, a different experiment can be more suited to predict the future behaviour of the system. The experimental goals will ultimately dictate what outputs should be observed or computed, and what the criteria are for a good experiment. The former is discussed in the next section (Section 2.4) while the latter is to be discussed later on (Chapter 4).

In this study, the experimental goals are to infer the values of (some or all of) the kinetic parameters (A_m, b_m, and $E_{a,m}$) of a subset of elementary reactions described in Table 2.1.

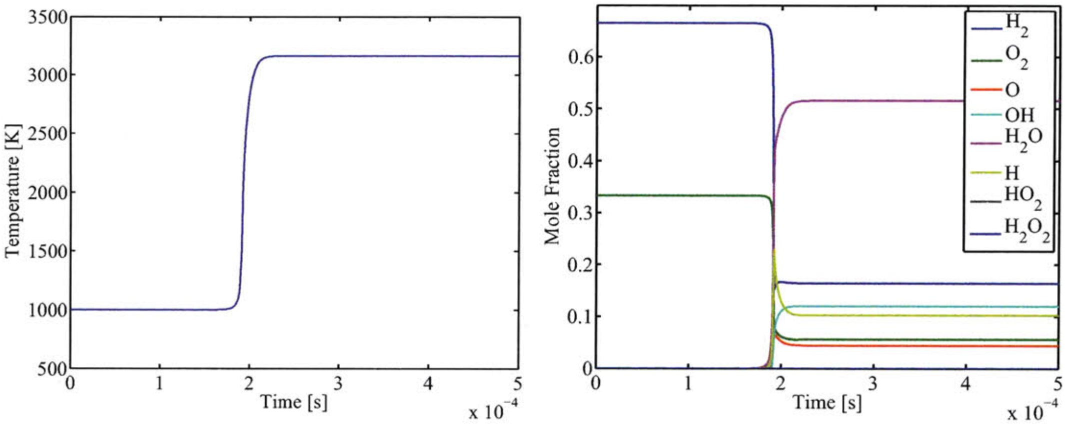
In the future, thermodynamic and transport parameters may also be explored.

2.4 Selection of Observables

Typical evolution profiles of the state variables are shown in Figure 2-1. In the following discussions, the initial conditions of the system shall be assumed to cause an ignition (explosion), unless otherwise specified.

The most complete and detailed set of observables of the system is simply the state variables as a function of time. To handle this numerically, one could, for example, discretize the time domain. However, too few discretization points would fail to capture the state behaviour, while too many discretization points would lead to an impractically high dimension of the observable vector. Also, as to be discussed at the end of Section 5.2, time-discretized states would be difficult to capture using polynomial chaos expansions. Therefore, one should transform the state variables to some new observables that in some sense compress the information, while retaining the information relevant to the experimental goals (analogous to sufficient statistics from information theory).

Given the experimental goals of inferring the kinetic parameter values, the observables must be able to reflect the variations in the kinetic parameters. For example,



(a) Temperature

(b) Species molar fractions

Figure 2-1: Typical evolution profiles of temperature and species molar fractions in a H_2-O_2 combustion.

temperature and species molar fractions at steady-state or equilibrium would not be able to reflect the kinetics of the system, although they would be good candidates for inferring thermodynamic parameter values. Another desirable property of the observables is that they are common and easy to measure in real experiments. For example, ignition delay is a very common and relatively easily measurable output in kinetics-related experiments, with ample data available in the literature. On the other hand, a characteristic time in which $[H_2O]$ reaches 75% of its equilibrium value would be more difficult to pinpoint, and almost never reported in the literature.

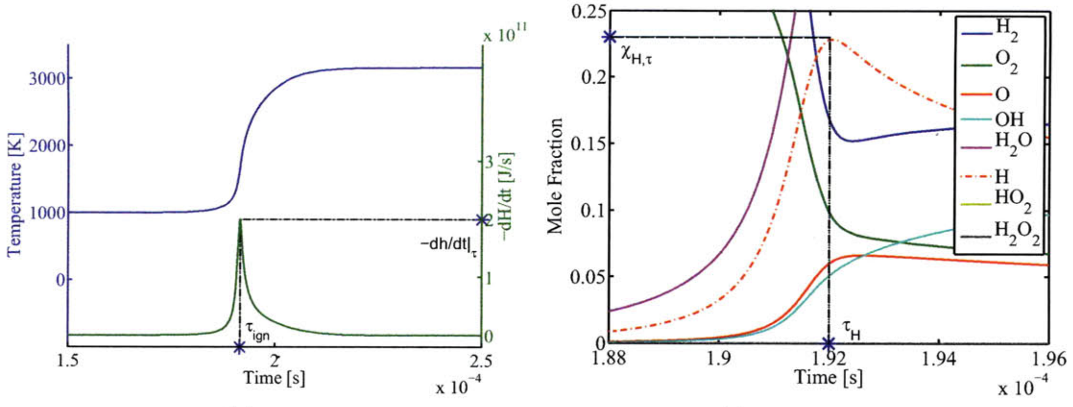
Taking the above factors into consideration, observables listed in Table 2.4 are selected for this study. The first 5 are characteristic times related to peak values, and the last 5 are the corresponding peak values. Note that $dh/dt < 0$ when enthalpy is released or lost by the system (i.e., exothermic). The species chosen for the observables are radical species (reactive due to unpaired electrons), which possess a peak in the mole fraction profiles (or a double peak in the case of H_2O_2). The peaks are caused by the following phenomenon. Prior to the ignition, radicals slowly accumulate in a pool by the chain-initiating reactions. After surpassing some threshold, they activate the chain-branching reactions, which further produce radicals very rapidly, leading to the ignition of the system. Soon after, reverse reactions balance the forward radical productions, while the chain-terminating reactions finally convert them to stable forms. Examples of τ_{ign}, τ_H, $\frac{dh}{dt}\big|_\tau$. and $\chi_{H,\tau}$ are shown in Figure 2-2. The time of the peak enthalpy release rate approximately matches the point when temperature rises most rapidly, as expected. As to be discussed later on in the thesis, the ln of the characteristic times shall be used in the actual implementation, that is, $\ln\tau$'s instead of τ's. This does not make any difference in the formulations except that the constructions of the polynomial chaos expansions in Chapter 5 would be for approximating the $\ln\tau$'s instead of the τ's.

Observable	Explanation
τ_{ign}	Ignition delay, defined as the time of peak enthalpy release rate.
τ_O	Characteristic time in which peak χ_O occurs.
τ_H	Characteristic time in which peak χ_H occurs.
τ_{HO_2}	Characteristic time in which peak χ_{HO_2} occurs.
$\tau_{H_2O_2}$	Characteristic time in which peak $\chi_{H_2O_2}$ occurs.
$\frac{dh}{dt}\big|_\tau$	Peak value of enthalpy release rate.
$\chi_{O,\tau}$	Peak value of χ_O.
$\chi_{H,\tau}$	Peak value of χ_H.
$\chi_{HO_2,\tau}$	Peak value of χ_{HO_2}.
$\chi_{H_2O_2,\tau}$	Peak value of $\chi_{H_2O_2}$.

Table 2.4: Selected observables for this study. Note that $dh/dt < 0$ when enthalpy is released or lost by the system.

2.5 Numerical Solution Tools

The governing equations are solved using Cantera version 1.7.0 [1, 33], which is an open-source chemical kinetics software. The Cantera input file used is presented in Appendix B. In particular, Cantera solves the system of ODEs with the help of CVODE [16], a suite of nonlinear differential algebraic equation solvers that solves stiff ODE systems implicitly, using the backward differentiation formulas. The validity and performance of the software are not assessed in this study, but extensive benchmark testings have been done by their developers [4]. One may view the software simply as



(a) τ_{ign}

(b) τ_H and $\chi_{H,\tau}$

Figure 2-2: Illustration of τ_{ign}, τ_H, and $\chi_{H,\tau}$.

third-party tools used in this study.

It has been occasionally encountered under some conditions that Cantera would fail. This may be caused by the fact that Cantera enforces the constant pressure condition via a high-gain controller for adjusting the volume, which can sometimes cause the system to "overreact", leading to negative volumes. Through some experimentation, most of these problems can be solved with"engineering solutions" such as relaxing the tolerances of the CVODE time integrator, and perturbing the final time to be integrated to.

Chapter 3

Bayesian Inference

The experimental goals have been defined in Section 2.3, which are to infer the kinetic parameter values of the mechanism reactions. These goals can then be used to quantify the goodness of an experimental design. However, before that can be done, the method of solving this inference problem needs to be introduced, as the indicator of goodness typically requires solving the inference problem itself. Furthermore, solving the inference problem is necessary in validating the final design optimization results.

3.1 Background

Parameter inference can be broadly divided into two schools of thought — Bayesian and non-Bayesian. The former treats the unknown as a random variable, incorporating both the experimenter's prior knowledge and belief, as well as observed data, via Bayes' theorem. The latter models the observed data as being parameterized by the unknown variable, which has a deterministic, albeit unknown, value. While both approaches have been studied extensively, there does not appear to be a clear superior method; rather, one method can be a more suitable choice depending on the problem structure. Some discussions about the advantages and disadvantages of the two approaches can be found in, for example, [25] and [92].

In this study, the Bayesian approach is selected. It can, for example, provide a natural structure for sequential parameter inference as well as sequential experimental design. An introduction to Bayesian analysis can be found in [84], while a more theoretical discussion can be found in [45].

Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space, where Ω is the sample space, $\mathcal{F}$ is the σ-field, and $\mathbb{P}$ is the probability measure. Let the vector of random variables $\boldsymbol{\theta} : \Omega \rightarrow \mathbb{R}^{n_{\boldsymbol{\theta}}}$ be the uncertain parameters of interest whose values are to be inferred, $\mathbf{y} = \{\mathbf{y}_l\}_{l=1}^{n_{\text{meas}}}$ be the set of n_{meas} data, where $\mathbf{y}_l : \Omega \rightarrow \mathbb{R}^{n_{\mathbf{y}}}$ is one particular datum, and $\mathbf{d} \in \mathbb{R}^{n_{\mathbf{d}}}$ be the experimental conditions. Here, $n_{\boldsymbol{\theta}}$ is the number of uncertain parameters, $n_{\mathbf{y}}$ is the number of observable categories (e.g., Table 2.4 results in $n_{\mathbf{y}}$=10), and $n_{\mathbf{d}}$ is the number of design variables.

At the heart of the Bayesian inference framework is, of course, Bayes' theorem. It can be expressed in this context as

$$p\left(\boldsymbol{\theta}|\, \mathbf{y}, \mathbf{d}\right) = \frac{p\left(\mathbf{y}|\, \boldsymbol{\theta}, \mathbf{d}\right) p\left(\boldsymbol{\theta}|\, \mathbf{d}\right)}{p\left(\mathbf{y}|\, \mathbf{d}\right)}, \tag{3.1}$$

where $p\left(\boldsymbol{\theta}|\, \mathbf{d}\right)$ is the prior probability density function (PDF), $p\left(\mathbf{y}|\, \boldsymbol{\theta}, \mathbf{d}\right)$ is the likelihood PDF, $p\left(\boldsymbol{\theta}|\, \mathbf{y}, \mathbf{d}\right)$ is the posterior PDF, and $p\left(\mathbf{y}|\, \mathbf{d}\right)$ is the evidence PDF. Moreover, the common assumption that the prior knowledge is independent of the experimental design can be made, simplifying $p\left(\boldsymbol{\theta}|\, \mathbf{d}\right) = p\left(\boldsymbol{\theta}\right)$. Upon obtaining the posterior PDF, point or interval estimates may also be constructed.

3.2 Prior and Likelihood

For the purpose of demonstration, the unknown kinetic parameters are chosen to be A_1 and $E_{a,3}$ (i.e., the pre-exponential factor of reaction R1 and the activation energy of reaction R3), while all other kinetic parameters are set to their recommended values, tabulated in Appendix A. In particular, instead of controlling A_1 directly, a transformation of $\tilde{A}_1 \equiv \ln\left(A_1/A_1^*\right)$ (where A_1^* is the recommended value of A_1) shall be used instead, which is useful in constraining A_1 to positive values only. Extension to additional parameters is easily generalizable.

The support of the parameter prior usually reflects constraints of physical admissibility, experimental limits, or regions of interest. Ranges for kinetic parameters, however, are not as intuitive as more familiar variables such as temperature. Therefore, some preliminary tests have been performed to determine "interesting regions" of these parameters using the error models constructed in the next section. More specifically, these regions are those having relatively large posterior values. Furthermore, a uniform prior is assigned across the support adhering to the principles of indifference and maximum entropy [41]. The uniform prior is also the non-informative Jeffreys prior [42] using the Gaussian likelihood models to be introduced shortly. The prior support is summarized in Table 3.1.

Parameter	Lower Bound	Upper Bound
$\tilde{A}_1$	-0.05	0.05
$E_{a,3}$	0	2.7196×10^7

Table 3.1: Prior support of the uncertain kinetic parameters $\tilde{A}_1$ and $E_{a,3}$. Uniform prior is assigned.

In constructing the likelihood, an additive error model is assumed for the observables

$$\mathbf{y}_l(\boldsymbol{\theta}, \mathbf{d}) = \mathbf{g}(\boldsymbol{\theta}, \mathbf{d}) + \boldsymbol{\epsilon}(\mathbf{d}), \forall l, \tag{3.2}$$

where $\mathbf{g}(\boldsymbol{\theta}, \mathbf{d})$ is the output from the computational model (i.e., Cantera), and $\boldsymbol{\epsilon}(\mathbf{d}) = \left(\epsilon_1(\mathbf{d}), \cdots, \epsilon_{n_y}(\mathbf{d})\right)$ is the additive error. Furthermore, the error shall be assumed to be i.i.d. zero-mean Gaussians $\epsilon_i \sim \mathcal{N}\left(0, \sigma_i^2(\mathbf{d})\right)$. This independence property conveniently makes the different entries in a datum vector to be independent conditioned on the parameters $\boldsymbol{\theta}$. Additionally, data measurements (i.e., the different $\mathbf{y}_l$'s) can be reasonably assumed to be independence conditioned on $\boldsymbol{\theta}$, and hence

$$\begin{aligned} p(\mathbf{y}|\,\boldsymbol{\theta},\mathbf{d}) &= \prod_{l=1}^{n_{\text{meas}}} p(\mathbf{y}_l|\,\boldsymbol{\theta},\mathbf{d}) \\ &= \prod_{l=1}^{n_{\text{meas}}} \prod_{i=1}^{n_{\mathbf{y}}} p(y_{l,i}|\,\boldsymbol{\theta},\mathbf{d}) \\ &= \prod_{l=1}^{n_{\text{meas}}} \prod_{i=1}^{n_{\mathbf{y}}} \frac{1}{\sigma_i\sqrt{2\pi}} \exp\left\{-\frac{(y_{l,i}-g_i(\boldsymbol{\theta},\mathbf{d}))^2}{2\sigma_i^2}\right\}. \end{aligned} \tag{3.3}$$

The σ_i's are allowed to be a function of the design variables (but not of uncertain parameters). The reason for this choice is that the observables in the combustion problem can vary over orders of magnitude when the design variables are changed within reasonable ranges — for example, such an H_2-O_2 combustion at $T = 900$ K has τ_{ign} on the order of 10^{-1} seconds, while at $T = 1000$ K, 10^{-3} seconds. Thus having the same error "width" at the two design conditions with such different observable values would seem unrealistic. For example, measurement error magnitudes generally increase with the measurement magnitudes. On the other hand, preliminary testings indicate that the observables are much less sensitive to the unknown parameters within the prior support set earlier, and thus the $\boldsymbol{\theta}$ dependence is not included in the σ_i's. Its inclusion, if desired, would be trivial to implement. Additionally, because of the large variations of the characteristic time observables, instead of directly using the τ's in the implementation, their natural log values, $\ln\tau$'s, are used for the rest of this thesis. However, the ϵ_i's are still Gaussian with respect to the non-log values. Thus, the only difference this would make is that, as to be introduced in Chapter 5, the polynomial chaos expansions would approximate the $\ln\tau$'s instead of the τ's.

The σ_i's are determined as follows. At the desired design conditions $\mathbf{d}$, a simulation at the recommended parameter values is performed to obtain the nominal observable values. For the peak-value observables ($\frac{dh}{dt}\big|_{\tau}$, $\chi_{O,\tau}$, $\chi_{H,\tau}$, $\chi_{HO_2,\tau}$, and $\chi_{H_2O_2,\tau}$), σ_i's are simply chosen to be 10% of the nominal observable values. For the characteristic-time observables (τ_{ign}, τ_O, τ_H, τ_{HO_2}, and $\tau_{H_2O_2}$), a common value of σ shall be used (because all five of these characteristic times typically have very similar magnitudes), which is represented by a linear model:

$$\sigma = a + b\tau_{0.75}. \tag{3.4}$$

$\tau_{0.75}$ is the characteristic interval length of the 75% peak values in the profile of the enthalpy release rate — this is illustrated in Figure 3-1. The purpose of a is to establish some minimum level of error, reflecting the resolution limit of timing technology. The purpose of b is to represent an assumed linear dependence on some width measure of a peak. The rationale is that pinpointing the maximum of a flatter peak would be experimentally more challenging, since the measurement error on the magnitude of the quantity would be more significant compared to the variation of the quantity across the peak — in other words, similar values would be more difficult to differentiate. This might be counter-intuitive since one often expects a very acute peak to be harder to detect. However, the detection issue is a matter of sensor resolution, which is reflected by the constant term a, not by b. For this study, $a = 10^{-5}$ seconds is a realistic choice, while $b = 10$ appears to work well after some experimenting.

3.3 Markov Chain Monte Carlo (MCMC)

In this section, all data and analysis are assumed to be from a fixed design condition, $\mathbf{d}$. With this assumption in mind, conditioning of $\mathbf{d}$ in PDFs shall be dropped in the notation for simplicity.

In general, the PDF of the posterior does not have an analytical form. The obvious method to construct the posterior is to simply form a grid in $\boldsymbol{\Theta}$ (where $\boldsymbol{\Theta}$ is the support of $p(\boldsymbol{\theta})$), and evaluate the posterior values at the grid points. While this can be done if $\boldsymbol{\Theta}$ is of very low dimensions, it becomes impractical for more than, say, two dimensions, as the number of grid points grows exponentially with the number of dimensions. Instead, a more economical method is to generate independent samples from the posterior PDF (e.g., via Monte Carlo). However, since the posterior generally

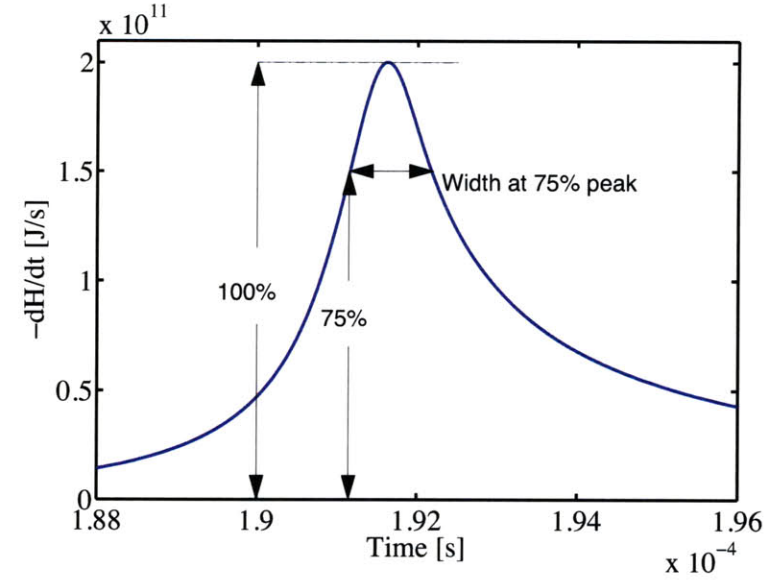


Figure 3-1: Illustration of $\tau_{0.75}$, the characteristic interval length of the 75% peak values in the profile of the enthalpy release rate.

does not have an analytic form, direct sampling from it would be difficult. Even if it does have some analytic form, the inverse-cumulative density function (CDF) sampling method cannot be used in multiple dimensions. The most robust method is perhaps the acceptance-rejection method, but that would almost always be very inefficient.

Markov chain Monte Carlo (MCMC) offers a solution, by constructing a Markov chain whose target distribution is the posterior, while trading off the independence of the samples. Nonetheless, a well-tuned MCMC can offer samples with very low correlation. The main advantage of MCMC is that the target distribution can be constructed solely based on point-wise evaluations of the *unnormalized* posterior. The resulting samples can then be used to either visually present the posterior or marginal posteriors, or to approximate expectations with respect to the posterior

$$\mathbb{E}_{p(\boldsymbol{\theta}|\mathbf{y})}\left[f\left(\boldsymbol{\theta}\right)\right] = \int_{\boldsymbol{\Theta}} f\left(\boldsymbol{\theta}\right) p\left(\boldsymbol{\theta}|\,\mathbf{y}\right)\, d\boldsymbol{\theta} \tag{3.5}$$

with the Monte Carlo estimates

$$\bar{f}_{n_M} = \frac{1}{n_{\mathrm{M}}} \sum_{t=1}^{n_{\mathrm{M}}} f\left(\boldsymbol{\theta}^{(t)}\right), \tag{3.6}$$

where n_{M} is the number of MCMC samples, and $\boldsymbol{\theta}^{(t)}$'s are the MCMC samples. For example, the popular minimum mean square error (MMSE) estimator is simply the mean of the posterior, while the corresponding Bayes risk is the posterior variance.

MCMC is supported by a strong theoretical foundation, and also requires finesse in its implementation — a well-implemented MCMC is an art itself. Awareness of its numerous variations, diagnostics, tricks, and caveats is essential in creating an effective MCMC algorithm. This thesis does not attempt to survey the enormous number of theoretical and practical topics of MCMC, but rather, the reader is referred to [5] for a brief review, and [31, 78] for detailed discussions.

3.3.1 Metropolis-Hastings (MH)

One simple and popular variation of MCMC is the Metropolis-Hastings (MH) algorithm, first proposed by Metropolis *et al.* [61], and later generalized by Hastings [38]. The algorithm is outlined in Algorithm 1.

Additionally, if the proposal is symmetric (i.e., $q\left(\boldsymbol{\theta}^{(t)}\middle|\boldsymbol{\theta}'\right) = q\left(\boldsymbol{\theta}'|\,\boldsymbol{\theta}^{(t)}\right)$), then the acceptance probability in Equation 3.7 reduces to

$$\alpha\left(\boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}'\right) = \min\left\{1, \frac{p\left(\boldsymbol{\theta}'|\,\mathrm{y}\right)}{p\left(\boldsymbol{\theta}^{(t)}\middle|\,\mathrm{y}\right)}\right\}, \tag{3.8}$$

which is known as the Metropolis algorithm [61].

3.3.2 Delayed Rejection Adaptive Metropolis (DRAM)

Two useful improvements to MH are the concepts of delayed rejection (DR) [34, 62] and adaptive Metropolis (AM) [37], which are recently combined together (DRAM) [35].

Algorithm 1: Metropolis-Hastings algorithm.

```
Initialize $\boldsymbol{\theta}^{(t)}$, where $t = 1$;
while $t < n_M$ do
    Sample candidate state $\boldsymbol{\theta}'$ from proposal PDF $q\left(\cdot|\,\boldsymbol{\theta}^{(t)}\right)$;
    Sample $U \sim \mathcal{U}(0,1)$;
    Compute
        $$\alpha\left(\boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}'\right) = \min\left\{1, \frac{p(\boldsymbol{\theta}'|\,\mathbf{y})\, q\left(\boldsymbol{\theta}^{(t)}\middle|\,\boldsymbol{\theta}'\right)}{p\left(\boldsymbol{\theta}^{(t)}\middle|\,\mathbf{y}\right) q\left(\boldsymbol{\theta}'|\,\boldsymbol{\theta}^{(t)}\right)}\right\}. \quad (3.7)$$
    if $U \leq \alpha\left(\boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}'\right)$ then
        $\boldsymbol{\theta}^{(t+1)} = \boldsymbol{\theta}'$ (accept);
    else
        $\boldsymbol{\theta}^{(t+1)} = \boldsymbol{\theta}^{(t)}$ (reject);
    end
    $t = t + 1$;
end
```

Delayed Rejection (DR)

The idea behind DR [34, 62] is that, if a proposed candidate in MH is to be rejected, instead of rejecting it right away, a second (or even higher) stage proposal is induced. The acceptance probability in the higher stages are computed such that the reversibility of the Markov chain is preserved. This multi-stage strategy can allow different proposals to be mixed, for example, with the first stage proposal to have a large proposal "width" to detect potential multi-modality, while the higher stages to have smaller "widths" to more efficiently explore local modes. Additionally, it has been shown [93] that DR improves MH algorithm in the Peskun sense [73], by uniformly reducing its asymptotic variance.

The first stage has the same acceptance probability as described in the MH algorithm

$$\begin{aligned} \alpha_1\left(\boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}_1'\right) &= \min\left\{1, \frac{p\left(\boldsymbol{\theta}' \middle| \mathbf{y}\right) q\left(\boldsymbol{\theta}^{(t)} \middle| \boldsymbol{\theta}'\right)}{p\left(\boldsymbol{\theta}^{(t)} \middle| \mathbf{y}\right) q\left(\boldsymbol{\theta}' \middle| \boldsymbol{\theta}^{(t)}\right)}\right\} \\ &= \min\left\{1, \frac{N_1}{D_1}\right\}, \end{aligned} \tag{3.9}$$

and the second stage has an acceptance probability

$$\begin{aligned} \alpha_2\left(\boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}_1', \boldsymbol{\theta}_2'\right) &= \min\left\{1, \frac{p\left(\boldsymbol{\theta}_2' \middle| \mathbf{y}\right) q_1\left(\boldsymbol{\theta}_1' \middle| \boldsymbol{\theta}_2'\right) q_2\left(\boldsymbol{\theta}^{(t)} \middle| \boldsymbol{\theta}_1', \boldsymbol{\theta}_2'\right) \left[1-\alpha_1\left(\boldsymbol{\theta}_2', \boldsymbol{\theta}_1'\right)\right]}{p\left(\boldsymbol{\theta}^{(t)} \middle| \mathbf{y}\right) q_1\left(\boldsymbol{\theta}_1' \middle| \boldsymbol{\theta}^{(t)}\right) q_2\left(\boldsymbol{\theta}_2' \middle| \boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}_1'\right) \left[1-\alpha_1\left(\boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}_1'\right)\right]}\right\} \\ &= \min\left\{1, \frac{N_2}{D_2}\right\}. \end{aligned} \tag{3.10}$$

In general, the ith stage has an acceptance probability

$$\begin{aligned} \alpha_i\left(\boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}_1', \cdots, \boldsymbol{\theta}_i'\right) &= \min\left\{1, \frac{p\left(\boldsymbol{\theta}_i' \middle| \mathbf{y}\right) q_1\left(\boldsymbol{\theta}_{i-1}' \middle| \boldsymbol{\theta}_i'\right) q_2\left(\boldsymbol{\theta}_{i-2}' \middle| \boldsymbol{\theta}_{i-1}', \boldsymbol{\theta}_i'\right) \cdots q_i\left(\boldsymbol{\theta}^{(t)} \middle| \boldsymbol{\theta}_1', \cdots, \boldsymbol{\theta}_i'\right)}{p\left(\boldsymbol{\theta}^{(t)} \middle| \mathbf{y}\right) q_1\left(\boldsymbol{\theta}_1' \middle| \boldsymbol{\theta}^{(t)}\right) q_2\left(\boldsymbol{\theta}_2' \middle| \boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}_1'\right) \cdots q_i\left(\boldsymbol{\theta}_i' \middle| \boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}_1', \cdots, \boldsymbol{\theta}_{i-1}'\right)}\right. \\ &\qquad \left. \frac{\left[1-\alpha_1\left(\boldsymbol{\theta}_i', \boldsymbol{\theta}_{i-1}'\right)\right]\left[1-\alpha_2\left(\boldsymbol{\theta}_i', \boldsymbol{\theta}_{i-1}', \boldsymbol{\theta}_{i-2}'\right)\right] \cdots \left[1-\alpha_{i-1}\left(\boldsymbol{\theta}_i', \cdots, \boldsymbol{\theta}_1'\right)\right]}{\left[1-\alpha_1\left(\boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}_1'\right)\right]\left[1-\alpha_2\left(\boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}_1', \boldsymbol{\theta}_2'\right)\right] \cdots \left[1-\alpha_{i-1}\left(\boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}_1', \cdots, \boldsymbol{\theta}_{i-1}'\right)\right]}\right\} \\ &= \min\left\{1, \frac{N_i}{D_i}\right\}. \end{aligned} \tag{3.11}$$

Upon reaching the ith stage, all previous stages must have led to rejection, and hence $N_j \leq D_j$ for $j = 1, \cdots, i-1$. Thus, $\alpha_j\left(\boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}_1', \cdots, \boldsymbol{\theta}_j'\right) = N_j/D_j$, which leads to a convenient recursive formula

$$D_i = q_i\left(\boldsymbol{\theta}_i' \middle| \boldsymbol{\theta}^{(t)}, \boldsymbol{\theta}_1', \cdots, \boldsymbol{\theta}_{i-1}'\right)\left(D_{i-1} - N_{i-1}\right). \tag{3.12}$$

Since each stage independently preserves the reversibility of the Markov chain, DR may be terminated after any finite number of stages. An alternative is to terminate with probability p_{DR} at each stage. Upon termination, the original state $\boldsymbol{\theta}^{(t)}$ is retained as in a rejection case.

Adaptive Metropolis (AM)

The tuning of the proposal "width" and "orientation" in MH is a tedious, but necessary task. It requires trial-and-error, and the resulting optimal parameters are different for different problems. For example, too large of a width would lead to high rejection probability, causing slow movement of the chain and thus poor mixing; too small of a width would lead to high acceptance probability, but each accepted state would be very close to the previous state, again leading to slow mixing. An automated way of tuning is desirable.

There exists numerous adaptation algorithms that are based on past samples in the chain history (e.g., [79, 94]). However, such alterations often destroy the Markov property of the chain, and hence the usual MCMC convergence results are no longer valid. This problem can be avoided if the adaptation is not performed constantly (e.g., only during a burn in period to tune the proposal parameters), or else the algorithm's ergodicity has to be proven separately. The AM algorithm by Haario *et al.* is an example of the latter, and is introduced below.

A multivariate Gaussian proposal centered at the current state, $q\left(\boldsymbol{\theta}' | \boldsymbol{\theta}^{(t)}\right) \sim \mathcal{N}\left(\boldsymbol{\theta}^{(t)}, \boldsymbol{\Sigma}\right)$, is assumed, with the proposal width and orientation reflected through its covariance matrix $\boldsymbol{\Sigma}$. Haario *et al.* [36] first proposed the Adaptive Proposal algorithm, where it updates $\boldsymbol{\Sigma}$ with the sample covariance matrix using samples in a fixed window of history. However, this algorithm was shown to be non-ergodic. Improvements are then made by Haario *et al.* [37] to introduce the AM algorithm, which uses all the past samples in history to form the sample covariance matrix, and is proven to be ergodic. This algorithm is described in detail below.

Let $\boldsymbol{\Sigma}_1$ be some initial proposal covariance matrix set by the user, the subsequent update at iteration t is then

$$\boldsymbol{\Sigma}^{(t)} = \begin{cases} \boldsymbol{\Sigma}_1, & t \leq n_A \\ s_d \mathrm{Cov}\left(\boldsymbol{\theta}^{(1)}, \cdots, \boldsymbol{\theta}^{(t-1)}\right) + s_d \varepsilon I_d, & t > n_A, \end{cases} . \qquad (3.13)$$

n_A is the iteration number in which the adaptive covariance matrix replaces the initial covariance matrix. The reason for this delay is that some initial samples are required for the sample covariance to have some significance, but of course the larger n_A is, the longer before AM starts to have its effect. At the same time, $\mathbf{\Sigma}_1$ cannot be totally unreasonable — for example, if all samples before n_A are rejected, the sample covariance would be singular, and does not give much useful information in adapting $\mathbf{\Sigma}^{(t)}$. I_d is the d-dimensional identity matrix, and $\varepsilon > 0$ is a small perturbation that makes sure $\mathbf{\Sigma}^{(t)}$ is non-singular. s_d is a scaling factor for the proposal. For example, $s_d = 2.4^2/d$ is the optimal value for Gaussian target distributions [26].

The sample covariance needs not be re-computed from all samples at each iteration, but can be simply updated through the formula

$$\operatorname{Cov}\left(\boldsymbol{\theta}^{(1)}, \cdots, \boldsymbol{\theta}^{(t)}\right) = \frac{1}{t-1}\left(\sum_{i=1}^{t} \boldsymbol{\theta}^{(t)} \left(\boldsymbol{\theta}^{(t)}\right)^T - t\bar{\boldsymbol{\theta}}^{(t)} \left(\bar{\boldsymbol{\theta}}^{(t)}\right)^T\right), \tag{3.14}$$

where

$$\bar{\boldsymbol{\theta}}^{(t)} = \frac{1}{t}\sum_{i=1}^{t} \boldsymbol{\theta}^{(i)} = \frac{t-1}{t}\bar{\boldsymbol{\theta}}^{(t-1)} + \frac{1}{t}\boldsymbol{\theta}^{(t)} \tag{3.15}$$

is the sample mean, and can be easily updated with the recursive formula above. Substituting Equations 3.14 and 3.15 into Equation 3.13 yields the recursive formula

$$\mathbf{\Sigma}^{(t+1)} = \frac{(t-2)}{(t-1)}\mathbf{\Sigma}^{(t)} + \frac{s_d}{t-1}\left[(t-1)\,\bar{\boldsymbol{\theta}}^{(t-1)}\left(\bar{\boldsymbol{\theta}}^{(t-1)}\right)^T - t\bar{\boldsymbol{\theta}}^{(t)}\left(\bar{\boldsymbol{\theta}}^{(t)}\right)^T + \boldsymbol{\theta}^{(t)}\left(\boldsymbol{\theta}^{(t)}\right)^T + \varepsilon I_d\right]. \tag{3.16}$$

One may also choose to update $\mathbf{\Sigma}$ every n_i iterations instead of every iteration.

Combining DR and AM

There are various ways to combine DR and AM [35]. The most direct method, as adopted by this study, is as follows.

- Perform DR at each iteration as described. At the ith rejection stage, one can simply set $q_i = \gamma q_{i-1}$, where γ is a scaling factor. The magnitude of γ reflects how rapidly the covariance resizes upon each rejection. Experience shows that choosing a slightly larger $\mathbf{\Sigma}_1$ and setting $\gamma \in (0, 1)$ works well (e.g., $\gamma = 0.1$).

- At the end of an iteration, only one sample would emerge — either the same sample as the previous iteration (i.e., rejected through all stages before terminating DR), or a sample accepted at some stage. That sample is used to update $\mathbf{\Sigma}^{(t)}$ using the AM method described.

3.3.3 Numerical Examples

Two simple test cases are presented to compare the performance between MH, DR, AM, and DRAM. The first case involves a series of multivariate Gaussians, which are relative easy target distributions for MCMC; the second case involves a variant of the Rosenbrock function, which has a very nonlinear, banana-shaped peak that provides a more difficult challenge to these algorithms. The examples are similar to those found in [35], but not exactly the same.

Multivariate Gaussians

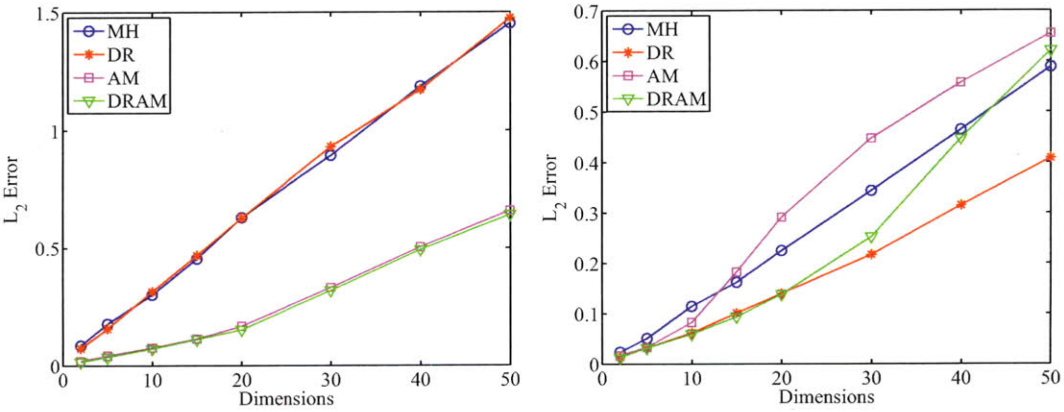
In this test case, correlated zero-mean multivariate Gaussians $\mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_{tar})$ of different dimensions are used as the target distributions. In order for the comparisons across dimensions to be meaningful, $\mathbf{\Sigma}_{tar}$ for each dimension is randomly generated, while fixing the condition number at 10. The initial proposal covariance matrices, the $\mathbf{\Sigma}_1$'s, are taken to be proportional to the $\mathbf{\Sigma}_{tar}$'s. Two situations are considered:

1. the proposal is too small, by setting $\mathbf{\Sigma}_1 = 0.01 s_d \mathbf{\Sigma}_{tar}$; and

2. the proposal is too large, by setting $\mathbf{\Sigma}_1 = 4 s_d \mathbf{\Sigma}_{tar}$.

In other words, the proposal covariances are well-oriented, but not well-scaled. The initial position is taken to be $(-1, \cdots, -1)$, away from the Gaussian mean. For DR, a maximum of 2 stages are allowed (i.e., a total of 2 levels of proposals), with $\gamma = 0.1$. For AM, $n_A = 500$ is selected. A chain length of 20,000 iteration is used for all runs, with no burn in. At each dimension, statistics are averaged over 100 independent MCMC runs.

The L_2 errors of the sample means are computed. The results are shown in Figure 3-2. The errors grow as dimension grows, due to the fixed length of the Markov chains. More interestingly, the AM and DRAM outperform MH and DR when $\mathbf{\Sigma}_1$ is too small, while all four algorithms have similar performance when $\mathbf{\Sigma}_1$ is too large. The reason for this observation is that, since DR only scales down the proposal covariance, DR has no effect when $\mathbf{\Sigma}_1$ is already too small.

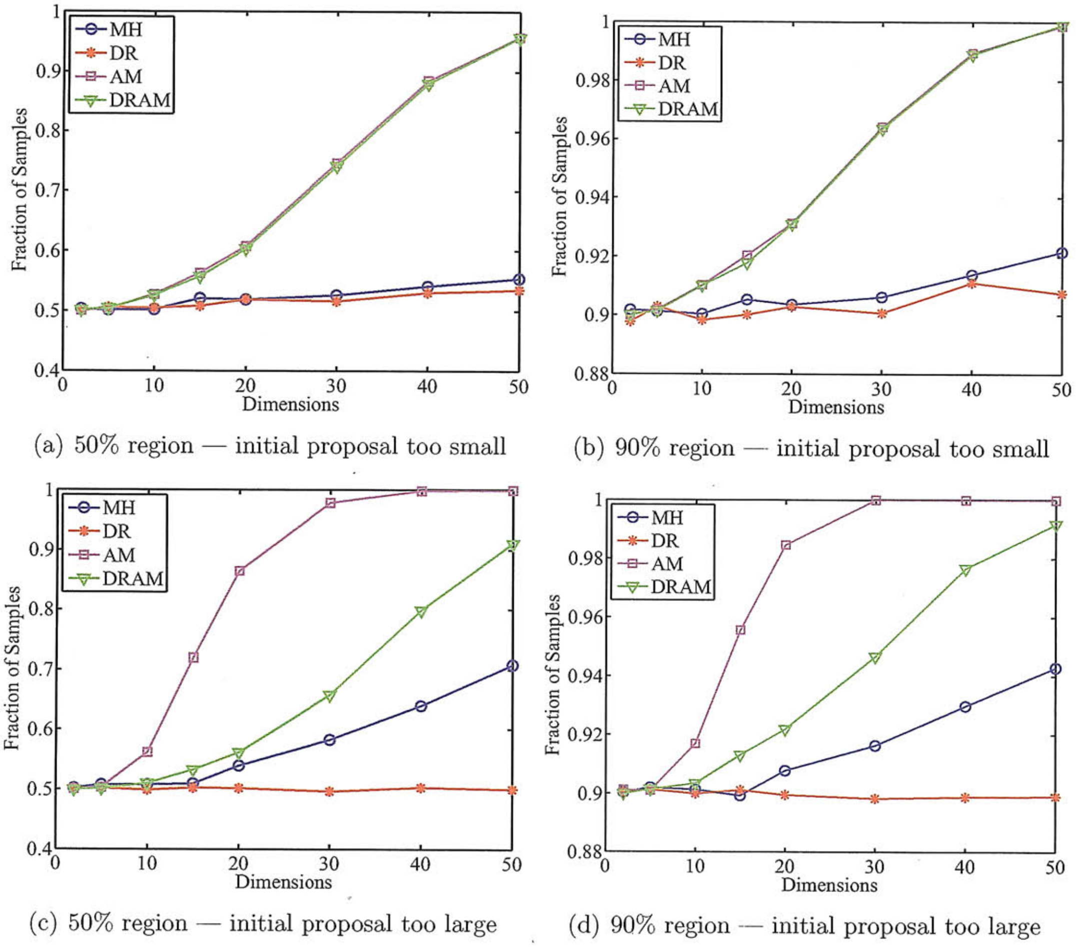


(a) Initial proposal too small

(b) Initial proposal too large

Figure 3-2: L_2 errors of the mean estimates for multivariate Gaussians.

Another analysis is to compute the fractions of sample points that fall within the 50% and 90% credible regions centered around the mean of the Gaussians (i.e., the origin). In a 2D contour, such region is simply an ellipse centered around the origin. The results are shown in Figure 3-3. In all cases, the AM and DRAM algorithms tend to oversample from the central regions of the Gaussians, especially as the number of dimensions increases. However, MH and DR are able to resist such phenomenon. This indicates a potential drawback in AM, that a too well-tuned proposal may induce oversampling in the high probability regions. One possible method to alleviate this problem in high dimensions is to choose a larger n_A or to adapt less frequently (e.g., adapt every n_i iterations).



(a) 50% region — initial proposal too small

(b) 90% region — initial proposal too small

(c) 50% region — initial proposal too large

(d) 90% region — initial proposal too large

Figure 3-3: Sample fractions within the 50% and 90% regions for multivariate Gaussians.

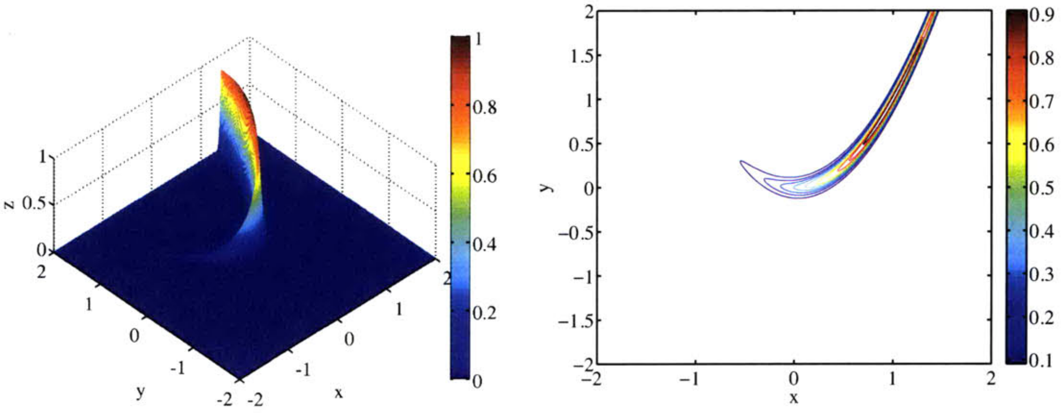
Rosenbrock Variant

Let the target (unnormalized) PDF of the MCMC be the following variant of the 2D Rosenbrock function for $(x, y) \in [-2, 2] \times [-2, 2]$:

$$f(x, y) = \exp\left\{-\left[(1-x)^2 + 100\left(y - x^2\right)^2\right]\right\}. \tag{3.17}$$

The original Rosenbrock function is often used to test optimization algorithms, and it has a sharp banana-shaped valley which is near-flat along the bottom of the valley.

In the variant, the negative inside the exponential “flips” the valley, turning it into a peak. The exponential ensures the function to be positive, and at the same time making the peak even sharper. The variant function is shown in Figure 3-4.



(a) Surface plot

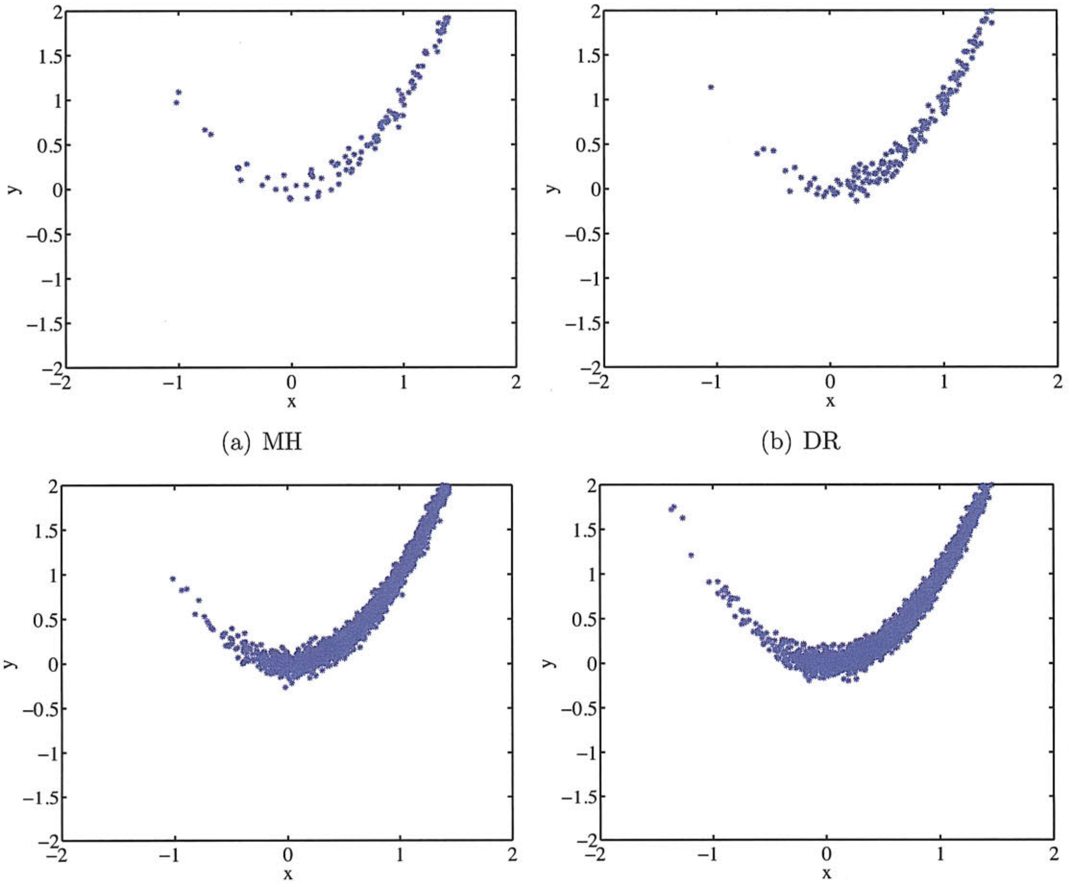
(b) Contour plot

Figure 3-4: The 2D Rosenbrock variant.

The MH, DR, AM, DRAM algorithms are run for 100,000 iterations, with a burn in of 1000 iterations. The initial proposal covariance matrix is $\mathbf{\Sigma}_1 = \text{diag}\,(10.0, 10.0)$ (which is much “wider” than the “thickness” of the banana peak, or even the prior support), and the initial position is $(-2, -2)$. For DR, a maximum of 2 stages are allowed (i.e., a total of 2 levels of proposals), with $\gamma = 0.1$. For AM, $n_A = 10$ is selected.

The last 20,000 samples from the algorithms are plotted in Figure 3-5, all of which resemble the banana-shaped peak from Figure 3-4. However, the plots for MH and DR appear less well mixed, indicating more rejections took place for them; this is further supported by the corresponding chain history of the x component shown in Figure 3-6, and the acceptance rates tabulated in Table 3.2.

The numbers of function evaluations in Table 3.2 reflect the computational time of the algorithms. Note that if a proposed coordinate falls outside the prior support, it is immediately rejected without evaluating $f\,(x, y)$, and this point would not be counted towards the total number of function evaluations. The results from the table show that adding the AM feature requires about 4 times more function evaluations, but the acceptance rate increases by a factor of 10. However, adding the DR feature requires about 2 times more function evaluations, but the acceptance rate is almost unchanged. These evidence supports DR's ineffectiveness.



(a) MH

(b) DR

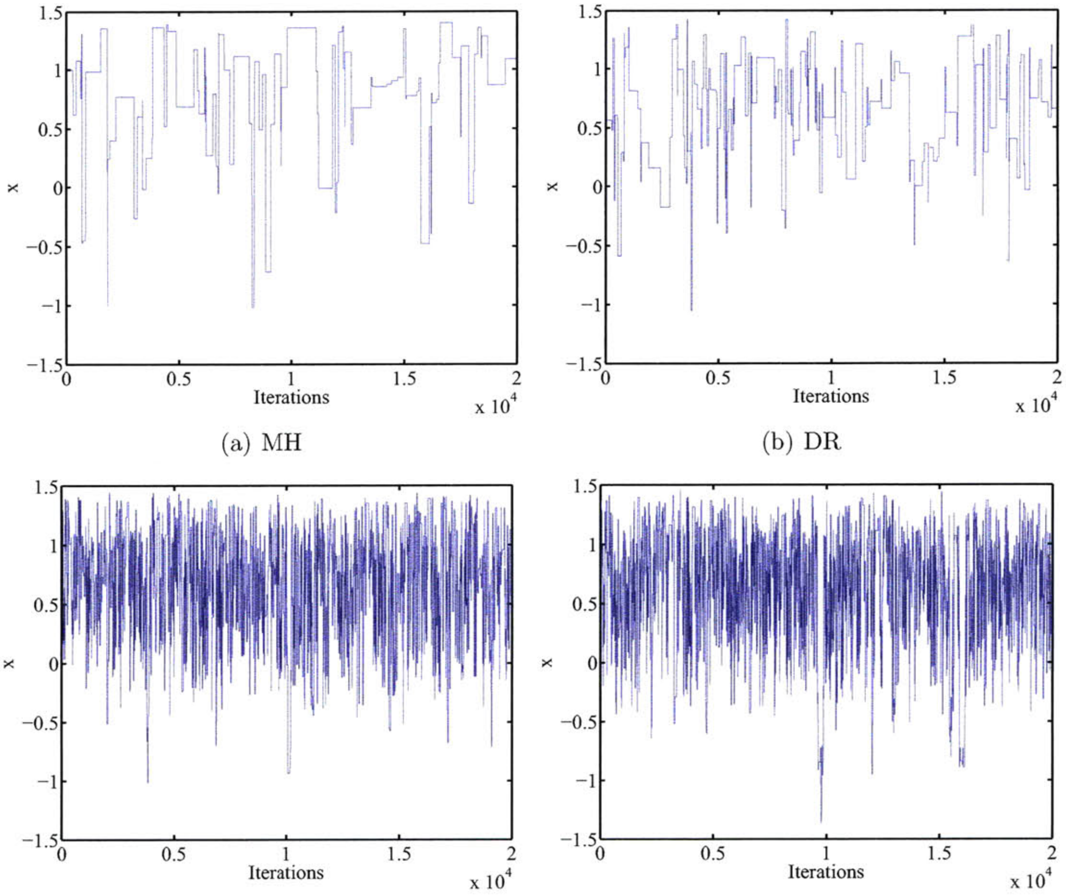
(c) AM

(d) DRAM

Figure 3-5: Last 20,000 samples for the 2D Rosenbrock variant.

Another method to quantify the mixing of a Markov chain is to compute its autocorrelation function (ACF)

$$\mathrm{ACF}\,(s) = \mathrm{Cov}\left(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(j)}\right), \tag{3.18}$$



(a) MH

(b) DR

(c) AM

(d) DRAM

Figure 3-6: Last 20,000 samples of component x for the 2D Rosenbrock variant.

which is a function of separation or lag, $s = i - j$, between the ith and jth samples in the chain. Ideally, if independent sampling can be achieved, the ACF value would equal to 1 when lag is 0, and 0 everywhere else. Samples from MCMC, of course, are not independent; however, the rate in which the ACF decays with s is an indicator of the degree of independence between the samples. For example, if the ACF decays to some tolerance value, say 0.1, at a lag of 40, then loosely speaking, every 40 MCMC samples may be viewed to be equivalent to a single independent sample. More rigorously, by

Algorithm	Acceptance Rate	Function Evaluations
MH	0.0060	21,116
DR	0.0077	41,319
AM	0.0880	84,911
DRAM	0.0940	160,136

Table 3.2: Acceptance rate and number of posterior function evaluations for the 2D Rosenbrock variant.

the Central Limit Theorem, the distribution of the estimator error is

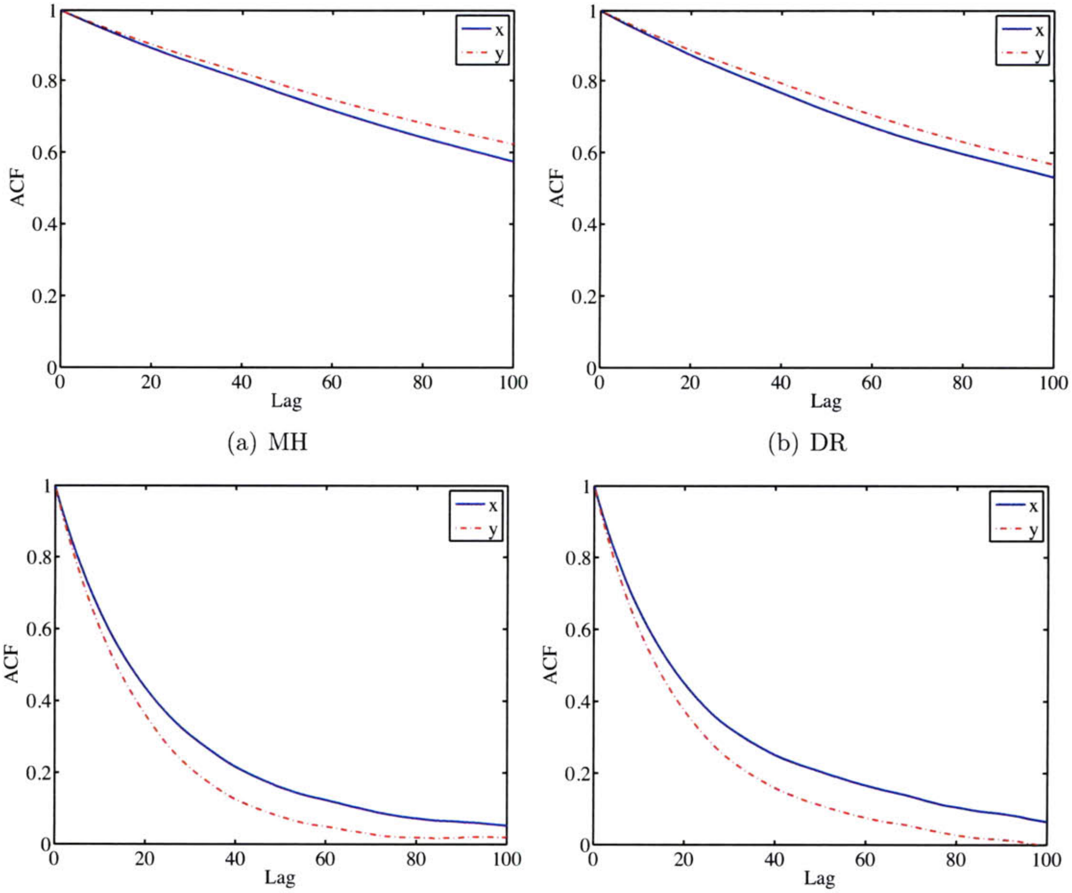
$$\left(\bar{f}_{n_M} - \mathbb{E}_{p(\boldsymbol{\theta}|\mathbf{y})}\left[f\left(\boldsymbol{\theta}\right)\right]\right) \rightarrow \mathcal{N}\left(0, \frac{\sigma^2}{\sqrt{n_M}}\right) \tag{3.19}$$

as $n_M \rightarrow \infty$, where σ is a constant. The magnitude of σ is given by

$$\sigma^2 = \mathrm{Var}_{p(\boldsymbol{\theta}|\mathbf{y})}\left(f\left(\boldsymbol{\theta}_0\right)\right) + 2\sum_{i=1}^{\infty} \mathrm{Cov}\left(\boldsymbol{\theta}_0, \boldsymbol{\theta}^{(i)}\right), \tag{3.20}$$

where $\boldsymbol{\theta}_0$ is distributed according to $p\left(\boldsymbol{\theta}|\,\mathbf{y}\right)$. $\mathrm{Var}_{p(\boldsymbol{\theta}|\mathbf{y})}\left(f\left(\boldsymbol{\theta}_0\right)\right)$ is a property of the posterior, and cannot be controlled; but $\sum_{i=1}^{\infty} \mathrm{Cov}\left(\boldsymbol{\theta}_0, \boldsymbol{\theta}^{(i)}\right)$, which is the integrated ACF, can be controlled to reduce the variance of the error. The ACF's for the four algorithms are shown in Figure 3-7. Clearly, the ACF decay is accelerated when the AM feature is introduced, but is rarely improved, if at all, when the DR feature is introduced.

The effect of ACF on the variance of the MC estimator can be seen in Figures 3-8 and 3-9, which plot the distributions of sample mean in x and y coordinates after repeating the previously described MCMC algorithms for 1000 times each. Clearly, the variance appears much smaller in AM and DRAM cases compared to MH and DR. Note that since the analytic means of Equation 3.17 in the finite domain are unavailable, it is thus not known whether or not these distributions have means equal to the true means, which would be an indicator of unbiasedness. However, since MH is known to be unbiased, and its distributions have the same means as the other three algorithms, then it can be safe to conclude that all four algorithms are indeed all



(a) MH

(b) DR

(c) AM

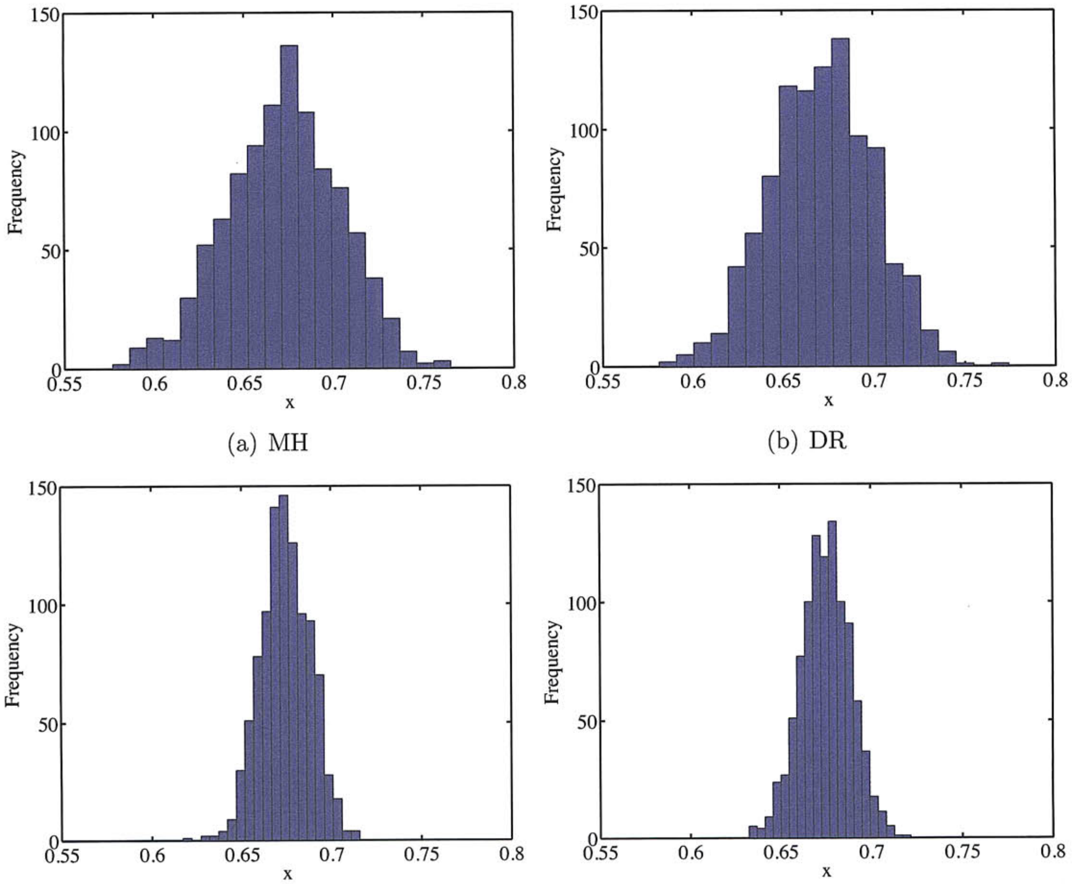
(d) DRAM

Figure 3-7: Autocorrelation functions for the 2D Rosenbrock variant.

unbiased, as expected by their constructions.

Conclusion

These numerical tests indicate that while the AM feature performs well in accelerating the mixing of the Markov chain, the DR feature is largely ineffective. Also, AM can induce over-sampling in high probability regions when the number of dimensions is high, but this may be mitigated by using a less aggressive adaptation schedule. In all remaining MCMC results in this thesis, only the AM algorithm is used.



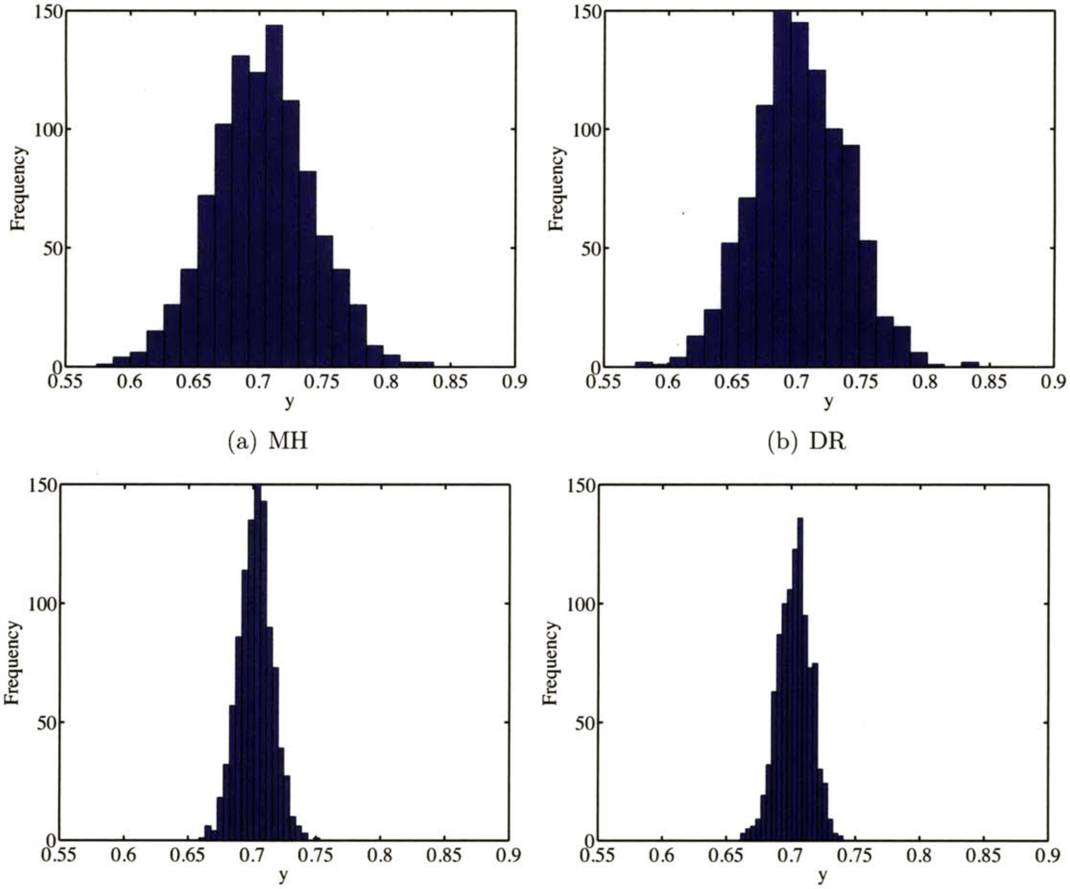
(a) MH

(b) DR

(c) AM

(d) DRAM

Figure 3-8: Distribution of the sample mean in x for 1000 repeated Markov chains.



(a) MH

(b) DR

(c) AM

(d) DRAM

Figure 3-9: Distribution of the sample mean in y for 1000 repeated Markov chains.

Chapter 4

Optimal Bayesian Experimental Design

After identifying the appropriate experimental observables in Chapter 2, an attempt can now be made to quantify the goodness of an experiment. Upon the development of such an indicator, numerical optimization can be invoked to find the optimal experimental design. This experiment can then be carried out, and its data analyzed as intended by the original experimental goals.

4.1 Background

What constitutes a good experiment? That, of course, depends on what the experimental goals are. For example, if the goal is to confidently infer an uncertain parameter, then intuitively, a good experiment might be one that minimizes the variance of the parameter's posterior. Before trying to construct an indicator for the combustion experiment, some background of current research in optimal experimental design is first introduced.

4.1.1 Optimal Linear Experimental Design

Optimal design theory for linear designs is well established, since the analysis can be carried out in closed-form. For example, the book by Atkinson and Donev [6] and the review paper by Chaloner and Verdinelli [13] provide a thorough treatment of the subject.

The linear design problem is formulated as follows. Consider a linear model, defined to have the form

$$\mathbf{g}\left(\boldsymbol{\theta},\mathbf{d}\right)=\mathbf{F}_L\left(\mathbf{d}\right)\boldsymbol{\theta}, \tag{4.1}$$

where $\boldsymbol{\theta}$ and $\mathbf{d}$ are the uncertain parameters and design variables, respectively, as described earlier in the thesis; $\mathbf{g}\in\mathbb{R}^{n_{\mathbf{y}}}$ is the vector of model outputs; and $\mathbf{F}_L\in\mathbb{R}^{n_{\mathbf{y}}\times n_{\boldsymbol{\theta}}}$ is the local design matrix. Note that the model outputs only need to be linear with respect to the uncertain coefficients $\boldsymbol{\theta}$, but not necessarily linear with respect to the design variables $\mathbf{d}$. The observed data $\mathbf{y}$ are assumed to have an additive noise

$$\mathbf{y}_l=\mathbf{g}\left(\boldsymbol{\theta},\mathbf{d}\right)+\boldsymbol{\epsilon},\ l=1,\cdots,n_{\text{meas}} \tag{4.2}$$

where the noise components ϵ_i's are assumed to be independent and have equal, finite variance. Given n_{meas} data points $\{\mathbf{y}_l\}_{l=1}^{n_{\text{meas}}}$ obtained under their corresponding experimental conditions $\{\mathbf{d}_l\}_{l=1}^{n_{\text{meas}}}$, Equation 4.1 can be extended to become

$$\begin{aligned}\mathbb{E}\begin{bmatrix}\mathbf{y}_1\\ \vdots\\ \mathbf{y}_{n_{\text{meas}}}\end{bmatrix}=\begin{bmatrix}\mathbf{g}\left(\boldsymbol{\theta},\mathbf{d}_1\right)\\ \vdots\\ \mathbf{g}\left(\boldsymbol{\theta},\mathbf{d}_{n_{\text{meas}}}\right)\end{bmatrix} &= \begin{bmatrix}\mathbf{F}_L\left(\mathbf{d}_1\right)\\ \vdots\\ \mathbf{F}_L\left(\mathbf{d}_{n_{\text{meas}}}\right)\end{bmatrix}\boldsymbol{\theta}\\ \Rightarrow\tilde{\mathbf{g}}\left(\boldsymbol{\theta},\mathbf{d}\right) &= \mathbf{F}\left(\mathbf{d}\right)\boldsymbol{\theta}\end{aligned} \tag{4.3}$$

where $\tilde{\mathbf{g}}\in\mathbb{R}^{n_{\mathbf{y}}n_{\text{meas}}}$, and $\mathbf{F}\in\mathbb{R}^{(n_{\mathbf{y}}n_{\text{meas}})\times n_{\boldsymbol{\theta}}}$ is the extended design matrix. An example with a quadratic regression model is presented in Example 4.1.1.

Example 4.1.1. Quadratic Regression Formulation

Consider a quadratic model with a scalar output g ($n_{\mathbf{y}} = 1$) in a single design variable d ($n_{\mathbf{d}} = 1$)

$$g(\boldsymbol{\theta}, d) = \theta_1 + \theta_2 d + \theta_3 d^2, \tag{4.4}$$

where $\boldsymbol{\theta} = (\theta_1, \theta_2, \theta_3)$ ($n_{\boldsymbol{\theta}} = 3$). Suppose resources allow 4 experiments ($n_{\text{meas}} = 4$) to be performed, then the experimental design formulation yields

$$\begin{bmatrix} g_1 \\ g_2 \\ g_3 \\ g_4 \end{bmatrix} = \begin{bmatrix} 1 & d_1 & d_1^2 \\ 1 & d_2 & d_2^2 \\ 1 & d_3 & d_3^2 \\ 1 & d_4 & d_4^2 \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix}. \tag{4.5}$$

□

Typically, n_{meas} is constrained by the resources available to conduct the experiments, but are chosen to at least satisfy $(n_{\mathbf{y}} n_{\text{meas}}) > n_{\boldsymbol{\theta}}$. Equation 4.3 can then be solved as a least-squares problem by forming its normal equations

$$\boldsymbol{\theta}_{lsq} = \left(\mathbf{F}^T\mathbf{F}\right)^{-1}\mathbf{F}^T\tilde{\mathbf{g}}, \tag{4.6}$$

where $\left(\mathbf{F}^T\mathbf{F}\right)^{-1}$ is called the information matrix. If the system were under-determined, then there is no unique solution. Physically, this means that not enough data is available to estimate the uncertain parameters — at best, only the relationships between some of the parameters may be estimated.

It can be shown that the covariance under random realizations of the noise $\boldsymbol{\epsilon}$ is

$$\text{Cov}\left(\boldsymbol{\theta}_{lsq}\right) = \sigma^2\left(\mathbf{F}^T\mathbf{F}\right)^{-1}, \tag{4.7}$$

where σ^2 is a positive constant that is independent of the design variable $\mathbf{d}$, and thus its exact value is irrelevant for the purpose of design optimization. A good design should make the covariance "small" in some sense, and Equation 4.7 hints that this can only be achieved by making $\mathbf{F}^T\mathbf{F}$ "large" in some sense.

A number of different ways have been proposed to "maximize" $\mathbf{F}^T\mathbf{F}$, giving the birth of the popular *alphabetical optimality criteria.* Here are some examples:

- A-optimality minimizes $\sum_i 1/\lambda_i$ (or maximizes $\mathrm{tr}\left(\mathbf{F}^T\mathbf{F}\right)$), where λ_i are the eigenvalues of $\mathbf{F}^T\mathbf{F}$;
- D-optimality minimizes $\prod_i 1/\lambda_i$ (or maximizes $\det(\mathbf{F}^T\mathbf{F})$);
- E-optimality minimizes $\max_i 1/\lambda_i$.

For nonlinear models, such closed-form analysis is typically not available. Various approximation methods have been proposed, such as locally linearizing the model, or approximating the posterior with different Gaussian distributions in which closed-form analysis may still be derived (e.g., [13]). However, a more general framework that is largely independent of the optimal linear design theory is usually preferred (e.g., [64]). This involves the development of a desired utility function to be optimized, whose value can often only be approximated through numerical techniques.

Applications of this optimal nonlinear experimental design framework are becoming more prevalent — for example, in the fields of astrophysics [54], material science (random fatigue model) [80], and geophysics (amplitude versus offset) [96]. However, the models used in these applications are all relatively simple to evaluate compared to the ODE system of the combustion problem described in Chapter 2. To date, no known literature has applied optimal nonlinear experimental design in such computationally expensive models, which, as evident by the result at the end of this chapter, becomes impractically expensive to perform.

Many algorithms have been developed to target at the acceleration of the design process. For example, Müller and Parmigiani [65] dedicated a paper discussing numerical techniques in estimating information theoretic measures. Clyde and Parmigiani [15] combined the design and parameter spaces into a so-called "augmented" space, which is then explored using a single MCMC process instead of having separate inference-related and optimization processes. This idea is further improved upon by Müller *et al.* [67] to introduce a simulated annealing type of update to the augmented space, thus improving the optimization performance of the process, while requiring inhomogeneous Markov chains. Müller and Parmigiani [66] also proposed a curve fitting scheme to the design space, which greatly accelerates the optimization procedure; however, the assumption on the fitting model can be arbitrary, especially for complicated models whose utility surface is not intuitive to expect.

The above methods mainly focus on the sampling and optimization techniques. However, as presented in the next chapter, this thesis tries to mitigate the computational cost through a different perspective, by using model reduction.

4.2 Optimal Nonlinear Experimental Design

4.2.1 Expected Utility

The combustion model in this thesis is certainly a highly nonlinear model. The approach taken to quantify the goodness of an experiment applies concepts from information theory. In particular, the goodness indicator will be based on the Kullback-Leibler (KL) divergence, whose usage in experimental design was first demonstrated by Lindley [52]. The relevant concepts from information theory are reviewed below.

The entropy (also known as the Shannon entropy or self-information) of a discrete random variable θ is defined to be

$$H(\theta) = -\sum_{\Theta} \mathbb{P}(\theta) \log \left[\mathbb{P}(\theta)\right]. \tag{4.8}$$

The entropy is always non-negative, and it can be loosely interpreted as an indication of the disorder (or lack of information) carried by the probability mass function (PMF) of θ. If $\log_2$ is used, it can be physically interpreted as the number of bits (say on a computer) required to describe the random variable. Without loss of generality, the natural log, ln, shall be used in all subsequent analysis.

The continuous random variable counterpart, the differential entropy, is defined to be

$$h(\theta) = -\int_{\Theta} p(\theta)\ln[p(\theta)]\, d\theta. \tag{4.9}$$

The interpretation of bits- (or nats-) representation is no longer suitable, as h may be negative (while H is always non-negative). However, one can still use this quantity as an indicator of the state of disorder (or lack of information) carried by the random variable, the very reason why h (or H) is known as the self-information.

To compare the amount of information between two PDFs, it is natural to use the relative entropy (also known as the Kullback-Leibler divergence or simply, information divergence), defined as

$$D\left(p_A(\theta)\,||p_B(\theta)\right) = \int_{\Theta} p_A(\theta)\ln\left[\frac{p_A(\theta)}{p_B(\theta)}\right] d\theta. \tag{4.10}$$

The KL divergence is always non-negative, but it is not strictly a distance metric because it is not symmetric (i.e., $D\left(p_A(\theta)\,||p_B(\theta_B)\right) \neq D\left(p_B(\theta)\,||p_A(\theta)\right)$). Nonetheless, it is still a useful indicator quantity, as it reflects the difference in information carried by the two PDFs. $D\left(p_A(\theta)\,||p_B(\theta)\right)$ is thus called the information gain about θ if $p_A(\theta)$ is used instead of $p_B(\theta)$. For example, placed in the Bayesian inference context, the mutual information between $\boldsymbol{\theta}$ and $\mathbf{y}$ is

$$I(\boldsymbol{\theta};\mathbf{y}) \equiv h(\boldsymbol{\theta}) - h(\boldsymbol{\theta}|\mathbf{y}) = D\left(p(\boldsymbol{\theta},\mathbf{y})\,||p(\boldsymbol{\theta})\,p(\mathbf{y})\right) \geq 0. \tag{4.11}$$

This implies that, although differential entropy may be positive or negative, obtaining additional data (i.e., conditioning on $\mathbf{y}$) can never increase the differential entropy (or decrease information). Information theory is a vast subject area, and the two simple concepts used in this framework are merely the tip of the iceberg. Curious readers who wish to see more detailed discussions on information theory are referred to [17, 55].

The KL divergence between the posterior and the prior is adopted as the utility u

to reflect the goodness of an experiment:

$$u(\mathbf{d}, \mathbf{y}) \equiv D\left(p\left(\boldsymbol{\theta}|\, \mathbf{y}, \mathbf{d}\right) || p\left(\boldsymbol{\theta}\right)\right) = \int_{\boldsymbol{\Theta}} p\left(\boldsymbol{\theta}|\, \mathbf{y}, \mathbf{d}\right) \ln \left[\frac{p\left(\boldsymbol{\theta}|\, \mathbf{y}, \mathbf{d}\right)}{p\left(\boldsymbol{\theta}\right)}\right] d\boldsymbol{\theta}. \tag{4.12}$$

The intuition is that a large KL divergence implies the data $\mathbf{y}$ have decreased the disorder (or increased information) of the belief of $\boldsymbol{\theta}$ by a large amount, and hence those data are more informative with respect to the experimental goal of inference. There exist other similar quantities, such as the broader family of f-divergences, that may be used as the utility. However, the KL divergence is chosen due to its strong foundation in information theory, and the fact it reduces to the D-optimality condition when applied to a linear design problem.

Equation 4.12 cannot be used directly, because it is a function of the data $\mathbf{y}$, which of course, are unknown when designing the experiment. Thus, an expectation is taken over $\mathbf{y}|\mathbf{d}$ to obtain the expected utility function, which is simply the utility averaged over all possible data realizations under a particular $\mathbf{d}$:

$$\boxed{U(\mathbf{d}) = \mathbb{E}_{\mathbf{y}|\mathbf{d}}\left[u(\mathbf{d}, \mathbf{y})\right] = \int_{\mathcal{Y}} \int_{\boldsymbol{\Theta}} p(\boldsymbol{\theta}|\mathbf{y}, \mathbf{d}) \ln \left[\frac{p(\boldsymbol{\theta}|\, \mathbf{y}, \mathbf{d})}{p(\boldsymbol{\theta})}\right] d\boldsymbol{\theta}\, p\left(\mathbf{y}|\, \mathbf{d}\right)\, d\mathbf{y},} \tag{4.13}$$

where $\mathcal{Y}$ is the support of $p\left(\mathbf{y}|\, \mathbf{d}\right)$. Finally, the expected utility needs to be maximized to find the optimal experimental design

$$\boxed{\mathbf{d}^* = \arg \max_{\mathbf{d} \in \mathcal{D}} U(\mathbf{d}),} \tag{4.14}$$

where $\mathcal{D}$ is the design space.

What if resources allow multiple n_{designs} designs to be carried out simultaneously? The answer is not simply to do all the experiments at $\mathbf{d}^*$, as that in general does not yield the optimal total information gain from all the experiments. Instead, the multiple experiments should be incorporated into the likelihood function, where now $\mathbf{d} \in \mathbb{R}^{n_{\text{designs}} n_{\mathbf{d}}}$ and $\mathbf{y}_i \in \mathbb{R}^{n_{\text{designs}} n_{\mathbf{y}}}$, and the data from the different experiments can be, for example, reasonably assumed to be conditionally independent given $\boldsymbol{\theta}$. The new optimal $\mathbf{d}^* \in \mathbb{R}^{n_{\text{designs}} n_{\mathbf{y}}}$ then carries all the n_{designs} design conditions which optimize the total information gain when simultaneously performed. One popular utility function is to use the predictive variance, mainly due to its ease of numerical estimation — but this utility is unable to perform multi-experimental design optimization, and would naïvely suggest to repeat all experiments at the single-experimental design optimal.

If the n_{designs} designs do not need to be carried out simultaneously, this gives rise to sequential experimental design. The concept is straightforward. A single optimal experimental design is initially computed and carried out, and the data are used to perform the inference. The resulting posterior is then used as the prior for designing the next experiment, and the process is repeated until all n_{designs} designs are performed. This sequential experimental design should be at least as good as when all the experiments are designed simultaneously, because of the extra information gained in the intermediate stages.

4.2.2 Numerical Methods

The expected utility almost always needs to be approximated numerically, due to the nonlinearity of the model. To do that, first, Equation 4.13 is rewritten as

$$\begin{aligned} U(\mathbf{d}) &= \int_{\mathcal{Y}} \int_{\boldsymbol{\Theta}} p(\boldsymbol{\theta}|\mathbf{y},\mathbf{d}) \ln \left[\frac{p(\boldsymbol{\theta}|\, \mathbf{y},\mathbf{d})}{p(\boldsymbol{\theta})} \right] d\boldsymbol{\theta}\, p(\mathbf{y}|\, \mathbf{d})\; d\mathbf{y}, \\ &= \int_{\mathcal{Y}} \int_{\boldsymbol{\Theta}} \ln \left[\frac{p(\mathbf{y}|\, \boldsymbol{\theta},\mathbf{d})}{p(\mathbf{y}|\, \mathbf{d})} \right] p(\mathbf{y}|\, \boldsymbol{\theta},\mathbf{d})\, p(\boldsymbol{\theta})\; d\boldsymbol{\theta}\, d\mathbf{y} \\ &= \int_{\mathcal{Y}} \int_{\boldsymbol{\Theta}} \{\ln [p(\mathbf{y}|\, \boldsymbol{\theta},\mathbf{d})] - \ln [p(\mathbf{y}|\, \mathbf{d})]\}\, p(\mathbf{y}|\, \boldsymbol{\theta},\mathbf{d})\, p(\boldsymbol{\theta})\; d\boldsymbol{\theta}\, d\mathbf{y}, \end{aligned} \tag{4.15}$$

where the second equality is due to the application of Bayes' theorem to the quantities both inside and outside the ln. Monte Carlo (MC) integration is then used to approximate the integral

$$U(\mathbf{d}) \approx \frac{1}{n_{\text{out}}} \sum_{i=1}^{n_{\text{out}}} \left\{ \ln \left[\mathbf{p} \left(\mathbf{y}^{(i)}|\, \boldsymbol{\theta}^{(i)}, \mathbf{d} \right) \right] - \ln \left[p\left(\mathbf{y}^{(i)}|\, \mathbf{d} \right) \right] \right\}, \tag{4.16}$$

where $\boldsymbol{\theta}^{(i)}$ are drawn from the prior $p(\boldsymbol{\theta})$; $\mathbf{y}^{(i)}$ are drawn from the likelihood $p\left(\mathbf{y}|\,\boldsymbol{\theta}=\boldsymbol{\theta}^{(i)}\right)$ given the $\boldsymbol{\theta}^{(i)}$ just sampled; and n_{out} is the number of samples in this "outer" MC approximation. The evidence, $p\left(\mathbf{y}^{(i)}|\,\mathbf{d}\right)$ is typically unknown, and an additional "inner" MC approximation needs to be made to approximate it:

$$\begin{aligned} p\left(\mathbf{y}^{(i)}|\,\mathbf{d}\right) &= \int_{\boldsymbol{\Theta}} p\left(\mathbf{y}^{(i)}|\,\boldsymbol{\theta},\mathbf{d}\right)p(\boldsymbol{\theta})\,d\boldsymbol{\theta} \\ &\approx \hat{p}\left(\mathbf{y}^{(i)}|\,\mathbf{d}\right)=\frac{1}{n_{\text{in}}}\sum_{j=1}^{n_{\text{in}}}p\left(\mathbf{y}^{(i)}|\,\boldsymbol{\theta}^{(i,j)},\mathbf{d}\right), \end{aligned} \tag{4.17}$$

where $\boldsymbol{\theta}^{(\cdot,j)}$ are drawn from the prior $p(\boldsymbol{\theta})$, and n_{in} is the number of samples in this "inner" MC approximation. As shown by Ryan [80], this MC estimator is biased. The selection of n_{out} and n_{in} provides a tradeoff between estimator bias and variance.

This double-nested MC estimation can quickly become enormously expensive. Some measures can be taken to limit the number of forward models evaluations required, by reusing samples between the outer and inner MC approximations, and across the outer MC approximation. In total, three different sampling techniques are proposed.

1. Let $n_{\text{out}}=n_{\text{in}}$, and use the same batch of $\boldsymbol{\theta}^{(i)}$, $i=1,\cdots,n_{\text{out}}$ samples for every outer MC approximation, as well as inner MC approximations for every $\mathbf{y}^{(i)}$.

2. Draw a fresh batch of n_{in} samples for every $\mathbf{y}^{(i)}$ in the inner MC approximation, but these samples are then carried over to be used for every outer MC approximation (i.e., for different $\mathbf{d}$'s).

3. Every single sample of $\boldsymbol{\theta}$ is sampled independently — this is the truly independent sampling.

The computational cost is $\mathcal{O}(n_{\text{out}})$ for the first sampling technique, and $\mathcal{O}(n_{\text{out}}n_{\text{in}})$ for the second and third sampling techniques. There is no computational advantage to choose option two over three. However, since only the difference between expected utility values at different designs is important, using the same sets of samples for the outer MC approximation (i.e., different $\mathbf{d}$'s) can help reduce the variance of the difference estimate. This is known as the variance reduction technique of common random numbers (e.g., [23, 32]). Reusing samples may potentially add bias, but as observed from the numerical example results in Section 4.2.4, the effect is very small.

4.2.3 Stochastic Optimization

One last numerical tool needed for the experimental design process is an optimization algorithm for Equation 4.14. More specifically, since only the MC approximation to the expected utility (which is the objective function) is available, the optimization method needs to be able to handle noisy objective functions — this topic is known as stochastic optimization (not to be confused with optimization method that uses random sampling techniques). Unfortunately, optimization is not yet implemented in the current code, pending further research and testing. Nonetheless, some discussion is given below. A more thorough discussion can be found in [91].

The simplest method of stochastic optimization is to use a large number of MC samples at each $\mathbf{d}$ to obtain a good estimate of the objective function, and then optimize according to any deterministic optimization algorithm. However, this is extremely expensive to do. A broad family of methods under the name of Stochastic Approximation (SA) are specially designed for noisy function optimization, such that the objective function needs not be well approximated. Two well known such algorithms are the Robbins-Monro [77] and Kiefer-Wolfowitz [47] algorithms. The former constructs a sequence that successively reduces the difference between the measured function value and the true optimum value, which is assumed to be known *a priori*. The latter uses finite-difference approximation to the gradient to aid the process.

Simultaneous Perturbation Stochastic Approximation (SPSA), developed by Spall [89, 90], is another SA method that has recently received considerable attention [87]. The method is similar to a steepest-descent method that uses centered-difference estimation of the gradient, except that SPSA only needs a total of two random perturbations to estimate the gradient regardless of the problem's dimension. The intuition justification is that the error from misdirection would average out over large number of iterations [90]. Further research in SPSA has incorporated the use of Hessian (approximated using only a total of two additional perturbations) [88], and common random numbers [48]. The inherent randomness in the objective function and finite-difference perturbations also allow a global optimal convergence property [56].

Optimization capability is not implemented in the current version of the code. Further research and testing is required to find the most suitable algorithm for the combustion experimental design problem, as there is no single algorithm that is superior for all problems (the so-called No Free Lunch theorem [101]). For now, the optimization is only done through a grid-search (by computing the objective function estimates in a tensor-product grid spanning the entire design space), but a reliable optimization method is absolutely essential when the design space is of high dimensions.

4.2.4 Numerical Example

The optimal nonlinear experimental design framework developed in this chapter is applied to a simple, 1D nonlinear model. Consider

$$y = g\left(\theta, d\right) + \epsilon = \theta^3 d^2 + \epsilon, \tag{4.18}$$

where $\epsilon \sim \mathcal{N}\left(0, \sigma_\epsilon^2\right)$. Let the prior be $\theta \sim \mathcal{U}\left(-1, 1\right)$, and the design space be limited to $d \in [-1, 1]$.

The first analysis explores the three proposed sampling techniques from the end of Section 4.2.2, using $n_{\text{out}} = 1001$ and $n_{\text{in}} = 1001$, with $\sigma_\epsilon^2 = 10^{-2}$. The expected utility results are shown in Figure 4-1, plotted using a 101-node grid in the design space. A few observations can be made.

1. The expected utility is maximized at $d = -1$ and $d = 1$, and minimized at $d = 0$. This can be intuitively explained by examining Equation 4.18. If a d with a small magnitude is chosen, then the observation y would be dominated by the noise ϵ, which would not yield much useful information on what the uncertain parameter

θ might be. In the extreme case where $d = 0$, the measurement is purely noise, which would yield no useful information at all. On the other hand, if a d with a large magnitude is selected, such that the noise is insignificant compared to g, then y would be very informative in inferring the value of θ.

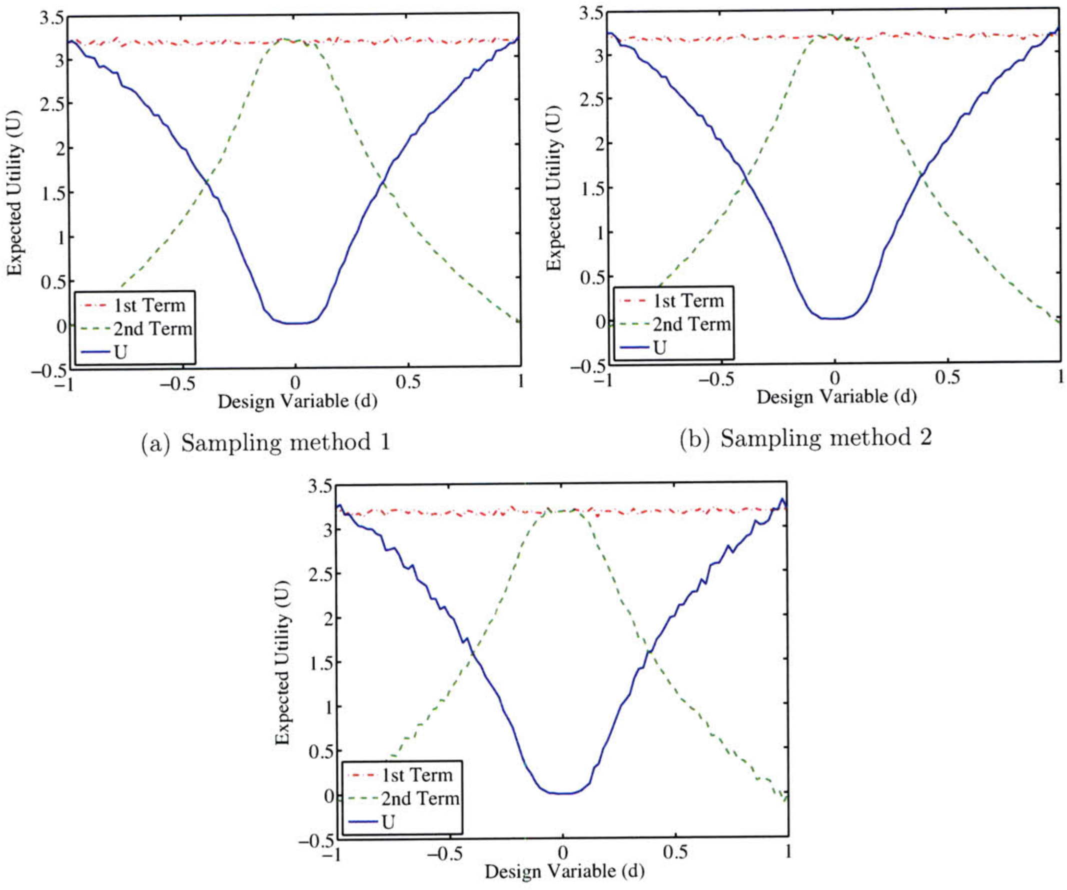
2. The three proposed sampling techniques have no significant visible difference. This suggests that the cheapest method (sampling method 1), by reusing the same samples for n_{out} and n_{in}, might be sufficient.

3. The noise model chosen for this run is independent of d (fixed at $\sigma_\epsilon^2 = 10^{-2}$). Thus, the first term in Equation 4.16 is expected to be independent of d as well, and the optimization is simply a minimization of the second term, consistent with a theorem proposed by Sebastiani and Wynn [83]. This is indeed observed, as shown by the "1st Term" graph lines.

For the rest of this thesis, only the first sampling method is used unless otherwise specified.

The second analysis explores the effect of noise intensity. Figure 4-2 compares the expected utility using $\sigma_\epsilon^2 = 10^{-4}$, 10^{-2}, and 10^{0}. As the noise level is increased, the expected utility becomes flatter. This is intuitively pleasing, since a smaller noise level is expected to lead to better inference. In fact, the "widths" of the valleys in the plots should be proportional to $\sqrt{\sigma_\epsilon}$, which can be shown through an order-of-magnitude analysis. Roughly speaking, $\epsilon \sim \mathcal{O}(\sigma_\epsilon)$ and $\theta \sim \mathcal{O}(1)$, then in order for g to dominate, $d \sim \mathcal{O}\left(\sqrt{\sigma_\epsilon}\right)$ is needed.

An interesting phenomenon is observed when the second and third sampling techniques are used for the $\sigma_\epsilon^2 = 10^{-4}$ case (not shown), where the expected utility becomes infinity near large magnitudes of d. This is likely happening for the other two noise levels as well, except is at d values beyond the $[-1, 1]$ range. The finiteness for the first sampling method is due to the fact that at least one sample is guaranteed to have a "reasonably large" likelihood, since the $\theta^{(i,j)}$ samples being reused are exactly the ones used to generate $\mathbf{y}^{(i)}$; whereas in the other two sampling techniques, all the samples



(a) Sampling method 1

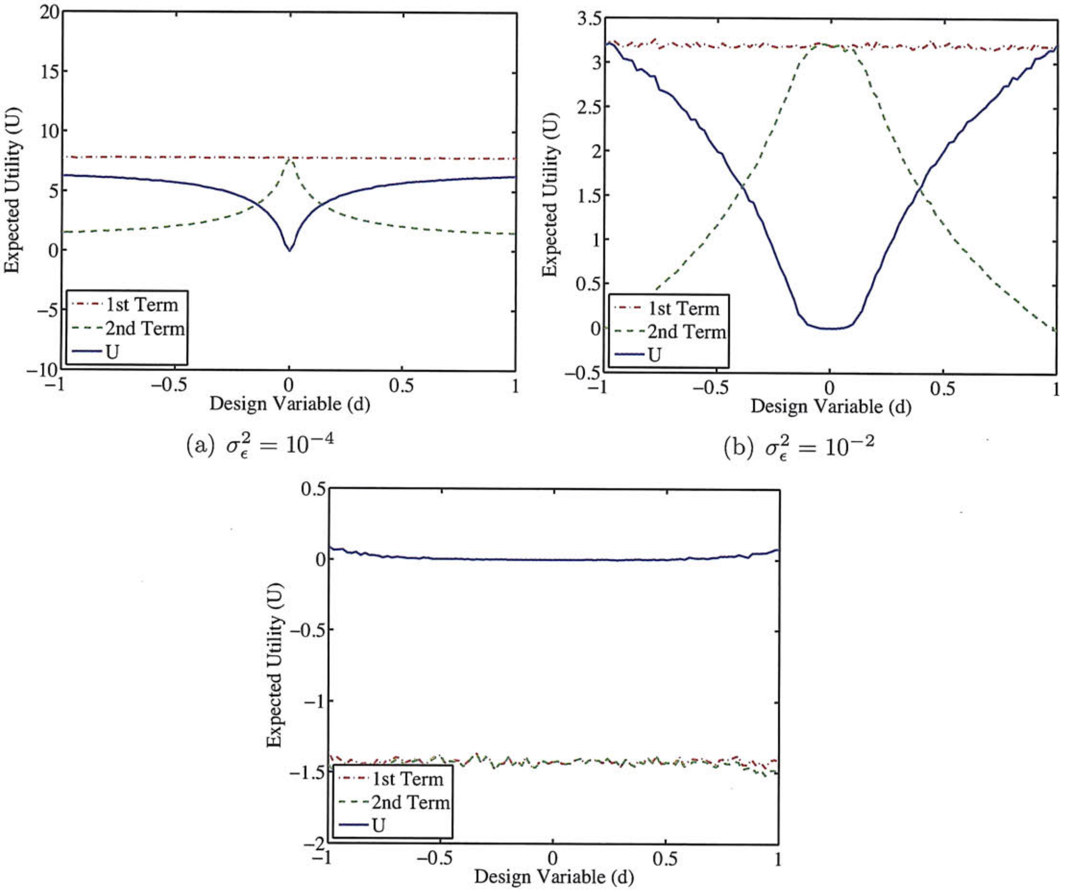
(b) Sampling method 2

(c) Sampling method 3

Figure 4-1: Expected utilities of the 1D nonlinear experimental design example with $\sigma_\epsilon^2 = 10^{-2}$, using different sampling techniques. The "1st Term" and "2nd Term" are the terms from Equation 4.16.

lead to near-zero likelihoods, causing the logarithm to blow up. This is a common problem in estimating low probability events using MC methods, as demonstrated by Example 4.2.1. In any case, the infinite values should not be of any trouble on a practical level, because they merely reflect the fact that at all those designs, very good inferences can be obtained.

Example 4.2.1. Monte Carlo Estimation of Low Probability Event



(a) $\sigma_\epsilon^2 = 10^{-4}$

(b) $\sigma_\epsilon^2 = 10^{-2}$

(c) $\sigma_\epsilon^2 = 10^{0}$

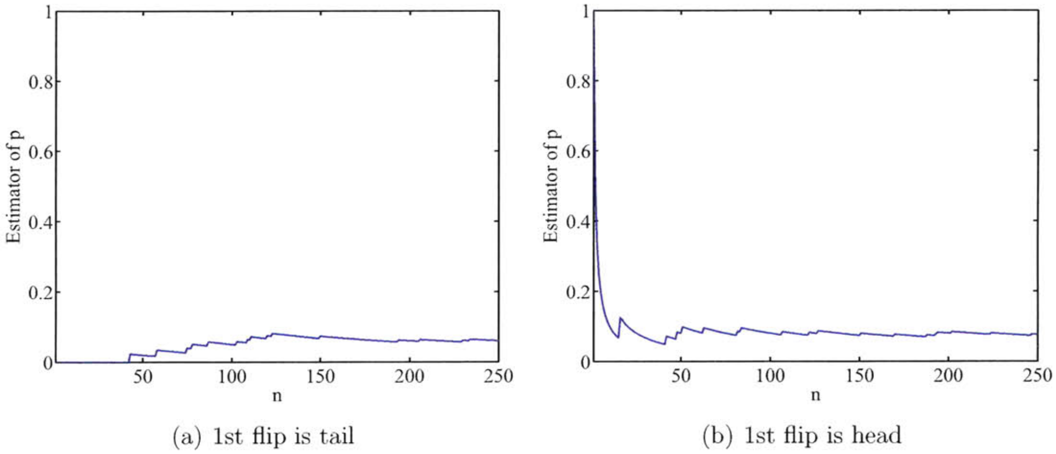
Figure 4-2: Expected utilities of the 1D nonlinear experimental design example using $\sigma_\epsilon^2 = 10^{-4}$, 10^{-2}, and 10^0.

A biased coin has a probability $p = 0.05$ of yielding heads. An MC estimator

$$\hat{p} = \frac{1}{n}\sum_{i=1}^{n} \mathbb{I}_{i\text{th flip} = \text{heads}} \tag{4.19}$$

is used, where $\mathbb{I}_{i\text{th flip} = \text{heads}}$ is the indicator function for the ith flip being heads. The estimator as a function of n is shown in Figure 4-3(a). $\hat{p}$ remains zero (and $\ln \hat{p} = -\infty$) for the first 40 flips, and only becomes non-zero after the first head occurs after a large sample size is used, and then slowly converge towards 0.05 (analogous to the second and third sampling techniques, which result in infinite expected utility values).

On the other hand, if the samples used are such that the first flip happens to be head, then $\hat{p} > 0$ (or $\ln \hat{p} > -\infty$) for all n, and the estimate still converges to 0.05 as n increases (analogous to the first sampling technique, which guaranteed finite expected utility values). This is shown in Figure 4-3(b). □



(a) 1st flip is tail

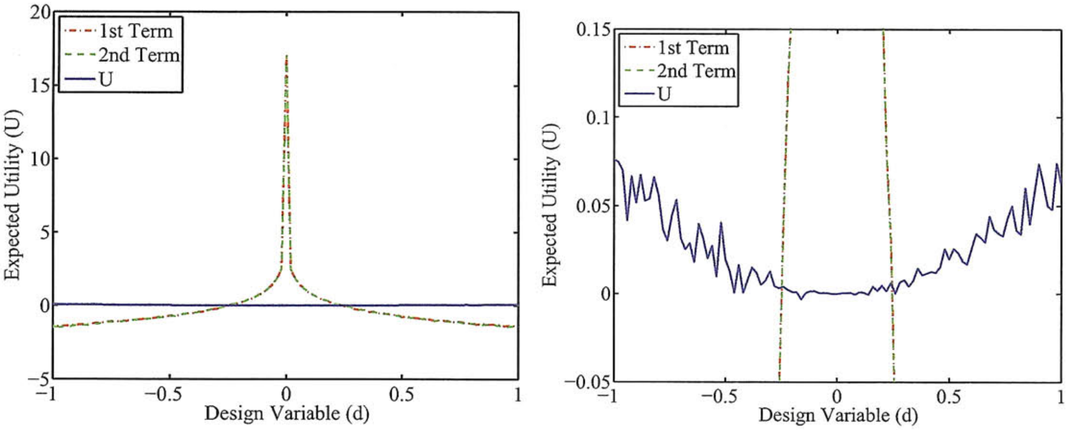
(b) 1st flip is head

Figure 4-3: $\hat{p}$ as a function n for Example 4.2.1. Figure (a) is where the first flip is tail, while Figure (b) is when the first flip is happens to be head.

The third analysis tests a design-dependent noise level, $\sigma_\epsilon^2 = |d| + 10^{-8}$. The expected utility is shown in Figure 4-4. As expected, the first term from Equation 4.16 is no longer d-independent. The expected utility is quite flat, since although a large magnitude d makes g large, it also brings a large noise level as well. In more complicated models and noises, the tradeoffs would not be very easy to determine only from analyzing the equations; however, the numerical framework would still be able to capture the relationships with ease.

4.3 Results and Issues

The nonlinear optimal experimental design framework is applied to the combustion problem with its underlying inference goal. First, only the initial temperature is allowed to vary in the design space $T_0 \in [900, 1050]$. The MC sampling size is chosen



(a) Full view

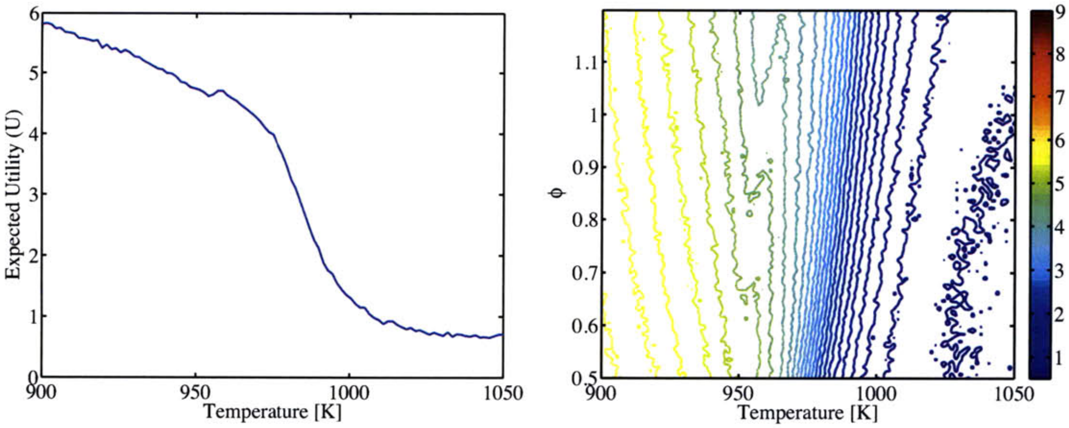
(b) Zoomed in

Figure 4-4: Expected utilities of the 1D nonlinear experimental design example using a design-dependent noise level $\sigma_\epsilon^2 = |d| + 10^{-8}$.

to be $n_{\text{out}} = n_{\text{in}} = 1001$ (recall that only the first sampling technique is used). Second, the design space is extended to two dimensions, now including the equivalence ratio $\phi \in [0.5, 1.2]$ as well. The same number of MC samples is used. The expected utilities are plotted on a 1001-node discretization in each dimension, shown in Figure 4-5 for both cases.

The optimal design appears to be at a low initial temperature and low equivalence ratio. One possible explanation for choosing a low temperature design is that the ignition time scales are much larger at low temperatures (because reactions occur more slowly at low temperatures), but the assumed error model decreases at a fast rate (recall assumed affine error model in Equation 3.4) such that the error variance becomes very small compared to the characteristic time values. However, this may be only one of the numerous factors that dictate the highly nonlinear relationship between the design condition and the expected utility, which is difficult to assess using only physical insights.

The computational resources required for these two tasks are enormous. The 1D design case requires hours to run when parallelized on about 50 3GHz CPUs; the 2D design case requires days to complete under the same parallelization. Considering this



(a) 1D design (T_0)

(b) 2D design (T_0 and ϕ)

Figure 4-5: Expected utilities of the combustion problem with 1 design variable T_0, and 2 design variables T_0 and ϕ.

is only a relatively simple 19-reaction mechanism, substantial speed-ups are needed in order for the framework to be practical. One possible solution is to use model reduction to find a computationally cheaper surrogate for the Cantera output $\mathbf{g}(\boldsymbol{\theta}, \mathbf{d})$ in Equation 3.2. This is explored in the next chapter.

Chapter 5

Polynomial Chaos

At the end of the previous chapter, the nonlinear optimal experimental design framework is applied to the combustion problem, in an attempt to determine the optimal design conditions whose results allow one to best infer the uncertain kinetic parameters. However, the Monte Carlo nature of the estimate of the objective function renders the computation unbearably expensive. If this framework were to have any applicability in more complicated real world experiments, it must be made orders of magnitude faster. One way to alleviate this problem is to perform model reduction. The model reduction method selected in this thesis is polynomial chaos (PC), a particular spectral expansion of random variables.

This chapter is outlined as follows. Section 5.1 provides some background information on model reduction methods and the development of PC, while Section 5.2 provides the mathematical formulation of PC. The non-intrusive spectral projection (NISP) method for computing the expansion coefficients is presented in Section 5.3, which motivates the need of high dimensional numerical integration, discussed in Section 5.4. Upon the appropriate selection of the integration method, a slight modification is made as it is incorporated into the NISP framework in Section 5.5. Finally, an algorithm is proposed in Section 5.6 to detect dimension anisotropy of the polynomial equivalent of the integrand, making the PC computation more efficient.

5.1 Background

5.1.1 Model Reduction

There are several ways to perform model reduction for a chemical system, some of which are discussed below.

- A less detailed chemical mechanism may be adopted, but it often has very limited regimes of applicability (e.g., only suitable for a particular range of T). In addition to its inability to capture detailed intermediate reactions (discussed in Chapter 2), it also does not take into account the uncertainty in those parameters which are eliminated as a result of the mechanism simplification.

- Adaptive chemistry (e.g., [81]) is an application of multiple levels of chemical mechanisms in reacting flows. The mechanisms are constructed *a priori* and stored, and the flow regime is divided into finite volumes, each of which will then adopt a suitable level of mechanism according to some adaptation scheme. As a result, the overall computation is made more efficient. The drawback of this method lies within its arbitrariness in forming the different levels of mechanisms and the adaptation scheme. Moreover, this method is not directly applicable to the zero-dimensional combustion problem considered in this thesis.

- Computational singular perturbation (CSP) (e.g., [49]) categorizes all the reactions into fast and slow groups, where the fast ones are solved as steady-state problems while the slow ones as time-dependent (ODE) ones. This effectively reduces the stiffness of the problem. In the inference framework, the reactions relevant to the uncertain kinetic parameters must be in the ODE set, for else any change to the kinetic parameters would have no effect if the relevant reactions are assumed to have reached equilibrium. CSP can be quite expensive if refinement is used to categorize the reactions. *A priori* knowledge may be incorporated, but this would be difficult to implement as the timescale of some reactions can change substantially with the kinetic parameters. Tabulating such prior information can quickly become infeasible especially if the dimension of the parameter space is large.

- Proper orthogonal decomposition (POD) (e.g., [14]) is similar to a low-rank approximation in singular value decomposition (SVD). The main drawback of POD is that it often eliminates the concentrations of minor species, which are almost always the driving forces of the reactions. One can enforce the important species to be always included, but that requires prior knowledge and can cause the POD to yield less computational savings. The difficulty is further compounded by the fact that the uncertain parameters are variables, and thus the number of snapshots required can grow exponentially with the parameter space dimension. For these reasons, POD is rarely found in chemistry applications.

The above model reduction methods are not used due to the shortcomings discussed. Instead, this thesis shall use a reduction method that is based on the spectral expansion of random variables, namely, the PC expansions. The main difference between PC and the other model reduction methods is that, while PC focuses on capturing the functional relationships between the uncertain input parameters and the model outputs by taking advantage of any regularity in these functions, the other reduction methods attempt to reduce the size and complexity of the model itself.

5.1.2 Polynomial Chaos

The idea of PC was first developed by Wiener [100]. Motivated by physical problems in statistical mechanics, he developed the homogeneous chaos expansion (also known as the Hermite chaos expansion), which uses Hermite polynomials in the random space as the trial basis. Since then, considerable research has taken place in the theoretical development of PC. For example, the PC formulation has been generalized to the Askey family of orthogonal polynomials basis (also known as the generalized polynomial chaos, or gPC) [104], and even to multiwavelet basis [51]; various techniques to perform arithmetics between PC expansions have also been developed [20]; and the use of PC to accelerate solving Bayesian inference problems has been extensive and successful [57–59].

One of the main attractions of PC is its ability to capture the potentially complicated probabilistic features of random variables (thus suitable for uncertainty quantification) while being computationally cheap to evaluation. It thus has received considerable attention in quantifying uncertainty in a wide range of engineering applications, such as Computational Fluid Dynamics (CFD) [40], reacting flow [76], geometric uncertainty in fluid flow [97], and heat conduction [105]. An introductory review can be found in [69], a comprehensive summary of the numerical techniques and implementation is presented in [102], and a thorough discussion on spectral methods for uncertainty quantification in general is provided in [50].

5.2 Formulation

Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space, where Ω is the sample space, $\mathcal{F}$ is the σ-field, and $\mathbb{P}$ is the probability measure. Then any (scalar) random variable $\theta : \Omega \to \mathbb{R}$, $\theta \in L_2(\Omega, \mathbb{P})$ (i.e., θ having finite variance) can be expressed as the following expansion

$$\begin{aligned}
\theta(\omega) &= a_0\Gamma_0 \\
&+ \sum_{i_1=1}^{\infty} a_{i_1}\Gamma_1(\xi_{i_1}(\omega)) \\
&+ \sum_{i_1=1}^{\infty}\sum_{i_2=1}^{i_1} a_{i_1 i_2}\Gamma_2(\xi_{i_1}(\omega), \xi_{i_2}(\omega)) \\
&+ \sum_{i_1=1}^{\infty}\sum_{i_2=1}^{i_1}\sum_{i_3=1}^{i_2} a_{i_1 i_2 i_3}\Gamma_3(\xi_{i_1}(\omega), \xi_{i_2}(\omega), \xi_{i_3}(\omega)) \\
&+ \cdots,
\end{aligned} \tag{5.1}$$

where $\omega \in \Omega$ is an element of the sample space, ξ_i are i.i.d. $\mathcal{N}(0, 1)$, Γ_i are known as the Polynomial Chaos of order i, and a_i are the expansion coefficients. This expansion is known as the polynomial chaos expansion, and is convergent in the mean-square sense [12].

Equation 5.1 is equivalent to the form

$$\theta(\omega) = \sum_{|\mathbf{i}|=0}^{\infty} \theta_{\mathbf{i}} \Psi_{\mathbf{i}}(\xi_1, \xi_2, \cdots) \tag{5.2}$$

where $\mathbf{i} = (i_1, i_2, \cdots), \forall i_j \in \mathbb{N}$, is an infinite-dimensional multi-index, $|\mathbf{i}| = i_1 + i_2 + \cdots$ is the L_1 norm, $\theta_{\mathbf{i}}$ are the expansions coefficients, and

$$\Psi_{\mathbf{i}}(\xi_1, \xi_2, \cdots) = \prod_{j=1}^{\infty} \psi_{i_j}(\xi_j). \tag{5.3}$$

The basis functions ψ_{i_j} are orthogonal polynomials of order i_j in the independent variable ξ_j, with respect to the PDF of ξ_j (i.e., $p(\xi_j)$). More specifically,

$$\mathbb{E}_{\xi}\left[\psi_m \psi_n\right] = \int_{\Xi} \psi_m(\xi)\, \psi_n(\xi)\, p(\xi)\; d\xi = \delta_{m,n} \mathbb{E}_{\xi}\left[\psi_m^2\right], \tag{5.4}$$

where Ξ is the support of $p(\xi)$. The form in Equation 5.2 is often preferred to Equation 5.1 as it facilitates the manipulation of PC expansions. For computational purposes, the infinite sum and infinite dimension must be truncated to some finite order p_θ and stochastic dimension n_s, leading to

$$\begin{aligned} \theta(\omega) &\approx \sum_{|\mathbf{i}|=0}^{p_\theta} \theta_{\mathbf{i}} \Psi_{\mathbf{i}}(\xi_1, \xi_2, \cdots, \xi_{n_s}) && (5.5) \\ \Psi_{\mathbf{i}}(\xi_1, \xi_2, \cdots, \xi_{n_s}) &= \prod_{j=1}^{n_s} \psi_{i_j}(\xi_j). && (5.6) \end{aligned}$$

The total number of terms in the expansion is thus

$$n_{PC} = \begin{pmatrix} n_s + p_\theta \\ p_\theta \end{pmatrix} = \frac{(n_s + p_\theta)!}{n_s! p_\theta!}. \tag{5.7}$$

The choice of p_θ is often influenced by the expected smoothness of the random variable, and the choice of n_s is dependent on the expected degrees of freedom (DOFs) to capture the stochasticity of the system. For example, in a Bayesian inference context, n_s is typically equal to $n_{\boldsymbol{\theta}}$. However, there are special cases — a χ^2-distribution can be easily represented by multiple independent Gaussian distributions, and a stochastic process may be represented by a finite number of modes after discretizing and performing the Karhunen-Loève (KL) expansion. Finally, the choice of p_θ and n_s are also constrained by the computational resource available, as n_{PC} grows very fast when these parameters are increased.

For the original Hermite chaos expansion, ξ_j are i.i.d. $\mathcal{N}(0,1)$, and $\psi_{i_j}(\xi_j)$ are Hermite polynomials in ξ_j. Xiu and Karniadakis [104] made generalizations such that $\psi_{i_j}(\xi_j)$ can be any orthogonal polynomials from the Askey family (collectively known as gPC). The PDF of ξ_j would then also be changed correspondingly in order to maintain the orthogonality property in Equation 5.4. Some examples are shown in Table 5.1, reproduced from [104]. The choices are typically made such that the PDFs of the known random variables can be conveniently expressed. For example, in Section 3.2, uniform prior is used for $\tilde{A}_1$ in the combustion problem, then $\xi_1 \sim \mathcal{U}(-1,1)$ and $\psi_{i_1}(\xi_1)$ of Legendre polynomials would be a good choice, as only an affine transformation on ξ_1 is required to capture the PDF of $\tilde{A}_1$ exactly. It is also possible to use, for example, an expansion of $\xi_1 \sim \mathcal{N}(0,1)$ to capture the uniform prior, but the (finite expansion) representation would never be exact, and a high order would be required to achieve reasonably low errors.

The methods to compute the PC expansion coefficients are broadly divided into two groups — intrusive and non-intrusive.

In the intrusive approach, the PC expansions of both the known and unknown random variables are directly substituted into the governing equations. This results in a larger system than the deterministic problem, with the unknowns being the PC coefficients of the unknown random variables. The advantage of this approach is that the new system, although larger and different from the deterministic case, needs only to be solved once. However, because the system is modified, the solver of the

	Random Variables	Wiener-Askey chaos	Support
Continuous	Gaussian	Hermite-chaos	$(-\infty,\infty)$
	Gamma	Laguerre-chaos	$[0,\infty)$
	Beta	Jacobi-chaos	$[a,b]$
	Uniform	Legendre-chaos	$[a,b]$
Discrete	Poisson	Charlier-chaos	$\{0,1,2,\cdots\}$
	Binomial	Krawtchouk-chaos	$\{0,1,\cdots,N\}$
	Negative binomial	Meixner-chaos	$\{0,1,2,\cdots\}$
	Hypergeometric	Hahn-chaos	$\{0,1,\cdots,N\}$

Table 5.1: The Wiener-Askey polynomial chaos and their underlying random variables [104].

deterministic system is no longer suitable, and needs to be modified as well. This task can be difficult in complicated systems, and even impossible if, for example, the source code for the deterministic case cannot be accessed. Another issue is how to express the manipulation of PC expansions in a single expansion. For example, multiplying two expansions require the computation of a third-order tensor

$$C_{lmn} = \mathbb{E}_{\xi}\left[\psi_l \psi_m \psi_n\right], \tag{5.8}$$

often done *a priori* and stored. However, more complicated operations cannot be easily handled, and other techniques, such as Taylor expansions, integration approaches, and sampling approaches, must to be used [20].

The non-intrusive approach involves computing the expansion coefficients by directly projecting the unknown random variable onto the basis functions. The main advantage of this method is that the solver for the deterministic case can be reused, and treated as a black box, not requiring any modifications. The main disadvantage is that the deterministic problem needs to be solved many times.

For the combustion problem in this thesis, the non-intrusive spectral projection (NISP), which is a type of non-intrusive methods, is chosen (detailed formulation is shown in Section 5.3). Its selection is mainly due to the flexibility it offers in choosing the observables in the combustion problem context. For example, consider the ignition delay τ_{ign}, which is a post-processing quantity that is not directly in the governing equations. If an intrusive approach were adopted, then new equations need to be invented and appended to the governing equations in order to compute τ_{ign} from the state variables. Consequently, PC expansions need to be constructed for each of the states at each time integration point — the result is often an intractably large system, with much more information than simply τ_{ign}. PC expansions for the state variables at different time discretizations are not only computationally expensive to construct, they are also almost certainly poor approximations, because most of the state variables would be multi-modal. To illustrate this, consider $T(t^*)$ as a function of the kinetic parameters. For certain (regions of) combinations of the parameters, the system would have ignited by the time t^* (and $T(t^*)$ would be close to the equilibrium temperature), while for others, the system would not have ignited by the time t^* (and $T(t^*)$ would be close to the initial temperature). Intermediate temperatures are very unlikely to be obtained due to the extremely small timescale of the ignition. As a result, a bifurcation exists in $T(t^*)$ as a function of the kinetic parameters, and it would be very difficult to capture using a polynomial basis.

5.3 Non-Intrusive Spectral Projection (NISP) for the Combustion Problem

In this section, the application of NISP to the combustion experimental design problem is discussed in detail. The known random variables are the input parameters, namely, $\boldsymbol{\theta} = \left(\tilde{A}_1, E_{a,3}\right)$. As discussed in Section 3.2, a uniform prior has been assigned, with the support described in Table 3.1. Therefore, the Legendre Chaos is a convenient choice according to Table 5.1, with ξ_1, ξ_2 i.i.d. $\mathcal{U}(-1,1)$. More specifically, the PC expansions for these parameters are simply

$$\theta_1 = 0.05\xi_1 \tag{5.9}$$

$$\theta_2 = 1.3598 \times 10^7 + 1.3598 \times 10^7 \xi_2. \tag{5.10}$$

The goal now is to construct the PC expansions for the unknown parameters, which are the observables from the model output (not the measured data $\mathbf{y}$), namely, $\mathbf{g}(\boldsymbol{\theta}, \mathbf{d}) = \left(\ln \tau_{ign}, \ln \tau_O, \ln \tau_H, \ln \tau_{HO_2}, \ln \tau_{H_2O_2}, \frac{dh}{dt}\big|_\tau, \chi_{O,\tau}, \chi_{H,\tau}, \chi_{HO_2,\tau}, \chi_{H_2O_2,\tau}\right)$. As discussed at the end of Section 2.4, ln of the characteristic time observables will be used in the PC expansions. Before this is done, the dependence on the design conditions $\mathbf{d}$ needs to be addressed.

If $\mathbf{d}$ is restricted to some finite set of possible design conditions, then it would be trivial to construct the PC expansions simply under those different realizations. However, the approach taken does not assume this constraint — instead, $\mathbf{d}$ is assumed to be a continuous variable. One possible method to accommodate this is to lay out a grid in the design space $\mathcal{D}$, construct a PC expansion at each of the grid points, and perform interpolation of the coefficients for off-grid design conditions. This method can become very computationally expensive, since numerous PC expansions need to be constructed; in fact, storage of the expansion coefficients can become intractable as the number of grid points increases exponentially with the stochastic dimension.

The approaches in the previous paragraph involve constructing numerous PC expansions in only 2 variables (since $\dim(\boldsymbol{\theta}) = 2$). Alternatively, one can treat $\mathbf{d}$ as (deterministic) "random variables" as well, thus effectively increasing the stochastic dimension, but now only need to solve for one single PC representation. Denoting $\xi_3 = T_0$ and $\xi_4 = \phi$, the new stochastic dimension is $n_s = 4$. For the sake of demonstration, uniform "priors" are assigned to $\mathbf{d}$ within the regions where the experiments are physically available. In particular, the support for the design space is shown in Table 5.2, where the bounds are arbitrarily, but reasonably chosen. An interesting question is what does this "prior" really mean? It is not truly a prior in the inference sense, because the design variables are not uncertain parameters. However, they may be interpreted as weight functions in emphasizing where in the design space the PC expansion should be made more accurate. For example, if one suspects more experiments need to be conducted in a particular small region of the design space, then a Gaussian "prior" may be used that is centered around that region. The resulting PC expansion will then have a smaller error in that region since the minimization of the L_2 error is weighed according to the Gaussian "prior". Using the Legendre Chaos, the "PC expansions" for the design variables are simply

$$d_1 = 975 + 75\xi_3 \tag{5.11}$$

$$d_2 = 0.85 + 0.35\xi_4. \tag{5.12}$$

Note that one would never sample ξ_3 and ξ_4, because there is no associated randomness in $\mathbf{d}$. When a particular $\mathbf{d}$ is desired, the corresponding ξ_3 and ξ_4 are simply computed by inverting Equations 5.11 and 5.12. One may interpret these equations simply as a "renaming" of the deterministic d_1 and d_2 to ξ_3 and ξ_4, respectively.

Parameter	Lower Bound	Upper Bound
T_0	900	1050
ϕ	0.5	1.2

Table 5.2: "Prior" support of the design variables T_0 and ϕ. Uniform prior is assigned.

Now the PC expansions of the unknown variables can finally be constructed. Performing a standard Galerkin projection of the random variables onto the desired basis functions and taking advantage of the orthogonality property, the coefficients are simply

$$g_{m,\mathbf{i}} = \frac{\mathbb{E}\left[g_m \Psi_{\mathbf{i}}\right]}{\mathbb{E}\left[\Psi_{\mathbf{i}}^2\right]} = \frac{\int_{\Xi} g_m\left(\boldsymbol{\theta}(\boldsymbol{\xi}), \mathbf{d}(\boldsymbol{\xi})\right) \Psi_{\mathbf{i}}(\boldsymbol{\xi}) p(\boldsymbol{\xi})\, d\boldsymbol{\xi}}{\int_{\Xi} \Psi_{\mathbf{i}}^2\left(\boldsymbol{\xi}\right) p\left(\boldsymbol{\xi}\right)\, d\boldsymbol{\xi}}, \quad m = 1, \cdots, n_{\mathbf{y}}, \tag{5.13}$$

where $g_{m,\mathbf{i}}$ is the PC expansion with multi-index $\mathbf{i}$ for the mth observable from the model (Cantera) output. The denominators can be further simplified to

$$\mathbb{E}\left[\Psi_{\mathbf{i}}^2\right] = \int_{\Xi} \Psi_{\mathbf{i}}^2\left(\boldsymbol{\xi}\right) p\left(\boldsymbol{\xi}\right)\, d\boldsymbol{\xi} = \prod_{j=1}^{n_s} \int_{\Xi} \psi_{i_j}^2\left(\xi_j\right) p\left(\xi_j\right)\, d\xi_j, \tag{5.14}$$

where the second equality is due to the independence of ξ_j's and Equation 5.6. Typically analytical formulas are available for this expression. For example, for Legendre

Chaos,

$$\int_{\Xi} \psi_{i_j}^2 \left(\xi_j\right) p\left(\xi_j\right) \, d\xi_j = \frac{2}{2i_j + 1}. \tag{5.15}$$

The numerator, on the other hand, almost always has no analytic form, and must be computed via numerical integration techniques — this is the heart of the NISP computation. Due to both the often high stochastic dimension, as well as the expensive evaluation of the forward model $\mathbf{g}$, an efficient method of high dimension numerical integration is essential. The next section is dedicated to explore the options available in this topic.

5.4 Numerical Integration in High Dimension

5.4.1 Overview

Many problems in the field of engineering involve integrating complicated functions, which is often difficult or even impossible to perform analytically. Consequently, numerical integration has become an indispensable tool, used in such diverse applications from making money on Wall Street [70] to computing the surface area of molecules [22]. In particular, the need for numerical integration, especially in potentially high dimensions, stems from computing the PC coefficients via the NISP method discussed in the previous section. Furthermore, the dimension of an NISP integral is typically equal to the number of parameters $n_{\boldsymbol{\theta}}$ in the model (with an additional $n_{\mathbf{d}}$ design variables for an experimental design problem). For example, if all the kinetic parameters from the H_2-O_2 mechanism in Table 2.1 are random variables, then the dimension of the NISP integrals would be $3 \times 19 = 57$ (plus $n_{\mathbf{d}} = 2$ if experimental design is desired). Numerical integration becomes increasingly challenging as the integral's dimension grows. The infamous "curse of dimensionality" refers to the fact that all quadrature rules have some dependence on the number of dimensions, leading to rapid increases in the number of required function evaluations with dimension.

There are many numerical integration methods available, and they can be broadly divided into two categories — sampling methods and quadrature methods. Sampling methods are stochastic methods that involve averaging of the integrand evaluations at a set of random points. Quadrature methods are deterministic methods that involve forming the products between the integrand evaluation and a corresponding weight at a set of predetermined quadrature points, and finally summing them. More specifically, the following five numerical integration options are described in the subsequent sections.

1. Monte Carlo (MC)

2. Quasi-Monte Carlo (QMC)

3. Tensor product quadrature (TPQ)

4. Sparse quadrature (SQ)

5. Dimension-adaptive sparse quadrature (DASQ)

In this section, the more traditional notation for integration is used (i.e., not that from the NISP problem). The integration problem involves computing an approximation to the integral

$$I = \int_{\mathcal{X}} f\left(\mathbf{x}\right) w\left(\mathbf{x}\right)\, d\mathbf{x}, \tag{5.16}$$

where f is the integrand function, w is some weight function, $\mathbf{x}$ is a d-dimensional independent variable, and $\mathcal{X}$ is the domain of integration.

In all implementations from this study, the Kahan algorithm [43] is used for all summations. This method keeps a running compensation value, which helps significantly reduce numerical errors when summing a sequence (especially a long sequence) of finite-precision floating point numbers compared to the naïve running sum.

5.4.2 Monte Carlo (MC)

MC integration involves the generation of sample points using pseudo-random numbers. Its general form is

$$I \approx V \frac{1}{n} \sum_{i=1}^{n} f(\mathbf{x}^{(i)}), \tag{5.17}$$

where V is the hyper-volume of $\mathcal{X}$, and the n sample points, $\mathbf{x}^{(i)}$, are sampled according to $w(\mathbf{x})$ (in this case, the weight function must be a proper PDF). Sampling technique is itself a major topic, especially if $w(\mathbf{x})$ is a complicated PDF. Methods such as important sampling [78], MCMC [31], and Latin hypercube [60] have been used extensively to improve both the flexibility and efficiency of sampling. In many cases, however, it is much simpler to group the troublesome $w(\mathbf{x})$ term with $f(\mathbf{x})$, and factor out a constant weight function with value equal to $1/V$ — this corresponds to sampling $\mathbf{x}^{(i)}$ from a uniform distribution. When sampling is implemented on a computer, the samples are generated through pseudo-random number generators. These samples are not truly random because they are computed from some deterministic algorithm. MC can be potentially improved by replacing the pseudo-random numbers with quasi-random numbers (see Section 5.4.3).

The convergence rate of MC is

$$e = \mathcal{O}\left(n^{-\frac{1}{2}}\right), \tag{5.18}$$

where e is some error measure of the integral estimate. The main advantages of MC are that its convergence rate is independent of dimensionality; it does not assume any integrand smoothness, thus is good for integrating discontinuous functions; and it can work for arbitrary shapes of $\mathcal{X}$, for example, by rejecting the sample points outside $\mathcal{X}$. The main disadvantages are that the $\sqrt{n}$ convergence rate is very slow; and it does not take advantage of any integrand smoothness, which contributes to its slow convergence.

Thorough discussions on various topics related to MC can be found in [78].

5.4.3 Quasi-Monte Carlo (QMC)

The QMC method is proposed by Morokoff and Caflisch [63]. It approximates the integral the same way as MC, shown in Equation 5.17, except that it uses low discrepancy (i.e., more uniformly distributed) sequences, known as quasi-random numbers, to generate the samples. In this thesis, only the Sobol' sequence [86] is used in QMC computations. Results from [63] suggest that QMC is in general superior to MC, but the advantages may be slight depending on the number of dimensions and the properties of the integrand.

The convergence rate of QMC is

$$e = \mathcal{O}\left(n^{-1}(\log n)^d\right). \tag{5.19}$$

The main advantages of QMC are that its convergence rate is faster than that of MC (but is now dependent on the number of dimensions); and similar to MC, it does not assume any integrand smoothness and that $\mathcal{X}$ can be arbitrarily shaped. The main disadvantages are that the quasi-random numbers are typically not readily available on computers, and must be implemented separately (this problem is further complicated by the fact that sequences for different dimensions are different, and must be generated separately); and similar to MC, it does not take advantage of any integrand smoothness, contributing to its slow convergence.

5.4.4 Tensor Product Quadrature (TPQ)

This section provides a motivation for sparse quadrature. A naïve construction of a quadrature rule in multiple dimensions is to simply apply a tensor product on the 1D quadrature rules

$$I \approx (Q_{n_1}^{(1)} \otimes Q_{n_2}^{(1)} \otimes \cdots \otimes Q_{n_d}^{(1)}) f = \sum_{i=1}^{n} w_i f(\mathbf{x}_i), \tag{5.20}$$

where $Q_{n_j}^{(1)}$ is the n_j-point 1D quadrature rule in the jth dimension, and $n = \prod_{j=1}^{d} n_j$ is the total number of quadrature points. Unfortunately, this number increases exponentially with the number of dimensions for some fixed accuracy

$$e = \mathcal{O}\left(n^{-\frac{r}{d}}\right), \tag{5.21}$$

where r is the highest order of bounded derivative of the integrand. For example, a measly 5-point rule in 1D translates to around 10 million points in 10 dimensions.

The main advantage of TPQ is that it has an intuitive and trivial extension from available 1D quadrature rules. The main (and fatal) disadvantage is that the exponential growth of n renders this method infeasible for integration of more than around 4 dimensions. This motivates the need of sparse quadrature, discussed in the next section.

5.4.5 Sparse Quadrature (SQ)

Details of the SQ performance and analysis can be found in, for example, [7, 29], while [11] provides some intuitive explanations with its 2D examples. The SQ construction is described below.

Smolyak Rule

The SQ is formed using the Smolyak rule [85], a general method that builds a sparse multi-dimensional quadrature (or grid) based on some 1D quadrature rule. Let

$$Q_l^{(1)} f = \sum_{i=1}^{n_l} w_i f(\mathbf{x}_i) \tag{5.22}$$

be the lth level (with n_l quadrature points) 1D quadrature rule. Theoretically, the level can be arbitrarily defined without compromising the validity of the Smolyak construction. For example, $n_l = l$ gives a very fine resolution in n as l is increased, whereas a more carefully chosen definition could take advantage of the nestedness of the quadrature rule (if it is nested) and thus reducing the overall computational cost. The latter greatly out-benefits the former if the goal is to simply obtain an accurate integral estimate, but there exist applications that value the former quality more (e.g., see Section 5.6).

Here are some examples of typical level definitions. For Clenshaw-Curtis (CC) quadrature, the level is related to the number of points by the formulas

$$\begin{aligned} n_l &= 2^{l-1}+1,\, l \geq 2 && (5.23) \\ n_1 &= 1, && (5.24) \end{aligned}$$

and for Gauss-Patterson (GP) quadrature (also known as Kronrod-Patterson) [71], by the formulas

$$\begin{aligned} n_l &= 2^{l}-1,\, l \geq 2 && (5.25) \\ n_1 &= 1. && (5.26) \end{aligned}$$

Non-nested quadrature rules, such as the Gaussian-Legendre (GL) quadrature, may be used in the Smolyak construction as well, but they do not have a natural definition of level.

The difference formulas are defined by

$$\Delta_k f = (Q_k^{(1)} - Q_{k-1}^{(1)})f, \qquad Q_0^{(1)} f = 0, \tag{5.27}$$

which are the "difference" of 1D quadrature rules between two consecutive levels. The "subtraction" is carried out by subtracting the weights at the quadrature points of the lower level. Example 5.4.1 provides a simple illustration.

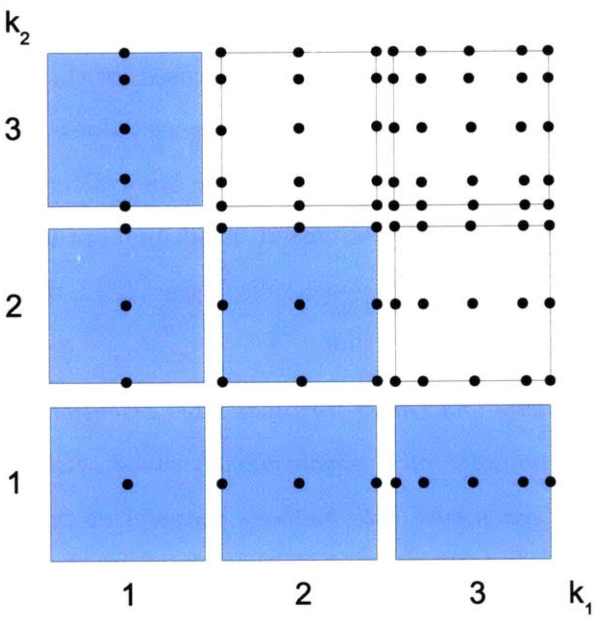


Figure 5-1: Illustration of the index sets of a 2D level 3 SQ rule. The dots are the approximate positions of quadrature points.

Example 5.4.1. Subtraction of Two Quadrature Rules

Suppose $Q_1^{(1)}$ is a 1-point rule with $x_1 = 0.0$ and $w_1 = 2.0$, and $Q_2^{(1)}$ is a 2-point rule with $x_1 = -1.0$, $x_2 = 1.0$, and $w_1 = w_2 = 1.0$, then $\Delta_2 = Q_2^{(1)} - Q_1^{(1)}$ would have abscissas $x_1 = -1.0$, $x_2 = 0.0$, $x_3 = 1.0$ with weights $w_1 = 1.0$, $w_2 = -2.0$, and $w_3 = 1.0$. Note the negative sign on w_2. □

Then, for $\mathbf{k} \in \mathbb{N}^d$, the sparse quadrature is defined to be

$$Q_L f = \sum_{|\mathbf{k}| \leq L+d-1} (\Delta_{k_1} \otimes \cdots \otimes \Delta_{k_d}) f, \tag{5.28}$$

where $|\mathbf{k}| = k_1 + \cdots + k_d$. Typically, the user would select some desired overall level L for the sparse quadrature. The resulting index sets (the $\mathbf{k}$'s) included in the summation thus form a simplex. An illustration of the index sets of a 2D level 3 SQ rule is shown in Figure 5-1.

Selection of 1D Quadrature Rule

In theory, any arbitrary 1D quadrature rule can be used for the Smolyak construction. However, CC rule is especially appealing because its nestedness induces computational saving in addition to the sparseness provided by the Smolyak construction. One might argue that GL quadrature (which is not nested) is optimal in the sense that the highest order of polynomial it can integrate exactly is higher than that of any other rules. However, Trefethen recently showed that

> *"the Clenshaw-Curtis and Gauss formulas have essentially the same accuracy unless f is analytic in a sizable neighborhood of the interval of the integration — in which case both methods converge so fast that the difference hardly matters"* [95].

Additionally, the ability to integrate polynomials exactly does not necessarily reflect the rule's general integration capability (c.f., Newton-Cotes). Another advantage of CC is its ease of construction. The 1D CC abscissas are simply

$$x_i = \cos\left(\frac{i\pi}{n}\right), \tag{5.29}$$

and the weights can be computed via FFT [27, 28], which only requires $\mathcal{O}\left(n \log n\right)$ in time and induces very little numerical roundoff. Nestedness and ease of construction render CC an especially popular choice.

Other nested quadrature rules can be used as well. For example, GP [71] has a systematic extension from two initial levels of arbitrary quadrature rule. However, the extension method [72], although carefully engineered to take advantage of the special structures of the problem, is vulnerable to round-off errors. An attempted implementation in double precision breaks down for higher than 3 levels. The failure is mainly due to two reasons: first is that it involves solving the roots of polynomials of degree that is roughly half the number of quadrature points (i.e., very high degree), and the second is that the computations of higher order extensions are dependent on the lower levels, so roundoff accumulates across all levels of extensions. The solution to this problem is to use high-precision arithmetic, and then tabulate the accurate results. This is exactly what Patterson has done, and nonetheless, the highest available level for GP found in the literature is 8 (i.e., 255 quadrature points).

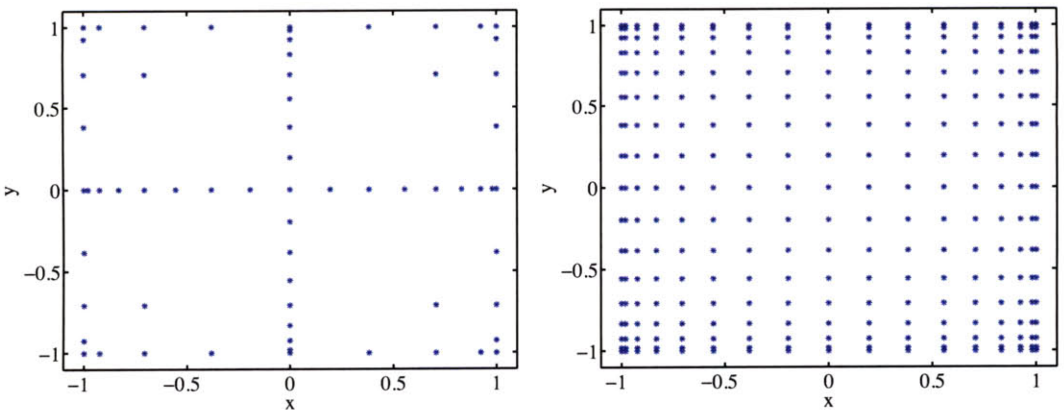
Performance

To get an idea of how much saving SQ is providing, Table 5.3 shows, for example, that the level 3 CC SQ in 10 dimensions has 58,828 fewer nodes than a TPQ that can integrate exactly all the polynomial orders that the SQ can. While SQ does not hold advantage over TPQ in 2D, the saving becomes substantial for higher dimensions. These results are identical to those in [103]. As an example, Figure 5-2 shows a 2D sparse grid constructed using the level 5 CC rule, and the corresponding tensor product grid constructed using the same 1D quadrature rule.

d	l	SQ n	TPQ n
2	1	1	1
	2	5	4
	3	13	9
	4	29	16
	5	65	25
10	1	1	1
	2	21	1024
	3	221	59049
	4	1581	1048576
20	1	1	1
	2	41	1048576
	3	841	$\approx 3.5 \times 10^9$
50	1	1	1
	2	101	$\approx 1.1 \times 10^{16}$
	3	5101	$\approx 7.2 \times 10^{23}$

Table 5.3: Examples of number-of-abscissas comparison between SQ and TPQ, using CC rule.

The fewer quadrature points in SQ is traded off in accuracy. For example, in 2D, if a SQ can integrate at most 2nd order polynomials (i.e., up to x^2, xy and y^2) exactly, it cannot integrate the mixed terms (i.e., x^2y, xy^2, and x^2y^2) exactly. However, a tensor

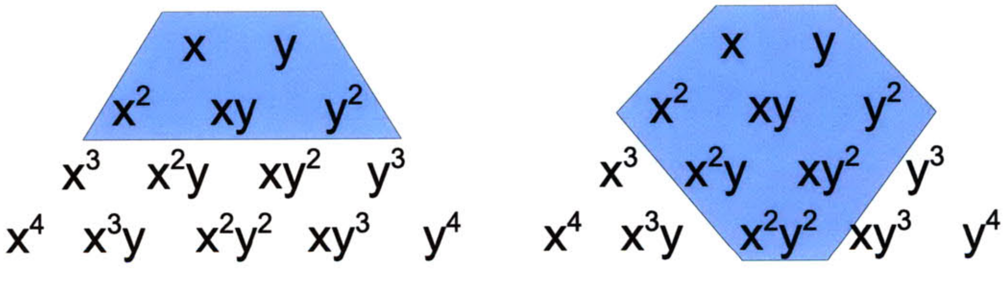


(a) Sparse quadrature

(b) Tensor product quadrature

Figure 5-2: 2D sparse grid constructed using the level 5 CC rule, and the corresponding tensor product grid constructed using the same 1D quadrature rule.

product quadrature can, by simply separate the mixed term integral into product of the two. This property is illustrated in Figure 5-3; the shaded terms are those that can be integrated exactly by the corresponding quadrature rule. Note that in higher dimensions, the pattern for SQ and its relationship to that of TPQ are not as simple [7].



(a) Sparse quadrature

(b) Tensor product quadrature

Figure 5-3: Polynomial exactness of a 2D SQ and of its corresponding TPQ.

The convergence rate for SQ is

$$e = \mathcal{O}\left(n^{-r}(\log n)^{(d-1)(r+1)}\right). \tag{5.30}$$

The main advantages of SQ are that it has a much weaker dependence on d than the TPQ, and it takes advantage of integrand smoothness. The main disadvantages are that it is less accurate than the TPQ, and it has small flexibility in choosing n (but better than TPQ in that respect) since n's resolution is dictated by the definition of level of the quadrature rule (for example, in 100D, level 3 CC rule has 20,201 points, but level 4 jumps to 1,353,801 points, and any intermediate numbers cannot be selected).

5.4.6 Dimension-Adaptive Sparse Quadrature (DASQ)

Original Algorithm

The DASQ was introduced by Gerstner and Griebel [30], and is a natural extension to the Smolyak rule. In Equation 5.28, instead of fixed summation over index sets $|\mathbf{k}| \leq L + d - 1$, the summation proceeds by choosing the index sets adaptively based on some error indicator. The algorithm is shown in Algorithm 2, reproduced from [30]. The idea is to divide all the indices into two sets: an old set, and an active set. A member of the active set is able to generate new candidates by increase its index, $\mathbf{i}$, in any dimension by 1 (e.g., $\mathbf{i} + \mathbf{e}_p, \forall p \leq d$, where $\mathbf{e}_p$ is the unit vector in the pth dimension). However, the candidate can only be accepted if all its backward neighbours (i.e., $\mathbf{i} + \mathbf{e}_p - \mathbf{e}_q, \forall p, q \leq d$) are in the old set — the so-called admissibility condition, which ensures the validity of the telescope sum expansion of the general sparse quadrature formulas using the difference formulas Δ_k. Finally, each index set has an error indicator, which is proportional to its corresponding summand value in Equation 5.28. Intuitively, if this term is contributing little to the overall integral estimate, then integral error due to this term would be small. The process iterates until the sum of error indicators for the active set members falls below some user-specified tolerance. More details, including proposed data structures for this algorithm, can be found in [30]. One drawback of DASQ is that parallelization can only be implemented within each index set, which is not as efficient as the parallelization in any of the other integration methods discussed so far, due to the adaptive nature of DASQ.

Algorithm 2: Dimension-adaptive sparse quadrature algorithm [30].

```
$\mathbf{i} = (1, \cdots, 1)$;
$\mathcal{O} = \emptyset$;
$\mathcal{A} = \{\mathbf{i}\}$;
$v = \Delta_{\mathbf{i}} f$;
$\eta = h_{\mathbf{i}}$;
while $\eta > TOL$ do
    select $\mathbf{i}$ from $\mathcal{A}$ with the largest $h_{\mathbf{i}}$;
    $\mathcal{A} = \mathcal{A} \setminus \{\mathbf{i}\}$;
    $\mathcal{O} = \mathcal{O} \cup \{\mathbf{i}\}$;
    $\eta = \eta - h_{\mathbf{i}}$;
    for $p = 1$ to $d$ do
        $\mathbf{j} = \mathbf{i} + \mathbf{e}_p$;
        if $\mathbf{j} - \mathbf{e}_q \in \mathcal{O}$ for all $q = 1, \cdots, d$ then
            $\mathcal{A} = \mathcal{A} \cup \{\mathbf{j}\}$;
            $s = \Delta_{\mathbf{j}} f$;
            $v = v + s$;
            $\eta = \eta + h_{\mathbf{j}}$;
        end
    end
end
Symbols:
$\mathcal{O}$ — old index set;
$\mathcal{A}$ — active index set;
$\Delta_{\mathbf{i}} f$ — integral increment $\otimes_{p=1}^{d} \Delta_{i_p} f$;
$h_{\mathbf{i}}$ — local error indicator;
$\eta$ — global error estimate $\sum_{\mathbf{i} \in \mathcal{A}} h_{\mathbf{i}}$;
$\mathbf{e}_p$ — $p$th unit vector;
TOL — error tolerance;
$v$ — computed integral value $\sum_{\mathbf{i} \in \mathcal{O} \cup \mathcal{A}} \otimes_{p=1}^{d} \Delta_{i_p} f$;
```

Reuse Nested Quadrature Function Evaluations

The original DASQ algorithm proposed by Gerstner and Griebel [30] does not address how nested quadrature points can be reused as adaptation proceeds. Unlike in SQ, where one can combine repeated quadrature points after the Smolyak construction, this cannot be done for DASQ due to the adaptive nature. This section proposes an algorithm to solve this problem. For demonstration purposes, the CC rule is used here. Other nested rules may be used as well, but the algorithm needs to be modified according to their abscissa structures.

This algorithm takes advantage of the special structure of the CC abscissas, whose formula is Equation 5.29. When increasing the level l by 1 (for $l \geq 2$), n is (almost) doubled. In particular, every other abscissa in the new level would also be in the previous level

$$\cos\left(\frac{i\pi}{n}\right) = \cos\left(\frac{j\pi}{2n}\right), \tag{5.31}$$

where $i = 0, 1, 2, \cdots, n$ and $j = 0, 2, 4, \cdots, 2n$. It is easy to see that the quadrature coordinates are monotonic with respect to the index i (or j). Hence, there will always be 1 new point between 2 old points when the level is increased by 1 (for $l \geq 2$).

Before presenting the algorithm, some terminology is first defined.

Definition 5.4.1. Position p — the set of ordinalities (in ascending order of the physical coordinate value, counting from 1) of the quadrature point of interest in each dimension, with respect to a particular index set **k**. For example, for the 2D index set $\mathbf{k} = (2, 2)$ (the center square in Figure 5-1), the quadrature points and their positions are listed in Table 5.4. More specifically, $(x, y) = (1.0, 0.0)$ is the 3rd point in x, and 2nd point in y, thus has position $\mathbf{p} = (3, 2)$. □

Definition 5.4.2. Position of a new function evaluation p_{new} — the ordinality of the new function value with respect to an index set **k**. The ordering is done in sequence of dimensions (i.e., sort by x first, then y). For example, in the 2D index set $\mathbf{k} = (2, 1)$ (the square to the right of the bottom-left square in Figure 5-1), the function evaluation at $(x, y) = (-1.0, 0.0)$ is the first new function value in that index set, therefore has $p_{new} = 1$, and at $(x, y) = (1.0, 0.0)$ is the second new function value in that index set, therefore has $p_{new} = 2$. $(x, y) = (0.0, 0.0)$ is not a few function evaluation since it has already been computed in index set (1,1). □

(x,y)	Position $\mathbf{p}$
(-1.0,-1.0)	(1,1)
(-1.0, 0.0)	(1,2)
(-1.0, 1.0)	(1,3)
(0.0,-1.0)	(2,1)
(0.0, 0.0)	(2,2)
(0.0, 1.0)	(2,3)
(1.0,-1.0)	(3,1)
(1.0, 0.0)	(3,2)
(1.0, 1.0)	(3,3)

Table 5.4: Physical coordinates and positions of quadrature points in the 2D index set (2,2).

Definition 5.4.3. Backtrace — an adjective describing properties or values related to the index set that is traced back by one step from the original index set. Some example usages include "backtrace path", "backtrace index set", "backtrace dimension" (the dimension in which the backtrace is performed), etc. □

The algorithm is presented in detail below and summarized in Algorithm 3. The starting point is when the function value of a particular quadrature point is requested during the computation of the tensor product of the Δ_{k_i}'s at a particular index set.

1. Determine whether this quadrature point is a new point by checking its position in this index set $\mathbf{k}$. In 1D, the potential new quadrature points may occur according to the rule presented in Table 5.5. In multiple dimensions, the new quadrature points take place at the intersection (or tensor product) of these 1D potential locations.

Index Set Value (k_i)	Positions p_i of New Points
1	1
2	1,3
>2	Even positions

Table 5.5: Positions p_i, $i = 1, \cdots, d$ of new points in 1D.

For example, consider the 2D index set (2,4). For the first index value $k_1 = 2$, potential new points are at positions 1 and 3 in that dimension. For the second index value $k_2 = 4$, potential new points are at positions 2, 4, 6, and 8 in that dimension. Therefore, the positions which require new function evaluations are (1,2), (1,4), (1,6), (1,8), (3,2), (3,4), (3,6), and (3,8).

(a) If it is a new point, go ahead and evaluate it, store it, and continue onto the next quadrature point.

(b) If it is not a new point, go to Step 2 to retrieve the previously-computed value.

2. Determine which dimension i^* to trace back by 1. Note that there can be multiple possible backtrace paths (e.g., $(x, y) = (0, 0)$ can be traced back via any path). This algorithm would find a possible path by searching in the ascending order of the dimensions.

   For any dimension of the index set, if the index value is 1, backtracing cannot take place in this dimension. If the index value is 2, backtracing can only occur if the quadrature point has position 2 in this dimension. If the index value > 2, backtracing in this dimension is possible only if the quadrature point has an odd-number position in this dimension.

3. Update to the backtrace index set $\mathbf{k}^*$ (simply decrease k_{i^*} by 1).

4. Find the memory storage location of $\mathbf{k}^*$. This is done by finding the index set within the backward neighbours of $\mathbf{k}$ that matches $\mathbf{k}^*$. This location is needed to access the new function evaluations with respect to $\mathbf{k}^*$.

5. Find the corresponding position $\mathbf{p}^*$ of the quadrature point with respect to $\mathbf{k}^*$. Modification only needs to be done on p_{i^*}.

   If $k_i^* = 1$, then $p_i^* = 1$. If k_i^* >=2, then $p_i^* = (p_i - 1)/2 + 1$.

6. Determine whether this quadrature point is a new quadrature point in $\mathbf{k}^*$. The same algorithm as Step 1 is used, except that $\mathbf{k}^*$ replaces $\mathbf{k}$.

   (a) If it is a new point, retrieve it. Go to Step 7.

   (b) If it is not a new point, let $\mathbf{k} = \mathbf{k}^*$, $\mathbf{p} = \mathbf{p}^*$, and go to Step 2.

7. Determine p_{new} in $\mathbf{k}^*$. Compute the values

$$a_d = 1 \tag{5.32}$$

$$a_i = \prod_{n=i+1}^{d} n_n, \qquad \text{for } i < d, \tag{5.33}$$

   where n_n is the number of new function evaluations for the nth dimension, obtained according to the rules presented in Table 5.5. Finally,

$$p_{new} = \sum_{i=1}^{d} \left\lfloor \frac{p_i - 1}{2} \right\rfloor a_i. \tag{5.34}$$

8. Extract the function value.

Algorithm 3: Algorithm summary for reusing nested quadrature function values in DASQ.

```
Check whether current quadrature point is a new point in $\mathbf{k}$;
if is a new point in $\boldsymbol{k}$ then
    evaluate it, store it, and continue onto the next quadrature point;
end
$\star$ Determine backtrace dimension $i^*$;
Update backtrace index set $\mathbf{k}^*$;
Find the memory storage location of $\mathbf{k}^*$, needed to access the stored function value;
Compute $\mathbf{p}^*$ with respect to $\mathbf{k}^*$;
Check whether the quadrature point is a new quadrature point in $\mathbf{k}^*$;
if is not a new point in $\boldsymbol{k}^*$ then
    $\mathbf{k} = \mathbf{k}^*$, $\mathbf{p} = \mathbf{p}^*$, go to $\star$;
end
Determine $p_{new}$ in $\mathbf{k}^*$;
Extract the function value;
```

One potential issue of this algorithm is that the recursive backtracing may become expensive when levels are high such that its cost surpasses that of an actual function evaluation. However, if the function is cheaper to evaluate than backtracing, then the overall DASQ time is likely to be small anyways and the less efficient backtracing method does not matter. Another potential issue is that the storage of all function evaluations can become very demanding. Some compromise can be made such as to stop further storage after certain memory limit has been reached, or to discard some storage from lower levels.

5.4.7 Numerical Results

Three numerical examples are considered to validate and compare the performance between the numerical integration algorithms discussed thus far. The first is an anisotropic 2D integral devised to illustrate the index set adaptation from the DASQ algorithm, the second is a 20D integral (smooth version) adopted from [63], and the last is a 100D extension to the first 2D example.

2D Integral

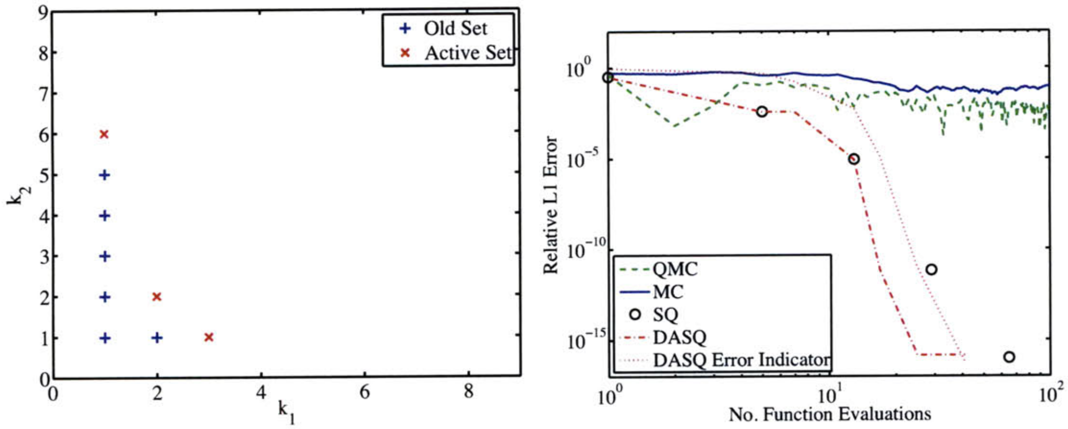
Consider the 2D integral

$$I = \int_{-1}^{1} \int_{-1}^{1} (x^2 + e^y)\, dx\, dy, \tag{5.35}$$

whose exact solution is

$$I = 2\left[\frac{2}{3} + \left(e^1 - e^{-1}\right)\right]. \tag{5.36}$$

The y dimension is much more difficult to integrate than the x dimension due to the exponential. Therefore, for DASQ, higher levels of final index sets are expected to favour the y dimension. The index sets and relative L_1 error (normalized by the true integral value) convergence are shown in Figure 5-4. The index distribution is indeed as expected, anisotropic favouring the y variable. In the error convergence plot, discrete points are used for SQ, because the intermediate number of function evaluations cannot be achieved due to the level definitions and simplex-like region of index sets. The error convergence suggests that DASQ is competitive with SQ, while both quadrature methods are substantially superior to MC and QMC in this low-dimensional problem.



(a) Final DASQ index sets

(b) Relative L_1 error

Figure 5-4: Final DASQ index sets and relative L_1 error of the 2D integral using various numerical integration schemes.

20D Integral

Consider the following integration problem adopted from [63]

$$I = \int_{[0,1]^{20}} \sum_{n=0}^{20} F^n(\mathbf{x})\, d\mathbf{x}, \tag{5.37}$$

where

$$F^n = \gamma^n \left(\prod_{i=1}^{n-1} x_i^{n-i} \right) \left(1 - \prod_{j=1}^{n} x_j \right), \tag{5.38}$$

and $\gamma \in (0,1)$ is a free parameter. Note that there is a typo in [63], where the first product in Equation 5.38 is incorrectly printed as a summation. The exact solution is

$$I = \frac{1}{\gamma}\left[1 - (1-\gamma)\exp(\gamma)\right]. \tag{5.39}$$

The relative L_1 error for $\gamma = 0.5$ is shown in Figure 5-5. A number of observations can be made. The QMC method is better than MC, although its integral estimate has large oscillations. On average, QMC is competitive to SQ, and can be better depending on where the estimate is in the oscillation cycle. DASQ is better than all other methods, indicating that adaptation is indeed contributing. The error indicator for DASQ appears to be approximating the true error, but it is not guaranteed to be always a lower or higher estimate. The non-monotonic convergence of the dimension-adaptive case implies that in some cases, refinement can actually lead to a worse estimate. This suggests that the error indicator can be further improved.

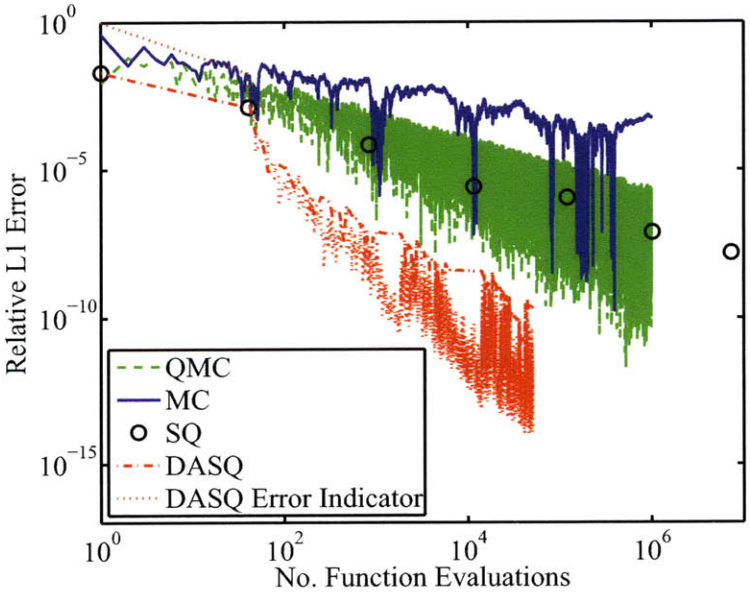


Figure 5-5: Relative L_1 error of the 20D integral using various numerical integration schemes.

100D Integral

A 100D integral is constructed by extending Equation 5.35

$$I = \int_{[-1,1]^{100}} \left(\sum_{j=1}^{50} x_j^2 + \sum_{k=51}^{100} e^{x_k} \right) d\mathbf{x}, \tag{5.40}$$

whose exact solution is

$$I = 2^{99} \left[50 \left(\frac{8}{3} \right) + 50 \left(e^1 - e^{-1} \right) \right] \approx 1.59 \times 10^{32}. \tag{5.41}$$

The errors are plotted in Figure 5-6. The comparisons between MC, QMC, and SQ are similar to the previous example, with MC and QMC gaining ground on the quadrature algorithms due to the higher dimension. However, a very interesting behaviour is observed for the DASQ algorithm: it first converges slowly, even slower than SQ while reaching a plateau, and then takes a sudden plunge. The plateau is caused by the fact that the algorithm is exploring the dimensions that have already resolved the x_i^2's exactly, but would not know it has resolved them until the local error indicators at one level higher are computed (which would be zero). Consequently, there are additional computations without any improvements in the actual error, a so-called "lag"; however, the error indicator still improves during the plateau. The sharp drop-off is due to the algorithm finally exploring the "correct" e^{x_k} dimensions, thus drastically improving the accuracy. The fact that the error indicator is a more conservative estimate in this case is important, for if it were less conservative (i.e., underestimate of the error), it could lead to premature termination of DASQ while the true error is still very large.

5.4.8 Conclusions

The sampling methods (MC and QMC) are only competitive to the quadrature methods (SQ and DASQ) when the integral dimension is very high (e.g., > 100), or if the integrand is expected to be non-smooth (discontinuous examples are not presented

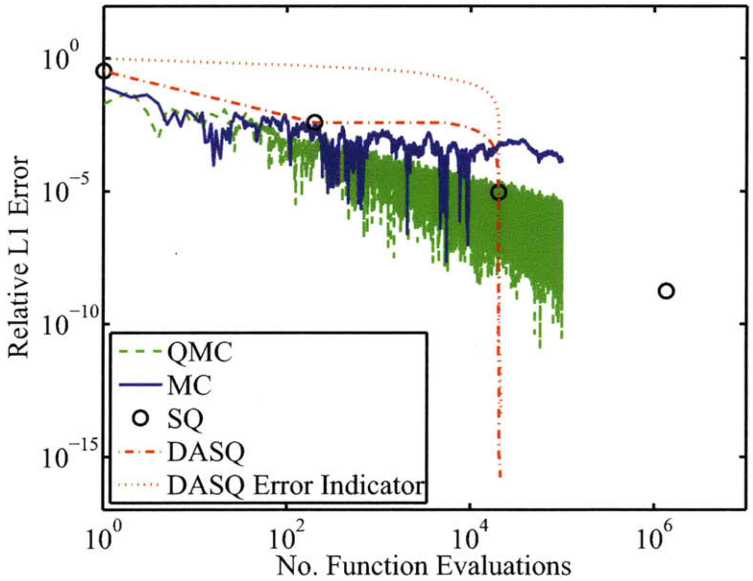


Figure 5-6: Relative L_1 error of the 100D integral using various numerical integration schemes.

here). If a sampling method is used, QMC holds an advantage over MC. If a quadrature method is used, DASQ has demonstrated advantage over SQ when the integrand is expected to be anisotropic.

For the rest of the thesis, only DASQ is used wherever numerical integration is required.

5.5 Implementation of DASQ in NISP

Armed with the selected numerical integration tool, DASQ, the PC expansion coefficients of the output variables in Equation 5.13 can now be computed. There are a total of n_{PC} (see Equation 5.7) PC coefficients for each output variable, for a total of $n_{\mathbf{y}}$ output variables, yielding a total of $n_{\text{coef}} = n_{PC} n_{\mathbf{y}}$ integrations. For the ease of notation, let $g_{m,\mathbf{i}}$, $m = 1, \cdots, n_{\mathbf{y}}, \forall \mathbf{i}$ be re-indexed by g_r, $r = 1, \cdots, n_{\text{coef}}$. It would be very inefficient to compute the g_r's from scratch independently, since some overlap in quadrature points may exist between the various integral computations. In other words, values of $\mathbf{g}(\boldsymbol{\theta}(\boldsymbol{\xi}), \mathbf{d}(\boldsymbol{\xi}))$ evaluation, which is the most expensive part of the computation, may be reused for different g_r computations.

To take advantage of this computational saving, the DASQ algorithm is altered to integrate all g_r's simultaneously. The idea is to simply dictate all the integrations via a single adaptation route, while using a "total effect" local error indicator $\tilde{h}_{\mathbf{i}}$ that is reflective of all the local error indicators ($h_{r,\mathbf{i}}$, $\forall r$) from the integrals. For example, some reasonable choices are to take the maximum or L_2 norm over all the $\tilde{h}_{r,\mathbf{i}}$'s over all r. The new algorithm, while retaining the same notations as Algorithm 2, is presented in Algorithm 4.

5.6 Detection of Polynomial Dimension Anisotropy Using DASQ

5.6.1 Motivation

The current method to truncate the infinite PC expansion to a finite expansion is through Equation 5.5, which retains all polynomial terms that have an overall order (i.e., L_1 norm of order in all dimensions) less or equal to some constant p_θ. This, however, may be very inefficient, as the random variable's dependence in some dimensions might be of lower order than others, exhibiting a polynomial dimension anisotropy. This motivates the need to estimate the polynomial equivalence of the random variable at some fixed error tolerance, and detect such anisotropy. The method proposed to perform this task uses DASQ. At this writing of this thesis, this algorithm has been developed and tested, but it is not yet incorporated into the combustion problem, and no results of the combustion problem using this algorithm is currently available.

The choice of quadrature rule is important in formulating the detection rule, because the goal now is no longer just to accurately integrate a function, but also to detect its polynomial equivalence. The CC rule is a good candidate. In general, an n point CC rule can integrate an $n-1$ degree polynomial exactly. However, as often

Algorithm 4: Dimension-adaptive sparse quadrature algorithm applied to non-intrusive spectral projection.

```
$\mathbf{i} = (1, \cdots, 1)$;
$\mathcal{O} = \emptyset$;
$\mathcal{A} = \{\mathbf{i}\}$;
for $r = 1$ to $n_{coef}$ do
    $v_r = \Delta_{\mathbf{i}} f_r$;
    Compute $h_{r,\mathbf{i}}$;
end
$\eta = \tilde{h}_{\mathbf{i}}$. For example, $\tilde{h}_{\mathbf{i}} = \max_r h_{r,\mathbf{i}}$;
while $\eta > TOL$ do
    select $\mathbf{i}$ from $\mathcal{A}$ with the largest $\tilde{g}_{\mathbf{i}}$;
    $\mathcal{A} = \mathcal{A} \setminus \{\mathbf{i}\}$;
    $\mathcal{O} = \mathcal{O} \cup \{\mathbf{i}\}$;
    $\eta = \eta - \tilde{h}_{\mathbf{i}}$;
    for $p = 1$ to $d$ do
        $\mathbf{j} = \mathbf{i} + \mathbf{e}_p$;
        if $\mathbf{j} - \mathbf{e}_q \in \mathcal{O}$ for all $q = 1, \cdots, d$ then
            $\mathcal{A} = \mathcal{A} \cup \{\mathbf{j}\}$;
            for $r = 1$ to $n_{coef}$ do
                $s_r = \Delta_{\mathbf{j}} f_r$;
                $v_r = v_r + s_r$;
                Compute $h_{r,\mathbf{j}}$;
            end
            Compute $\tilde{h}_{\mathbf{j}}$;
            $\eta = \eta + \tilde{h}_{\mathbf{j}}$;
        end
    end
end
```

neglected to state in literature, the rule gains an extra degree of accuracy if n is odd — that is, an n point rule can integrate an n degree polynomial exactly if n is odd. This is a subtle but important property in constructing the detection rule later on. Consequently, every addition of 1 quadrature point leads to at most 2 extra degree of polynomial accuracy, depending on whether or not the number of points is odd. This relationship is shown in Table 5.6(a).

The GL and GP rules are poor candidates because they are "too" accurate — that is, their degree of polynomial exactness jumps several degrees when the number

n-Point Rule	Max Degree Exactness
1	0,1
2	1
3	2,3
4	3
5	4,5
$\vdots$	$\vdots$

(a) Original CC rule

n-Point Rule	Max Degree Exactness
1	0
2	1
3	2,3
4	3
5	4,5
$\vdots$	$\vdots$

(b) Shifted CC rule

Table 5.6: Relationship between the number of quadrature points in a CC rule and the maximum exact polynomial degrees.

of quadrature point is increased by one. This essentially decreases the resolution of polynomial degree that can be detected. The Newton-Cotes (NC) rule has the same resolution as the CC rule, but it is plagued by Runge phenomena at high orders if the integrand is not polynomial. Composite NC cannot be used because it does not have a guaranteed polynomial exactness.

Therefore, in this detection rule, only the CC rule is used.

5.6.2 Preliminary Observations

Some preliminary testings are performed, with the following notable observations.

- In order for the adaptive rule to proceed beyond the first index set $(1, \cdots, 1)$, the integrand must have a non-zero value when evaluated at the origin (otherwise the error indicator returns zero, and adaptation terminates). If this is the case, a constant offset (say α) can be artificially introduced to the integrand. The hypervolume of the integration domain multiplied by α can then be simply subtracted out from the final estimate at the end to extract the true integral estimate. Additionally, α should be carefully chosen, for example, to have a similar order of magnitude as the integrand evaluations, in order to avoid numerical errors.

- There are often situations where a higher resolution of polynomial exactness in Table 5.6(a) is desired. This can be achieved if the property of an n-point rule being able to integrate an n degree polynomial exactly when n is odd, is destroyed. One possible way to do this is to perturb some of the abscissas from the CC rule, and recompute the weights. However, this is expensive to do, and if not carefully done, can introduce undesirable behaviour such as the Runge phenomenon, especially at high degrees.

Instead, higher resolution at a low degree shall be targeted. High-dimension low-order representations are frequent in PC expansions, thus, the resolution for the 0-1 degrees of polynomial is most important. These two degrees can be easily distinguished by simply shifting the 1-point CC rule's abscissa by some non-zero constant while remaining within the integration domain. One can very easily visualize how this 1-point rule can no longer integrate an affine function exactly. The higher order rules remain untouched, retaining the original 2-degree resolution. The new relationship between the number of quadrature points and the polynomial exactness is shown in Table 5.6(b). This new rule shall be referred to as the **shifted CC rule**.

- For any symmetric quadrature rule, the 2-point rule for an odd function integrand would yield $w_1 f(x_1) + w_2 f(x_2) = w\left(f(x_1) - f(x_1)\right) = 0$, since $w_1 = w_2 = w$ and $f(x_1) = -f(x_2)$. Consequently, the quadrature rule in that dimension would not advance beyond the 2-point rule. This is problematic because then the polynomial equivalence of that odd function cannot be detected.

- The final index sets are only dictated by the highest equivalent polynomial orders (e.g., x^7 and $x^7 + x^5$ would yield identical results). This is expected since when a certain rule that is able to integrate a particular polynomial degree exactly, it can also integrate all the lower degrees exactly.

5.6.3 Detection Rule

The final detection rule for polynomial equivalence using DASQ is as follows.

A shifted version of the CC rule incrementing one abscissa at a time is used in the DASQ algorithm. Given the final index sets of some fixed tolerance, only the old set is used to infer the equivalent polynomial degrees according to the relationship from Table 5.6(b).

The shifted CC rule is described in the second bullet point in Section 5.6.2.

Here are some remarks regarding the detection rule.

- The active set reflects the phenomenon of lagging, which is due to the fact that the error indicator can only detect when a polynomial has been integrated exactly after reaching the rule that is one abscissa more than necessary (the same phenomenon that caused the plateau in the 100D example in Section 5.4.7). Thus, the active set can be safely ignored.

- If a polynomial is used as the integrand, DASQ would never terminate on an even number of abscissas (e.g., both degrees 2 and 3 would terminate on the 3-point rule). However, this is not true for general non-polynomial integrands, and thus rules for even number of abscissas still need to be established.

- After the highest order polynomials are detected, all the lower order terms also need to be included because the adaptation is only driven by the highest degree terms, and thus unable to detect lower order terms.

- Another, perhaps very expensive, method to increase resolution is to simply run the algorithm twice — first with the original integrand, then second with the original integrand multiplied by $\prod_{i=1}^{d}(x_i + t)$, where t is some shifting constant to disrupt odd functions. For example, if in the first run the terminating n is 3, and in the second run is also 3, then the degree is 2; but if in the second run is 5, then the degree is 3. If this method is used, the shifted CC rule is no longer necessary.

  After some testing, this method is deemed too expensive. Additionally, this rule is still not fail-proof for polynomial integrands, and can occasionally introduce overestimates of polynomial degrees. Therefore, it is not used.

5.6.4 Numerical Results

The following two test cases are used to validate the detection rule described in Section 5.6.3.

$$\text{Case 1} \quad \int_{[-1,1]^{10}} [(x_1+s)+(x_3+s)+(x_4+s)+(x_8+s)+(x_1+s)(x_3+s) \\ +(x_1+s)(x_8+s)+(x_4+s)(x_7+s)+1]\, d\mathbf{x} \tag{5.42}$$

$$\text{Case 2} \quad \int_{[-1,1]^{10}} \big[(x_1+s)+(x_3+s)^2+(x_4+s)+(x_8+s)^4+(x_1+s)^3(x_3+s) \\ +(x_1+s)^2(x_8+s)^2+(x_4+s)^3(x_7+s)^5+(x_1+s)(x_3+s)^2(x_4+s)^3 \\ +(x_4+s)^2(x_7+s)(x_8+s)^5+1\big]\, d\mathbf{x}, \tag{5.43}$$

where $s = 0.123$ is an arbitrary shifting constant. Since these cases are 10D, the final index sets are not presented.

For Case 1, the detected polynomial terms contain 1, x_1, x_3, x_4, x_7, x_8, x_4x_7, x_1x_8, x_1x_3 (and all lower order combinations), which cover all the terms in the integrand. For Case 2, there are a total of 176 final polynomial terms, and they indeed cover all the polynomial terms in the integrand.

Chapter 6

Results

Combining and applying the tools introduced thus far to the combustion problem described in Chapter 2, this chapter provides the main results of this thesis. In particular, the mechanics from Chapters 3, 4, and 5 are used in reverse order. First, PC expansions to the observables of the combustion system are constructed. This is done by using DASQ to compute the integrals in Equation 5.13. Second, the PC expansions are used in the optimal nonlinear experimental design framework, to obtain the optimal experimental conditions shown in Equation 4.14, whose results allow the best inference of the uncertain kinetic parameters. Third, the inference problem is solved at a selected number of experimental conditions from the design space in order to validate the experimental design results. Finally, a short discussion on the computational savings from using PC expansion is presented.

6.1 PC Expansion Construction

The PC expansions for the 10 observables presented in Table 2.4 are constructed via Equation 5.13. Note that as mentioned at the end of Section 2.4 and beginning of Section 5.3, ln of the characteristic time observables are used in the PC expansions. The numerator in that equation is computed using the modified DASQ algorithm described in Section 5.5 (recall the denominators can be computed analytically). More specifically, a global order of $p_\theta = 4$ is chosen for all of the expansions, and DASQ is stopped once a total of $n_{\text{quad}} = 1000$ function evaluations has been exceeded. These choices are not intuitive to reach, and they are a result of numerous trial-and-errors. For example, one can initially start from a low p_θ and low n_{quad} for the DASQ, solve the inference problem at a particular fixed experimental condition using the resultant PC expansions, and compare the posterior to the exact posterior obtained from when the original Cantera model is used. p_θ and n_{quad} can then be increased until the posteriors appear to have a good match. Example comparisons between posteriors from a PC expansion and the Cantera model can be observed from Figures 6-5 and 6-6 later in this chapter.

A more rigorous analysis is performed by computing the relative L_2 error for different combinations of p_θ and n_{quad}:

$$e_m = \frac{\int_{\Xi} \left| g_m\left(\boldsymbol{\theta}\left(\boldsymbol{\xi}\right), \mathbf{d}\left(\boldsymbol{\xi}\right)\right) - g_m^p\left(\boldsymbol{\xi}\right)\right|^2 p\left(\boldsymbol{\xi}\right)\, d\boldsymbol{\xi}}{\int_{\Xi} \left| g_m\left(\boldsymbol{\theta}\left(\boldsymbol{\xi}\right), \mathbf{d}\left(\boldsymbol{\xi}\right)\right)\right|^2 p\left(\boldsymbol{\xi}\right)\, d\boldsymbol{\xi}}, \quad m = 1, \cdots, n_{\mathbf{y}}, \tag{6.1}$$

where g_m and g_m^p are the original Cantera model output and the PC expansion, respectively, for the mth observable; and $\boldsymbol{\theta}$ and $\mathbf{d}$ as functions of $\boldsymbol{\xi}$ are from Equations 5.9, 5.10 and 5.11, 5.12, respectively. The L_2 error is a better indicator than comparing the posteriors because it reflects the output variables directly as they are ones being approximated by the PC expansions. However, this quantity is expensive to compute, especially when the dimension of the parameter space is high, since it needs additional numerical integration for Equation 6.1. The best indicator is perhaps the final optimal design itself, since that is the ultimate solution of the design problem. However, it would be impractically expensive to compute as it involves solving an optimal experimental design problem for multiple p_θ's and n_{quad}'s.

The integrals in the L_2 error expression are computed using a level 15 CC SQ rule (3,502,081 abscissas) — such a high accuracy integration rule is used in an attempt to prevent additional integration errors in computing the L_2 errors. Figures 6-1 and 6-2 show the $\log_{10}$ of the L_2 error contours with respect to p_θ and n_{quad}. Note that n_{quad}

values are approximate, as DASQ is immediately terminated at the iteration after passing n_{quad}, which might not be exactly equal to n_{quad}.

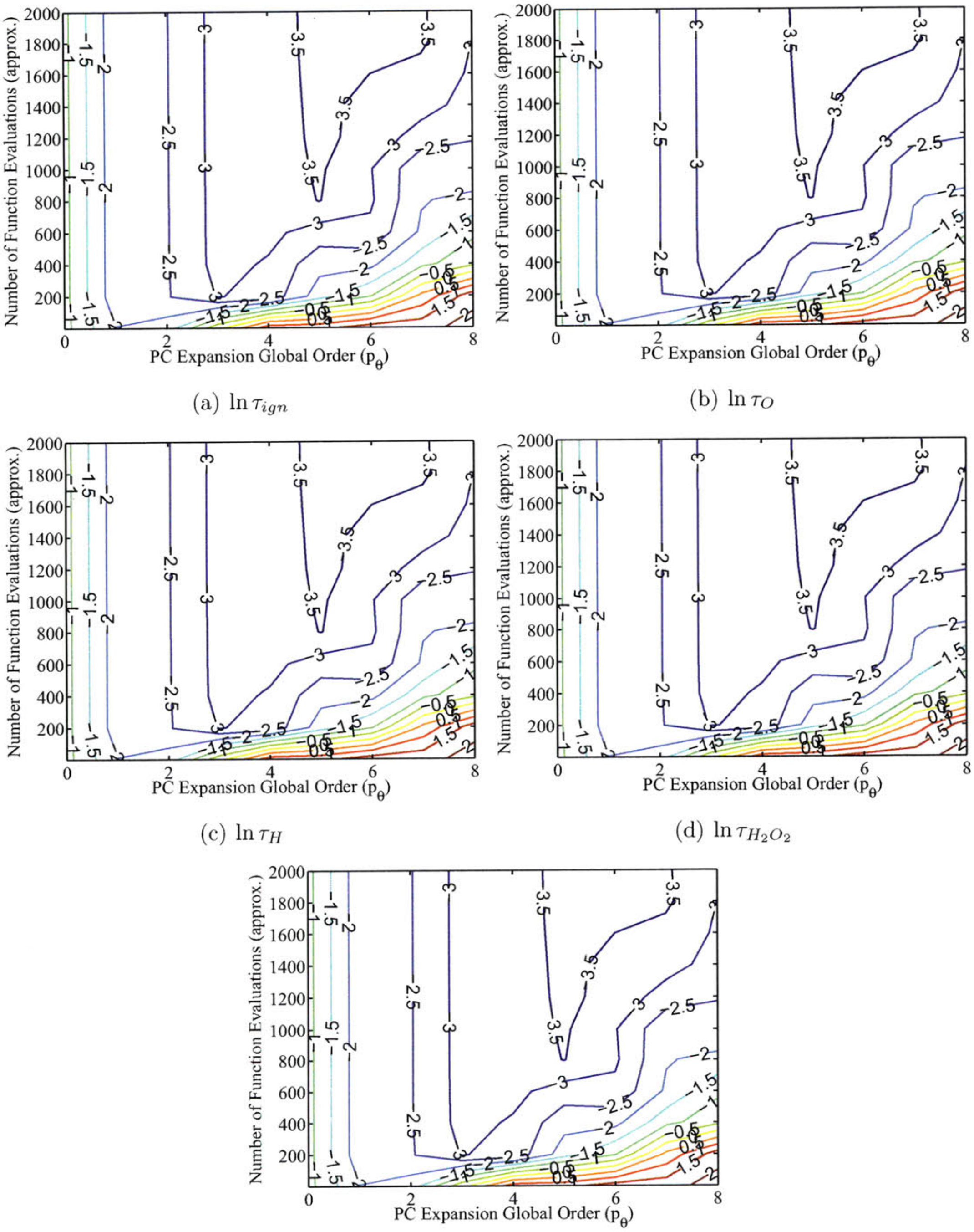
When n_θ is small, the errors actually increase as p_θ is increased. The reason is that those errors are dominated by aliasing (integration) error from the construction of PC expansions, and additional higher order terms are essentially meaningless and further contribute to the total errors. However, when a large enough n_θ is used such that the truncation error is dominating, the additional higher order terms indeed help converge to the correct functions and decrease the errors. Ideally, an n_θ for a particular desirable p_θ should be selected such that no significant advantages can be gained when n_θ is increased further. This occurs at the "corners" of these contour plots, but these locations are difficult to pinpoint *a priori.* For a fixed, high enough n_θ, the asymptotic exponential convergence of PC expansions for smooth functions can be observed.

The errors for the ln of the characteristic time observables are higher than those for the peak value variables, except for the peak value of H_2O_2. This is likely due to that the ln of the characteristic time observables are less regular with respect to the two particular uncertain kinetic parameters selected. The peak value of H_2O_2 may be more difficult to capture, because it carries a double peak in its concentration during the ignition.

6.2 Optimal Experimental Design

The optimal experimental design problem stated in Equation 4.14 is solved with the PC expansions replacing the original Cantera model, but with $n_{\text{in}} = n_{\text{out}} = 10001$. The expected utility contours for both cases are presented in Figure 6-3 using a 1001×1001 uniform grid, with the first subfigure reproduced from Figure 4-5(b) for comparison.

The contours from the PC expansions are overall quite similar to those from the original model, and most importantly, they yield the same optimal experimental design at $(T_0^*, \phi^*) = (900, 0.5)$. However, the PC expansions result is unable to capture the finer details, and has introduced some additional features that do not exist. For



(a) $\ln \tau_{ign}$

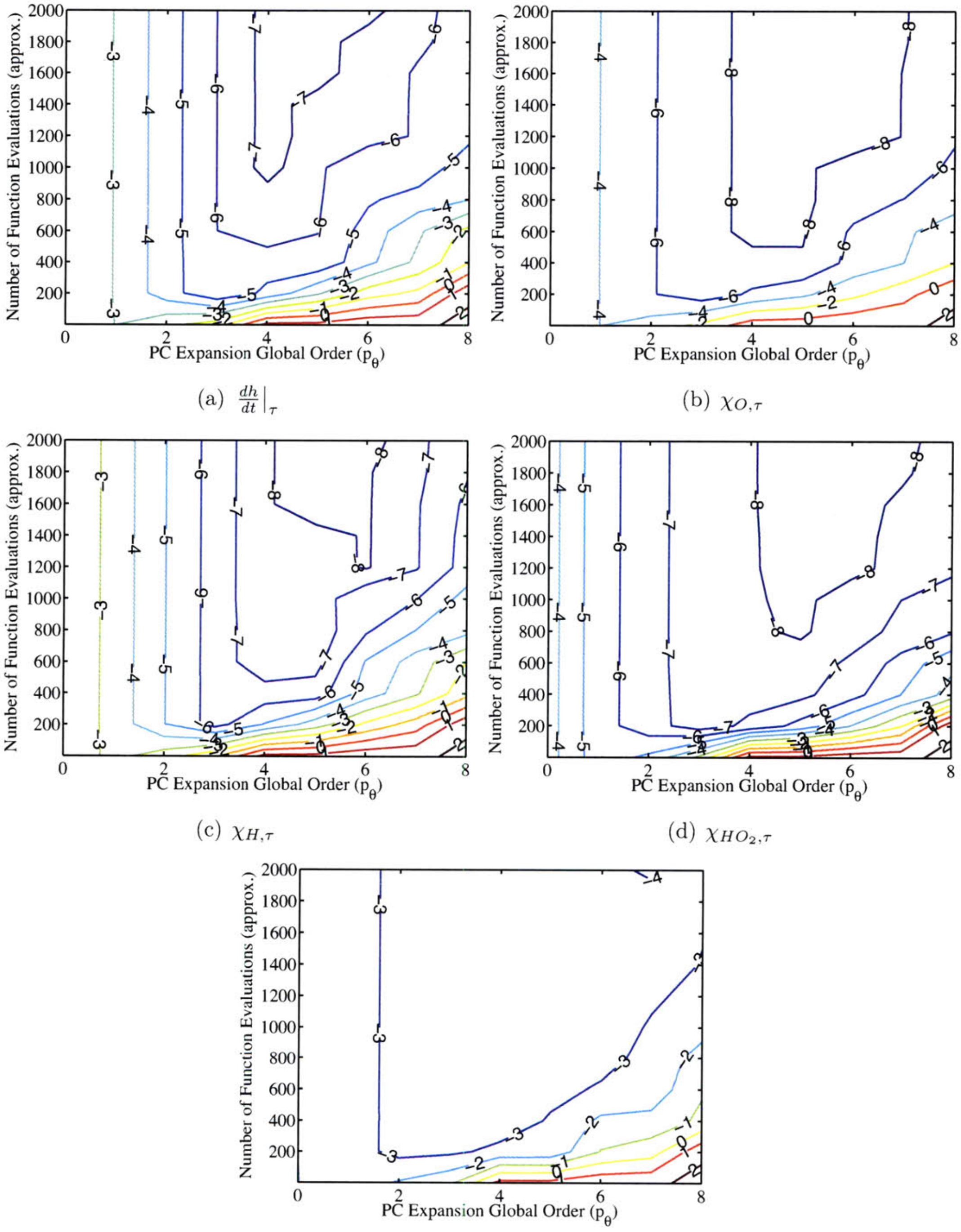
(b) $\ln \tau_O$

(c) $\ln \tau_H$

(d) $\ln \tau_{H_2O_2}$

(e) $\ln \tau_{ign}$

Figure 6-1: $\log_{10}$ of the L_2 errors of the characteristic time observables.



(a) $\frac{dh}{dt}\big|_\tau$

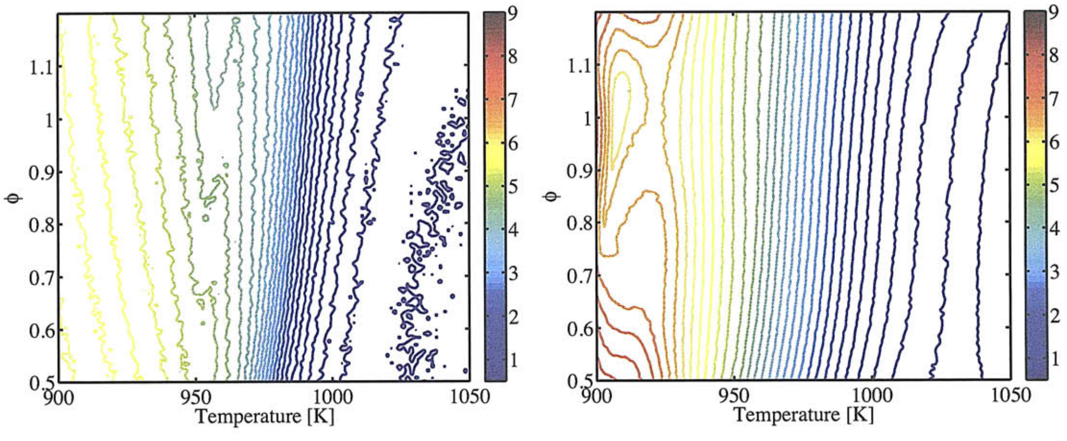
(b) $\chi_{O,\tau}$

(c) $\chi_{H,\tau}$

(d) $\chi_{HO_2,\tau}$

(e) $\chi_{H_2O_2,\tau}$

Figure 6-2: $\log_{10}$ of the L_2 errors of the peak value observables.



(a) Using original model (Cantera)

(b) Using PC expansions

Figure 6-3: Expected utility contours of the combustion problem with 2 design variables T_0 and ϕ, using the original Cantera model (reproduced from Figure 4-5(b)), and using the PC expansions with $p_\theta = 4$ and $n_{\text{quad}} = 1000$.

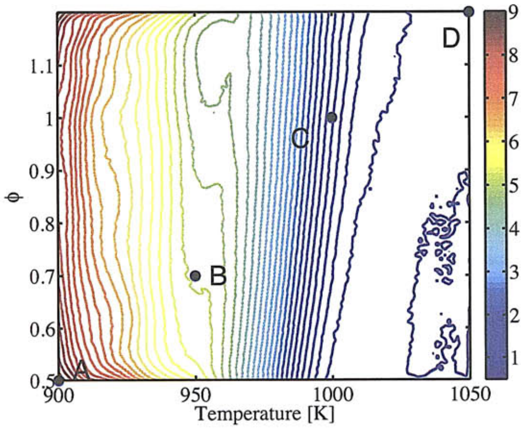


Figure 6-4: Expected utility contours of the combustion problem with 2 design variables T_0 and ϕ, using "the overkill" — PC expansions with $p_\theta = 12$ and $n_{\text{quad}} = 25,000$. The four designs at points A, B, C, and D are used for validating the experimental design methodology by solving the inference problem, described in Section 6.3.

example, the ridge near 950 K is not captured, while a local minimum at around $(T_0, \phi) = (910, 1.0)$ is introduced. These imply that perhaps higher p_θ and n_{quad} are needed to capture the finer details, and yield even better results, .

To test this hypothesis, a high accuracy PC expansion of $p_\theta = 12$ and $n_{\text{quad}} = 25,000$ (dubbed "the overkill") is constructed. Its L_2 errors are shown in Table 6.1. It is used to solve the optimal experimental design problem, and its expected utility contours are presented in Figure 6-4. This time, p_θ and n_{quad} are both high enough to capture the ridge (although not perfectly) and eliminate the local minimum.

Observable	$\log_{10}$ of the L_2 Error
$\ln \tau_{ign}$	-4.79
$\ln \tau_O$	-4.79
$\ln \tau_H$	-4.79
$\ln \tau_{H_2O_2}$	-4.79
$\ln \tau_{ign}$	-4.79
$\frac{dh}{dt}\Big\vert_\tau$	-7.30
$\chi_{O,\tau}$	-10.1
$\chi_{H,\tau}$	-8.71
$\chi_{HO_2,\tau}$	-9.90
$\chi_{H_2O_2,\tau}$	-4.81

Table 6.1: $\log_{10}$ of the L_2 errors of "the overkill" — PC expansions with $p_\theta = 12$ and $n_{\text{quad}} = 25,000$.

6.3 Validation of the Experimental Design Results

Now that the expected utility results are available, how reliable are they in reflecting the goodness of an experiment? This can be answered by solving the inference problems at a number of design points. In particular, designs at points A, B, C, and D are selected. Their conditions are presented in Table 6.2, and are illustrated on the design space in Figure 6-4. Since the expected utility is decreasing from designs A through D, then intuitively, the posterior from the inference problem should reflect the most confidence (since the prior is uniform) at A, less at B and C, and least at D.

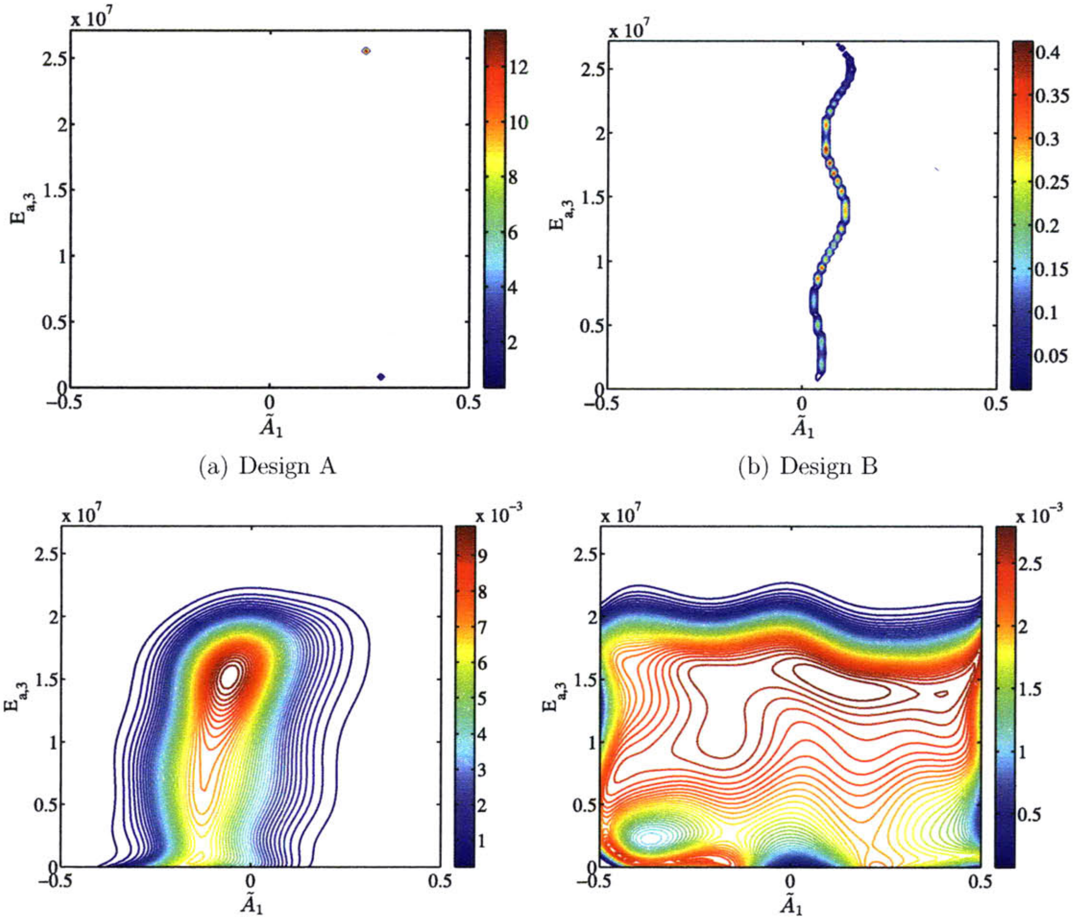
At each design, one single datum is generated from $g_m(\boldsymbol{\theta}, \mathbf{d})$ (not $g_m^p(\boldsymbol{\theta}, \mathbf{d})$) using the recommended values of the kinetic parameters tabulated in Table A, and according to the likelihood described in Section 3.2. Typically, the inference problems would be

Design Point	T_0	ϕ
A	900	0.5
B	950	0.7
C	1000	1.0
D	1050	1.2

Table 6.2: Design conditions at design points A, B, C, and D.

solved using MCMC as described in Section 3.3, especially if the dimension of the parameter space is high. However, for better demonstration purposes, and since only two uncertain parameters are targeted in the combustion study of this thesis, the posteriors are constructed by computing the values on a uniform grid (101 × 101) in the parameter space. The posteriors computed using "the overkill" are shown in Figure 6-5, and for comparisons, the posteriors computed using the original Cantera model (and with the same data from "the overkill" case) are shown in Figure 6-6. With the exception of design A, these two sets of posteriors agree reasonably well, implying that the PC expansions are not only suitable for experimental design, they can also be used for solving the inference problems as well. Posterior for design A is especially sensitive to the forward model approximation error, because it is such an informative set of design conditions that the assumed measurement error is small compared to the PC approximation error.

As expected, the kinetic parameters can be most confidently inferred (almost deterministically) from design A, less confidently in B and C, and least confidently in D. This implies that the expected utility is indeed a good indicator for the goodness of an experiment, and the experimental design methodology is functioning. One reason why the posterior at design A is so close to being deterministic is because the error width from the likelihood model is too small compared to the error due to the approximation from the PC expansions. In the future, error models should not only try to capture measurement noise, but also the inadequacy in modelling. Design A's posterior is also clearly not centered around the recommended values, and this can simply be explained by that, in additional to the PC approximation error, the datum



(a) Design A

(b) Design B

(c) Design C

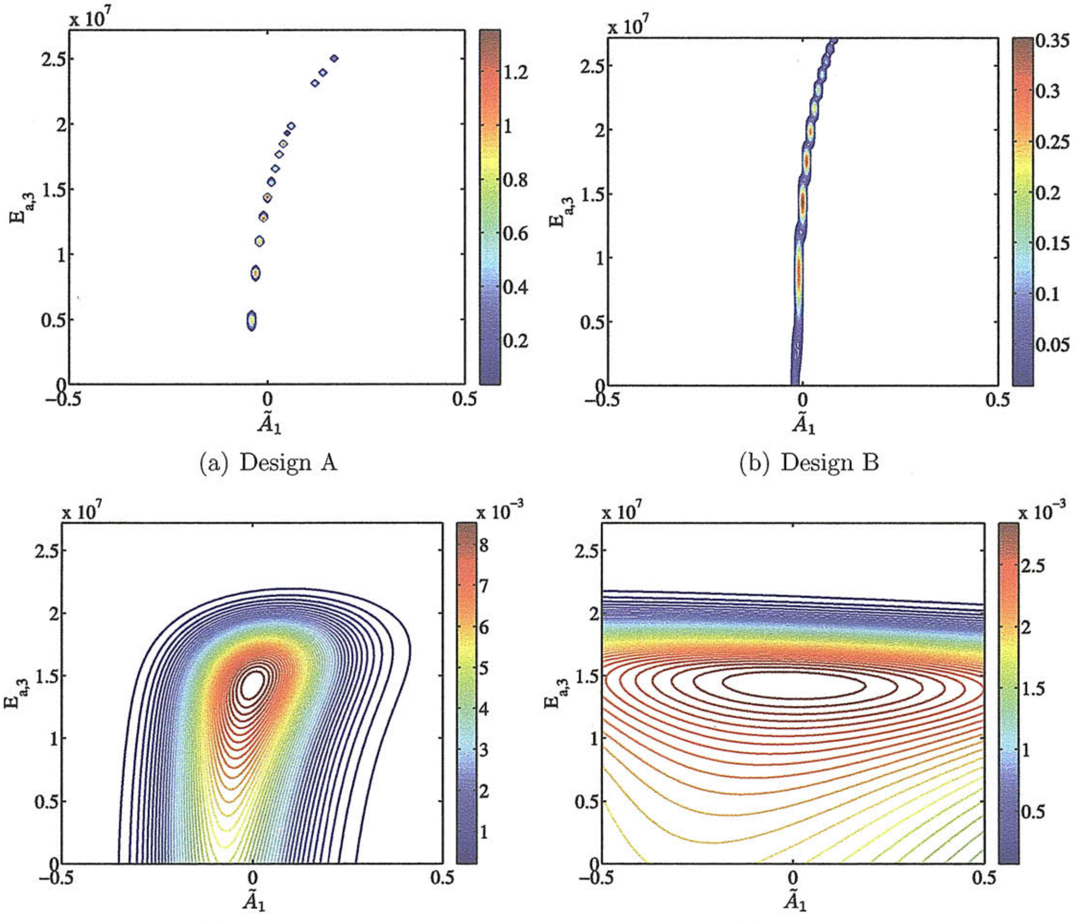
(d) Design D

Figure 6-5: Posterior of the inference problem at the four chosen designs to validate the experiment design methodology, constructed using "the overkill" — PC expansions with $p_\theta = 12$, $n_{\text{quad}} = 25,000$.

is randomly generated, and that only a finite number (one in this case) is available.

6.4 Computational Savings

The original Cantera model has a cost of about 0.2 second per evaluation on a 3GHz CPU, whereas the $p_\theta = 12$, $n_{\text{quad}} = 25,000$ PC expansions are about 600 times faster, while the $p_\theta = 4$, $n_{\text{quad}} = 1,000$ PC expansions are about 6000 times faster. However,



(a) Design A

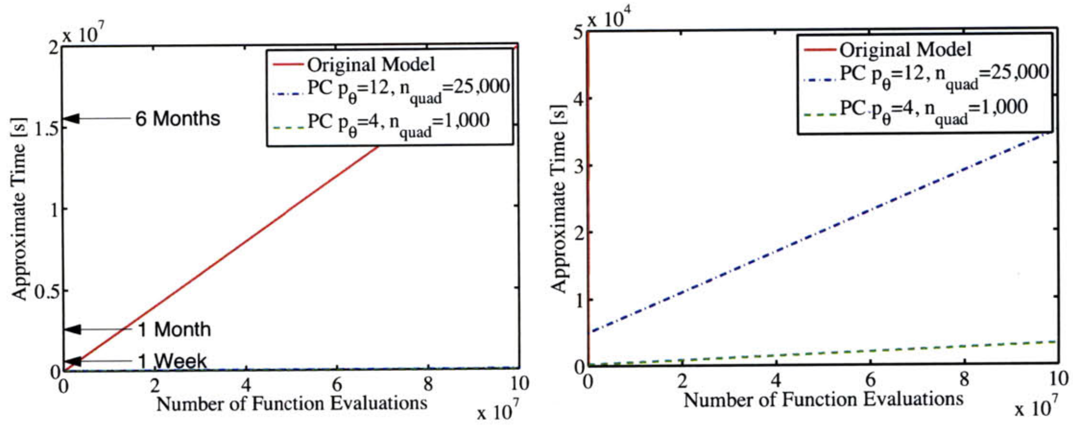
(b) Design B

(c) Design C

(d) Design D

Figure 6-6: Posterior of the inference problem at the four chosen designs to validate the experiment design methodology, constructed using the original Cantera model.

the PC expansions require an up-front investment for construction. This cost can be reduced by using a lower order polynomial and fewer quadrature points, but the tradeoff is accuracy. PC expansions are thus worth-while to invest given a sufficiently large number of required samples. For example, the break-even point for the $p_\theta = 12$, $n_{\text{quad}} = 25,000$ PC expansions is 25,038 forward evaluations. In this study, about 100 million samples are used to perform the final experimental design computations, which translate to a saving of almost 8 months if a single CPU were used. An illustration is shown in Figure 6-7; the tremendous savings are evident.



(a) Full view

(b) Zoomed in

Figure 6-7: Approximate computational time required for the optimal experimental design problem. Figure (b) is the zoomed-in (note the different scale on the y axis) view.

Chapter 7

Conclusions and Future Work

7.1 Summary

Motivated by the need of better understanding of alternative fuels, parameter inference needs to be performed in order to reduce the uncertainty in chemical kinetic parameters, which in turn improves the reliability of numerical models. In order to do this, experiments must be done to obtain data. Moreover, experiments can be designed to optimize the usefulness of their data, and this thesis has developed the framework and tools necessary to perform this numerically.

Figure 7-1 provides a simple visual summary of the important aspects in this thesis. Initially, an expensive model is used to design an optimal experiment such that the resulting data are most informative for inferring uncertain parameters of interest. Unfortunately, the computations become infeasible due to the large number of samples required, and a cheaper surrogate model using polynomial chaos expansions is constructed. The expansions are then substituted into the experimental design framework to compute the optimal design. Data can then be obtained from performing that experiment, and used to infer the parameter values. Similarly, when solving the inference problems, the PC expansions can be used to replace the expensive forward model and further accelerate the process.

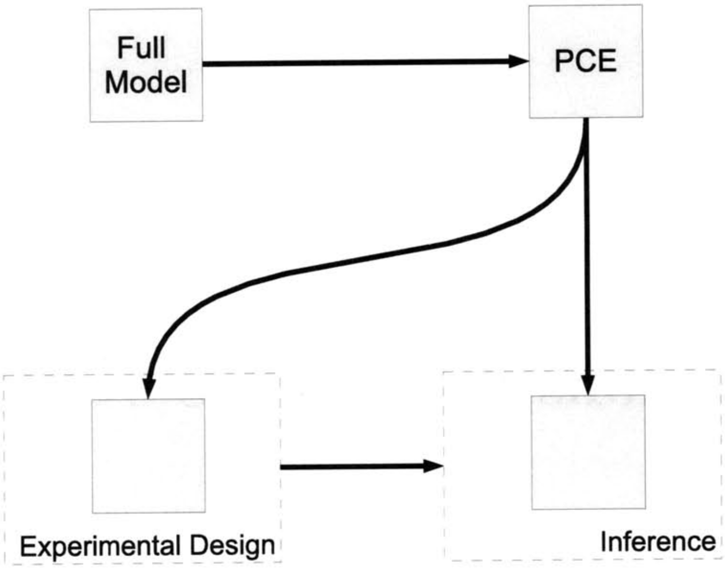


Figure 7-1: Visual summary of important aspects in this thesis.

7.2 Conclusions

This thesis demonstrates that optimal experimental design for nonlinear models may be performed quickly with the help of computationally cheaper PC expansions. The framework developed is very general, and can be applied to many different models. The combustion model considered, however, is especially difficult due to its high nonlinearity, and involves model observables that vary over many orders of magnitude. According to author's knowledge, this is the first work with the application of statistical experimental design to a detailed chemical kinetic system, and the use of PC expansions to detailed chemistry and experimental design.

7.3 Future Work

Stochastic optimization discussed in Section 4.2.3 has yet to be implemented in this framework, and its presence is necessary in order for optimal experimental design to be practical for more than, say, three uncertain parameters. The proposed detection method of polynomial dimension anisotropy using DASQ, described in Section 5.6, is also ready for immediate implementation, and it would make both the construction and evaluation of PC expansions even more efficient. The approximation PC expansions can also be further improved by *hp*-adaptation, where the parameter space and polynomial orders are adaptively refined based on some error indicator.

The error (or likelihood) models can be improved to better match reality. For example, as pointed out near the end of Section 6.3, they should try to capture model inadequacy in addition to just measurement error. Up until this point, the design conditions are also assumed to be deterministic, in the sense that if a desired set of conditions is chosen, then those conditions are achieved exactly. This is often not realistic due to factors such as experimental setup. Uncertainty in design variables themselves thus would make the framework even better.

Finally, the optimal experimental design methodology has a natural extension to sequential experimental design. In particular, tools better suited for sequential data or dynamically evolving systems, such as sequential Monte Carlo, should be explored further.

Appendix A

Recommended Kinetic Parameter Values

Reaction No.	Elementary Reaction			A^*	b^*	E_a^*
R1	$H + O_2$	$\Longleftrightarrow$	$O + OH$	1.915×10^{11}	0.00	6.878×10^{7}
R2	$O + H_2$	$\Longleftrightarrow$	$H + OH$	5.080×10^{1}	2.67	2.632×10^{7}
R3	$H_2 + OH$	$\Longleftrightarrow$	$H_2O + H$	2.160×10^{5}	1.51	1.435×10^{7}
R4	$OH + OH$	$\Longleftrightarrow$	$O + H_2O$	1.230×10^{1}	2.62	-7.866×10^{6}
R5	$H_2 + M$	$\Longleftrightarrow$	$H + H + M$	4.577×10^{16}	-1.40	4.368×10^{8}
R6	$O + O + M$	$\Longleftrightarrow$	$O_2 + M$	6.165×10^{9}	-0.50	0.000×10^{0}
R7	$O + H + M$	$\Longleftrightarrow$	$OH + M$	4.714×10^{12}	-1.00	0.000×10^{0}
R8	$H + OH + M$	$\Longleftrightarrow$	$H_2O + M$	2.240×10^{16}	-2.00	0.000×10^{0}
R9	$H + O_2 + M$	$\Longleftrightarrow$	$HO_2 + M$	6.170×10^{13}	-1.42	0.000×10^{0}
R10	$HO_2 + H$	$\Longleftrightarrow$	$H_2 + O_2$	6.630×10^{10}	0.00	8.912×10^{6}
R11	$HO_2 + H$	$\Longleftrightarrow$	$OH + OH$	1.690×10^{11}	0.00	3.657×10^{6}
R12	$HO_2 + O$	$\Longleftrightarrow$	$O_2 + OH$	1.810×10^{10}	0.00	-1.674×10^{6}
R13	$HO_2 + OH$	$\Longleftrightarrow$	$H_2O + O_2$	1.450×10^{13}	-1.00	0.000×10^{0}
R14	$HO_2 + HO_2$	$\Longleftrightarrow$	$H_2O_2 + O_2$	3.020×10^{9}	0.00	5.816×10^{6}
R15	$H_2O_2 + M$	$\Longleftrightarrow$	$OH + OH + M$	1.202×10^{14}	0.00	1.904×10^{8}
R16	$H_2O_2 + H$	$\Longleftrightarrow$	$H_2O + OH$	1.000×10^{10}	0.00	1.502×10^{7}
R17	$H_2O_2 + H$	$\Longleftrightarrow$	$HO_2 + H_2$	4.820×10^{10}	0.00	3.326×10^{7}
R18	$H_2O_2 + O$	$\Longleftrightarrow$	$OH + HO_2$	9.550×10^{3}	2.00	1.661×10^{7}
R19	$H_2O_2 + OH$	$\Longleftrightarrow$	$HO_2 + H_2O$	7.000×10^{9}	0.00	5.983×10^{6}

Table A.1: 19-reaction hydrogen-oxygen mechanism. Reactions involving M are three-body interactions, where M is a wild-card with different efficiencies corresponding to different species. The recommended values of the kinetic parameters are shown in the last three columns [106], and their units are those described in Chapter 2.

Appendix B

Cantera Input File

The following is the Cantera (version 1.7.0) input file for the 19-reaction H_2O_2 mechanism. Note that the units for this file use the cm–mol–s–cal–K system, instead of the m–kmol–s–J–K system used in this thesis.

```
units(length="cm", time="s", quantity="mol", act_energy="cal/mol")

ideal_gas(name = "h2mech",
          elements = "H  O  N",
          species = """H2  O2  O  OH  H2O  H  HO2  H2O2  N2""",
          reactions = "all"
         )

#------------------------------------------------------------
#  Species data
#------------------------------------------------------------

species(name = "H2",
    atoms = " H:2 ",
    thermo = (
       NASA( [  200.00,  1000.00], [  2.344331120E+00,   7.980520750E-03,
               -1.947815100E-05,   2.015720940E-08,  -7.376117610E-12,
               -9.179351730E+02,   6.830102380E-01] ),
       NASA( [ 1000.00,  6000.00], [  2.932865790E+00,   8.266079670E-04,
               -1.464023350E-07,   1.541003590E-11,  -6.888044320E-16,
               -8.130655970E+02,  -1.024328870E+00] )
             )
    # note = "TPIS78"
       )

species(name = "O2",
    atoms = " O:2 ",
    thermo = (
       NASA( [  200.00,  1000.00], [  3.782456360E+00,  -2.996734150E-03,
                9.847302000E-06,  -9.681295080E-09,   3.243728360E-12,
               -1.063943560E+03,   3.657675730E+00] ),
       NASA( [ 1000.00,  6000.00], [  3.660960830E+00,   6.563655230E-04,
               -1.411494850E-07,   2.057976580E-11,  -1.299132480E-15,
               -1.215977250E+03,   3.415361840E+00] )
             )
    # note = "TPIS89"
       )

species(name = "O",
    atoms = " O:1 ",
    thermo = (
       NASA( [  200.00,  1000.00], [  3.168267100E+00,  -3.279318840E-03,
                6.643063960E-06,  -6.128066240E-09,   2.112659710E-12,
                2.912225920E+04,   2.051933460E+00] ),
       NASA( [ 1000.00,  6000.00], [  2.543636970E+00,  -2.731624860E-05,
               -4.190295200E-09,   4.954818450E-12,  -4.795536940E-16,
                2.922601200E+04,   4.922294570E+00] )
             )
    # note = "L 1/90"
       )

species(name = "OH",
    atoms = " O:1  H:1 ",
    thermo = (
       NASA( [  200.00,  1000.00], [  3.992015430E+00,  -2.401317520E-03,
                4.617938410E-06,  -3.881133330E-09,   1.364114700E-12,
                3.615080560E+03,  -1.039254580E-01] ),
       NASA( [ 1000.00,  6000.00], [  2.838646070E+00,   1.107255860E-03,
               -2.939149780E-07,   4.205242470E-11,  -2.421690920E-15,
                3.943958520E+03,   5.844526620E+00] )
             )
    # note = "TPIS78"
       )

species(name = "H2O",
    atoms = " H:2  O:1 ",
    thermo = (
       NASA( [  200.00,  1000.00], [  4.198640560E+00,  -2.036434100E-03,
                6.520402110E-06,  -5.487970620E-09,   1.771978170E-12,
               -3.029372670E+04,  -8.490322080E-01] ),
       NASA( [ 1000.00,  6000.00], [  2.677037870E+00,   2.973183290E-03,
               -7.737696900E-07,   9.443366890E-11,  -4.269009590E-15,
               -2.988589380E+04,   6.882555710E+00] )
             )
    # note = "L 8/89"
       )

species(name = "H",
    atoms = " H:1 ",
    thermo = (
       NASA( [  200.00,  1000.00], [  2.500000000E+00,   0.000000000E+00,
                0.000000000E+00,   0.000000000E+00,   0.000000000E+00,
                2.547365990E+04,  -4.466828530E-01] ),
       NASA( [ 1000.00,  6000.00], [  2.500002860E+00,  -5.653342140E-09,
                3.632517230E-12,  -9.199497200E-16,   7.952607460E-20,
                2.547365890E+04,  -4.466984940E-01] )
             )
    # note = "L 5/93"
       )

species(name = "HO2",
    atoms = " H:1  O:2 ",
    thermo = (
       NASA( [  200.00,  1000.00], [  4.301798010E+00,  -4.749120510E-03,
                2.115828910E-05,  -2.427638940E-08,   9.292251240E-12,
                2.948080400E+02,   3.716662450E+00] ),
       NASA( [ 1000.00,  6000.00], [  4.172287280E+00,   1.881176470E-03,
               -3.462774080E-07,   1.946578530E-11,   1.762542940E-16,
                6.181029640E+01,   2.957677460E+00] )
             )
    # note = "L 5/89"
       )

species(name = "H2O2",
    atoms = " H:2  O:2 ",
    thermo = (
       NASA( [  200.00,  1000.00], [  4.276112690E+00,  -5.428224170E-04,
                1.673357010E-05,  -2.157708130E-08,   8.624543630E-12,
               -1.775429890E+04,   3.435050740E+00] ),
       NASA( [ 1000.00,  6000.00], [  4.573335370E+00,   4.049840700E-03,
               -1.294794790E-06,   1.972817100E-10,  -1.134028460E-14,
               -1.805481210E+04,   7.042784880E-01] )
             )
    # note = "L 2/93"
       )

species(name = "N2",
    atoms = " N:2 ",
    thermo = (
       NASA( [  200.00,  1000.00], [  3.531005280E+00,  -1.236609870E-04,
               -5.029994370E-07,   2.435306120E-09,  -1.408812350E-12,
               -1.046976280E+03,   2.967474680E+00] ),
       NASA( [ 1000.00,  6000.00], [  2.952576260E+00,   1.396900570E-03,
               -4.926316910E-07,   7.860103670E-11,  -4.607553210E-15,
               -9.239486450E+02,   5.871892520E+00] )
             )
    # note = "TPIS78"
       )

#------------------------------------------------------------
#  Reaction data
#------------------------------------------------------------

#  Reaction 1
reaction(  "H + O2 <=> O + OH",  [1.91500E+14, 0, 16440])

#  Reaction 2
reaction(  "O + H2 <=> H + OH",  [5.08000E+04, 2.67, 6290])

#  Reaction 3
reaction(  "H2 + OH <=> H2O + H",  [2.16000E+08, 1.51, 3430])

#  Reaction 4
reaction(  "OH + OH <=> O + H2O",  [1.23000E+04, 2.62, -1880])

#  Reaction 5
three_body_reaction( "H2 + M <=> H + H + M",  [4.57700E+19, -1.4, 104400],
         efficiencies = " H2:2.5  H2O:12 ")

#  Reaction 6
three_body_reaction( "O + O + M <=> O2 + M",  [6.16500E+15, -0.5, 0],
         efficiencies = " H2:2.5  H2O:12 ")

#  Reaction 7
three_body_reaction( "O + H + M <=> OH + M",  [4.71400E+18, -1, 0],
         efficiencies = " H2:2.5  H2O:12 ")

#  Reaction 8
three_body_reaction( "H + OH + M <=> H2O + M",  [2.24000E+22, -2, 0],
         efficiencies = " H2:2.5  H2O:6.3 ")

#  Reaction 9
three_body_reaction( "H + O2 + M <=> HO2 + M",  [6.17000E+19, -1.42, 0],
         efficiencies = " H2:2.5  H2O:12 ")

#  Reaction 10
reaction(  "HO2 + H <=> H2 + O2",  [6.63000E+13, 0, 2130])

#  Reaction 11
reaction(  "HO2 + H <=> OH + OH",  [1.69000E+14, 0, 874])

#  Reaction 12
reaction(  "HO2 + O <=> O2 + OH",  [1.81000E+13, 0, -400])

#  Reaction 13
reaction(  "HO2 + OH <=> H2O + O2",  [1.45000E+16, -1, 0])

#  Reaction 14
reaction(  "HO2 + HO2 <=> H2O2 + O2",  [3.02000E+12, 0, 1390])

#  Reaction 15
three_body_reaction( "H2O2 + M <=> OH + OH + M",
         [1.20200E+17, 0, 45500],
         efficiencies = " H2:2.5  H2O:12 ")

#  Reaction 16
reaction(  "H2O2 + H <=> H2O + OH",  [1.00000E+13, 0, 3590])

#  Reaction 17
reaction(  "H2O2 + H <=> HO2 + H2",  [4.82000E+13, 0, 7950])

#  Reaction 18
reaction(  "H2O2 + O <=> OH + HO2",  [9.55000E+06, 2, 3970])
```

```
#  Reaction 19
reaction(  "H2O2 + OH <=> HO2 + H2O",  [7.00000E+12, 0, 1430])
```

Bibliography

[1] Cantera 1.7.0 website. http://sourceforge.net/projects/cantera/.

[2] Nitrogen dioxide: Health. United Stated Environmental Protection Agency. http://www.epa.gov/air/nitrogenoxides/health.html, last updated June 2009.

[3] Particulate matter: Health and environment. United Stated Environmental Protection Agency. http://www.epa.gov/air/particlepollution/health.html, last updated May 2008.

[4] Cantera workshop, July 2004. Presentation slides.

[5] C. Andrieu, N. de Freitas, A. Doucet, and M. I. Jordan. An introduction to MCMC for machine learning. *Machine Learning*, 50(1):5–43, January 2003.

[6] A. C. Atkinson and A. N. Donev. *Optimum Experimental Designs*. Oxford Statistical Science Series. Oxford University Press, 1992.

[7] V. Barthelmann, E. Novak, and K. Ritter. High dimensional polynomial interpolation on sparse grids. *Advances in Computational Mathematics*, 12(4):273–288, March 2000.

[8] D. L. Baulch, C. T. Bowman, C. J. Cobos, R. A. Cox, T. Just, J. A. Kerr, M. J. Pilling, D. Stocker, J. Troe, W. Tsang, R. W. Walker, and J. Warnatz. Evaluated kinetic data for combustion modeling: Supplement II. *Journal of Physical and Chemical Reference Data*, 34(3):757–1398, 2005.

[9] D. L. Baulch, C. J. Cobos, R. A. Cox, P. Frank, G. Hayman, T. Just, J. A. Kerr, T. Murrells, M. J. Pilling, J. Troe, R. W. Walker, and J. Warnatz. Evaluated kinetic data for combustion modeling. supplement I. *Journal of Physical and Chemical Reference Data*, 23(6):847–1033, 1994.

[10] S. Bringezu, H. Schütz, M. O'Brien, L. Kauppi, R. W. Howarth, and J. McNeely. Towards sustainable production and use of resources: Assessing biofuels. Technical report, United Nations Environment Programme, 2009.

[11] H.-J. Bungartz and S. Dirnstorfer. *High Order Quadrature on Sparse Grids*, pages 394–401. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2004.

[12] R. H. Cameron and W. T. Martin. The orthogonal development of non-linear functionals in series of Fourier-Hermite functionals. *The Annals of Mathematics*, 48(2):385–392, 1947.

[13] K. Chaloner and I. Verdinelli. Bayesian experimental design: A review. *Statistical Science*, 10(3):273–304, 1995.

[14] A. Chatterjee. An introduction to the proper orthogonal decomposition. *Current Science*, 78(7):808–817, 2000.

[15] M. A. Clyde, P. Müller, and G. Parmigiani. Exploring expected utility surfaces by Markov chains. Technical Report 95-39, Duke University, 1995.

[16] S. D. Cohen and A. C. Hindmarsh. Cvode, a stiff/nonstiff ODE solver in C. *Computers in Physics*, 10(2):138–143, 1996.

[17] T. M. Cover and J. A. Thomas. *Elements of Information Theory.* John Wiley & Sons, Inc., 2nd edition, 2006.

[18] D. F. Davidson and R. K. Hanson. Interpreting shock tube ignition data. *International Journal of Chemical Kinetics*, 36(9):510–523, 2004.

[19] D. F. Davidson and R. K. Hanson. Recent advances in shock tube/laser diagnostic methods for improved chemical kinetics measurements. *Shock Waves*, 2009.

[20] B. J. Debusschere, H. N. Najm, P. P. Pébay, O. M. Knio, R. G. Ghanem, and O. P. Le Maître. Numerical challenges in the use of polynomial chaos representations for stochastic processes. *SIAM Journal on Scientific Computing*, 26(2):698–719, 2004.

[21] F. L. Dryer and M. Chaos. Ignition of syngas/air and hydrogen/air mixtures at low temperatures and high pressures: Experimental data interpretation and kinetic modeling implications. *Combustion and Flame*, 152(1-2):293–299, 2008.

[22] F. Eisenhaber, P. Lijnzaad, P. Argos, C. Sander, and M. Scharf. The double cubic lattice method: Efficient approaches to numerical integration of surface area and volume and to dot surface contouring of molecular assemblies. *Journal of Computational Chemistry*, 16(3):273–284, 2004.

[23] S. Gal, R. Y. Rubinstein, and A. Ziv. On the optimality and efficiency of common random numbers. *Mathematics and Computers in Simulation*, 26(6):502–512, 1984.

[24] W. C. Gardiner and D. B. Olson. Chemical kinetics of high temperature combustion. *Annual Review of Physical Chemistry*, 31(1):377–399, 1980.

[25] A. Gelman. Objections to Bayesian statistics. *Bayesian Analysis*, 3(3):445–478, 2008. Comments by Bernardo, Kadane, Senn, Wasserman, and rejoinder by Gelman.

[26] A. Gelman, G. Roberts, and W. Gilks. Efficient Metropolis jumping rules. In *Bayesian Statistics 5: Proceedings of the Fifth Valencia International Meeting*, pages 599–608. Oxford University Press, 1996.

[27] W. M. Gentleman. Implementing Clenshaw-Curtis quadrature, I methodology and experience. *Communications of the ACM*, 15(5):337–342, 1972.

[28] W. M. Gentleman. Implementing Clenshaw-Curtis quadrature, II computing the cosine transformation. *Communications of the ACM*, 15(5):343–346, 1972.

[29] T. Gerstner and M. Griebel. Numerical integration using sparse grids. *Numerical Algorithms*, 18:209–232, 1998.

[30] T. Gerstner and M. Griebel. Dimension-adaptive tensor-product quadrature. *Computing*, 71(1):65–87, 2003.

[31] W. R. Gilks, S. Richardson, and D. J. Spiegelhalter. *Markov Chain Monte Carlo in Practice*. Interdisciplinary Statistics. Chapman & Hall/CRC, 1996.

[32] P. Glasserman and D. D. Yao. Some guidelines and guarantees for common random numbers. *Management Science*, 38(6):884–908, 1992.

[33] D. G. Goodwin. *Cantera C++ User's Guide*. California Institute of Technology, October 2002.

[34] P. J. Green and A. Mira. Delayed rejection in reversible jump Metropolis-hastings. *Biometrika*, 88(4):1035–1053, 2001.

[35] H. Haario, M. Laine, A. Mira, and E. Saksman. DRAM: Efficient adaptive MCMC. *Statistics and Computing*, 16(4):339–354, December 2006.

[36] H. Haario, E. Saksman, and J. Tamminen. Adaptive proposal distribution for random walk Metropolis algorithm. Technical report, University of Helsinki, 1999.

[37] H. Haario, E. Saksman, and J. Tamminen. An adaptive Metropolis algorithm. *Bernoulli*, 7(2):223–242, 2001.

[38] W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

[39] D. A. Hickman and L. D. Schmidt. Production of syngas by direct catalytic oxidation of methane. *Science*, 259(5093):343–346, 1993.

[40] S. Hosder, R. Walters, and R. Perez. A non-intrusive polynomial chaos method for uncertainty propagation in CFD simulations. In *44th AIAA Aerospace Sciences Meeting and Exhibit*, 2006. AIAA paper 2006-891.

[41] E. T. Jaynes and G. L. Bretthorst. *Probability Theory: the Logic of Science.* Cambridge University Press, 2003.

[42] H. Jeffreys. *Theory of Probability.* Oxford University Press, 3rd edition, 1961.

[43] W. Kahan. Further remarks on reducing truncation errors. *Communications of the ACM*, 8(1):40, 1965.

[44] R. J. Kee, M. E. Coltrin, and P. Glarborg. *Chemically Reacting Flow: Theory & Practice.* John Wiley & Sons, Inc., 2003.

[45] M. C. Kennedy and A. O'Hagan. Bayesian calibration of computer models. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 63(3):425–464, 2001.

[46] J. L. Kerrebrock. *Aircraft Engines and Gas Turbines.* The MIT Press, 2nd edition, 1992.

[47] J. Kiefer and J. Wolfowitz. Stochastic estimation of the maximum of a regression function. *The Annals of Mathematical Statistics*, 23(3):462–466, 1952.

[48] N. L. Kleinman, J. C. Spall, and D. Q. Naiman. Simulation-based optimization with stochastic approximation using common random numbers. *Management Science*, 45(11):1570–1578, 1999.

[49] S. H. Lam and D. A. Coussis. Understanding complex chemical kinetics with computational singular perturbation. *Symposium (International) on Combustion*, 22(1):931–941, 1989.

[50] O. P. Le Maître and O. M. Knio. *Spectral Methods for Uncertainty Quantification: With Applications to Computational Fluid Dynamics.* Springer, 2010.

[51] O. P. Le Maître, O. M. Knio, H. N. Najm, and R. G. Ghanem. Uncertainty propagation using Wiener-Haar expansions. *Journal of Computational Physics*, 197(1):28–57, 2004.

[52] D. V. Lindley. On a measure of the information provided by an experiment. *The Annals of Mathematical Statistics*, 27(4):986–1005, 1956.

[53] B. Liu, Z. Huang, K. Zeng, H. Chen, X. Wang, H. Miao, and D. Jiang. Experimental study on emissions of a spark-ignition engine fueled with natural gas-hydrogen blends. *Energy & Fuels*, 22(1):273–277, 2008.

[54] T. J. Loredo and D. F. Chernoff. Bayesian adaptive exploration. In *Statistical Challenges of Astronomy*, pages 57–69. Springer, 2003.

[55] D. J. C. MacKay. *Information Theory, Inference, and Learning Algorithms.* Cambridge University Press, 2002.

[56] J. L. Maryak and D. C. Chin. Global random optimization by simultaneous perturbation stochastic approximation. *Johns Hopkins APL Technical Digest*, 25(2):91–100, 2004.

[57] Y. M. Marzouk and H. N. Najm. Dimensionality reduction and polynomial chaos acceleration of Bayesian inference in inverse problems. *Journal of Computational Physics*, 228(6):1862–1902, April 2009.

[58] Y. M. Marzouk, H. N. Najm, and L. A. Rahn. Stochastic spectral methods for efficient Bayesian solution of inverse problems. *Journal of Computational Physics*, 224(2):560–586, June 2007.

[59] Y. M. Marzouk and D. Xiu. A stochastic collocation approach to Bayesian inference in inverse problems. *Communications in Computational Physics*, 6(4):826–847, October 2009.

[60] M. D. McKay, R. J. Beckman, and W. J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21(2):239–245, 1979.

[61] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092, 1953.

[62] A. Mira. On Metropolis-Hastings algorithms with delayed rejection. *Metron - International Journal of Statistics*, 59(3-4):231–241, 2001.

[63] W. J. Morokoff and R. E. Caflisch. Quasi-Monte Carlo integration. *Journal of Computational Physics*, 122:218–230, 1995.

[64] P. Müller. Simulation based optimal design. In *Bayesian Statistics 6: Proceedings of the Sixth Valencia International Meeting*, pages 459–474. Oxford University Press, 1998.

[65] P. Müller and G. Parmigiani. Numerical evaluation of information theoretic measures. Technical Report 93-A05, Duke University, 1993.

[66] P. Müller and G. Parmigiani. Optimal design via curve fitting of Monte Carlo experiments. *Journal of the American Statistical Association*, 90(432):1322–1330, December 1995.

[67] P. Müller, B. Sansó, and M. De Iorio. Optimal Bayesian design by inhomogeneous Markov chain simulation. *Journal of the American Statistical Association*, 99(467):788–798, September 2004.

[68] N. Z. Muradov and T. N. Veziroğlu. From hydrocarbon to hydrogen-carbon to hydrogen economy. *International Journal of Hydrogen Energy*, 30(3):225–237, 2005.

[69] H. N. Najm. Uncertainty quantification and polynomial chaos techniques in computational fluid dynamics. *Annual Review of Fluid Mechanics*, 41(1):35–52, 2009.

[70] S. H. Paskov and J. F. Traub. Faster valuation of financial derivatives. *The Journal of Portfolio Management*, 22(1):113–123, 1995.

[71] T. N. L. Patterson. The optimum addition of points to quadrature formulae. *Mathematics of Computation*, 22(104):847–856, s21–s31, 1968.

[72] T. N. L. Patterson. An algorithm for generating interpolatory quadrature rules of the highest degree of precision with preassigned nodes for general weight functions. *ACM Transactions on Mathematical Software*, 15(2):123–136, 1989.

[73] P. H. Peskun. Optimum Monte-Carlo sampling using Markov chains. *Biometrika*, 60(3):607–612, 1973.

[74] E. L. Petersen, D. M. Kalitan, A. B. Barrett, S. C. Reehal, J. D. Mertens, D. J. Beerer, R. L. Hack, and V. G. McDonell. New syngas/air ignition data at lower temperature and elevated pressure and comparison to current kinetics models. *Combustion and Flame*, 149(1-2):244–247, 2007.

[75] A. J. Ragauskas, C. K. Williams, B. H. Davison, G. Britovsek, J. Cairney, C. A. Eckert, J. Frederick, William J., J. P. Hallett, D. J. Leak, C. L. Liotta, J. R. Mielenz, R. Murphy, R. Templer, and T. Tschaplinski. The path forward for biofuels and biomaterials. *Science*, 311(5760):484–489, 2006.

[76] M. T. Reagan, H. N. Najm, R. G. Ghanem, and O. M. Knio. Uncertainty quantification in reacting-flow simulations through non-intrusive spectral projection. *Combustion and Flame*, 132(3):545–555, 2003.

[77] H. Robbins and S. Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3):400–407, 1951.

[78] C. P. Robert and G. Casella. *Monte Carlo Statistical Methods.* Springer Verlag, 2004.

[79] G. O. Roberts and J. S. Rosenthal. Examples of adaptive MCMC. *Journal of Computational and Graphical Statistics*, 18(2):349–367, 2009.

[80] K. J. Ryan. Estimating expected information gains for experimental designs with application to the random fatigue-limit model. *Journal of Computational and Graphical Statistics*, 12(3):585–603, September 2003.

[81] D. A. Schwer, P. Lu, and W. H. Green. An adaptive chemistry approach to modeling complex kinetics in reacting flows. *Combustion and Flame*, 133(4):451–465, 2003.

[82] J. J. Scire, R. A. Yetter, and F. L. Dryer. Flow reactor studies of methyl radical oxidation reactions in methane-perturbed moist carbon monoxide oxidation at high pressure with model sensitivity analysis. *International Journal of Chemical Kinetics*, 33(2):75–100, 2001.

[83] P. Sebastiani and H. P. Wynn. Bayesian experimental design and Shannon information. In *In 1997 Proceedings of the Section on Bayesian Statistical Science*, pages 176–181, 1997.

[84] D. S. Sivia and J. Skilling. *Data Analysis: a Bayesian Tutorial.* Oxford University Press, 2006.

[85] S. A. Smolyak. Quadrature and interpolation formulas for tensor products of certain classes of functions. In *Dokl. Akad. Nauk SSSR*, volume 4, page 123, 1963.

[86] I. M. Sobol. On the distribution of points in a cube and the approximate evaluation of integrals. *USSR Computational Mathematics and Mathematical Physics*, 7(4):86–112, 1967.

[87] J. C. Spall. Simultaneous perturbation stochastic approximation website. http://www.jhuapl.edu/SPSA/, last updated July 2008.

[88] J. C. Spall. Accelerated second-order stochastic optimization using only function measurements. In *Proceedings of the 36th Conerence on Decision & Control*, pages 1417–1424. IEEE, 1997.

[89] J. C. Spall. Implementation of the simultaneous perturbation algorithm for stochastic optimization. *IEEE Transactions on Aerospace and Electronic Systems*, 34(3):817–823, 1998.

[90] J. C. Spall. An overview of the simultaneous perturbation method for efficient optimization. *Johns Hopkins APL Technical Digest*, 19(4):482–492, 1998.

[91] J. C. Spall. *Introduction to stochastic search and optimization: estimation, simulation, and control.* John Wiley and Sons, 2003.

[92] P. B. Stark and L. Tenorio. A primer of frequentist and Bayesian inference in inverse problems. In *Large Scale Inverse Problems and Quantification of Uncertainty.* John Wiley and Sons, 2010.

[93] L. Tierney. Markov chains for exploring posterior distributions. *The Annals of Statistics*, 22(4):1701–1728, 1994.

[94] L. Tierney and A. Mira. Some adaptive Monte Carlo methods for Bayesian inference. *Statistics in Medicine*, 18:2507–2515, 1999.

[95] L. N. Trefethen. Is Gauss quadrature better than Clenshaw-Curtis? *SIAM Review*, 50(1):67–87, 2008.

[96] J. van den Berg, A. Curtis, and J. Trampert. Optimal nonlinear Bayesian experimental design: an application to amplitude versus offset experiments. *Geophysical Journal International*, 155(2):411–421, November 2003.

[97] R. W. Walters. Towards stochastic fluid mechanics via polynomial chaos. In *41st Aerospace Sciences Meeting and Exhibit*, 2003. AIAA paper 2003-413.

[98] B. L. Wang, H. Olivier, and H. Grönig. Ignition of shock-heated H2-air-steam mixtures. *Combustion and Flame*, 133(1-2):93–106, 2003.

[99] C. K. Westbrook and F. L. Dryer. Chemical kinetic modeling of hydrocarbon combustion. *Progress in Energy and Combustion Science*, 10(1):1–57, 1984.

[100] N. Wiener. The homogeneous chaos. *American Journal of Mathematics*, 60(4):897–936, 1938.

[101] D. H. Wolpert and W. G. Macready. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82, 1997.

[102] D. Xiu. Fast numerical methods for stochastic computations: A review. *Communications in Computational Physics*, 5(2-4):242–272, 2009.

[103] D. Xiu and J. S. Hesthaven. High-order collocation methods for differential equations with random inputs. *SIAM Journal of Scientific Computing*, 27(3):1118–1139, 2005.

[104] D. Xiu and G. E. Karniadakis. The Wiener-Askey polynomial chaos for stochastic differential equations. *SIAM Journal of Scientific Computing*, 24(2):619–644, 2002.

[105] D. Xiu and G. E. Karniadakis. A new stochastic approach to transient heat conduction modeling with uncertainty. *International Journal of Heat and Mass Transfer*, 46:4681–4693, 2003.

[106] R. A. Yetter, F. L. Dryer, and H. A. Rabitz. A comprehensive reaction mechanism for carbon monoxide/hydrogen/oxygen kinetics. *Combustion Science and Technology*, 79:97–128, 1991.