Bayesian Design of Experiments for Complex Chemical Systems

by

Kenneth T. Hu

B.S., Carnegie Mellon University (2006)
M.S.CEP, Massachusetts Institute of Technology (2010)

Submitted to the Department of Chemical Engineering
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Chemical Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2011

Author ....
Department of Chemical Engineering
March 25, 2011

Certified by ....
Gregory McRae
Hoyt C. Hottel Professor of Chemical Engineering
Thesis Supervisor

Accepted by ....
William M. Deen
Professor of Chemical Engineering
Chairman, Committee for Graduate Students

Bayesian Design of Experiments
for Complex Chemical Systems

by

Kenneth T. Hu

Submitted to the Department of Chemical Engineering
on March 25, 2011, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Chemical Engineering

Abstract

Engineering design work relies on the ability to predict system performance. A great deal of effort is spent producing models that incorporate knowledge of the underlying physics and chemistry in order to understand the relationship between system inputs and responses. Although models can provide great insight into the behavior of the system, actual design decisions cannot be made based on predictions alone. In order to make properly informed decisions, it is critical to understand uncertainty. Otherwise, there cannot be a quantitative assessment of which predictions are reliable and which inputs are most significant. To address this issue, a new design method is required that can quantify the complex sources of uncertainty that influence model predictions and the corresponding engineering decisions.

Design of experiments is traditionally defined as a structured procedure to gather information. This thesis reframes design of experiments as a problem of quantifying and managing uncertainties. The process of designing experimental studies is treated as a statistical decision problem using Bayesian methods. This perspective follows from the realization that the primary role of engineering experiments is not only to gain knowledge but to gather the necessary information to make future design decisions. To do this, experiments must be designed to reduce the uncertainties relevant to the future decision. The necessary components are: a model of the system, a model of the observations taken from the system, and an understanding of the sources of uncertainty that impact the system.

While the Bayesian approach has previously been attempted in various fields including Chemical Engineering the true benefit has been obscured by the use of linear system models, simplified descriptions of uncertainty, and the lack of emphasis on the decision theory framework. With the recent development of techniques for Bayesian statistics and uncertainty quantification, including Markov Chain Monte Carlo, Polynomial Chaos Expansions, and a prior sampling formulation for computing utility functions, such simplifications are no longer necessary. In this work, these methods have been integrated into the decision theory framework to allow the application of Bayesian Designs to more complex systems.

The benefits of the Bayesian approach to design of experiments are demonstrated on three systems: an air mill classifier, a network of chemical reactions, and a process simulation based on unit operations. These case studies quantify the impact of rigorous modeling of uncertainty in terms of reduced number of experiments as compared to the currently used Classical Design methods. Fewer experiments translate to less time and resources spent, while reducing the important uncertainties relevant to decision makers. In an industrial setting, this represents real world benefits for large research projects in reducing development costs and time-to-market. Besides identifying the best experiments, the Bayesian approach also allows a prediction of the value of experimental data which is crucial in the decision making process. Finally, this work demonstrates the flexibility of the decision theory framework and the feasibility of Bayesian Design of Experiments for the complex process models commonly found in the field of Chemical Engineering.

Thesis Supervisor: Gregory McRae
Title: Hoyt C. Hottel Professor of Chemical Engineering

Acknowledgments

It's hard to believe that I'm finally leaving the McRae lab. It has been five years of both excitement and frustration but I can honestly say I've had a wonderful time at MIT. I want to thank the many remarkable people who have shaped my life here.

First I thank my advisor, Gregory McRae. He has helped me develop my analytical skills and given me the perspective to take on any challenges. I have benefited from the advice and knowledge he has passed on and learned to find my own way. I want to thank him for giving me guidance and life lessons when I needed them and for not counting vacation days. Thanks to the rest of the McRae lab – Arman Haidari, Bo Gong, Kunle Adeyeme, Ingrid Berkelmans, Mihai Anton, Alex Lewis, Chuang-Chuang Lee and Carolyn Seto for the helpful discussions and inspiration, and Joan Chisholm for always listening and putting up with us.

My thesis committee, Professor William Green and Professor Youssef Marzouk, have given me plenty of advice for navigating the academic world and invaluable technical help. Also a special thanks to Professor Marzouk and the UQ reading group, in particular Xun Huan, who have all helped me gain a broader appreciation for the work I've done.

I acknowledge the support of Eni S.p.A. who have funded my Ph.D. work. Roberto Bagatin, Ernesto Rocarro, Letizia Bua, Irene Rapone, and Fabio Oldani, all provided help with my thesis work and hosted me in Novara. It was a great experience and I hope to return someday.

Beyond the basement, I want to thank all my friends and classmates, the Mooseheads, and the Landau IM teams. I have so many treasured memories and I will miss being here with all of you. I'm also glad I found the AJJ club. Remember, the beatings will continue until morale improves.

To Lauren: the past few years would have been incomplete without you. I needed your encouragement, your smiles, and the regular doses of culture more than I could have known.

To my family: your guidance has taught me who I want to be and your support has helped me achieve all that I have. Thank you.

- Ken

Contents

List of Figures

List of Tables

List of Examples

Chapter 1

The Future of Experimental Design

We learn about systems by building, testing, and observing. This is true whether the system is a machine, social group, or computer simulation. Experiments are one way we can control the way information is gathered from a system. Examples include measurements or observations, surveys of experts, or the evaluation of the computer model. The common connection is that information is revealed by running the experiment.

Design of experiments is the exercise of selecting specific experiments in order to gather relevant information. If infinite resources and time were available then we would not be concerned with experimental design. We could simply run experiments until we obtain useful results. In reality, experiments must be selected intelligently in order to collect information in an efficient and effective manner. Unfortunately, it is difficult to know exactly what experiments should be used and how many are required – each study has unique goals and therefore requires different information.

In engineering fields we have a wealth of knowledge about the systems we use. We understand many of the underlying principles that dictate system performance or at least have empirical observations that can describe the system. It is commonplace and even required to use this knowledge for process design and optimization but it is rarely seen in the design of experimental studies. This represents a great opportunity because incorporating this knowledge into the design of experiments process can greatly reduce the required number of experiments - saving time, resources, and money.

The goal of this thesis is to change the way information is gathered by focusing on the treatment of uncertainty. Much attention has been paid to improving process modeling, numerical methods, and optimization. The treatment of uncertainty during the design process often overlooked. The vast majority of experiments being done today either use Classical Design of Experiments or skip the explicit design of experiments step altogether. These experiments are still guided by the expertise of the researcher, but prior knowledge, models, and uncertainty do not explicitly influence the design. This results in the inefficient use of resources and time.

1.1 The Experimental Process and Experimental Design

To better understand how to design experiments, we first examine how an experimental study is carried out. In a study, design, experiments, and analysis are iterated to gather evidence until the goals are reached or resources are exhausted. This is illustrated in Figure 1-1.

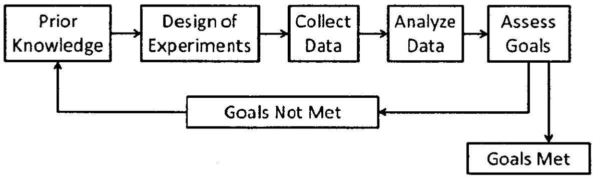


Figure 1-1: Flowchart of the experimental study process showing the role of design of experiments

Ideally, we could somehow determine how useful every possible experiment will be and run only the most useful. Unfortunately, this information only after the experiments are run and data is analyzed. Therefore, more practical design of experiment approaches have been developed. Design of experiments approaches are used to determine the best experiments to run, however, it is important to understand that the root of this problem is estimating the usefulness of a future experiment. Each design of experiments approach accomplishes this in a different way; this thesis focuses on the Bayesian approach.

1.2 The Bayesian Approach to Design of Experiments

In order to incorporate all the prior knowledge and treat uncertainties, we take a general approach that models the entire experimental process. The design step is posed as a decision problem, where the engineer must select the best experiment out of all possible experiments. The framework from decision theory organizes and connects various ideas: how to describe the process knowledge and uncertainty, value information, and use optimization methods. This is called the Bayesian approach or simply Bayesian Designs because many of the methods rely on Bayes' Theorem (See Chapter 6 in particular). These connections are illustrated in Figure 1-2.

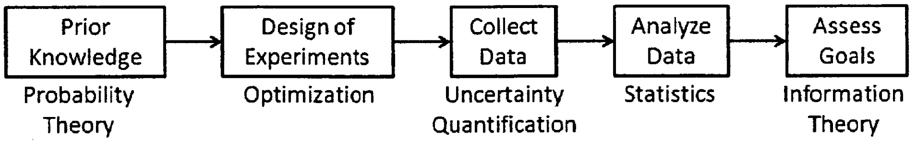


Figure 1-2: The steps in the experimental process can be matched to components of decision theory

Applications in the pharmaceutical and energy industries will serve as initial case studies, however, the techniques illustrated here can be applied to any field and any system. Examples include the design of experiments to characterize equipment, improve system performance, compare models of a system, or even compare two technologies. The ideas are also applicable to other design work such as process design and optimization.

1.3 Motivating Study

The pharmaceutical industry aptly demonstrates the impact of experimental design. Pharmaceutical companies face two critical challenges: lengthy product development times and process variation. The first is driven by economic concerns - the first product in the market captures and retains a disproportionate amount of the market share, even if a superior product is introduced later. The second is due to government regulation of drug quality. The root cause of both product development times and variability in product quality is the difficulty of gathering useful information. Before a drug can be sold, a reliable and consistent manufacturing process must be designed, built, and validated. Each step takes experiments and each experiment costs resources and time.

The case study which motivated this work was an experimental project carried out by an industrial partner. They had used Classical Design of Experiments to run four studies for the scale up of a mechanical separation process. Using classical statistical methods and scale up procedures, they developed a model for the production scale equipment that did not accurately predict the true performance. As a result, an additional study was required which increased the time and money spent.

We will examine why this scale up failed and how the experiments could be improved using Bayesian approach to Design of Experiments in Chapter 8.

1.4 Thesis Structure

The building blocks required for decision theory are introduced in Chapters 2, including probability theory, uncertainty, information theory, statistic/ estimation theory, and optimization. More detailed discussion and related topics are also included in the Appendices.

After the background chapters, the various approaches to design of experiments are described in Chapters 4 and 5. The relationships between experimental studies and decision theory shown in Figure 1-2 are emphasized, focusing on the treatment of uncertainty and current process knowledge. The methodologies used in this work are then detailed in Chapter 3: Methods for Uncertainty Quantification, and Chapter 6: Bayesian Methods for Estimation and Design. Finally several examples and case studies are used to showcase the methods and design of experiments framework in Chapters 7 through 10, followed by the conclusions and discussion.

1.5 Thesis Contributions

This thesis provides a substantially different perspective on design work than traditional Chemical Engineering Process Systems Engineering methods. The focuses is on uncertainty and decision theory rather than optimization and modeling. This perspective follows from the realization that experiments are not always carried out for gaining knowledge. The true goal is to gather the necessary information to make an informed decision. The framework provided in this work gives a clearer picture of the important factors that affect the system and the impact on the future decisions.

The first contribution is the synthesis of the ideas and methodology for properly treating uncertainty for design of experiments on chemical engineering systems. While the Bayesian approach has been previously attempted in various fields including chemical engineering [29, 28, 62, 63], the true benefit has been obscured by the use of linear system models, simplified descriptions of uncertainty, and de-emphasis of the decision theory framework. In this work, no simplifying assumptions are taken and the decision theory framework is followed strictly. This allows a quantitative assessment of the decision making benefits gained by properly treating uncertainty. The methods required include: Bayesian probabilistic modeling of uncertainty, use of prior knowledge, use of non-Gaussian distributions and corresponding Information Theory metrics, Markov Chain Monte Carlo methods, and Polynomial Chaos Expansions. Many of these techniques have not previously been applied to the design of experiments.

The second contribution of this thesis is the application to common chemical engineering systems including chemical kinetics models, a mechanical separations unit, and a process flowsheet. By applying these methods to practical chemical engineering systems the benefits of the Bayesian Design of Experiments approach over the commonly used Classical Designs approach are clearly established.

Chapter 2

Background Theory

This chapter presents the required background theory at the level necessary to understand the remainder of the thesis. Section 2.1 gives a definition for uncertainty and introduces the necessary supporting concepts. Then Sections 2.2 and 2.3 provide the background on probability, statistics, and information theory that provide the basis for Bayesian methods. Finally, Section 2.4 ties all the concepts together to describe the framework used for design of experiments and Section 2.5 discusses the optimization tools that are used in this thesis. Chapters 3 and 6 contain additional details of the most important topics.

2.1 Uncertainty and Modeling

Uncertainty describes the state of imperfect knowledge. For example, the state of a physical system at any instant is well defined and certain but our knowledge about the system is quite uncertain. It is impossible to exactly know the system states because we know only what our instruments and observations tell us. Decisions are often made with incomplete knowledge - we never have enough information about the system to know for sure which decision is best. Therefore it is critical to understand how much knowledge we have and how to improve that knowledge through experiments. In this section, we will discuss the significance, modeling, and quantification of uncertainty and information.

2.1.1 Modeling Uncertainty

A system model relates the system inputs, outputs, states, and parameters to one another. The model is a mathematical representation of the knowledge we have of the system. Using these models helps engineers understand which inputs, states, or parameters have an impact on the outputs. Since our knowledge is often incomplete, all these quantities have some degree of uncertainty; there can be uncertainty in the physical basis of the model, the mathematical representation, the model parameters, etc. The uncertainty impacts the validity of the model and the ability to make decisions based on the model. Before relying on model predictions for design work, engineers must first have a sense of the quality or the uncertainty of those predictions. We need to be able to relate uncertainty the inputs, states, parameters, etc. to the uncertainty in the outputs and measurable variables.

Still, uncertainty is a vague and subjective concept. Given different levels of understanding and knowledge, two observers of the same event may have different amounts of uncertainty. This presents enormous problems for modeling. To address this issue, uncertainty must be represented in a way that is consistent between observers but still have physical meaning.

2.1.2 Uncertainty and Probability Theory

There are several tools that have been used to characterize uncertainty [65, 18, 19], but here we will use probability theory. Probability theory is mathematically rigorous and provides the necessary consistency. For example, although two people can disagree over what probability of a future event, they must agree on the meaning of that probability. In addition, probability is a natural concept to apply to physical systems. While other methods also share these properties, probability theory is an appropriate and widely accepted tool for treating uncertainty in engineering systems. This is sometimes called a Bayesian perspective of uncertainty, in which uncertain quantities are represented by Random Variables. See the books *Understanding Uncertainty* by David Lindley [51] and *Probability Theory* by Jaynes [43] for more discussion.

When applying probability theory to engineering systems, we narrow our scope to parametric uncertainties. In these situations we are most interested in *true* values or *optimal* values of parameters. Unfortunately, we are uncertain what these values are. Rather than describing a fixed value and assuming it is the true value, we use probability theory to describe the the uncertain knowledge of the true value. The key point is that this is modeling our understanding of the parameter, not the parameter - which has a fixed true value.

Statistics and Probability in Engineering

Statistics are used to describe past observations and probability theory to predict future events. Observations of a system always have some variability which is described by a statistical model. These models are then used to estimate parameters and finally to predict future observations. In terms of uncertainty, observations allow us to make inferences about the uncertain state of the system. Using our imperfect knowledge of the system, we can imperfectly estimate the unknown parameters. Using the parameters and the model, we can determine the uncertainty in the model outputs.

Although there is a great deal in common between the classical model development strategy and our current uncertainty framework, there is one major distinction. The classical approach always assumes that uncertainties can be described by Gaussian probability distributions. The Gaussian distribution is useful because it is remarkably simple to define and manipulate, yet it describes naturally varying processes well. Unfortunately, there are many instances where the Gaussian distribution is inappropriate. To be treated accurately, uncertainty must be described with more fidelity so we will use the entire probability density function. Many commonly used statistical techniques cannot treat non-Gaussian distributions, so we will need to develop more flexible methods. These issues will be discussed further in Section 2.2.

2.1.3 Sources of Uncertainty

There are countless sources of uncertainty when modeling a complex system. To attempt to understand them, they are placed in two categories as modified from a report by the Intergovernmental Panel on Climate Change [59] and an article by Draper [19].

Data Observations

Data can never be treated as a true measurement of the system because the measurement tools are imperfect. There is some uncertain discrepancy between the observations and the true system state. What engineers and scientists strive for is to keep the uncertainty low enough so that the observation conveys useful information about the system. To model this, we must understand the sources of uncertainty, including but not limited to: random fluctuations in the measurement equipment, influences from factors outside the system, incorrect interpretation of the observation, and observation of the wrong signal. The great difficult with modeling these sources is that they are often unknown and undetectable. As discussed below, these are typically all lumped together along with terms which do not belong.

Modeling

It is an unfortunate fact that models are never completely correct. The physical world has so many factors which influence system performance that a model cannot capture them all. We must accept this fact and simply try to understand how this affects our confidence in the model. This is called the model output uncertainty.

First of all, the model structure is incorrect. Even models built on physical principles will either be missing terms or have the wrong mathematical representation. This is notoriously difficult to detect, much less model, because it is convoluted with other sources of uncertainty. For this reason, model inadequacy is often lumped into the observation uncertainty because the unexplained variations in the model output are mistaken for observation errors. There are other approaches treating model inadequacy [47] which are not explored here.

Secondly, there is uncertainty associated with the numerical computation of model outputs. This is assumed to be negligible throughout this work. This is a safe assumption, as models should always be validated and verified.

Lastly, there is parametric uncertainty. The parameters have physical meaning and are therefore assumed to have a true value, which remains unknown to us. By representing our knowledge of these parameters with Random Variables, we introduce uncertainty into the system model. This is the only source of model uncertainty which is treated rigorously here.

The Quantification of Uncertainty

After lumping many sources of uncertainty together we are left with two main sources: observations and model parameters. Uncertainty quantification is the analysis of the impact these uncertainties have on the model outputs. Uncertainty refers to the probability density function of the model outputs, while sensitivity is the contribution or relative contribution of a single parameter to the overall model output uncertainty. Methods for uncertainty quantification are discussed in Chapter 3.

2.1.4 Recap

Now that the ideas are in place to define, model, and quantify uncertainty we return to give an engineering perspective. When we attempt to quantify the uncertainty in a model, the purpose is typically to gain insight into a design problem. In a more general sense, uncertainty analysis helps to make decisions. To show how this is done, some background is required on the various tools that come together to solve decision problems.

2.2 Probability Theory

In the previous section, probability theory was introduced as a tool to describe uncertainty in physical systems. This section gives a basic introduction to probability at the level required to understand the remainder of the thesis. Further material is in Appendix A. In particular, Appendix Sections A.2.1 and A.2.2 discuss two crucial concepts which will allow a more thorough understanding of the methods in Chapter 3.

2.2.1 Probability for Quantifying Uncertainty

Probability theory can be used to characterize a wide variety of quantities: future events, uncertain or stochastic quantities, populations, etc. This work deals with parameters in engineering models which are predominantly uncertain quantities that must be a real number (as opposed to abstract concepts).

The purpose of probability theory is illustrated for two applications: a future event and an unknown quantity. When analyzing a future event, it is clear that there are many potential outcomes. Probability theory is used to describe the chances that each particular outcome will actually occur. An example is a coin flip where the two outcomes are heads or tails. Uncertain quantities are less intuitive. A quantity x is called uncertain when it has a fixed *true* or *optimal* value, which is unknown. Probability theory can describe how much we know about this optimal value, denoted x^*. There is no physical event with a concrete outcome. Instead, probability theory describes the uncertainty about the true value. In a simple sense, probability theory describes the chances that a particular value will turn out to be the true value.

We will represent an uncertain quantity, x, with the Random Variable $X(\omega_X)$. For every possible value x, $X(\omega_X)$ describes the probability that $\{\mathsf{x} = x^*\}$. This is still called an outcome, even though there is no intuitive event that has occurred.

The Probability Density Function

One way to define a Random Variable $X(\omega_X)$ is the probability density function:

$$f_X(\mathsf{x}) = \text{Relative probability that } \mathsf{x} \text{ is the true value}$$

A (real-valued, continuous) Random Variable is a function which maps outcomes and values to a corresponding probability density. There are three pieces here: the value x, the outcome ω_X, and probability density f_X. These three pieces make up the Random Variable $X(\omega_X)$. This is illustrated in Example 1.

The set of all possible values is the outcome space Ω_X. For a continuous Random Variable, there are infinitely many valid outcomes in Ω_X so it does not make sense to assign absolute probabilities to each of them. Instead, densities are used to compare the relative probabilities between outcomes. A probability density can range from 0 to ∞.

In the case where there are discrete, specified outcomes, a probability mass function is used instead. This assigns an absolute probability to each outcome.

$$p_X(\omega_X) = \text{Absolute probability of outcome } \omega_X$$

In the coin flip example mentioned above, this could be describe by:

$$p_C(\omega_C) = \begin{cases} \frac{1}{2} & \text{heads} \\ \frac{1}{2} & \text{tails} \end{cases}$$

Example 1: Modeling an Uncertain Quantity

Say we plan to use a ball bearing from a certain manufacturer and need to predict its performance.

We build a model which depends on its radius. The manufacturer guarantees that every bearing has radius r between 4.99 mm and 5.01 mm and that 40% of their bearings are between 4.999 mm and 5.001 mm.

We need to represent the uncertainty in the radius of a particular bearing, for use in the model. In this physical example, the radius could be any distance and so a continuous Random Variable is appropriate. Because we lack additional information about the probability densities, we choose to describe the radius with Random Variable $R(\omega_R)$. The probability density function is:

$$f_R(\mathsf{r}) = \begin{cases} \frac{4}{10}\frac{1}{0.002} & 4.999 < \mathsf{r} < 5.001 \\ \frac{1}{2}\frac{6}{10}\frac{1}{0.018} & 4.99 < \mathsf{r} < 4.999 \\ \frac{1}{2}\frac{6}{10}\frac{1}{0.018} & 5.001 < \mathsf{r} < 5.01 \end{cases}$$

$f_R(\mathsf{r})$ is shown in Figure 2-1. The outcome space is $\Omega_R = [4.99, 5.01]$ and each value r within the inner bounds is 9 times as probable as the other values.

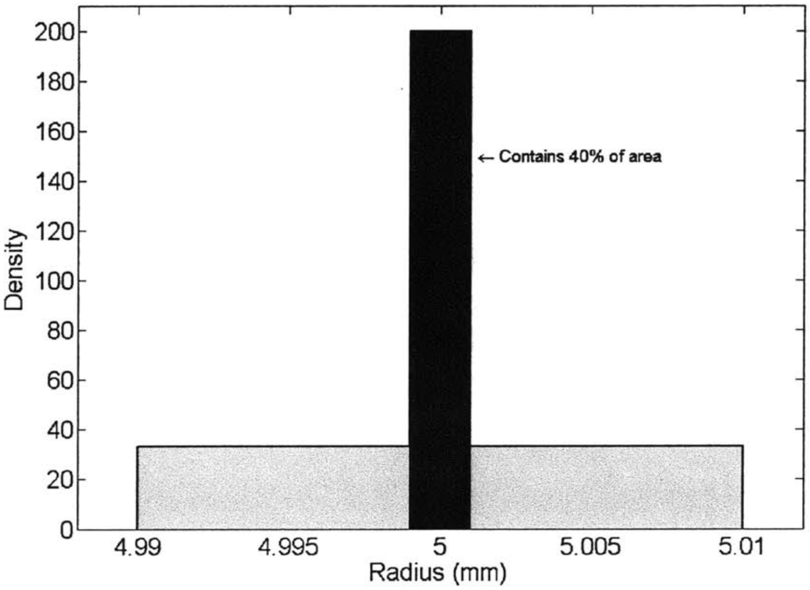


Figure 2-1: The probability density function of the uncertain bearing radius in Example 1

Section A deals with more advanced concepts in probability theory and reconciles general probability theory with the narrow definition that we use.

2.2.2 Statistics

Statistics as a field is concerned with interpreting data. In this work, we use statistics in two ways: estimating model parameters given a new dataset, and estimating properties of a hypothetical population represented by a probability distribution. The estimation techniques are discussed later in Section 6.2. Here we define some basic statistics of probability distributions.

Mean/ Expected Value

As with most statistics, the expected value of a Random Variable is computed using the probability density function. For continuous Random Variables:

$$\mu = E_X\left[X\left(\omega\right)\right] = \int\limits_{\Omega_X} \left(\mathsf{x}\right) f_X\left(\mathsf{x}\right) d\mathsf{x}$$

This is simply the weighted average of all the values the Random Variable can take. The subscript in $E_X\left[X\left(\omega\right)\right]$ indicates that the values of the argument $X\left(\omega\right)$ will be weighted by the probability density function of X. This is more clear below:

$$E_X\left[g\left(X\left(\omega\right)\right)\right] = \int\limits_{\Omega_X} g\left(\mathsf{x}\right) f_X\left(\mathsf{x}\right) d\mathsf{x}$$

Here, the Expected Value are the values of the argument, which is a function of the values x, are weighted by probability density function $f_X\left(\mathsf{x}\right)$.

Mode

The mode is the value (or values) with highest probability density.

$$\arg\max_{\mathsf{x}} f_X\left(\mathsf{x}\right)$$

Median

The median is the value which has half the probability mass below and half the mass above.

$$\int_{-\infty}^{\mathsf{x}^{median}} f_X(\mathsf{x})\, d\mathsf{x} = \int_{\mathsf{x}^{median}}^{\infty} f_X(\mathsf{x})\, d\mathsf{x} = 0.5$$

Variance

The variance of a Random Variable $X(\omega)$ measures the spread of the values x around the mean μ.

$$\begin{aligned} \sigma^2 = \operatorname{var}[X(\omega)] =& E_X\left[X^2\right] - (E_X[X])^2 \\ \sigma^2 =& \int_{\Omega_X} (\mathsf{x} - E_X[X])^2 f_X(\mathsf{x})\, d\mathsf{x} \end{aligned} \tag{2.1}$$

Skewness

The skewness of a Random Variable is a measure of how the values are balanced around the mean μ. For example, a small density of values far greater than μ can be balanced by a large density of values slightly less than μ. This would cause a positive skew. For *some* densities this is analogous to symmetry but a density can be asymmetric and unskewed.

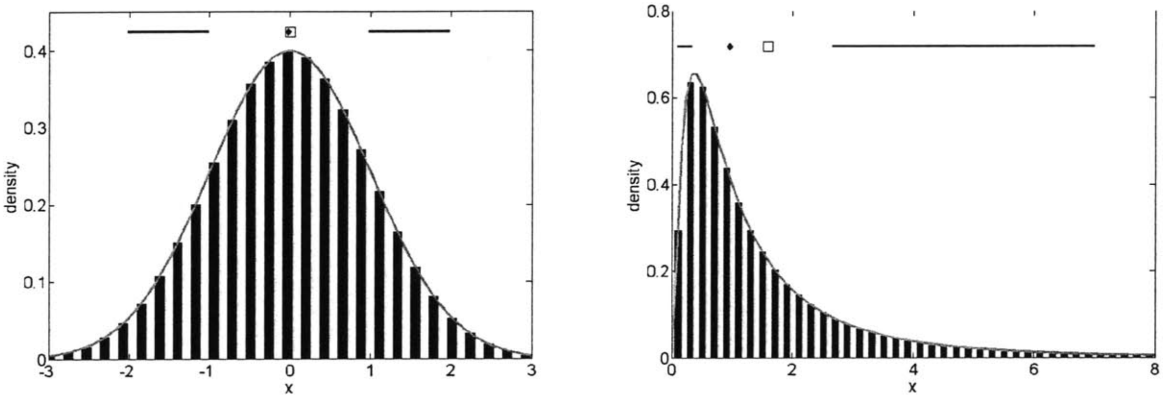
$$\gamma = E_X\left[\left(\frac{X-\mu}{\sigma}\right)^3\right]$$

Descriptive Worth

Each statistic provides a different description of a Random Variable. No single statistic can completely characterize all Random Variables - some statistics are very informative for particular Random Variables but do not capture interesting behavior of other densities. For example, mean and variance completely define the Gaussian and continuous Uniform distributions, but can be confusing on bimodal distributions. Therefore it is important when describing uncertainty to consider which metrics are most appropriate and can give valuable descriptions of a wide range of Random Variables. This will come up again in Section 2.3.

Example 2: Visual Descriptions of Uncertainty Using Probability Theory

Figure 2-2 shows the standard Gaussian distribution and the standard lognormal (the log of this Random Variable is the standard Gaussian). The credible intervals are a Bayesian equivalent of confidence intervals. They show the mean (squares), median (diamonds), 68.2% credible interval (inner end points of lines), and 95.4% credible interval (outer end points). The x-credible interval is centered on the median, and covers $\frac{x}{2}\%$ of the values above and below the median.



(a) Gaussian

(b) Lognormal

Figure 2-2: Example 2 – Random Variables representing uncertainty are shown as histograms, functions, and credible intervals.

2.2.3 The Gaussian Distribution

The Gaussian distribution is the most commonly used Random Variable for describing uncertainty. One reason is that natural populations often have attributes that are well approximated by the Gaussian distribution. In addition, it is a convenient modeling tool because it has very nice statistical properties. For example, summing Gaussian Random Variables results in another Gaussian Random Variable. Also, the entire distribution can be completely defined by two statistics: mean and variance.

To explain why the Gaussian distribution is not always appropriate, we describe the central limit theorem and contrast it with the phenomena associated with physical systems. An in depth discussion can be found in the Ph.D. Thesis of de Mann [17].

The Central Limit Theorem

Let $Y_m(\omega_Y)$ for $m = 1 \ldots M$ be a series of Random Variables, all of which have the identical, non-Gaussian distribution with mean μ and finite variance σ^2. Then let $S_M(\omega_Y) = \frac{1}{M} \sum_{i=1}^{M} Y_m(\omega_Y)$. The central limit theorem states that as $M \to \infty$, the distribution of $S_M(\omega_Y)$ will approach a normal distribution $S_\infty(\omega_Y) \sim N(\mu, \sigma)$. By rescaling, $Z_M(\omega_Y) = \frac{1}{\sqrt{M}} \sum_{i=1}^{M} \left[\frac{Y_m(\omega_Y)-\mu}{\sigma}\right]$ will approach the standard Gaussian $Z_\infty \sim N(0,1)$.

Example 3: The Central Limit Theorem

The Central Limit Theorem is illustrated by plotting the probability density function of $S_M(\omega_Y)$ for increasing values of M for two distributions: the uniform distribution $Y_m(\omega_Y) \sim U(0,1)$ in Figure 2-3 and the exponential distribution $Y_m(\omega_Y) \sim \text{Exp}(1)$ in Figure 2-4.

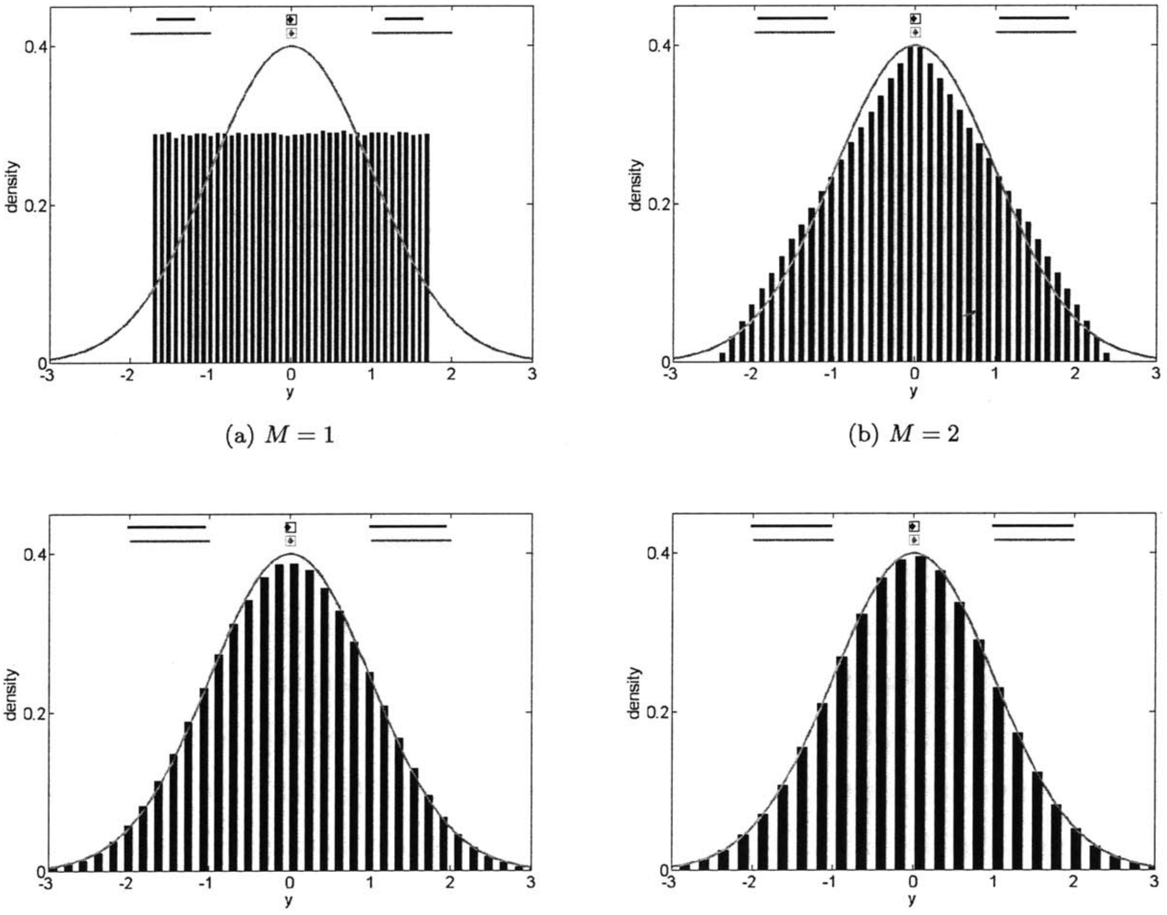
The closer that the original distribution of $Y_m(\omega_Y)$ is to Gaussian, the smaller M needs to be for $Z_M(\omega_Y)$ to approach the normal distribution.

The Central Limit Theorem and Physical Systems

There are two problems with using the central limit theorem to justify modeling physical phenomena with Gaussian distributions. This assumes that all the underlying variations that cause the uncertainty are identically distributed and present in infinite number. The first assumption can be relaxed under certain conditions but the second is the larger concern. There may be a very large number of sources of uncertainty but only a handful will be significant. In that case, we are not looking at an infinite sum of Random Variables but a finite and rather small sum. This new scenario is examined in Example 4.

Example 4: When the Central Limit Theorem Does Not Apply

Here we look at the sum of eight different distributions (Gaussian, χ^2, lognormal, exponential, uniform, beta, Poisson, and gamma). This is a more realistic scenario that might be observed in practice in that there are only a small number of significant uncertainties. Figure 2-5a shows an example where many sources of uncertainty combine to produce a Gaussian uncertainty. This occurs when each of the Random Variables has (approximately) equal variance. Figure 2-5b shows



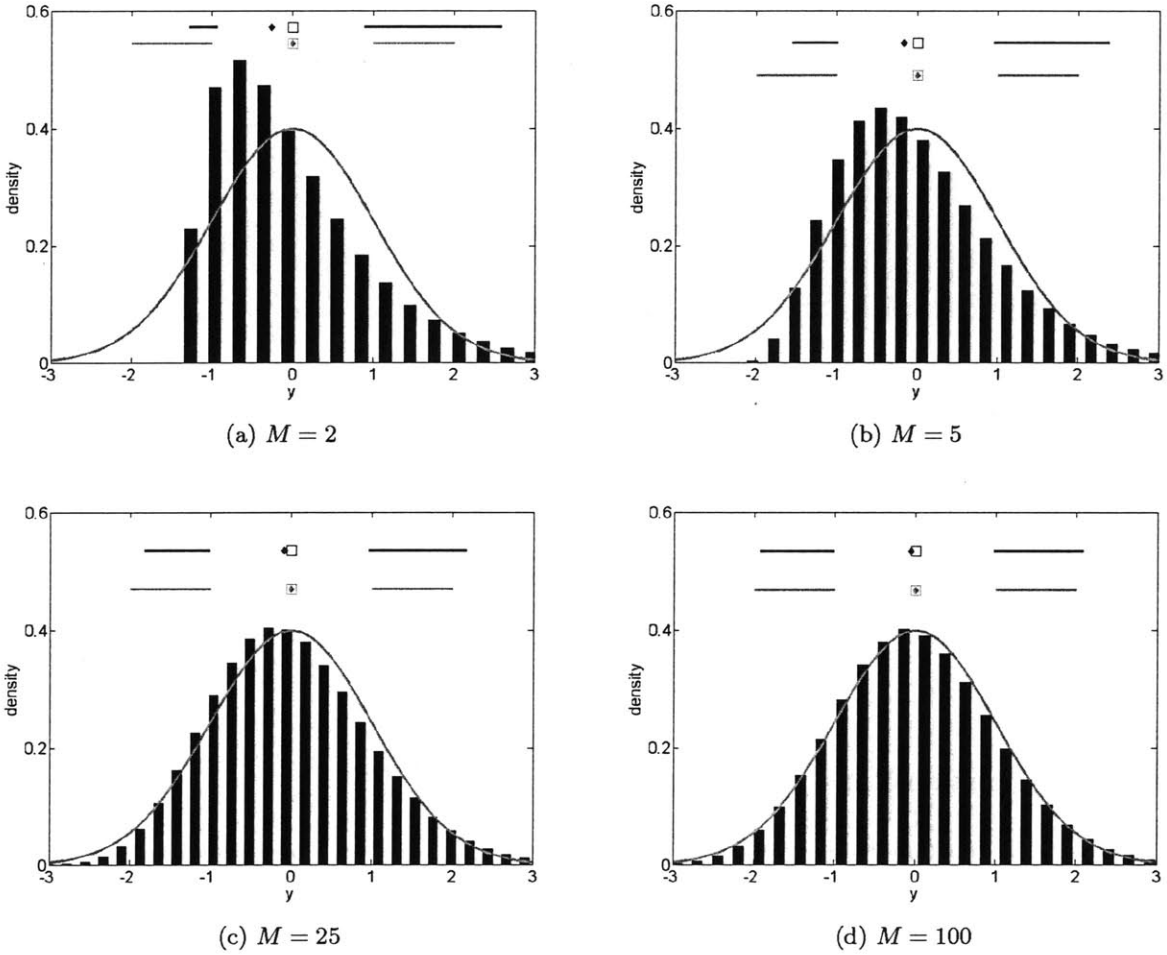
(a) $M = 1$

(b) $M = 2$

(c) $M = 5$

(d) $M = 25$

Figure 2-3: Example 3 – The scaled sum of M uniform Random Variables (bars & top credible interval) approaches the standard Gaussian (— & bottom credible interval) as $M \to \infty$



(a) $M = 2$

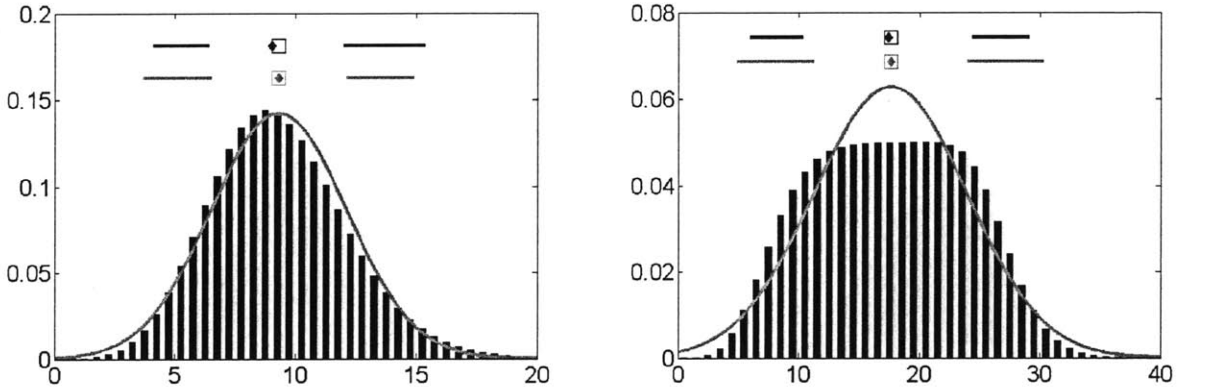
(b) $M = 5$

(c) $M = 25$

(d) $M = 100$

Figure 2-4: Example 3 – The scaled sum of M exponential Random Variables (bars & top credible interval) approaches the standard Gaussian (— & bottom credible interval) as $M \to \infty$

an example where many different distributions are summed together, however, one of them has a larger variance than the rest. Note that these sums are not scaled, so the target is not the standard Gaussian. The Gaussian density function shown is fitted to the mean and variance of of the summation density.



(a) All variances equal

(b) Uniform Random Variable has much larger variance

Figure 2-5: Example 4 – The unscaled sum of eight Random Variables (bars & top credible interval) versus a Gaussian distribution (—) neither is a perfect Gaussian distribution

Intuitively, if one source of uncertainty is larger than the others, it will dominate the sum. This example shows that while the Gaussian distribution is a good approximation for some systems, it is not universal.

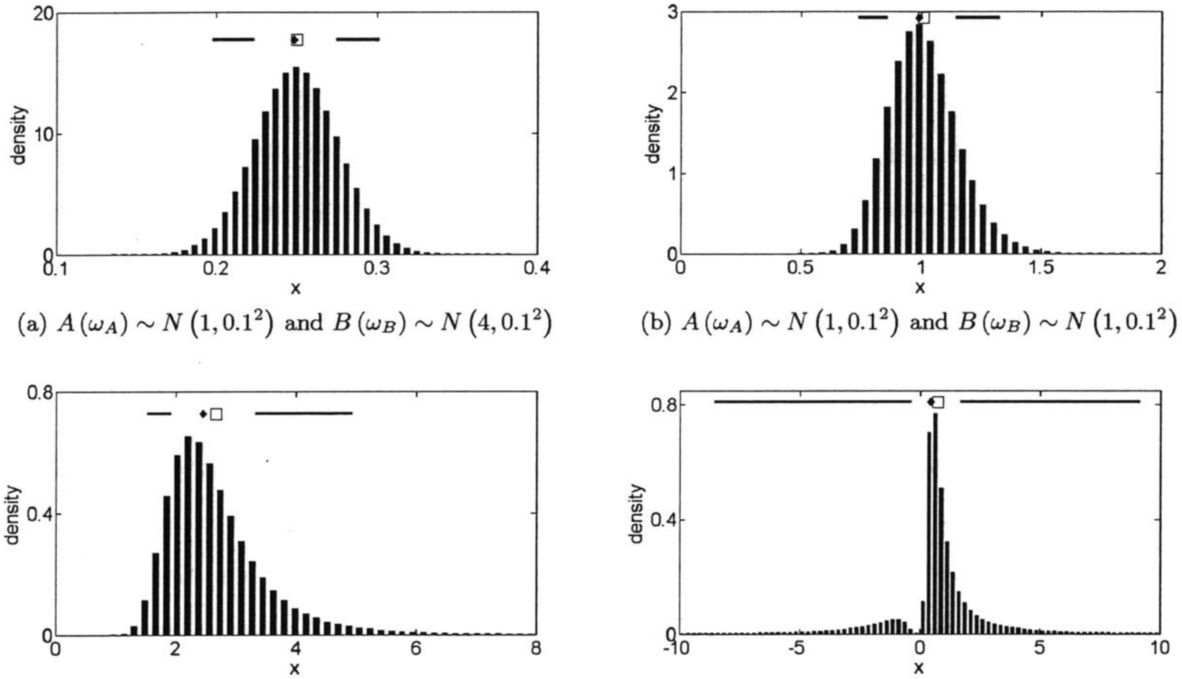
Functions of Gaussian Random Variables

In addition to the fact that not all sources of uncertainty can be represented with Gaussian Random Variables, a non-linear model will very rarely have Gaussian model prediction uncertainty. This is illustrated in Example 5 with the simplest possible model.

Example 5: Gaussian Parametric Uncertainties in a Linear Model

Our model is $ax = b$. Both a and b are uncertain parameters with Gaussian probability density functions, and we want to know the density of x.

This is modeled as $X(\omega_X) = \frac{A(\omega_A)}{B(\omega_B)}$. The results are shown in Figure 2-6 for several choices of $A(\omega_A)$ and $B(\omega_B)$.



(a) $A(\omega_A) \sim N(1, 0.1^2)$ and $B(\omega_B) \sim N(4, 0.1^2)$

(b) $A(\omega_A) \sim N(1, 0.1^2)$ and $B(\omega_B) \sim N(1, 0.1^2)$

(c) $A(\omega_A) \sim N(1, 0.1^2)$ and $B(\omega_B) \sim N(0.4, 0.1^2)$

(d) $A(\omega_A) \sim N(1, 0.3^2)$ and $B(\omega_B) \sim N(1, 1^2)$

Figure 2-6: Example 5 – Probability density function of the ratio of two Gaussian Random Variables, $X(\omega_X) = \frac{A(\omega_A)}{B(\omega_B)}$

Not only is the output non-Gaussian, the probability density function can vary widely depending on the parametric uncertainties. A Gaussian approximation of the output would be a very poor choice.

2.2.4 Rigorous Modeling of Uncertainty

Probability Theory allows for a rigorous description of uncertainty. However, in the past the Gaussian distribution has been used to describe almost all uncertainties. This greatly simplifies the analysis but as shown in the above examples, this is not always a valid assumption. Accurate analysis must begin with an accurate description of uncertainty, meaning that full probability density functions must be used. Variance is a useful statistic that can describe the spread of an uncertain parameter, but as was discussed in Section 2.2.2 it is not the only metric. Other statistics have been created that are better suited for describing the property we are most interested in: information.

2.3 Information Theory

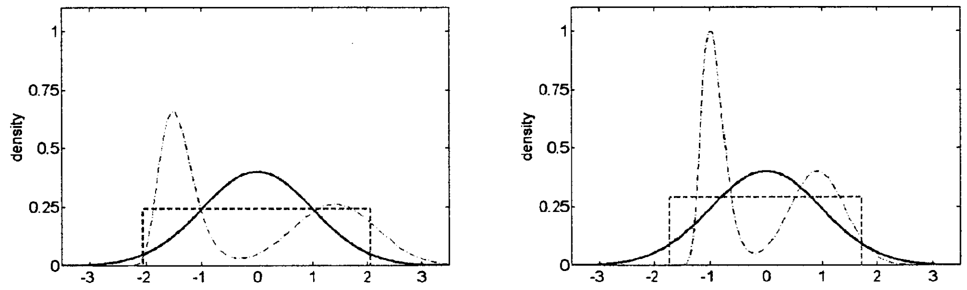
Information theory was developed in the 1940's to answer questions of how fast and how reliable communication methods could be. Along the way, the theory also established metrics for the information content of a random event. While this rather abstract idea has its uses for data compression, it has had a wide ranging impact on practical devices like music players and cell phones. The problem of interest is how to compare two random events and measure them on some common basis. According to information theory, Random Variables can be ranked in terms of the information content they convey [52].

In the Bayesian approach to modeling, probability theory and Random Variables represent the uncertainty of a parameter and the knowledge we have about that parameter. Information theory statistics are then used to describe the information content of these Random Variable. For decision theory and design of experiments, these statistics are necessary to quantitatively measure the usefulness of each experiment. This is the basis of determining the best experiments during the design process.

By far the most common statistic for describing uncertainty is variance. When dealing with Gaussian Random Variables, variance is the only necessary statistic for describing uncertainty, however as stated in Section 2.2 not all uncertainties should be represented with Gaussian Random Variables. This section discusses the properties of the variance statistic and its limitations. Then the ideas of Shannon Information and the entropy statistic are introduced. Appendix B has more details about both as well as some supporting material.

2.3.1 The Variance Statistic

The variance of any Random Variable was given by Equation 2.1. Figure 2-7 shows two sets of probability density functions to illustrate this statistic.



(a) Same variance

(b) Same information content

Figure 2-7: Two sets of probability density functions illustrating the two uncertainty statistics - all have mean 0

Variance is related to the spread of values around the mean. The three probability density functions in Figure 2-7a have variance of 1. The Gaussian Random Variables have most of their mass concentrated around the mean, but the tails of the distributions greatly increase the variance. The uniform Random Variable has no tails but this is balanced by having more density at an intermediate distance from the mean. The third is a mixture (combination) of a lognormal and Gaussian distributions. It has less density at the mean and has two modes, which greatly increases the variance.

If these densities in Figure 2-7a were representing an uncertain parameter, what would they tell us? The Gaussian and mixture densities indicate that the true value is much more likely to be near the the modes, respectively 0 and -1 and 0.9. The uniform density does not indicates a most likely true value but does limit the possibilities to between ± 1.7. Intuitively, these three densities convey a different amount of information about the parameter. Although variance does describe one facet of how much information a variable contains, it does not convey this interpretation of information. Figure 2-7b shows the same distributions, but with new parameters. The mixture and Uniform Random Variables now have higher variances; they are set to have equal *information content*, averaged over the whole density. This statistic is explained below.

2.3.2 Shannon and Differential Entropy

Shannon Entropy is an information metric defined for discrete random variables. Higher entropy corresponds to lower information content. The analogous statistic for continuous random variables is the differential entropy. The formula of differential entropy of the random variable Θ is:

$$h\left(X\left(\omega\right)\right) = -\int\limits_{\Omega_X} f_X\left(\mathsf{x}\right) \log f_X\left(\mathsf{x}\right) d\mathsf{x}$$

For uncertain parameters, high entropy indicates that a lot of data is required to learn the true value. Lower entropy indicates that a lot is already known about the uncertain parameter and less data would be required to learn the true value.

2.3.3 Kullback-Leibler Divergence

Like probability density functions, the differential entropy is a relative quantity because its absolute value is not meaningful. It must be stated along with a reference point. One way to do this is to simple compare the differential entropy of two Random Variables. This gives a measure of information gain. Another statistic is the Kullback-Leibler Divergence:

$$D_{KL}\left(X\left(\omega_\Xi\right) \parallel Y\left(\omega_\Xi\right)\right) = -\int\limits_{\Omega_\Xi} f_X\left(\xi\right) \log \frac{f_X\left(\xi\right)}{f_Y\left(\xi\right)} d\xi \qquad (2.2)$$

The Kullback-Leibler divergence D_{KL} is conceptually similar to a 'distance' between two probability density functions that share a underlying probability space, in this case denoted by the shared value ξ. This occurs, for example, when the two Random Variables being compared represent two characterizations of the same uncertain parameter. The Kullback-Leibler divergence will be used as an information metric for design of experiments. Instead of information gain from one density to another, this is a measure of how different the densities are. It can be interpreted as the information penalty that arises from using $Y(\omega_\Xi)$ to represent an uncertain parameter when $X(\omega_\Xi)$ is the correct representation. For this reason, maximizing the decrease in differential entropies:

$$\begin{aligned}-\Delta h &= -\left(h\left[X\left(\omega_\Xi\right)\right] - h\left[Y\left(\omega_\Xi\right)\right]\right)\\ &= \int\limits_{\Omega_\Xi} f_X(\omega)\log f_X(\xi) - f_Y(\xi)\log f_Y(\xi)\,d\xi \end{aligned} \tag{2.3}$$

is analogous (but not equal) to maximizing the Kullback-Leibler Divergence in Equation 2.2.

2.3.4 Use in Design of Experiments

Optimization is based on comparison and to compare two abstract objects like probability density functions, they must be reduced to a real number. Bayesian designs rely on the two information metrics: differential entropy and Kullback-Leibler Divergence to compare two probability density functions. This will be discussed in Chapter 5.

2.4 Decision Theory

Decision theory establishes a quantitative and objective framework for making decisions under uncertain conditions. In this section, only the Bayesian perspective on statistical decision theory is presented. The alternative, frequentist perspective is similar but does describe the concepts of uncertainty and risk in the same way. A more complete reference is *Statistical Decision Theory and Bayesian Analysis* by Berger [7].

2.4.1 How Decisions are Made

Similar to uncertainty quantification, decision making is a ubiquitous process yet it is neglected in most engineering applications. The simplest explanation of statistical decision theory is that the positive and negative results from all possible decisions are compared and the best decision is the one where the positives outweigh the negatives. The key point is that every action can be assigned value on many different scales (profit, convenience, etc.). In order to compare all outcomes, all values must be put on a common basis.

The framework is remarkably simple, which may explain why it is not used explicitly in most decision making scenarios. Most companies will have a decision making procedure that involves combinations of experiments, consultants, economic analyses, etc. The problem, as mentioned in Chapter 1, is that these procedures often lack a rigorous treatment of uncertainties. Instead, the common approach is to evaluate each action by running a small set of scenarios: best case, baseline, and worst case. This simplistic modeling of uncertainty gives the decision maker a very incomplete picture of the consequences of his or her actions. Because the Bayesian perspective enables the characterization of complex sources of uncertainty using probability theory (see Chapter 3) an equally rigorous framework is required to analyze the uncertainty and make an informed decision.

The framework is illustrated in Example 6 with simple, discrete uncertainties and actions. This is easily extended to the case with an infinite number of possible actions and complex uncertainties.

Example 6: Commuting in the Rain

As an example, on a given morning you must decide between two actions: bike or drive to work. Each has advantages and disadvantages including: cost of gasoline and parking, time of commute, or safety. It is difficult to weigh all these factors objectively. To complicate matters, you know the weatherman has predicted that all day there is a 21% chance of rain but you observe the skies are clear right now. All these aspects play a part in making decisions - the uncertainty, the potential benefits, and the downsides. The role of decision theory is to formalize the process by providing a framework to address each aspect in an objective manner.

2.4.2 Components of a Decision Problem

In the above example we have a single decision with two possible *actions*: take the bike or take the car. In addition there is one uncertain *parameter*: whether or not it is raining. Given these factors, there are four possible consequences: bike, bike in the rain, drive, or drive in the rain. This can be visualized using a decision tree, as shown in Figure 2-8. Decisions take place on squares, the uncertain parameters resolve at circles, and the consequences are represented by triangles. In addition, some initial information may be available before the decision, represented by diamonds.

The decision tree lays out all the possible consequences of the decision. The rest of the problem is in the details of what the consequences mean and how to deal with the uncertainty. In order to compare consequences, a utility function must be constructed, which translates all the advantages

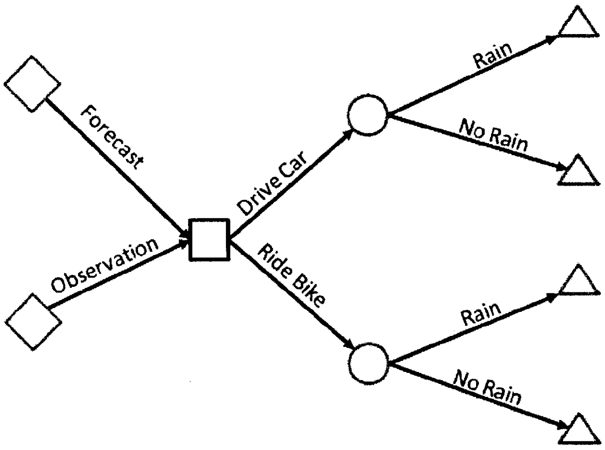


Figure 2-8: Example 6 – The decision tree for the decision of whether to bike or drive, given uncertain weather conditions

and disadvantages of the consequence into a real number. It is common to think of this in terms of money - gas and parking cost money, time and exercise and comfort have a monetary value, getting rained on has monetary costs. Finally, probability theory is used to describe the impact of uncertain parameters. This takes into account the prior knowledge - the weather forecast, as well as any new data - clear skies in the morning.

A decision can only be made after the calculation of the consequences, their utilities, and their probabilities. The simplest method is to select the action that leads to the highest expected utility. This is shown in the following example.

Example 6a: Formal Statement and Solution

To restate (and expand) the above problem: Decision - how to commute to work?

- Set of Actions $\mathcal{A} = \{a_1\text{: takethecar}, a_2\text{: bike}\}$
- Uncertain parameter $\Theta = \begin{cases} \theta_{11} & \text{rain all day} \\ \theta_{12} & \text{rain in the morning} \\ \theta_{21} & \text{rain in the afternoon} \\ \theta_{22} & \text{no rain all day} \end{cases}$

- Set of Consequences $C = \begin{cases} c_1 & = \{a_1, \theta_1 1\} \\ c_2 & = \{a_1, \theta_1 2\} \\ c_3 & = \{a_1, \theta_2 1\} \\ c_4 & = \{a_1, \theta_2 2\} \\ c_5 & = \{a_2, \theta_1 1\} \\ c_6 & = \{a_2, \theta_1 2\} \\ c_7 & = \{a_2, \theta_2 1\} \\ c_8 & = \{a_2, \theta_2 2\} \end{cases}$

  This is often arrayed in a table, as in Table 2.1
- Utility function $U(c)$ for all $c \in C$

  See Table 2.1. Driving without rain is the usual experience so it is given a baseline of 0. The utility or enjoyment of each consequence is then determined relative to this baseline.
- Prior information about the parameters $p_\Theta(\theta)$. This is the information from the weatherman.

  A probability mass function must be assigned: $p_\Theta = \begin{cases} \theta_{11} & 0.01 \\ \theta_{12} & 0.1 \\ \theta_{21} & 0.1 \\ \theta_{22} & 0.79 \end{cases}$
- Initial observations $D^{(i)}$ made before the decision: clear skies in the morning.
- Model that relates the initial observations and parameters $D^{(i)} = \mathcal{M}[\Theta]$. A simple mathematical model for this example is that observing clear skies in the morning makes it highly (90%) unlikely that it will rain in the morning, but does not affect the likelihood of rain in the afternoon. This inference shows that your own prediction for rain is then:

$$p_\Theta|X = \begin{cases} \theta_{11} & 0.001 \\ \theta_{12} & 0.011 \\ \theta_{21} & 0.111 \\ \theta_{22} & 0.877 \end{cases}$$

See Section 6.2 for more details on Bayesian parameter estimation.

Table 2.1: Table of Consequences and Associated Utilities

	Take the car a_1	Take the bike a_2
rain all day θ_{11}	$U(c_1) = -20$	$U(c_5) = -100$
rain in the morning θ_{12}	$U(c_2) = -10$	$U(c_6) = -80$
rain in the afternoon θ_{21}	$U(c_3) = -10$	$U(c_7) = -20$
no rain all day θ_{22}	$U(c_4) = 0$	$U(c_8) = 20$

Result

Now we have all the information and several ways to represent it. A high level visual representation is the decision tree, and a mathematical representation is the utility of each consequence and probability of each parameter outcome. These are combined in Figure 2-9

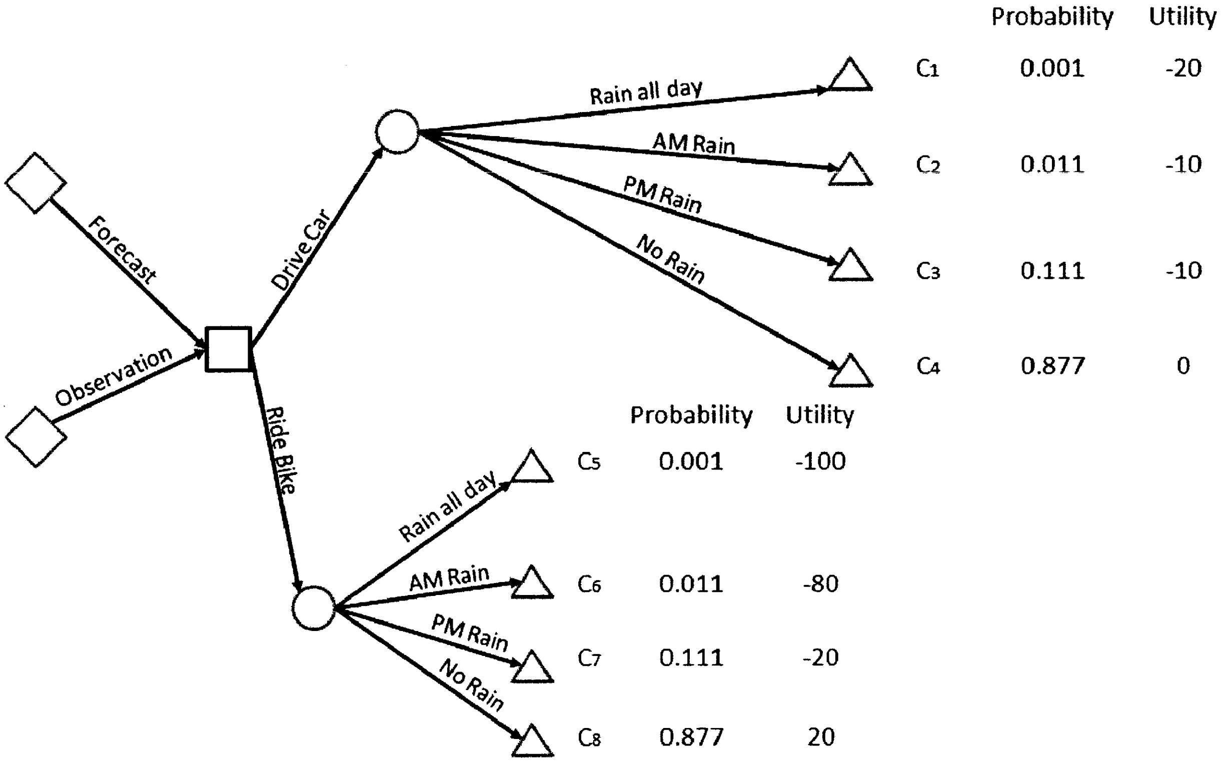


Figure 2-9: Example 6 – Detailed decision tree showing all consequences, their probabilities, and their utilities

Finally we can determine the expected utility for each decision, averaged over the uncertain parameter. The expected utility of driving is ≈ -1 and the expected utility of biking is 14. So biking is the best action.

Note about Initial Observations

In this work, there is never an initial observation before the decision is made, this detail is only included to match the existing literature. Note that prior to the decision, an initial observation can be assimilated into the existing prior information as shown in Example 6 and described in Section 6.2. This will simplify the terminology - prior information will be taken to mean any knowledge about the system, including previously observed data.

2.4.3 Risk Informed Decision Making

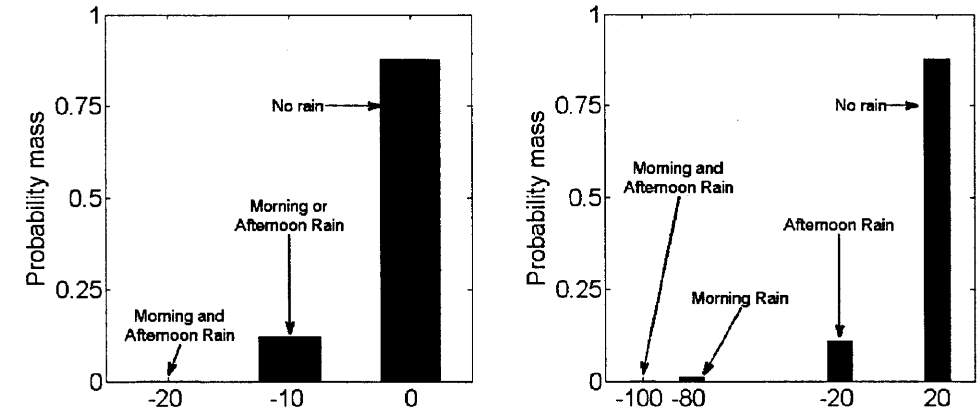
The basic framework for Bayesian decision theory can be condensed into four steps:

1. Define all the potential actions
2. Propagate parametric uncertainty through the model to identify all possible consequences and their probabilities of occurring
3. Define a utility function to rank all possible consequences
4. Choose an action by comparing distributions of utility

While the framework is simple, the steps can be quite difficult - requiring both numerical methods and engineering and economic judgments. Many choices of utility functions seem arbitrary so it helps to explain the concepts behind them, namely: utility and risk.

Utility is a quantitative measure of satisfaction, usefulness, attractiveness, quality, etc. for an option or consequence. It is an objective way to compare two choices - but the selection of a utility function is quite subjective. We define risk to be the combined influence of utility and probability. It is described for every outcome by the distribution of utilities. For Example 6, the utility distributions for both actions are shown in Figure 2-10.

The final decision is based on an assessment of the utility distribution according to a risk metric. In the above example, and by far the most common risk metric in use, is the Expected Value. This is an objective metric but its selection is an imperfect process. As mentioned within the discussion of statistics in Section 2.2.2 there is no single perfect metric.



(a) Utility of Driving

(b) Utility of Biking

Figure 2-10: Example 6 – Probabilistic description of risk: probability mass functions of utility for both potential actions

Risk Metrics

Using Expected Value as the risk metric should be a choice, not a default. It corresponds with the goal of achiving the best average outcome, when the decision maker is neither conservative (risk averse) - unwilling to take large negative risks even with the possibility of gains, or aggressive (risk seeking) - willing to tolerate the possibility of large losses, if they are balanced by the possibility of large gains.

In reality, most decision makers are not risk neutral. The choice of risk metrics should reflect the attitude of the decision maker. Various other risk metrics exist to match these attitudes, including robust metrics (minmax), chance constrained metrics, mean variance, and others. There are not many references that discuss these, as the theory is still being developed. The best source is ***Robust** Optimization* by Ben-Tal et al [6].

These risk metrics correspond to the objective functions used in the optimization algorithms, discussed in the next section.

2.5 Optimization

Design work relies on optimization to select the best option. Process design is the attempt to determine the best configuration of equipment for a specific task. This requires an optimization over all possible configurations. In the same way, model based design of experiments attempts to determine the most useful experiments - the best designs. The difficultly with optimization within the decision theory framework is that we deal with uncertain model outputs and data predictions. When dealing with uncertainty the traditional, deterministic optimization methods often fail. This requires a new approach which falls under the umbrella term "optimization under uncertainty".

2.5.1 Deterministic Optimization

Deterministic optimization is a mature field, with a wide range of methods suitable for design problems. These are applicable for problems where the objective function can be evaluated accurately and its derivative can be computed analytically or numerically. A relevant example is sequential quadratic programming which can solve nonlinear, convex problems with constraints on the design variables. Good references are *Convex Optimization* by Boyd and Vandenberghe [12] and *Convex Analysis and Optimization* by Bertsekas et al. [8]. Sequential quadratic programming is useful for convex problems - those with convex objective functions and convex design spaces.

If the problem is non-convex then a global optimization method is required. These problems must be broken down into a sequence of convex problems whose solutions converge to the solution of the non-convex problem. For more details see *Global optimization : deterministic approaches* by Horst and Hoang [38].

2.5.2 Optimization Under Uncertainty

Optimization under uncertainty deals with problems with objective functions that are functions of uncertain quantities. These uncertainties are represented with Random Variables, as discussed in Section 2.1. In Decision Theory, this is the distribution of utilities corresponding to the uncertain consequences of a particular action. To be optimized, this distribution must be reduced to a real number, i.e., a summary statistic, by applying a risk metric.

There are three important features of the decision problems of interest here. First, the actions are the values of continuous design variables. On a physical system, these are the 'knobs' that the experimenter can adjust to change the system properties. The design space is therefore restricted by the physical limitations on the system. It is possible that the design space is non-convex if the design variables cannot be varied independently. Secondly, the summary statistics should be continuous because even if the model output has discontinuities, small perturbations of the design variables should not radically change the *distribution* of uncertain model outputs. However, the statistics are computed numerically, with computational cost increasing with desired accuracy. Lastly, the summary statistic used as an objective function is not guaranteed to be convex. Proving convexity would require an analytical solution to the summary statistic, which rarely exists.

The features of this optimization problem presents difficulties for the deterministic optimization methods. The non-convexity can be dealt with with global optimization concepts, but the inaccuracy of the objective function poses a fundamental problem: the derivative or ranking of objective function values cannot be trusted. The challenges here have lead to the development of methods for noisy optimization. The issue at hand is the signal-to-noise ratio where the signal is the change in objective function over the design space. If the uncertainty in computing the objective function (the noise) is too high the optimization algorithm will not be able to distinguish between different designs. Reference include *Iterative Methods for Optimization* by Kelley [46] and *Introduction to Stochastic Search and Optimization* by Spall [74].

2.5.3 Genetic Algorithms

The term Genetic Algorithm covers a wide array of methods that perform optimization by tracking a population as it evolves [85]. The population grows by adding new members and declines when the unfit members die. There are many variations on how members are added and subtracted from the population, but the overall concept is very simple. In the decision theory context, each member of a species is an action and the fitness is measured by the objective function. This class of methods is attractive because it does not require any knowledge about the models and it is very robust to noisy evaluation of the objective function. The drawback is that there is no way to guarantee global or local optimality so the solutions are always approximate.

2.5.4 Nelder-Mead and Implicit Filter Algorithms

Similar to genetic algorithms, the Nelder-Mead and Implicit Filter algorithms [11] do not need any information besides objective function evaluations. No derivative or convexity information is used. Instead, these algorithms search the design space with a stencil - basically an N-dimensional structure within the N dimensional design space. The stencil moves around the design space according to some heuristic until a local optimum is found. Because there is some structure to the search, the algorithms can determine when an optimum is reached and terminate, however there is still no guarantee of global optimality. These methods are less robust to noise than genetic algorithms.

2.5.5 Global and Greedy Strategies

Designing an experimental study is not just a question of which experiments to run, but also how many. When the design is formulated as an optimization problem in the decision theory framework, the number of experiments must be specified. If designs of different size are required, there are two strategies: global and greedy.

The global strategy is to formulate and solve each optimization problem independently. If there are V design variables and P design points are required, then the optimization problem will be a search over $V \times P$ dimensions. This is repeated for all the desired values of P. Despite the name, this does not guarantee a global optimum. The term global indicates that the search is over all possible designs, instead of a restricted set of designs.

A greedy strategy is to compute the designs in series, with each successive design building on the last. For example, if we first compute a design with P design points, it is a search over $V \times P$ dimensions. We will call the solution of this problem design $x^{(P)}$. The next design we wish to compute has $P+2$ design points. Instead of searching $V \times (P+2)$ dimensions, we assume that this design includes $x^{(P)}$ plus two new points. This optimization problem is only a search over $V \times 2$ dimensions and is much easier to solve. However, this is not guaranteed to be the global optimum because most of the design space is not searched.

2.5.6 Optimization for Design of Experiments

For the Design of Experiments applications in this thesis, both a genetic algorithm and implicit filter algorithm were applied using both global and greedy strategies. Global optimization is not used because the algorithms are not effective and we are not concerned with locating an absolute global minimum. Instead, the focus of this work is on formulating the objective function so that it truly represents the value of interest. This requires the most accurate modeling, characterization of uncertainty, utility, and risk. Although a global optimum is desirable, it is a secondary issue

- an approximate solution to the correct optimization problem is better than a guaranteed global optimum of a simplified optimization problem.

Chapter 3

Methods for Uncertainty Quantification

A major component of decision making is analyzing uncertainty and understanding its impact. This chapter discusses the mathematical methods use for uncertainty analysis or quantification. These methods play a major role in the design of experiments introduced in Chapter 5. The first step is to state some examples which will be used throughout the thesis to illustrate the concepts and algorithms. Then we introduce the uncertainty quantification problem and frame it such that the methods can be applied to any model. Finally two classes of methods are discussed: Monte Carlo and Polynomial Chaos Expansions. The last few sections focus on the Polynomial Chaos Expansions, since these will be used extensively throughout the remainder of the thesis.

The terminology used here is consistent with the Bayesian methods chapters. It may be helpful to read the first two sections of Chapter 6 before continuing.

3.1 Examples

Throughout this chapter, examples are used to illustrate the concepts and methods. After the methods are explained, the techniques are also demonstrated on more complex models.

Example 7: Target Practice

We start with a simple target practice problem. There is a target 100 m away, and we want to hit it with a cannonball. Unfortunately, the cannon we have is rather imprecise and the exit speed and angle of the cannonball are uncertain. We want to know the probability of striking within 1 m of the target given our uncertain aim.

The model is an Ordinary Differential Equation:

$$\begin{aligned} h''(t) &= -g \\ d'(t) &= s\cos\theta \\ h'(t=0) &= s\sin\theta \\ h(t=0) &= 0 \\ d(t=0) &= 0 \end{aligned} \tag{3.1}$$

Where $d(t)$ and $h(t)$ are the horizontal distance and vertical height that the cannonball has traveled since being fired, and s and θ are the speed and angle at which the cannonball leaves the cannon. The problem is illustrated in Figure 3-1.

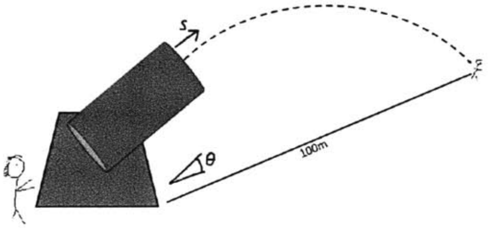


Figure 3-1: Example 7 – Firing a cannon with uncertain initial angle and speed

Initial conditions on the cannon (initial speed and angle) are uncertain; we will represent them using Random Variables distributed as $S(\omega) \sim N(32,1)\,\mathrm{m/s}$ and $\Theta(\omega) \sim U\left(\frac{\pi}{8}, \frac{\pi}{3}\right)$. The initial speed has an obvious effect on the impact distance - faster initial speed means the cannonball will fall farther away. The initial angle is less obvious. The cannon's maximum range is reached with an initial angle of $\frac{\pi}{4}$. At lower angles the cannonball falls to the ground too quickly even though it is traveling faster horizontally. At higher angles there is not enough forward speed to take advantage of the longer airtime. The effect of these initial conditions affects the impact distance in a nonlinear fashion.

Example 8: Sequential Chemical Reactions

Here we have a (physically inaccurate) instance of sequential reactions. Pentane converts to isopentane, and then to neopentane, as shown in Figure 3-2.

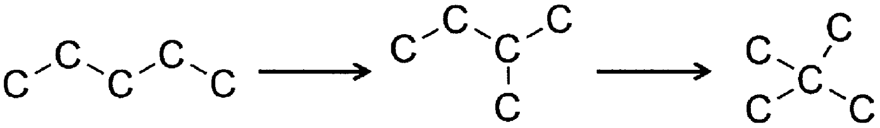


Figure 3-2: Example 8 – Two sequential isomerizations of Pentene, which is abstracted to $A \to B \to C$

We will abstract this to the hypothetical reaction $A \to B \to C$, where A, B, C are the dimensionless concentration of each species: $A = \frac{[Pentene]}{[Pentene]_{t=0}}$, $B = \frac{[isopentene]}{[Pentene]_{t=0}}$ and $C = \frac{[neopentene]}{[Pentene]_{t=0}}$. A very common chemical engineering problem is how to maximize the yield of species B. First we build a model by assuming that each reaction is elementary.

$$\begin{aligned} \frac{dA}{dt} &= -k_1 A \\ \frac{dB}{dt} &= k_1 A - k_2 B \\ \frac{dC}{dt} &= k_2 B \\ A(t=0) = 1, &B(t=0) = C(t=0) = 0 \end{aligned} \tag{3.2}$$

Next we assume that the kinetic parameters are constant with time (but unknown), and we derive the equations:

$$\begin{aligned} A(t) &= \exp(-k_1 t) \\ B(t) &= \frac{k_1}{k_2 - k_1}\left[\exp(-k_1 t) - \exp(-k_2 t)\right] \\ C(t) &= \frac{k_1}{k_2 - k_1}\left[\exp(-k_2 t) - 1\right] - \frac{k_2}{k_2 - k_1}\left[\exp(-k_1 t) - 1\right] \end{aligned} \tag{3.3}$$

The concentration profiles for certain values of the kinetic parameters is shown in Figure 3-3.

With a bit of calculus, we find that the maximum of B occurs at time

$$t_{\max} = \frac{\log k_1 - \log k_2}{k_1 - k_2} \tag{3.4}$$

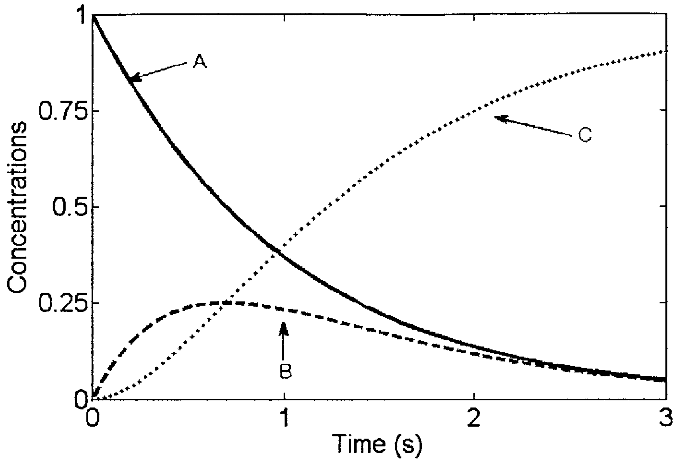


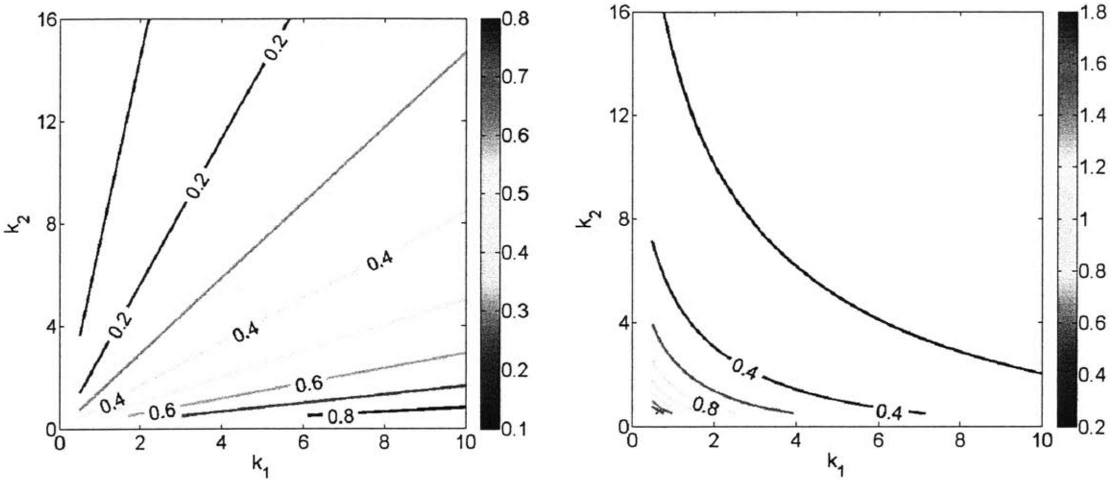
Figure 3-3: Example 8 – Concentration profiles versus time for the mean values of the kinetic parameters

However, before designing a reactor based on this information, we would like to know the effect of uncertainty in the kinetic parameters on the optimal residence time. Instead of relying on Equation 3.4, we wish to quantify the uncertainty in the concentration of B as a function of time.

Based on prior experiments, the kinetic parameters can be represented by the independent Random Variables:

$$K_1(\omega_1) \sim logN(1, 0.5)$$
$$K_2(\omega_2) \sim logN(2, 0.25)$$

As shown in Figure 3-4, increasing k_1 drives the optimal point to shorter times and higher concentrations of B, while increasing k_2 drives the optimal point to shorter times and lower concentrations of B. Given that the concentration of B may be uncertain, we may be interested in a *robust* design - one which ensures that the concentration of B will meet a minimum requirement.



(a) B_{max}

(b) t_{max} (sec)

Figure 3-4: Example 8 – Maximum concentration of Species B and corresponding time as a function of kinetic parameters k_1 and k_2

3.2 Formulating an Uncertainty Quantification Problem

We have discussed how and why Random Variables can be used to characterize uncertain quantities. This section describes how to use Random Variables to *quantify* the uncertainty in the model output. In order to properly quantify uncertainty, we need to frame the problem in the right way. The general procedure for solving uncertainty quantification problems is shown in Figure 3-5.

3.2.1 The System Model

Begin with a model that depends on independent variables x and parameters (θ). Independent variables are inputs to the model that vary, while parameters are fixed. The typical modeling process is to define the structure of the model and then address what values the independent variables and parameters will take. Here we assume that the model structure has been set and the uncertain parameters identified. Any parameters that are considered certain or known are referred to as constants and are not discussed here.

A generic model can then be written as an operator or function $\mathcal{M}$ that maps the inputs (parameters and variables) to the model outputs.

$$y = \mathcal{M}\left[\theta, x\right]$$

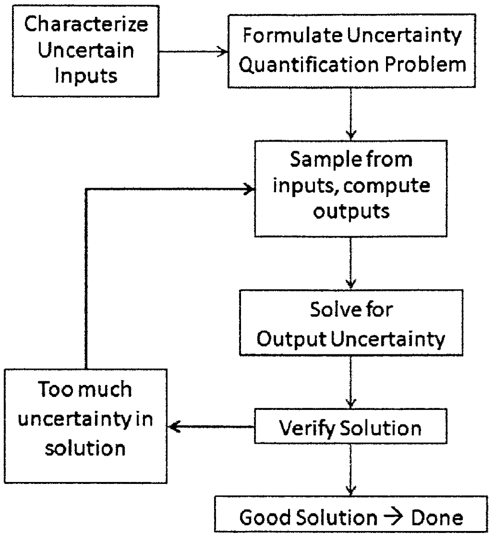

Characterize Uncertain Inputs
Formulate Uncertainty Quantification Problem
Sample from inputs, compute outputs
Solve for Output Uncertainty
Too much uncertainty in solution
Verify Solution
Good Solution → Done

Figure 3-5: Procedure for Solving Uncertainty Quantification Problems

For simplicity, only one model output is discussed. Each model output has different uncertainty, so the quantification process must be repeated for each.

Since the parameters and the outputs are uncertain, they are represented here by Random Variables.

$$Y\left(\omega_\Theta, x\right) = \mathcal{M}\left[\Theta\left(\omega_\Theta\right), x\right] \tag{3.5}$$

By framing the problem in this way, we see the model uncertainty $Y\left(\omega_\Theta, x\right)$ will vary with x but is caused only by $\Theta\left(\omega_\Theta\right)$. Now the problem is how to characterize the output Random Variable.

This framework is meant for uncertainties that do not change. That means that if the uncertainty of an input depends on the value of an independent variable, it is actually stochastic, not uncertain. In some cases, a stochastic quantity can be represented uncertain parameters and deterministic quantities. For example if the quantity $Q\left(\omega_Q\right)$ had uncertainty that grew with variable x, it would have to be decomposed into a function of an uncertain parameter and a function of x:

$$Q\left(\omega_Q\right) \approx \Theta\left(\omega_\Theta\right) g\left(x\right)$$

Then the model could be rewritten as:

$$Y\left(\omega_\Theta; x\right) = \mathcal{M}\left[\Theta\left(\omega_\Theta\right); x\right]$$

Where the deterministic portion $g(x)$ is lumped into the model equations.

3.2.2 Uncertain Parameters

When formulating an uncertainty quantification problem, the parametric uncertainties $\Theta(\omega_\Theta)$ must be known or assumed. In many engineering applications existing knowledge is enough to characterize the uncertain inputs using the Maximum Entropy Principle [43], or statistical modeling. Whether this method is used or experts are consulted, we assume from this point on that the input uncertainties are completely defined. A discussion of how to characterize prior knowledge is included in Section 6.1.1.

Example 7a: Formulating an Uncertainty Quantification Problem

In this problem the uncertain parameters are the initial speed and angle. We define the Random Variables $S(\omega_S)$ and $\Theta(\omega_T)$ to represent these. The outcome space is:

$$\begin{aligned}\Omega_{S,T} =& \{\omega_S, \omega_T : S(\omega_S) = \mathsf{s}, \Theta(\omega_T) = \theta\}\\ &\text{for } -\infty \leq \mathsf{s} \leq \infty \text{ and } \frac{\pi}{8} \leq \theta \leq \frac{\pi}{3}\end{aligned}$$

Because there are two Random Variables, the probability density function is a two dimensional surface. In general, these functions can be very complex, however, in this case the Random Variables are independent and so the cumulative distribution function and the joint probability density function are the products of the one dimensional functions.

$$\begin{aligned}F_{S\Theta}(\mathsf{s},\theta) =& P(S(\omega_S) < \mathsf{s}, \Theta(\omega_T) < \theta)\\ =& P(S(\omega_S) < \mathsf{s}) P(\Theta(\omega_T) < \theta)\\ =& \frac{1}{2}\left[1 + erf\left(\frac{\mathsf{s}-32}{\sqrt{2(1)^2}}\right)\right]\left[\frac{\theta - \frac{\pi}{8}}{\frac{\pi}{3} - \frac{\pi}{8}}\right]\\ f_{S\Theta}(\mathsf{s},\theta) =& \frac{\partial^2 F_{S\Theta}}{\partial \mathsf{s} \partial \theta} = \frac{d}{d\theta}\left(\frac{dF_{S\Theta}}{d\mathsf{s}}\right)\\ =& \frac{1}{\sqrt{2(1)^2}} \exp\left(-\frac{(\mathsf{s}-32)^2}{2(1)^2}\right)\left[\frac{1}{\frac{\pi}{3} - \frac{\pi}{8}}\right]\end{aligned}$$

See Figure 3-6 for a plot of these Random Variables.

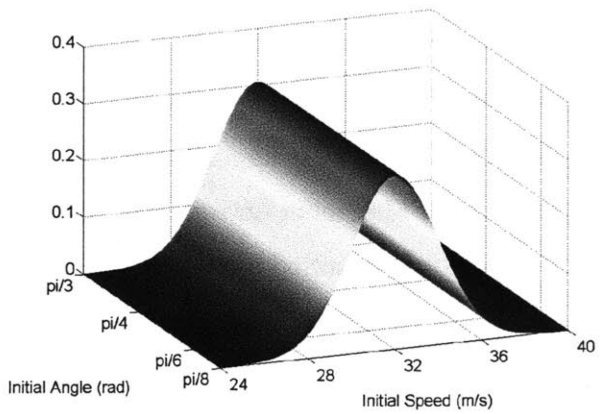


Figure 3-6: Example 7 – The joint probability density function of the initial angle and speed

3.2.3 Propagation and Solution

The uncertain parameters are defined and the relationship between the inputs and outputs is defined by the model. The next step is to determine the model output uncertainty $Y(\omega_\Theta, u)$. The uncertainty from parameters $\Theta(\omega_\Theta)$ must be propagated through the model by evaluating the model at various values θ. This data is used to estimate the model output's probability space - all the possible values of the model output and their probability densities. This estimate of the model output uncertainty is the goal of uncertainty quantification problems. From this we can determine any statistic of interest about the model output, for example: mean, variance, or more complex statistics like the probability that the model predicts failure. The uncertainty propagation process is depicted in Figure 3-7.

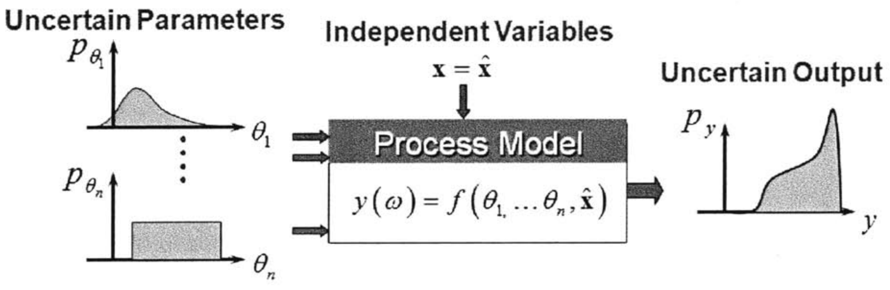


Figure 3-7: Propagation of uncertain inputs through a process model results in uncertain outputs, which can be quantified

3.2.4 Verification

For most uncertainty quantification problems we cannot determine the true solution, only approximations of the true solution. So the final step in the process is to verify the approximate solutions in order to be confident they are correct enough.

3.2.5 Solution Methods

There are many ways to carry out the uncertainty propagation step. The properties we are looking for in a method are:

1. Accurate solutions
2. Computationally efficient - minimizing the number of required model evaluations
3. Easy to verify
4. Can be applied to non-linear models
5. Can be applied to black box models
6. Can handle non-Gaussian inputs

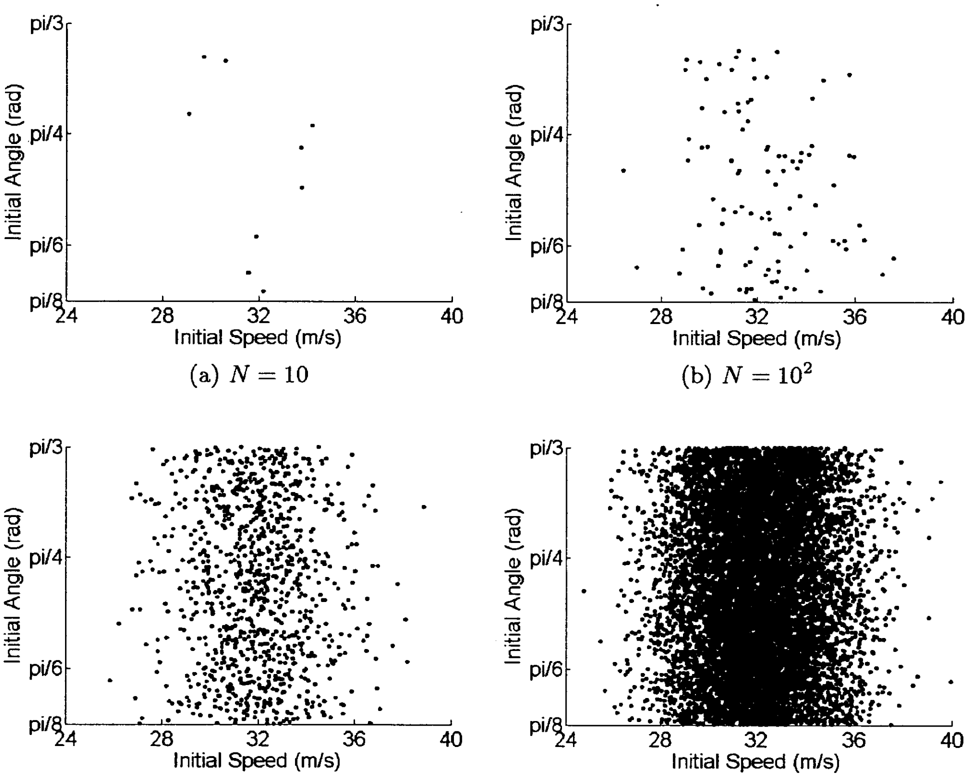
Two methods that satisfy almost all the above criteria are Monte Carlo and Polynomial Chaos Expansions. These will be discussed and compared in Sections 3.3 and Section 3.5.

3.3 Uncertainty Quantification with Monte Carlo

After formulating an uncertainty quantification problem, there exist several methods to find the solution. Monte Carlo is the simplest and most robust method; unfortunately it is also the most computationally expensive. There are several other references for the Monte Carlo method, so only the basics are shown here.

3.3.1 Algorithm

Monte Carlo techniques rely on repeated, random sampling of the parameters. All the possible combinations of parameters are defined by their joint probability density function, as in Figure 3-6. The details of how to sample from such a density function are described well by Robert and Casella [69]. Often times it takes thousands of samples in order to fully 'explore' the probability space of the parameters. Figure 3-8 shows how well the parameter probability space of Example 7 is explored. Compare this visually with Figure 3-6 which shows the true density.



(a) $N = 10$

(b) $N = 10^2$

(c) $N = 10^3$

(d) $N = 10^4$

Figure 3-8: Example 7 – Visualization of the joint parameter density function with N Monte Carlo samples

For each sample of the parameter probability space, the model output is computed at a particular set of independent variables $u = u_0$. The samples together approximate the probability density function of the model output: $Y(\omega_X, u_0)$.

Monte Carlo methods are called robust because they can handle any kind of joint probability density function as parameters. Many other methods assume that the parameter is normally distributed, or smooth and continuous, or that each parameter is independent. These assumptions restrict the types of models that can be analyzed accurately.

3.3.2 Probability Spaces

In the language of probability theory, Monte Carlo samples from the probability space of the parameters $\{\Omega_\Theta, \Sigma, P\}$ and generates samples from the probability space of the output $\{\Omega_Y, \Sigma, P\}$. The goal of uncertainty quantification is to characterize the probability space of the output.

Example 7b: Target Practice Input Uncertainty

Now have two uncertain dimensions, initial speed and angle. The parameter uncertainty is defined by the joint probability density function of the parameters, $f_{S,\Theta}(s, \theta)$, shown in Figure 3-6. This has an outcome space that contains every possible combination of speed and angle. This gives us a wide range of trajectories that the cannonball might follow. One hundred such trajectories are shown in Figure 3-9.

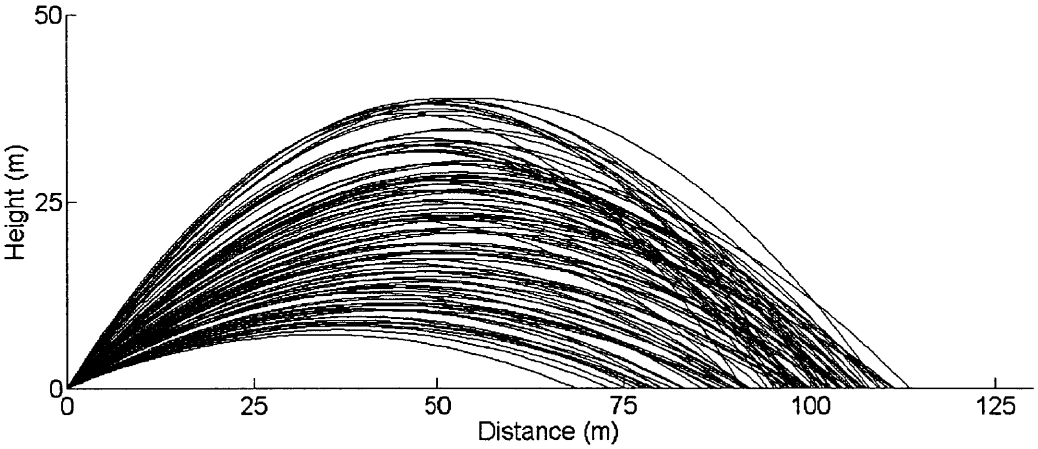


Figure 3-9: Example 7 – A sample of possible trajectories due to uncertain initial conditions. Notice the wide spread of possible impact positions also shown in Figure 3-10.

Starting from Equation 3.1, there is an analytical solution for the distance from the cannon where the cannonball hits, given initial speed s and angle θ.

$$d = \frac{2}{g} s^2 \sin\theta \cos\theta \tag{3.6}$$

The properly-framed, uncertain model output is:

$$\begin{aligned} D\left(\omega_{S,\Theta}\right) =& \mathcal{M}\left[S\left(\omega_S\right),\Theta\left(\omega_\Theta\right)\right] \\ D\left(\omega_{S,\Theta}\right) =& \frac{2}{g} S(\omega_S)^2 \sin\left(\Theta\left(\omega_\Theta\right)\right) \cos\left(\Theta\left(\omega_\Theta\right)\right) \end{aligned} \tag{3.7}$$

In the second step, the probability density function of the output is generated using 10^7 Monte Carlo samples, shown in Figure 3-10.

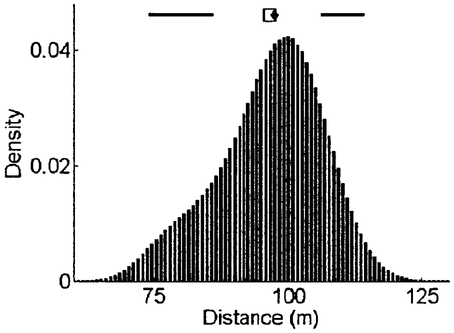


Figure 3-10: Example 7 – The probability density function of the impact distance, computed with 10^7 Monte Carlo samples

The probability density function is non-Gaussian, and would have been difficult to predict just from the parameters.

3.3.3 The Monte Carlo Solution

The results of Monte Carlo is a large set of model output values, which are equivalent to samples of $Y\left(\omega_\Theta, u\right)$. The samples can be used to calculate statistics of the model output like the mean and variance. To visualize the output uncertainty, the samples can be binned and displayed as a histogram, or normalized into a probability density function $f_Y\left(\mathrm{y}; u\right)$. However, before the result is analyzed we must make sure that the solution is correct.

3.3.4 Verification of Monte Carlo Solutions

The Monte Carlo solution is guaranteed to converge to the true solution as the number of samples approaches infinity. When the analytical solutions are not easily available, the Monte Carlo solution with many, many samples is regarded as the 'true' solution. This is because Monte Carlo methods are robust for all outcome spaces and are easy to verify. Again, the issue is computational expense. A single Monte Carlo simulation requires many model evaluations and the verification process takes many simulations. Here we discuss the theory behind the verification process and illustrate this on Example 7.

The Central Limit Theorem and Monte Carlo

A Monte Carlo simulation samples the uncertain parameters and computes the model output. This can be viewed as taking samples from the output Random Variable. If these samples are used to calculate statistics of the output Random Variable, the calculation of the statistic will require a summation over all the samples. For example, the sample mean and sample variance are given by:

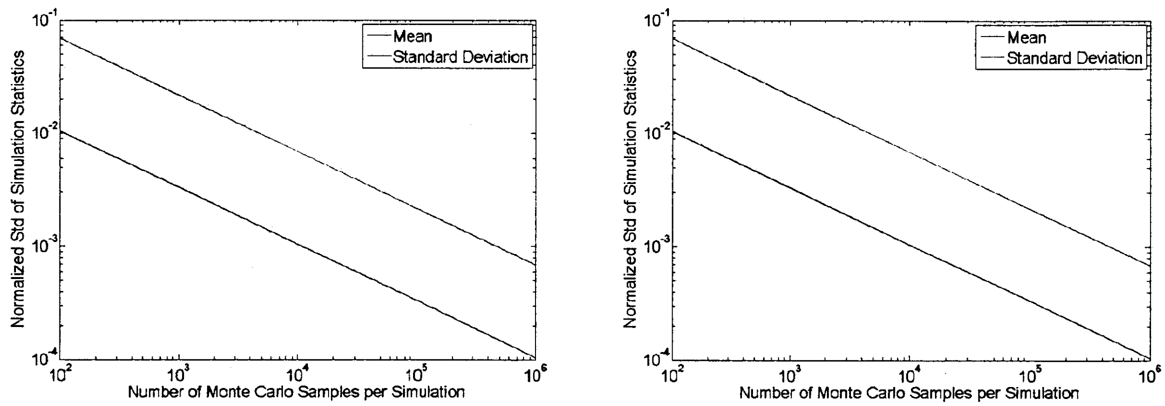
$$\hat{\mu} = \frac{1}{N}\sum_{i=1}^{N} Y_i \quad \text{and} \quad \hat{\sigma}^2 = \frac{1}{N}\left[\sum_{i=1}^{N} Y_i^2\right] - \left[\frac{1}{N}\sum_{i=1}^{N} Y_i\right]^2$$

In the context of the Central Limit Theorem, the N output samples are $Y_i\left(\omega_y\right)$ and the statistics are analogous to the summation Random Variable S_N. It is known that one particular Monte Carlo simulation, with N samples, will have statistics that are realizations from a summation Random Variable, and the variance or imprecision of that statistic calculation can be estimated. As $N \to \infty$, all the imprecision of all calculated statistics will approach zero and the Monte Carlo simulation will converge in the L^2 sense.

The problem with this verification strategy is that the imprecision in the estimation of the statistics is quantified with μ and σ which are parameters of the unknown output Random Variable. The determination of these parameters requires a Monte Carlo simulation of a Random Variable S_N that is itself computed with a Monte Carlo simulation. This means an ensemble of thousands of simulations of thousands of samples. In practice, shortcut methods are used however they can introduce significant bias [69].

Example 7c: Target Practice Verification

After characterizing the uncertainty in the model output, we wish to know the quality of the statistics we compute. For example, a single Monte Carlo simulation will enable us to estimate the mean and variance of the model output, however that estimate is also uncertain. The uncertainty is reduced as more Monte Carlo samples are used. An empirical test with an ensemble size of 10^4 is shown in Figure 3-11. With only 100 samples, the estimate of the mean has a normalized standard deviation of 1%, while the estimate of the variance is worse at nearly 10%. The uncertainty in these statistics generally is proportional to $\sqrt{\frac{1}{N_{MC}}}$.



(a) Standard Deviation

(b) Differential Entropy

Figure 3-11: Example 7 – Uncertainty in statistics of the Monte Carlo solution, indicating the quality of solution increases with number of samples

From the probability density function of the output, we can see that most impacts will be around 100 m. The probability of hitting the target is

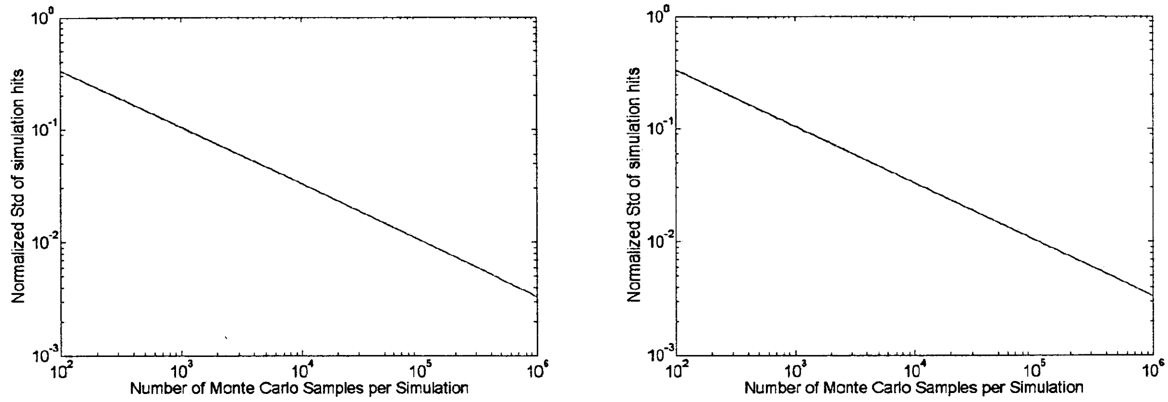
$$P\left(99\,\mathrm{m} < D\left(\omega_{S,\Theta}\right) < 101\,\mathrm{m}\right) = \int_{99}^{101} f_D\left(\mathrm{d}\right)d\mathrm{d} \approx 0.0843 \tag{3.8}$$

This accuracy metric (called A) is validated in Figure 3-12 with the same ensembles as Figure 3-11. The normalized standard deviation in the Monte Carlo solution of this statistic $\frac{\sigma_A}{\mu_A}$ does not drop below 1% until there 10^5 samples are used.

Depending on the application, an 8.4% probablity of hitting the target may not be good enough. Better get a bigger cannon!

3.3.5 Recap

Looking at the desired features of uncertainty quantification methods back in Section 3.2.5, Monte Carlo satisfies nearly all. Especially attractive is the accuracy and ability to handle any



(a) Standard Deviation

(b) Differential Entropy

Figure 3-12: Example 7 – Uncertainty of accuracy statistic A from Monte Carlo solution, indicating the quality of solution increases with number of samples

model and parameter uncertainty. Despite the guarantee of convergence to the true solution, Monte Carlo is not always an appropriate method for uncertainty quantification. It requires thousands of samples in order to be reliable and this may be too expensive to compute. In that case, we require alternative methods which are robust and accurate like Monte Carlo, and are more efficient in solving the uncertainty quantification problem.

3.4 Polynomial Chaos Expansions

We have seen that Random Variables are a natural way to describe uncertainty and that quantification of this uncertainty with Monte Carlo can be quite expensive. We would like alternative methods that satisfy the requirements in Section 3.2.5, but can be solved more efficiently. The key to reducing the number of model evaluations is to represent the problem in such a way that each model evaluation gives you more information. In the standard Monte Carlo method, each model evaluation tells you only the value of the model output - this is not very efficient. Variations of Monte Carlo (Latin Hypercube/ Importance Sampling) modify the representation of uncertainty so that each model evaluation reveals not only the value of the model output, but also the weight of that value in the probability density function. Of course, this comes at the price of bias in the solution. Similarly, we find that changing the representation of Random Variables using Polynomial Chaos Expansions can drastically reduce the cost of solving uncertainty quantification problems [35, 76]. This section will provide the theory and several examples, and the application to uncertainty quantification is described in the following sections.

3.4.1 Introduction to Polynomial Chaos Expansions

Two examples are presented to illustrate the ideas behind functional expansions. In the first, Fourier Series are used as a deterministic analogy. Just like a deterministic function can be approximated as a sum of basis functions depending on an independent variable, a Target Random Variable can be approximated as a sum of Basis Functionals that depend on Basis Random Variables. After the examples, Polynomial Chaos Expansions are defined and a detailed explanation follows.

Example 9: Fourier Series

In this example we wish to approximate the function $f(x) = x$ for $-1 \leq x \leq 1$

Coefficients are $c_n = 2\frac{(-1)^{n+1}}{n}$ and $f(x) = 2\sum\limits_{n=1}^{\infty} \frac{(-1)^{n+1}}{n} \sin(n\pi x)$. Figure 3-13 shows the series with increasing number of terms.

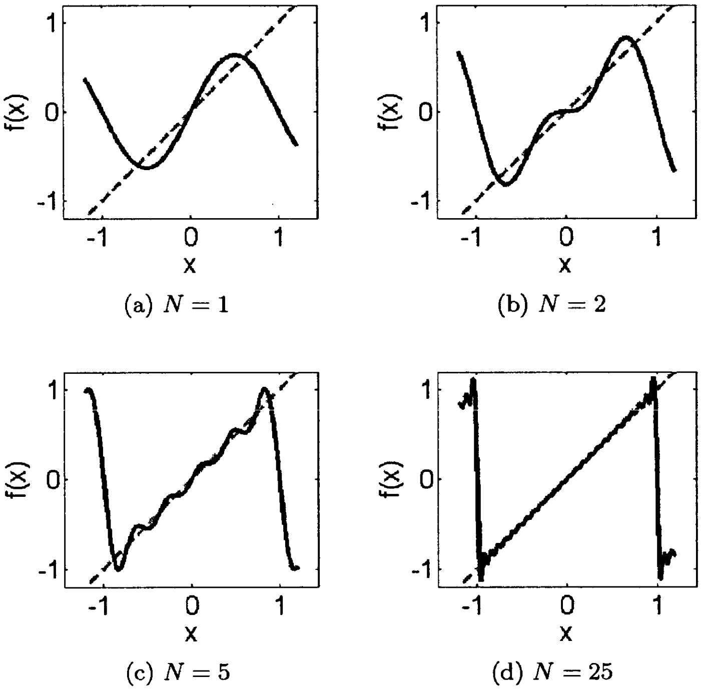
Summary of Fourier Series

We would like to approximate the arbitrary function $f(x)$ with a finite range R. This is the target function. x is an independent variable defined on a segment of the real line, $D \in \Re$. Let us call this the basis variable. The series consists of a sum of simple, well-known functions of x: $f_n(x)$. These are the basis functions, which map from the space of x to the space of $f(x)$. In Fourier Series, the basis functions are sine and cosine waves with different frequencies. Using linear combinations of these basis functions, we can approximate the target function with an expansion

$$f(x) = \hat{f}(x) = \sum_{n=0}^{\infty} c_n f_n(x)$$

With infinite terms, the series converges to the target function in the L^2 sense, meaning

$$\int_D \left[f(x) - \sum_{n=0}^{\infty} c_n f_n(x) \right]^2 dx = 0$$



(a) $N = 1$

(b) $N = 2$

(c) $N = 5$

(d) $N = 25$

Figure 3-13: Example 9 – Convergence of the N-term Fourier Series (—) to the target function $f(x) = x$ (- -) on the domain $-1 \leq x \leq 1$

Example 8a - Sequential Reactions: Polynomial Chaos Expansion of parameter K_1

From previous data we have decide to model the uncertainty of parameters K_1 with a lognormal Random Variable $K_1(\omega_1) \sim \log N(1, 0.5)$. The analytical form of the pdf is:

$$f_{K_1}(\mathsf{k}) = \frac{1}{\mathsf{k}\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(\log \mathsf{k} - \mu)^2}{2\sigma^2}\right) \tag{3.9}$$

To approximate this with a Polynomial Chaos Expansion, start with a standard normal Random Variable $Z(\omega_Z) \sim N(1, 0)$. Rewrite $K_1(\omega_1)$ in terms of $Z(\omega_Z)$ and coefficients $\mathbf{k}$:

$$\hat{K}_1(\omega_Z) = k_0 + k_1 Z(\omega_Z) + k_2\left(Z(\omega_Z)^2 - 1\right) + k_3\left(Z(\omega_Z)^3 - 3Z(\omega_Z)\right)\dots \tag{3.10}$$

with $\mathbf{k} = [3.0785, 1.5401, 0.38500.0642, \dots]^T$

$\hat{K}_1(\omega_Z)$ is a direct representation of $K_1(\omega_1)$, written in terms of the Random Variable $Z(\omega_Z)$. Indirect representations, such as the analytical form of the probability density function, are in terms of value of the Random Variable, k . $\hat{K}_1(\omega_Z)$ can be visualized using Monte Carlo in the same way as an indirect representation. The values of the parameter are sampled from the distribution $Z(\omega_Z)$ and the resulting evaluations of the Polynomial Chaos Expansion are samples from $\hat{K}_1(\omega_Z)$. The resulting histogram is shown in Figure 3-14d along with the true pdf $f_{K_1}(\mathsf{k})$ (solid red line). Section 3.4.5 shows how the coefficients were computed.

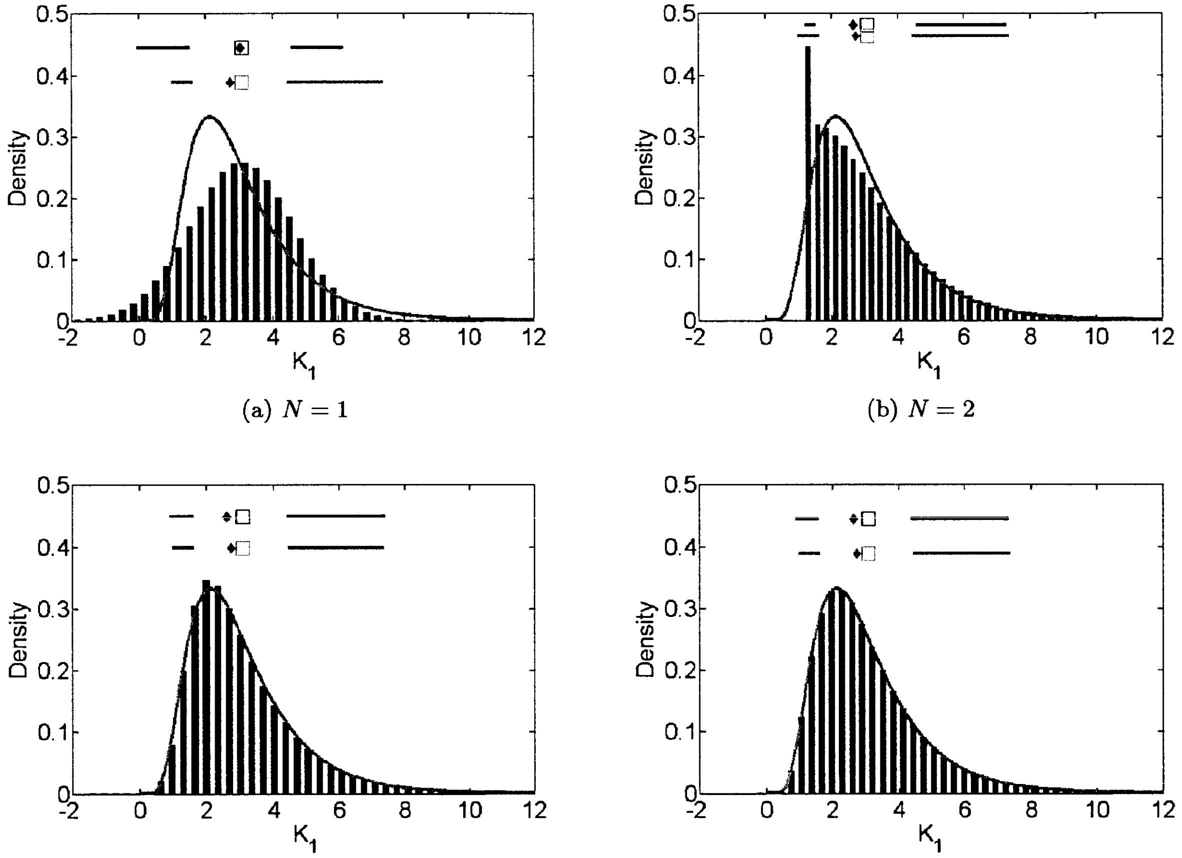
Note that the expansion is more complex than the indirect form of original Random Variable, however, we will see that this representation makes uncertainty calculations much more efficient.

Formal Definition

Expand the above example to the general case. Polynomial Chaos Expansions are used to approximate any square integrable Target Random Variable (meaning it has finite variance).

The Polynomial Chaos Expansion of Target Random Variable $\hat{X}(\omega)$ is:

$$X(\omega) = \hat{X}\left(\vec{\xi}\right) = \sum_{\vec{k}=\vec{0}}^{\vec{\infty}} c_{\vec{k}} \Psi_{\vec{k}}\left(\vec{\xi}\right) \tag{3.11}$$



(a) $N = 1$

(b) $N = 2$

(c) $N = 3$

(d) $N = 4$

Figure 3-14: Example 8 – Polynomial Chaos Expansions with order N (bars & top credible interval) versus the true parameter density (line & bottom credible interval)

where

- $\vec{\Xi}$ is the vector of Basis Random Variables
  $\vec{\xi}$ are the values of $\vec{\Xi}$
- $\vec{k}$ is a multi index
- $\Psi_{\vec{k}}$ are *Basis Functionals* which are indexed by $\vec{k}$.
- $\mathbf{c}$ is a vector, indexed by $\vec{k}$

3.4.2 Detailed Description

Truncating the Expansion

In practice, we cannot deal with infinite numbers of Random Variables or expansions with infinite terms. Therefore the expansion is truncated and becomes an approximation:

$$X(\omega_X) \approx \hat{X}\left(\vec{\xi}\right) = \sum_{\vec{k}:\left|\vec{k}\right| \leq P} \mathrm{c}_{\vec{k}} \Psi_{\vec{k}}\left(\vec{\xi}\right) \text{ for } \vec{\Xi} = \{\Xi_1 \ldots \Xi_N\} \tag{3.12}$$

The number of Basis Random Variables is limited to N, which is called the *dimension* of the expansion. In addition to truncating the dimension, the *order* of the expansion must be limited. The Basis Functionals are restricted to polynomials of order P or lower. With this constraint, the total number of Basis Functionals is given by: $K = count\left(\vec{k}\right) = \begin{pmatrix} n+p \\ p \end{pmatrix}$.

It is also possible to truncate the order anisotropically in each dimension. In this case, the notation gets very messy. P_n will denote the maximum order of Basis Functionals containing the nth Basis Random Variable, but a fuller description will usually be necessary to prevent confusion. The total number of Basis Functionals will always be denoted K even though it will not follow the above formula for isotropic order truncation. The choice of P is a matter of expansion convergence and is discussed in Section 3.5.5. The number of dimensions N is problem specific and this is discussed in Section 3.4.5.

The Multi-Index

The summation in a Polynomial Chaos Expansion is over a multi-index, $\vec{k}$. The length of the multi-index is $\left|\vec{k}\right| = K$. Each element is unique and identifies a Basis Functional and a coefficient

$\mathbf{c}_{\vec{k}}$. The elements themselves are sequences of numbers, and the sequence is indexed by n. The length of each sequence is N and a particular number in the sequence $\vec{k}$ is denoted $\vec{k}(n)$. It takes a N number sequence to define one Basis Functional and the order of that Basis Functional is given by: $p = \sum_{n=1}^{N} \vec{k}(n)$. The nth number in each sequence corresponds to the nth Basis Random Variable.

Example 10: The Multi-index

Say $N = 3$. The multi-indecies, $\vec{k}$, up to order $P = 2$ are: $[0,0,0]$, $[1,0,0]$, $[0,1,0]$, $[0,0,1]$, $[2,0,0]$, $[1,1,0]$, $[1,0,1]$, $[0,2,0]$, $[0,1,1]$, $[0,0,2]$.

So $K = \begin{pmatrix} 5 \\ 2 \end{pmatrix} = 10$.

Basis Random Variables

Let $\vec{\Xi} = \{\Xi_n(\xi_n)\}$ for $n = 1 \ldots N$ be a set of orthonormal Basis Random Variables. Random Variables X and Y are orthonormal if $E[XY] = 0$ and $\text{var}(X) = \text{var}(Y) = 1$. Basis Random Variables are chosen to be simple, commonly used Random Variables, with well known properties and closed form expressions for their moments. This is not mathematically necessary, it is simply for convenience.

Each $\Xi_n(\omega)$ has its own probability space - they are all different Random Variables, although they can take the same form ie: $\Xi_1(\omega)$ and $\Xi_2(\omega)$ could both be Standard Normal Random Variables.

Notation

For most Random Variables like $X(\omega_X)$, we show the dependency on the outcomes ω_X to make it clear that X is random. We also make a distinction between outcomes ω_X and values x, although for real valued Random Variables they are equivalent, in order to distinguish a Random Variable $X(\omega_X)$ from a deterministic function $X(x)$. This is not necessary for Basis Random Variables so both these features are dropped. Only the value ξ is used, and $\Xi(\omega_\Xi) = \Xi(\xi) = \Xi$. Functions of Basis Random Variables and (equivalently) Random Variables whose underlying probability space is taken from a Basis Random Variable DO show the dependence on the value ξ for emphasis.

Orthogonal Polynomials

Every Random Variable $X(\omega)$ has a series of orthogonal polynomials $\{\psi\}$ for which the inner product of any pair is zero.

$$\langle\psi_i(X)\,\psi_j(X)\rangle = \int\limits_{\Omega_X} \psi_i(\mathsf{x})\,\psi_j(\mathsf{x})\,f_X(\mathsf{x})\,d\mathsf{x} = 0, \text{ for } i \neq j \tag{3.13}$$

In the bracket notation, the weighting with respect to $X(\omega)$ is implied. The bracket notation is used for direct representation, while the integral on the right hand side is indirect - in terms of the value x.

Example 11: Hermite Polynomials

The Hermite polynomial series is orthogonal with respect to a standard normal Gaussian Random Variable with probability density function $f_X(\mathsf{x}) = \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{\mathsf{x}^2}{2}\right)$ for all $\mathsf{x} \in \Re$. These polynomials were used as Basis Functionals in Section 3.4.1. The first four terms of the series (0th through 3rd order) are:

$$\psi_0(x) = 1$$
$$\psi_1(x) = x$$
$$\psi_2(x) = x^2 - 1$$
$$\psi_3(x) = x^3 - 3x$$

For example, the 2nd and 3rd order polynomials are orthogonal:

$$\int\limits_{\Omega_X} \psi_2(\mathsf{x})\,\psi_3(\mathsf{x})\,f_X(\mathsf{x})\,d\mathsf{x} = 0$$
$$\int\limits_{-\infty}^{\infty} \left(\mathsf{x}^3 - 3\mathsf{x}\right)\left(\mathsf{x}^2 - 1\right)\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{\mathsf{x}^2}{2}\right))d\mathsf{x} = 0$$
$$\int\limits_{-\infty}^{\infty} \left(\mathsf{x}^5 - 4\mathsf{x}^3 + 3\mathsf{x}\right)\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{\mathsf{x}^2}{2}\right)d\mathsf{x} = 0$$

The normal pdf is even, symmetric about $\mathsf{x} = 0$. The polynomial is odd, symmetric about the origin. Therefore the product is odd, and the integral is 0.

Example 12: Legendre Polynomials

The shifted Legendre polynomial series is orthogonal with respect to a uniform Random Variable $X(\omega_X) \sim U(0,-1)$. The probability density function is $f_X(\mathrm{x}) = \begin{cases} 1, & \text{if } \mathrm{x} \in [0,-1] \\ 0, & \text{else} \end{cases}$.

$$\begin{aligned}
\psi_0(x) &= 1 \\
\psi_1(x) &= 2x - 1 \\
\psi_2(x) &= 6x^2 - 6x + 1 \\
\psi_3(x) &= 20x^3 - 30x^2 + 12x - 1
\end{aligned}$$

For example the 1st and 3rd order polynomials are orthogonal:

$$\begin{aligned}
&\int\limits_{\Omega_X} \psi_1(\mathrm{x})\,\psi_3(\mathrm{x})\,f_X(\mathrm{x})\,d\mathrm{x} = 0 \\
&\int\limits_0^1 (2\mathrm{x}-1)\left(20\mathrm{x}^3 - 30\mathrm{x}^2 + 12\mathrm{x} - 1\right)(1)d\mathrm{x} = 0 \\
&\int\limits_0^1 \left(40\mathrm{x}^4 - 80\mathrm{x}^3 + 54\mathrm{x}^2 - 14\mathrm{x} - 1\right)d\mathrm{x} = 0 \\
&\frac{40}{5} - \frac{80}{4} + \frac{54}{3} - \frac{14}{2} + 1 = 0
\end{aligned}$$

There are many other named, polynomial series. Some are listed here, along with the Random Variable to which they are orthogonal

- Normal distribution - Probabilist's Hermite polynomials
- Uniform [0,1] - Shifted Legendre polynomial
- Uniform [-1,1] - Legendre polynomials
- Exponential - Laguerre polynomials
- Gamma - Generalized Laguerre polynomials (Exponential is a special case)

- Beta - Jacobi polynomials
- Poisson - Charlier polynomials
- Binomial - Krawtchouk polynomials

Using Arbitrary Basis Random Variables

It is possible to use an arbitrary Random Variable, $X\left(\omega_X\right)$, as a Basis Random Variable. To do so, the correct orthogonal polynomial series would need to be constructed. Let $f_X\left(\mathsf{x}\right)$ be the probability density function of $X\left(\omega\right)$. Then for every pair of polynomials in the series ψ,

$$\int\limits_{\Omega_X} \psi_i\left(\mathsf{x}\right)\psi_j\left(\mathsf{x}\right)f_X\left(\mathsf{x}\right)d\mathsf{x} = \delta_{ij}c_i \tag{3.14}$$

where: $c_i = \left\langle \psi_i(X)^2 \right\rangle = \int\limits_{\Omega_X} \psi_i\left(\mathsf{x}\right)\psi_i\left(\mathsf{x}\right)f_X\left(\mathsf{x}\right)d\mathsf{x}$.

Any polynomial can be defined by its coefficients. The Mth order polynomial has $M+1$ coefficients ${a_m}^{(M)}$ for $m = 0 \ldots M$. The polynomial can be written as the sum of monomials:

$$\psi_M\left(X\right) = \sum_{m=0}^{M} {a_m}^{(M)}X^m$$

The goal is to determine the coefficients ${a_m}^{(M)}$ for each value of M, such that that the resulting polynomials are orthogonal. These polynomial series are generated recursively by Gram-Schmidt orthogonalization. The zeroth and first order polynomials are chosen to be:

$$\begin{aligned}\psi_0\left(X\right) =& 1\\ \psi_1\left(X\right) =& X - \left\langle X\psi_0\left(X\right)\right\rangle = X - \int\limits_{\Omega_X}\left(\mathsf{x}\right)\left(1\right)f_X\left(\mathsf{x}\right)d\mathsf{x}\end{aligned}$$

The higher order polynomials are derived by multiplying the highest order by X and subtracting off the (normalized) projections into the lower orders. This generates a polynomial that is orthogonal to all polynomials of lower order. To save space we drop the dependence on X.

$$\psi_{i+1} = X\psi_i - \left[\frac{\langle X\psi_i\psi_i\rangle}{\langle\psi_i\psi_i\rangle}\right]\psi_i - \left[\frac{\langle X\psi_i\psi_{i-1}\rangle}{\langle\psi_{i-1}\psi_{i-1}\rangle}\right]\psi_{i-1} - \sum_{k=0}^{i-2}\left[\frac{\langle X\psi_i\psi_k\rangle}{\langle\psi_k\psi_k\rangle}\right]\psi_k \tag{3.15}$$

Starting with the last term:

$$\langle X\psi_i\psi_k\rangle = 0 \text{ for } k \leq i-2$$

Use the property that ψ_i is orthogonal to all polynomials of order $\leq i-1$ because such a polynomial could be written as $\sum\limits_{j=0}^{i-1} c_j\psi_j$ and all ψ are orthogonal. Because $X\psi_k$ is a polynomial of order $\leq i-1$, the inner product $\langle(\psi_i)(X\psi_k)\rangle$ is zero.

Next, break down the 3rd term and write out each polynomial in terms of their coefficients:

$$\begin{aligned}
&\langle X\psi_i\psi_{i-1}\rangle \\
&\qquad = \langle(\psi_i)(X\psi_{i-1})\rangle \\
&\qquad = \left\langle\left(\sum_{j=0}^{i} a_jX^j\right)\left(\sum_{k=1}^{i} b_{k-1}X^k\right)\right\rangle \\
&\qquad = \left\langle\left(\sum_{j=0}^{i} a_jX^j\right)\left(\sum_{k=1}^{i} \frac{a_i}{b_{i-1}}b_{k-1}X^k\right)\right\rangle\frac{b_{i-1}}{a_i} \\
&\qquad = \left\langle\left(\sum_{j=0}^{i} a_jX^j\right)\left(\left[\sum_{k=1}^{i} a_kX^k\right]+a_0\right)\right\rangle\frac{b_{i-1}}{a_i} \\
&\qquad\quad + \left\langle\left(\sum_{j=0}^{i} a_jX^j\right)\left(\left[\sum_{k=1}^{i-1}\left(\frac{a_i}{b_{i-1}}b_{k-1}-a_k\right)X^k\right]-a_0\right)\right\rangle \\
&\qquad = \langle\psi_i\psi_i\rangle\frac{b_{i-1}}{a_i}+0
\end{aligned}$$

Again, this takes advantage of the property that ψ_i is orthogonal to all polynomials of order $\leq i-1$. If $\psi_0(X) = 1$, then the coefficient of the highest order term in each polynomial is 1, so $\frac{b_{i-1}}{a_i} = 1$ and the recursive formula for the orthogonal polynomial series is:

$$\begin{aligned}
&\psi_0(X) = 1 \\
&\psi_1(X) = X - \langle X\rangle \\
&\psi_{i+1} = X\psi_i - \left[\frac{\langle X\psi_i\psi_i\rangle}{\langle\psi_i\psi_i\rangle}\right]\psi_i - \left[\frac{\langle\psi_i\psi_i\rangle}{\langle\psi_{i-1}\psi_{i-1}\rangle}\right]\psi_{i-1} \text{ for } i \geq 2
\end{aligned}$$

Example 13: Gram-Schmidt Orthogonalization

Use the uniform Random Variable $X(\omega_X) \sim [0,1]$ to generate the Shifted Legendre orthogonal polynomial series. Let $f_X(\mathsf{x}) = \begin{cases} 1 & 0 \leq \mathsf{x} \leq 1 \\ 0 & \mathit{else} \end{cases}$

$$\psi_0 = 1$$

$$\psi_1 = X - \langle X \rangle = X - \int_0^1 \mathsf{x} d\mathsf{x} = X - \frac{1}{2}$$

$$\begin{aligned}
\psi_2 &= X\psi_1 - \left[\frac{\langle X\psi_1\psi_1 \rangle}{\langle \psi_1\psi_1 \rangle}\right]\psi_1 - \left[\frac{\langle \psi_1\psi_1 \rangle}{\langle \psi_0\psi_0 \rangle}\right]\psi_0 \\
&= X\left(X - \frac{1}{2}\right) - \left[\frac{\langle X\left(X - \frac{1}{2}\right)\left(X - \frac{1}{2}\right)\rangle}{\langle \left(X - \frac{1}{2}\right)\left(X - \frac{1}{2}\right)\rangle}\right]\left(X - \frac{1}{2}\right) - \left\langle \left(X - \frac{1}{2}\right)\left(X - \frac{1}{2}\right)\right\rangle \\
&= X\left(X - \frac{1}{2}\right) - \left[\frac{\langle X^3 - X^2 + \frac{1}{4}X\rangle}{\langle X^2 - X + \frac{1}{4}\rangle}\right]\left(X - \frac{1}{2}\right) - \left\langle X^2 - X + \frac{1}{4}\right\rangle \\
&= X^2 - \frac{1}{2}X - \left[\frac{\frac{1}{24}}{\frac{1}{12}}\right]\left(X - \frac{1}{2}\right) - \frac{1}{12} \\
&= X^2 - X + \frac{1}{6}
\end{aligned}$$

etc.

For historical reasons, the named polynomial series are often multiplied by a constant, as seen in comparison with Example 12. This does not affect the orthogonality at all.

While the Gram-Schmidt process for generating polynomials is quite straightforward, in practice the recursive nature leads to numerical problems. One very attractive feature of the named polynomial series is that general formulas exist to compute the coefficients explicitly. For example, the Shifted Legendre Polynomials of order m are given by:

$$\psi_m(x) = (-1)^m \sum_{i=0}^{m} \begin{pmatrix} m \\ i \end{pmatrix} \begin{pmatrix} m+i \\ i \end{pmatrix} (-x)^i \qquad (3.16)$$

Notation

The notation used here for an orthogonal polynomials series is $\psi(\xi_n)$. The series is distinguished from others by its underlying dependence on the Basis Random Variable (just like any functional of a Random Variable), so $\psi(\xi_i)$ is not the same series as $\psi(\xi_j)$. The order of polynomials within the series are denoted using subscripts.

- $\psi_p(\xi_n)$ is a pth order polynomial from a series that is orthogonal with respect to Ξ_n
- If $\xi_1 \sim N(0,1)$ then $\psi_3(\Xi_1) = {\Xi_1}^3 - 3\Xi_1$ - the 3rd order Hermite Polynomial
- If $\xi_2 \sim U(0,1)$ then $\psi_2(\Xi_2) = 6{\Xi_2}^2 - 6\Xi_2 + 1$ - the 2nd order Shifted Legendre Polynomial

Basis Functionals

Let $\Psi_{\vec{k}}\left(\vec{\xi}\right)$ be Basis Functionals, which depend on the N Basis Random Variables. Because they are functions of Random Variables, they are themselves Random Variables, and they take the outcome space of the Basis Random Variables. Every Basis Functional is the product of N orthogonal polynomials chosen from N orthogonal polynomial series. The Basis Functionals are indexed by $\vec{k}$ and each element in $\vec{k}$ is a sequence, indexed by $n = 1 \ldots N$, that defines the Basis Functional.

$$\Psi_{\vec{k}}\left(\vec{\xi}\right) = \prod_{n=1}^{N} \psi_{\vec{k}(n)}(\xi_n) \qquad (3.17)$$

The nth orthogonal polynomial, $\psi_{\vec{k}(n)}(\xi_n)$, is of order $\vec{k}(n)$ and depends only on the nth Basis Random Variable Ξ_n. The magnitude of $\vec{k}$ is the total polynomial order of the Basis Functional.

Example 14: Basis Functionals

A 2-dimensional example: $\Xi_1 \sim N(0,1)$ and $\Xi_2 \sim U(0,1)$ and let $\vec{k} = [3,2]$. Then

$$\Psi_{[3,2]}(\xi_1,\xi_2) = \psi_3(\xi_1)\,\psi_2(\xi_2) = \left({\Xi_1}^3 - 3\Xi_1\right)\left(6{\Xi_2}^2 - 6\Xi_2 + 1\right) \qquad (3.18)$$

Polynomial Chaos and Orthogonality

The term Polynomial Chaos refers to the set of all orthogonal Basis Functionals with an infinite set of Basis Random Variables. This is not of much practical use, but it is important to understand the effect of orthogonal terms.

- The Polynomial Chaos Γ is the combined set of all $\Gamma_p\left(\vec{\Xi}\right)$, for $p = 0 \ldots \infty$
- $\Gamma_p\left(\vec{\Xi}\right)$ is the Polynomial Chaos of order p

- The pth Polynomial Chaos is the set of all the Basis Functionals with polynomial order p, or $\left|\vec{k}\right| = \sum\limits_{i=1}^{\infty} \vec{k}(i) = p$ for all Basis Functionals in the pth Polynomial Chaos
- Because the Basis Random Variables and the Basis Functionals are orthogonal, every Basis Functional in Γ is orthogonal to all others with respect to the probability space $\left\{\hat{\Omega}_{\vec{\Xi}}, \Sigma, P\right\}$
- This means that the inner product on $L^2(\Omega)$ is zero, for all non-self combinations $\left\langle \Psi_{\vec{i}} \Psi_{\vec{j}} \right\rangle = \int\limits_{\hat{\Omega}} \Psi_{\vec{i}}\left(\vec{\xi}\right) \Psi_{\vec{j}}\left(\vec{\xi}\right) f_{\vec{\Xi}}\left(\vec{\xi}\right) d\vec{\xi} = \delta_{\vec{i}\vec{j}} \left\langle \Psi_{\vec{i}} \Psi_{\vec{i}} \right\rangle$
- Work out the self inner product:
  $\int\limits_{\hat{\Omega}} \Psi_{\vec{i}}\left(\vec{\xi}\right) \Psi_{\vec{i}}\left(\vec{\xi}\right) f_{\vec{\Xi}}\left(\vec{\xi}\right) d\vec{\xi} = \int\limits_{\hat{\Omega}} \left(\prod\limits_{n=1}^{N} \psi_{\vec{k}(n)}\left(\vec{\xi}\right)\right)^2 \prod\limits_{n=1}^{N} f_{\Xi_n}(\xi_n) d\xi_n$
  $= \int\limits_{\hat{\Omega}} \prod\limits_{n=1}^{N} \left[\psi_{\vec{k}(n)}\left(\vec{\xi}\right)\right]^2 f_{\Xi_n}(\xi_n)\, d\xi_n$, each differentiation variable is independent, so bring the integral inside the product
  $= \prod\limits_{n=1}^{N} \int\limits_{\Omega_{Xi_n}} \left[\psi_{\vec{k}(n)}\left(\vec{\xi}\right)\right]^2 f_{\Xi_n}(\xi_n)\, d\xi_n$
- The product of 1D integrals is much easier to compute

Polynomial Chaos Expansions

A Random Variable $X(\omega)$ is represented as a linear combination of known Basis Functionals which depend on Basis Random Variables

$$X(\omega) \approx \hat{X}\left(\vec{\xi}\right) = \sum_{\vec{k}:\left|\vec{k}\right| \le P} \mathrm{c}_{\vec{k}} \Psi_{\vec{k}}\left(\vec{\xi}\right) \tag{3.19}$$

$$\text{where } \vec{\Xi} = \{\Xi_1 \ldots \Xi_N\}$$

An expanded notation makes it clear that there are a number of Basis Functionals with order p

$$\hat{X}\left(\vec{\xi}\right) = \sum_{p=0}^{P} \sum_{\vec{k}:\left|\vec{k}\right| = p} \mathrm{c}_{\vec{k}} \Psi_{\vec{k}}\left(\vec{\xi}\right) \tag{3.20}$$

A further expanded notation shows the role of the multi-index.

$$\hat{X}\left(\vec{\Xi}\right) = \sum_{p=0}^{P} \sum_{\vec{k}:\left|\vec{k}\right| = p} \left\{ \mathrm{c}_{\vec{k}} \prod_{n=1}^{N} \psi_{\vec{k}(n)}(\xi_n) \right\} \tag{3.21}$$

In words: sum over all Polynomial Chaos orders, of the sum over Basis Functionals with order p, which are each composed of the products of a real valued coefficient and one-dimensional orthogonal polynomials dependent on the n Basis Random Variable and have order given by the nth entry of multi index $\vec{k}$.

With infinite terms, the series converges to the target function in the L^2 sense, meaning

$$\left\langle \left[X(\omega) - \sum_{p=0}^{\infty} \sum_{\vec{k}:|\vec{k}|=p} \left\{ c_{\vec{k}} \prod_{n=1}^{N} \psi_{\vec{k}(n)}(\xi_n) \right\} \right]^2 \right\rangle = \int_{\Omega_X} \left[f_X(x) - f_{\hat{X}}\left(\vec{\xi}\right) \right]^2 dx = 0 \qquad (3.22)$$

As the order goes to infinity, the outcome spaces Ω_X and $\hat{\Omega}_X$ become equal (every value x can be mapped to a value of $\vec{\xi}$) and the probability density functions $f_X(x)$ and $f_{\hat{X}}\left(\vec{\xi}\right)$ have no L^2 error.

3.4.3 Probability Spaces and Representation

The Target Random Variable has a probability space, which we will describe with outcome space Ω_T and sigma algebra Σ and probability measure P. A Polynomial Chaos Expansion will have a different probability space, $\left\{\Omega_{\hat{T}}, \Sigma, P\right\}$ which is derived from the probability space of the Basis Random Variables $\left\{\Omega_{\vec{\Xi}}, \Sigma, P\right\}$. The key concept is that the Polynomial Chaos Expansion is a Random Variable whose underlying probability space is actually that of the Basis Random Variable. Every outcome of the Polynomial Chaos Expansion can be mapped back to an outcome of the Basis Random Variables. That way, instead of manipulating $\{\Omega_T, \Sigma, P\}$ or $\left\{\Omega_{\hat{T}}, \Sigma, P\right\}$ which may be difficult, a Polynomial Chaos Expansion is used so we can manipulate $\left\{\Omega_{\vec{\Xi}}, \Sigma, P\right\}$, which is simple.

How does this work? Taken together, the Basis Functionals map outcomes from $\Omega_{\vec{\Xi}}$ to $\Omega_{\hat{T}}$ in order to match with Ω_T. Then the coefficients of the Polynomial Chaos Expansions are determined so that the probability measure of the expansion matches that of the Target Random Variable.

For instance the Target Random Variable could be the measurement of temperature with outcomes $\{\omega_T : T(\omega_T) < t\}$. The physical meaning is that each temperature value has some probability of being measured. The Basis Random Variables on the other hand are completely without context. When the Basis Functionals are applied to $\Omega_{\vec{\Xi}}$, they map to a new outcome space $\Omega_{\hat{T}}$ that has the same physical meaning as Ω_T. So representing the Target Random Variable with a Polynomial Chaos Expansion substitutes a physically intuitive outcome space Ω_T with a similar outcome space

$\Omega_{\hat{T}}$, which is derived from an abstract outcome space $\Omega_{\vec{\Xi}}$.

In addition, truncating the order of the expansion means that the underlying outcome space of the Polynomial Chaos Expansion Ω_{Γ} might not completely match Ω_T. It might not span the entire outcome space of the Target Random Variable, and/or it may include a space with zero probability under the Target Random Variable. This means that the resulting probability measures will not be exact.

3.4.4 Other Perspectives of Polynomial Chaos Expansions

The probability density function of the Target Random Variable can be visualized by Monte Carlo sampling from the Polynomial Chaos Expansion. This is possible because the simple polynomial form is easy to evaluate, so Monte Carlo is not computationally demanding.

Example 15: Polynomial Chaos Expansion of a Lognormal Random Variable

In Section 3.4.1 a lognormal Target Random Variable is expanded with a 4th order Polynomial Chaos Expansion:

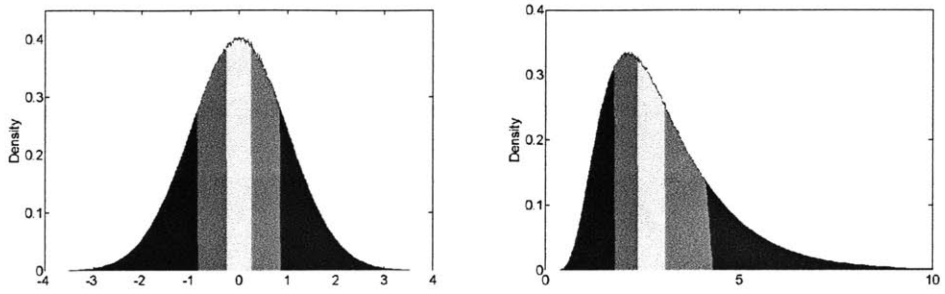
$$\hat{X}\left(\vec{\xi}\right) = x_0 + x_1\Xi + x_2\left(\Xi^2 - 1\right) + x_3\left(\Xi^3 - 3\Xi\right) + x_4\left(\Xi^4 - 6\Xi^2 + 3\right)\ldots \tag{3.23}$$

where Ξ is a standard normal Random Variable. Conceptually, the Polynomial Chaos Expansion can be thought of as a transformation of the outcome space. Every outcome from Ω_{Ξ} is nonlinearly shifted in order to represent an outcome from the Target Random Variable's outcome space Ω_T. Because of the nonlinearity, the most probable values of the Basis Random Variable do not correspond to the most probable values of the target Random Variable.

Power Series Approximation

In terms of direct representation, the lognormal can be exactly described as

$$X(\omega) = \exp\left(\frac{1}{2}Z(\omega) + 1\right)$$



(a) Gaussian Basis Random Variable

(b) 4^{th} order Polynomial Chaos Expansion

Figure 3-15: Lognormal Polynomial Chaos Expansion with values color coded by corresponding Basis Random Variable value

Also the exponential function can be written as a power series:

$$\exp(y) = \sum_{n=0}^{\infty} \frac{y^n}{n!}$$

A fourth order approximation would be:

$$\exp(y) \approx 1 + y + \frac{1}{2}y^2 + \frac{1}{6}y^3 + \frac{1}{24}y^4$$

substitute in

$$y = \frac{1}{2}z + 1$$

then

$$\exp\left(\frac{1}{2}z + 1\right) \approx \frac{65}{24} + \frac{4}{3}z + \frac{5}{16}z^2 + \frac{1}{24}z^3 + \frac{1}{384}z^4$$

This approximation is very close to the Polynomial Chaos Expansion result as seen below Equation 3.10, but the orthogonal grouping of terms in the Polynomial Chaos Expansion makes the coefficients easier to compute.

Truncated Exponential Random Variable

Use a Polynomial Chaos Expansion of a exponential Basis Random Variable to approximate a truncated exponential with support $[a, b] = [-1, -0.1]$ and mean $\mu = -0.75$. The pdf is given by

$$f_X(x) = C_1 C_2 \exp(-C_2 x)$$

where the constants are calculated by solving:

$$\begin{aligned}\mu &= \frac{\exp(-bC_2)(bC_2+1) - \exp(-aC_2)(aC_2+1)}{C_2\left[\exp(-bC_2) - \exp(-aC_2)\right]} \\ C_1 &= \exp(-bC_2) - \exp(-aC_2)\end{aligned}$$

The orthogonal polynomials for the exponential Basis Random Variable $\Xi \sim Exp(1)$ are the Laguerre Polynomials and the 4th order expansion is:

$$\begin{aligned}X \approx & x_0 + x_1(-\Xi+1) + x_2\frac{1}{2}\left(\Xi^2 - 4\Xi + 2\right) + \ldots \\ & x_3\frac{1}{6}\left(-\Xi^3 + 9\Xi^2 - 18\Xi + 6\right) + x_4\frac{1}{24}\left(\Xi^4 - 16\Xi^3 + 72\Xi^2 - 96\Xi + 24\right)\end{aligned}$$

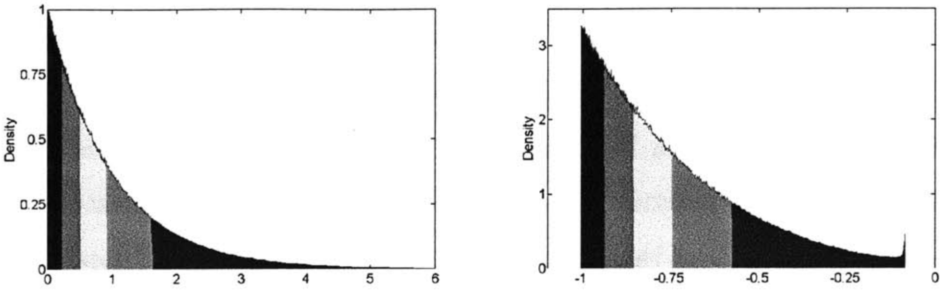
Coefficients are: $\mathbf{x} = \begin{bmatrix} -0.75 \\ -0.2050 \\ -5.108 \times 10^{-2} \\ 3.150 \times 10^{-4} \end{bmatrix}$ The Polynomial Chaos Expansion is shown in Figure 3-16.

Notice that the approximation has some artifacts. The portion of the tail that was 'cut off' has been shifted to other values, producing an unexpected spike. This should disappear as higher order Polynomial Chaos Expansions are used.

3.4.5 Uncertain parameters

Choosing Basis Random Variables

Uncertainty quantification with Polynomial Chaos Expansions is much more efficient when the Basis Random Variables are similar to the target Random Variable. While this is a nice heuristic for matching a known distribution, it is not helpful when the target Random Variable is unknown.



(a) Exponential Basis Random Variable (b) 6^{th} order Polynomial Chaos Expansion

Figure 3-16: Truncated Exponential Polynomial Chaos Expansion with values color coded by corresponding Basis Random Variable value

Therefore, the common practice for uncertainty quantification problems is to choose the Basis Random Variables based on the uncertain parameters and hope that the output is also similar. This can require some trial and error before a Basis Random Variable is found that achieves exponential convergence. For this reason, it is rare to use arbitrary Basis Random Variables because the expense of generating an orthogonal polynomial series could be wasted. For the common, Wiener–Askey family of Random Variables, the orthogonal polynomial series are well known and in fact have explicit generating functions so the recursive generation is unnecessary. This is discussed by Xiu and Karniadakis [90].

Approximating a Known Random Variable with a Polynomial Chaos Expansion

As shown in the next section, the Polynomial Chaos Expansions are much easier to manipulate in Uncertainty Quantifications problems. Therefore, we may wish to substitute a target Random Variable with a Polynomial Chaos Expansion. By using a Polynomial Chaos Expansion, we are reducing an entire probability density function to a small number of coefficients. We do this by matching statistics of the Polynomial Chaos Expansion to the target Random Variable. The more statistics that are matched, the better the Polynomial Chaos Expansion approximation will be. The statistics are typically the mean, and the central moments. If the moments of the Basis Functionals are known analytically, as they are for most common Basis Random Variables, this is formulated fairly easily into an optimization problem. The unknown coefficients are varied in order to minimize the least squares error of the target moments and the moments of the Polynomial Chaos Expansion.

Example 8b - Generating a Polynomial Chaos Expansion from Statistics

In this example we have an uncertain kinetic parameter for which we know the mean m and the variance s^2. Say we want to use a lognormal distribution to represent the uncertainty in this parameter. This is a common practice in kinetics because it forces the lognormal distribution is strictly positive. The moments of this distribution are: $\mu = \exp\left(m + \frac{s^2}{2}\right)$, and the next three central moments are:

$$\begin{aligned}
cm_2 &= \left(-1 + \exp s^2\right) \exp\left(2m + s^2\right) \\
cm_3 &= \exp\left(3m + \frac{3}{2}s^2\right)\left(-1 + \exp s^2\right)\left(\exp\left(2m + s^2\right) + 2\right) \\
cm_4 &= \exp\left(4m + 2s^2\right)\left(-1 + \exp s^2\right)^2 \times \ldots \\
&\quad \left(\exp 4s^2 + 2\exp 3s^2 + 3\exp 2s^2 - 3\right)
\end{aligned}$$

We choose the Basis Random Variable to be the standard normal due to its similarity to the target Random Variable. This is written out as $\hat{K}_1(\xi) = k_0 + k_1\Xi + k_2\left(\Xi^2 - 1\right) + k_3\left(\Xi^3 - 3\Xi\right)$. We can take advantage of the formula for moments of the standard normal Random Variable:

$$E\left[Z^n\right] = \begin{cases} 0, & \text{if } n \text{ is odd} \\ \prod_{i=1}^{\frac{n}{2}} i - 1, & \text{if } n \text{ is even} \end{cases} \tag{3.24}$$

The mean is then

$$\begin{aligned}
E_\Xi\left[\hat{K}_1(\xi)\right] &= E_\Xi\left[k_0 + k_1\Xi + k_2\left(\Xi^2 - 1\right) + k_3\left(\Xi^3 - 3\Xi\right)\right] \\
&= k_0 + k_2(1 - 1) = k_0
\end{aligned}$$

Similarly, the next three central moments, in terms of the coefficients $\mathbf{k}$, are:

$$\begin{aligned}
cm_2 =& k_1^2 + 2k_2^2 + 6k_3^2 \\
cm_3 =& 2k_2(3k_1^2 + 18k_1k_3 + 4k_2^2 + 54k_3^2) \\
cm_4 =& 3k_1^4 + 24k_1^3k_3 + 60k_1^2k_2^2 + 252k_1^2k_3^2 + 576k_1k_2^2k_3 + \dots \\
& 1296k_1k_3^3 + 60k_2^4 + 2232k_2^2k_3^2 + 3348k_3^4
\end{aligned}$$

The coefficients are computed as $\arg\min\limits_{\mathbf{k}} \sum\limits_{orders} (cm_{target} - cm_{PCE})^2$. This because of bilinear terms the objective is non-convex. Fortunately, the objective function is very cheap to compute, so a sampling algorithm was applied. The results were: $\mathbf{k1} = \begin{bmatrix} 3.0785 \\ 1.5401 \\ 0.3850 \\ 0.0642\dots \end{bmatrix}$, and the Polynomial Chaos Expansion is shown in Figure 3-14.

3.4.6 Recommended Reading

There are two textbooks on the subject of Polynomial Chaos Expansions [48, 88] that are more accessible than the original works by Wiener [86] and Ghanem and Spanos [35]. Papers by Xiu [87] and Najm [64] are good introductions, while Eldred et al. [21, 23] have compared the various methods that utilize Polynomial Chaos Expansions for uncertainty quantification.

3.4.7 Recap

This section showed how changing the representation of a Random Variable into one that utilized orthogonality reduces the problem of characterizing a Random Variable to the problem of computing a few coefficients. The previous section streamlined the formulation of uncertainty quantification problems. Now these ideas will be combined in order to solve uncertainty quantification problems far more efficiently - by reducing the problem of characterizing an unknown Random Variable from its dependencies to a problem of computing coefficients.

3.5 Uncertainty Quantification with Polynomial Chaos Expansions

Polynomial Chaos Expansions were introduced in Section 3.4 as an alternative way to represent Random Variables. Although the representation is more complex, the consequence is easier manipulation of the Random Variable. Figure 3-17 shows the entire procedure for performing uncertainty quantification with Polynomial Chaos Expansions. This begins with the integration of Polynomial Chaos Expansions into the framework shown in Section 3.2, then the calculation of the coefficients that will specify the output, and finally the verification of the solution. The framework and the solution approach are shown in this section, and theory and implementation of various methods are described in the following sections.

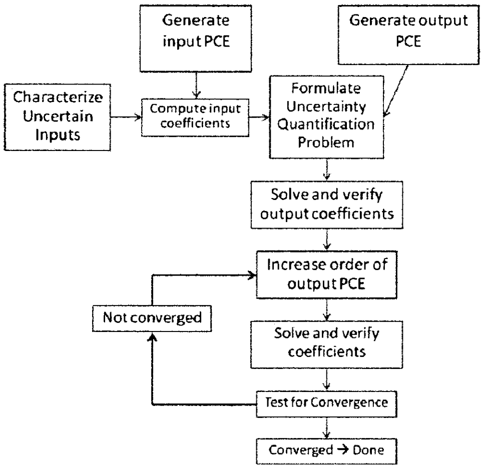

Generate input PCE
Generate output PCE
Characterize Uncertain Inputs
Compute input coefficients
Formulate Uncertainty Quantification Problem
Solve and verify output coefficients
Increase order of output PCE
Not converged
Solve and verify coefficients
Test for Convergence
Converged → Done

Figure 3-17: Overview of Polynomial Chaos Expansion Solutions to Uncertainty Quantification Examples

3.5.1 Formulating the problem

The model must be formulated for uncertainty quantification as in Section 3.2. This means that the model is an operator that maps deterministic variables u (treated as constants) and uncertain parameters $X(\omega)$, to model outputs $Y(\omega)$.

$$Y(\omega) = \mathcal{M}[X(\omega), u] \tag{3.5}$$

Let N be the number of uncertain parameters, and U be the number of variables. $Y(\omega)$ is a real-valued Random Variable defined from the model above. So model $\mathcal{M}$ maps from $\Re^{N+U} \to \Re$, which means the model has $N + U$ degrees of freedom.

The formulation is ready for uncertainty quantification, however, we can also change the way the Random Variables are represented in order to make the problem easier to solve. We know that the parameters are described by a joint probability density function. It is assumed that all the uncertain parameters are independent. Therefore each parameter can be described separately by a marginal probability density function.

$$f_X(\vec{x}) = \prod_{n=1}^{N} f_{X_n}(x_n) \tag{3.25}$$

Where $X(\omega)$ is a vector of uncertain parameters, which is decomposed into the product of N marginal distributions. Each of the N uncertain parameters is substituted with a one dimensional Polynomial Chaos Expansion. Each expansion is based on a different Basis Random Variable ξ_n.

$$X_n(\omega) \approx \hat{X}_n(\xi_n) = \sum_{p=0}^{P_n} x_{n,p} \psi_p(\xi_n) \text{ for } n = 1 \ldots N \tag{3.26}$$

where the coefficients $\mathbf{x}$ are known.

In addition, the output is replaced with an N dimensional Polynomial Chaos Expansion based on the same Basis Random Variables as the parameters.

$$Y(\omega, u) \approx \hat{Y}\left(\vec{\xi}, u\right) = \sum_{p=0}^{P} \sum_{\vec{k}:|\vec{k}|=p} y_{\vec{k}}(u) \Psi_{\vec{k}}\left(\vec{\xi}\right) \tag{3.27}$$

We emphasize that the output is still dependent on the variables u and so the coefficients will also vary with u.

The original uncertainty quantification problem is now approximated as:

$$\begin{aligned}
\hat{Y}\left(\vec{\xi}\right) &= \mathcal{M}\left[\hat{X}\left(\vec{\xi}\right), u\right] \\
\sum_{p=0}^{P}\sum_{\vec{k}:\left|\vec{k}\right|=p} y_{\vec{k}}(u)\,\Psi_{\vec{k}}\left(\vec{\xi}\right) &= \mathcal{M}\left[\left\{\sum_{p=0}^{P_n} x_{n,p}\psi_p\left(\xi_n\right)\right\}_{n=1\ldots N}, u\right] \\
\sum_{p=0}^{P}\sum_{\vec{k}:\left|\vec{k}\right|=p} \left\{y_{\vec{k}}(u)\prod_{n=1}^{N}\psi_{\vec{k}(n)}\left(\xi_n\right)\right\} &= \mathcal{M}\left[\left\{\sum_{p=0}^{P_n} x_{n,p}\psi_p\left(\xi_n\right)\right\}_{n=1\ldots N}, u\right]
\end{aligned} \tag{3.28}$$

The goal of uncertainty quantification problems is to characterize the output Random Variable. In the original formulation of the problem this was done with indirect representation - by determining the probability of every possible output value or in other words, characterizing the probability density function. In this new formulation, the output Random Variable is represented directly, and the problem is reduced to the calculation of the unknown coefficients.

No matter what representation is used, the output must be computed at each value of u. Each time the outputs are computed, however, u is held constant so in order to simplify the notation we will drop the dependence on u.

The way that the problem has been represented has reduced it to the calculation of coefficients of a Polynomial Chaos Expansion. These coefficients are computed using the Method of Weighted Residuals [26, 81]. The Method of Weighted Residuals selects coefficients that minimize the expected value of the residual. Different weighting functions result in different solution methods, and methods can generally be organized into two classes: Galerkin projection methods and collocation methods.

3.5.2 Formulating a Problem with Polynomial Chaos Expansions

Following Figure 3-17, we characterize the uncertain parameters (Section 3.4.5), choose a Basis Random Variable, generate the Polynomial Chaos Expansion, and compute the coefficients as in Section 3.4.5. Next, we use the same Basis Random Variables to generate the output Polynomial Chaos Expansion. Finally we substitute the Polynomial Chaos Expansions into the model.

Example 7d: Using Polynomial Chaos Expansions for Uncertainty Quantification

The uncertain model is:

$$D(\omega) = \frac{2}{g} S(\omega)^2 \sin(\Theta(\omega)) \cos(\Theta(\omega)) \tag{3.7}$$

The uncertain parameters are $S(\omega) \sim N(32, 1)\,\mathrm{m\,s^{-1}}$ and $\Theta(\omega) \sim U\left(\frac{\pi}{8}, \frac{\pi}{3}\right)$. These relatively easy to represent as Polynomial Chaos Expansions, because they are linear transformations of the standard normal and standard uniform Random Variables.

We will select our Basis Random Variables as: $\Xi_1 \sim N(0, 1)$ and $\Xi_2 \sim U(0, 1)$. The probability density functions are:

$$\begin{aligned} f_{\Xi_1}(\xi_1) &= \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{{\xi_1}^2}{2}\right) && \text{for } -\infty < \xi_1 < \infty \\ f_{\Xi_2}(\xi_2) &= 1 && \text{for } 0 \leq \xi_2 \leq 1 \end{aligned}$$

The parameter Polynomial Chaos Expansions are shown below. Because the parameters are simple, these expansions are exact matches.

$$\begin{aligned} \hat{S}(\xi_1) &= s_0 + s_1 \Xi_1 = 32 + (1)\,\Xi_1 \\ \hat{\Theta}(\xi_2) &= t_0 + t_1 \Xi_2 = \frac{\pi}{8} + \left(\frac{\pi}{3} - \frac{\pi}{8}\right) \Xi_2 \end{aligned}$$

The output Polynomial Chaos Expansion is chosen to be order 3.

$$\begin{aligned} \hat{D}\left(\vec{\xi}\right) =& d_{[0,0]} + d_{[1,0]} \Psi_{[1,0]}\left(\vec{\xi}\right) + d_{[0,1]} \Psi_{[0,1]}\left(\vec{\xi}\right) \\ &+ d_{[2,0]} \Psi_{[2,0]}\left(\vec{\xi}\right) + d_{[0,2]} \Psi_{[0,2]}\left(\vec{\xi}\right) + d_{[1,1]} \Psi_{[1,1]}\left(\vec{\xi}\right) \\ &+ d_{[3,0]} \Psi_{[3,0]}\left(\vec{\xi}\right) + d_{[0,3]} \Psi_{[0,3]}\left(\vec{\xi}\right) \\ &+ d_{[2,1]} \Psi_{[2,1]}\left(\vec{\xi}\right) + d_{[1,2]} \Psi_{[1,2]}\left(\vec{\xi}\right) \end{aligned}$$

$$\begin{aligned} \hat{D}\left(\vec{\xi}\right) =& d_{[0,0]} + d_{[1,0]} \Xi_1 + d_{[0,1]} (2\Xi_2 - 1) \\ &+ d_{[2,0]} \left(\Xi_1^2 - 1\right) + d_{[0,2]} \left(6\Xi_2^2 - 6\Xi_2 + 1\right) + d_{[1,1]} (\Xi_1)(2\Xi_2 - 1) \\ &+ d_{[3,0]} \left(\Xi_1^3 - 3\Xi\right) + d_{[0,3]} \left(20\Xi_2^3 - 30\Xi_2^2 + 12\Xi_2 - 1\right) \\ &+ d_{[2,1]} (\Xi_1)^2 (2\Xi_2 - 1) + d_{[1,2]} (\Xi_1)(2\Xi_2 - 1)^2 \end{aligned} \tag{3.29}$$

Now substitute $\hat{D}\left(\vec{\xi}\right)$, and $\hat{S}\left(\xi_1\right)$, and $\hat{\Theta}\left(\xi_2\right)$ into Equation 3.7:

$$\hat{D}\left(\vec{\xi}\right) \approx \frac{2}{g}(s_0 + s_1\Xi_1)^2 \sin\left(t_0 + t_1\Xi_2\right)\cos\left(t_0 + t_1\Xi_2\right) \tag{3.30}$$

Now the model is properly framed and represented. The solution and verification methods are demonstrated in following sections.

Example 8c - Formulation with Polynomial Chaos Expansions

We wish to quantify the uncertainty in the dimensionless concentration B, given by Equation 3.3, and determine how the uncertainty evolves with time. The parameters are given as lognormal Random Variables. The Polynomial Chaos Expansion approximations of these parameters are found using the steps in Section 3.4.5 with 4th order Polynomial Chaos Expansions. The first kinetic parameter is shown in Section 3.4.1. For the second kinetic parameter, the mean and $2^{nd} \sim 5^{th}$ central moments of the target distribution are matched to those of the Polynomial Chaos Expansion. The target Random Variable is $K_2\left(\omega_2\right) \sim logN(2, 0.25)$, and its statistics are: $\mu = 7.624$ and $cm = \begin{bmatrix} 3.748 \\ 5.648 \\ 57.55 \\ 274.2 \end{bmatrix}$. The analytical expressions for the moments of a 4^{th} order Polynomial Chaos Expansion are much more complex than the 3^{rd} order example in Section 3.4.5, because the number of terms grows exponentially with order. However, the analytic solutions are still easy if the moments of the Basis Random Variable are known. The least squares estimates of the coefficients are: $\mathbf{k_2} = \begin{bmatrix} 7.624 \\ 1.929 \\ 0.2458 \\ 6.309 \times 10^{-4} \\ 7.422 \times 10^{-3} \end{bmatrix}$.

Output

We choose a 6^{th} order Polynomial Chaos Expansion for the output. With two dimensions, that results in 28 coefficients and Basis Functionals. Since both Basis Random Variables are standard normal Random Variables, the orthogonal polynomial series are both Hermite. Since the output depends on the variable time, in order to use Polynomial Chaos Expansions for this uncertainty quantification problem, the time domain must be discretized and the coefficients computed at each time step. The first few terms are shown below.

$$\begin{aligned}\hat{B}\left(\vec{\xi},t\right) &= b_{[0,0]}\left(t\right) + b_{[1,0]}\left(t\right)\Psi_{[1,0]}\left(\vec{\xi}\right) + b_{[0,1]}\left(t\right)\Psi_{[0,1]}\left(\vec{\xi}\right)\\ &+ b_{[2,0]}\left(t\right)\Psi_{[2,0]}\left(\vec{\xi}\right) + b_{[0,2]}\left(t\right)\Psi_{[0,2]}\left(\vec{\xi}\right) + b_{[1,1]}\left(t\right)\Psi_{[1,1]}\left(\vec{\xi}\right)\ldots\end{aligned}$$

Substituting in the Basis Functionals:

$$\begin{aligned}\hat{B}\left(\vec{\xi},t\right) &= b_{[0,0]}\left(t\right) + b_{[1,0]}\left(t\right)\Xi_1 + b_{[0,1]}\left(t\right)\Xi_2\\ &+ b_{[2,0]}\left(t\right)\left(\Xi_1{}^2 - 1\right) + b_{[0,2]}\left(t\right)\left(\Xi_2{}^2 - 1\right) + b_{[1,1]}\left(t\right)\Xi_1\Xi_2\ldots\end{aligned}$$

Framework

Starting with Equation 3.3, we formulate the uncertainty quantification problem, then substitute in the Polynomial Chaos Expansions.

$$\begin{aligned}B\left(t\right) =& \frac{k_1}{k_2 - k_1}\left[\exp\left(-k_1 t\right) - \exp\left(-k_2 t\right)\right]\\ B\left(\omega_1,\omega_2,t\right) =& \frac{K_1\left(\omega_1\right)}{K_2\left(\omega_2\right) - K_1\left(\omega_1\right)}\left[\exp\left(-K_1\left(\omega_1\right)t\right) - \exp\left(-K_2\left(\omega_2\right)t\right)\right]\\ \hat{B}\left(\vec{\xi},t\right) =& \frac{\hat{K}_1\left(\xi_1\right)}{\hat{K}_2\left(\xi_2\right) - \hat{K}_1\left(\xi_1\right)}\left[\exp\left(-\hat{K}_1\left(\xi_1\right)t\right) - \exp\left(-\hat{K}_2\left(\xi_2\right)t\right)\right]\end{aligned}$$

Now we are ready to solve for the unknown coefficients of the output Polynomial Chaos Expansion.

3.5.3 The Method of Weighted Residuals

The Method of Weighted Residuals is an approach to solving differential equations and integrals. The idea is to construct a set of basis functions and then match them to the true solution by minimizing a residual. First, errors and residuals must be defined, then the approach itself is described.

Errors and Residuals

Ideally, the best coefficients should minimize the error ε. The direct (Equation 3.31) and indirect (Equations 3.32 and 3.33) representations are shown below.

$$\varepsilon\left(\mathbf{y}, \omega_Y, \vec{\xi}\right) = Y\left(\omega_Y\right) - \hat{Y}\left(\mathbf{y}, \vec{\xi}\right) \tag{3.31}$$

$$\varepsilon\left(\mathbf{y}, \mathsf{y}, \vec{\xi}\right) = f_Y\left(\mathsf{y}\right) - f_{\hat{Y}}\left(\mathbf{y}, \vec{\xi}\right) \tag{3.32}$$

$$\varepsilon\left(\mathbf{y}, \mathsf{y}\right) = f_Y\left(\mathsf{y}\right) - f_{\hat{Y}}\left(\mathbf{y}, \mathsf{y}\right) \tag{3.33}$$

The error depends on the coefficients $\mathbf{y}$ and the values y . The target output Random Variable $Y\left(\omega_Y\right)$ and the Polynomial Chaos Expansion $\hat{Y}\left(\vec{\xi}\right)$ have different outcome spaces, but both map to values y with the same physical meaning, which relates the two indirect representations. Every value $\vec{\xi} \in \Omega_{\xi}$ can be mapped to a value $\mathsf{y} \in \hat{\Omega}_Y$. In the indirect representations, the dependence on the coefficients $\mathbf{y}$ is hidden within the probability density function of the Polynomial Chaos Expansion.

Because the error itself is a Random Variable, it is often reduced to a statistic for reporting. The most common statistic is the L^2 norm.

$$\begin{aligned}\left\|\varepsilon\left(\mathbf{y}, \omega_Y, \vec{\xi}\right)\right\| &= \left\|Y\left(\omega_Y\right) - \hat{Y}\left(\vec{\xi}\right)\right\| \\ &= \left\|\mathcal{M}\left[X\left(\omega_x\right)\right] - \hat{Y}\left(\vec{\xi}\right)\right\|\end{aligned} \tag{3.34}$$

$$\begin{aligned}\left\|\varepsilon\left(\mathbf{y}, \mathsf{y}\right)\right\| &= \left\|f_Y\left(\mathsf{y}\right) - f_{\hat{Y}}\left(\mathsf{y}\right)\right\| \\ &= \left[\int\limits_{\Omega_Y} \left[f_Y\left(\mathsf{y}\right) - f_{\hat{Y}}^{(P)}\left(\mathsf{y}\right)\right]^2 f_Y\left(\mathsf{y}\right) dy\right]^{\frac{1}{2}}\end{aligned} \tag{3.35}$$

The error is defined over the value y . To compute the error, any values that do not have a corresponding outcome in $\hat{\Omega}_Y$ are assigned zero probability, $f_{\hat{Y}}^{(P)} = 0$. In terms of functional approximation, zero L^2 error would mean that the probability density functions of the output Random Variable and the output Polynomial Chaos Expansion are the same. Unfortunately, in general the error cannot be computed because the target output is unknown. Therefore, the residual is used as a substitute.

Define the Residual in direct form as:

$$R\left(\mathbf{y},\vec{\xi}\right) = \mathcal{M}\left[X\left(\vec{\xi}\right)\right] - \hat{Y}\left(\vec{\xi}\right) \tag{3.36}$$
$$R\left(\mathbf{y},\vec{\xi}\right) = \mathcal{M}\left[\left\{\sum_{p=0}^{P_n} x_{np}\psi_p\left(\xi_n\right)\right\}_{n=1\ldots N}\right] - \sum_{p=0}^{P}\sum_{\vec{k}:\left|\vec{k}\right|=p} y_{\vec{k}}\Psi_{\vec{k}}\left(\vec{\xi}\right)$$

The residual depends on the output coefficients and is defined over the outcome space of the Basis Random Variables. Instead of matching the output Polynomial Chaos Expansion to the target model output, it is matched to the model evaluated with the parameter Polynomial Chaos Expansions. The only unknowns are now the coefficients $\mathbf{y}$.

The Method of Weighted Residuals

In the Method of Weighted Residuals, the coefficients $\mathbf{y}$ are selected such that the expected value of the weighted residual is zero.

$$E_{\vec{\xi}}\left[R\left(\mathbf{y},\vec{\xi}\right) r_i\left(\vec{\xi}\right)\right] = 0 \text{ for } \forall i \tag{3.37}$$

Weighting functions $r_i\left(\vec{\xi}\right)$ are functions of the values of the Basis Random Variables. There are many different ways to weight the residual, which gives different kinds of minimization. Typically $|i| \geq K$, so there is at least one weighting function for each coefficient. The classes of methods that are derived using the Method of Weighted Residuals are the Galerkin Projection Methods (Section 3.6) and Collocation Methods (Section 3.7).

3.5.4 Probability Spaces and Polynomial Chaos Expansions

The target parameters $X\left(\omega_X\right)$ in the space Ω_X are projected onto the space spanned by their Polynomial Chaos Expansion representation: $\hat{\Omega}_X$. The outcomes ω_X are replaced with outcomes ξ. Any error from the projection is due to the finite order of the expansion. The same projection is done for the output Polynomial Chaos Expansion. This is equivalent to saying $Y\left(\omega_Y\right)$ in the space Ω_Y is decomposed into orthogonal components in the space Ω_Ξ. As discussed in Section 3.4.3, the use of Polynomial Chaos Expansions disrupts the physical intuition about the Random Variables.

In Section 3.5.1 there are several probability spaces in play.

- The target parameter probability spaces, $\{\Omega_{X_n}, \Sigma, P\}$ for $n = 1 \ldots N$, which are known
- The target output probability space, $\{\Omega_Y, \Sigma, P\}$ which is unknown
- The Basis Random Variable probability spaces $\{\Omega_{\xi_n}, \Sigma, P\}$ for $n = 1 \ldots N$, which are known
- The parameter Polynomial Chaos Expansions' probability spaces $\left\{\hat{\Omega}_{X_n}, \Sigma, P\right\}$
- The output Polynomial Chaos Expansion's probability space $\left\{\hat{\Omega}_Y, \Sigma, P\right\}$

Start by breaking down Equation 3.28. The original model relates uncertain parameters to uncertain output. There's an intuitive connection between the outcome spaces of the parameters and the outcome space of the output. In Equation 3.28, Polynomial Chaos Expansions are substituted for the target Random Variables. Both parameters and output Polynomial Chaos Expansions retain the same physical meaning, but are both now derived from the probability space of the Basis Random Variables. There is not an intuitive causal connection with the Basis Random Variables, only a mathematical relationship.

The key is to compute the unknown output probability space $\left\{\hat{\Omega}_Y, \Sigma, P\right\}$ such that it is consistent with the model connection to the known parameter probability space $\left\{\hat{\Omega}_{X_n}, \Sigma, P\right\}$.

3.5.5 Verification of the Polynomial Chaos Expansion Solution

Although Monte Carlo simulations can provide the true solution to uncertainty quantification problems, they cannot be relied upon for verification of Polynomial Chaos Expansion solutions because the primary purpose of using Polynomial Chaos Expansions is to limit the number of model evaluations. Another verification strategy is to test the Polynomial Chaos Expansions for convergence and assume that when an expansion converges it will has found the correct solution.

Errors, Residuals, and Convergence

The L^2 norm of the error of the output Polynomial Chaos Expansion $\hat{Y}\left(\vec{\xi}\right)$ with respect to the target output $Y(\omega)$ was given in Equation 3.35.

$$\|\varepsilon(\mathrm{y}, \mathrm{y})\| = \int\limits_{\Omega_Y} \left[f_Y(\mathrm{y}) - f_{\hat{Y}^{(P)}}(\mathrm{y})\right]^2 f_Y(\mathrm{y})\, dy \tag{3.35}$$

The residual of the output Polynomial Chaos Expansion $\hat{Y}\left(\vec{\xi}\right)$ with respect to model predictions $\mathcal{M}\left[X\left(\vec{\xi}\right)\right]$ was given in Equation 3.36:

$$R\left(\mathbf{y},\vec{\xi}\right) = \hat{Y}\left(\vec{\xi}\right) - \mathcal{M}\left[X\left(\vec{\xi}\right)\right] \tag{3.36}$$

It has been shown that as the order of a Polynomial Chaos Expansion increases to infinity, the L^2 norm of the error decreases to zero [13]. Let $\hat{Y}^{(P)}\left(\vec{\xi}\right)$ be the Pth order Polynomial Chaos Expansion approximation of $Y(\omega)$.

$$\lim_{P\to\infty}\left\|Y(\omega) - \hat{Y}^{(P)}\left(\vec{\xi}\right)\right\| = 0$$

We would like to know the rate of error convergence as P increases and the minimum order required to accurately describe the target Random Variable. Unfortunately the error cannot be calculated, so this convergence property cannot be tested. Instead, we define the expansion order difference Δ^P:

$$\Delta^P = \left\|\hat{Y}^{(P)}\left(\vec{\xi}\right) - \hat{Y}^{(P-1)}\left(\vec{\xi}\right)\right\| \tag{3.38}$$

and examine its convergence as P increases. When Δ^P meets some criterion, we say that the expansion has converged and we assume that the error is negligible. Another possible metric would be the difference between expansions of orders P and $P-2$. This could be advantageous because the errors often depend on whether the expansion's highest-order term is even or odd.

The expansion order difference is defined using the L^2 norm. Some early papers advocated computing differences only at a limited number of points $\vec{\xi}^{(l)}$ for $l = 1\ldots N_l$ in order to approximate the L^2 norm:

$$\sum_{l=1}^{N_l}\left[\hat{Y}^{(P+1)}\left(\vec{\xi}^{(l)}\right) - \hat{Y}^{(P)}\left(\vec{\xi}^{(l)}\right)\right]^2$$

or use the difference in statistics like mean and variance:

$$\left|E\left[\hat{Y}^{(P+1)}\left(\vec{\xi}\right)\right] - E\left[\hat{Y}^{(P)}\left(\vec{\xi}\right)\right]\right|$$
$$\left|\mathrm{var}\left[\hat{Y}^{(P+1)}\left(\vec{\xi}\right)\right] - \mathrm{var}\left[\hat{Y}^{(P)}\left(\vec{\xi}\right)\right]\right|$$

These alternatives would save some calculations however, the comparison in Equation 3.38 is more stringent and is quite cheap to compute using Monte Carlo. So there is no reason to use other convergence metrics.

Verification Strategy

The verification strategy is illustrated in Figure 3-17. An output Polynomial Chaos Expansion of order P is used to solve the problem as discussed in Sections 3.4-sec:results. For every solution, the computed coefficients must be verified as discussed below. Then the problem is solved a second time with an output Polynomial Chaos Expansion of order $P+1$ and the expansion order difference Δ^P is computed. If the convergence criterion is met, then the $P+1$ order expansion is the verified solution. Otherwise, the process is repeated for the $P+2$ order and so on.

Expansion Convergence

The criterion for convergence of the output Polynomial Chaos Expansion is:

$$\Delta^P = \left\| \hat{Y}^{(P)}\left(\vec{\xi}\right) - \hat{Y}^{(P-2)}\left(\vec{\xi}\right) \right\| \leq \tau \tag{3.39}$$

The algorithm is the same no matter which method is used to compute the coefficients. However, there are some notable differences which may have an impact on the choice of solution methods, these differences are discussed with the solution methods.

3.5.6 Effect of errors

Errors in the parameters

If the parameter Polynomial Chaos Expansions do not match the target parameter Random Variables, this will add to the error in the output Polynomial Chaos Expansion. However, this is impossible to determine from the residual, it can only be detected by comparison to a Monte Carlo uncertainty quantification solution. This must be caught and fixed when generating the parameter Polynomial Chaos Expansion as discussed in Section 3.4.5.

Errors from Model Evaluation

Complex models will have some error in their solvers, which prevents the model from being calculated exactly. This is assumed to have negligible effect compared to errors from the uncertainty quantification analysis.

3.5.7 Recap

Verification is an important and often overlooked step in the solution of uncertainty quantification problems. The process typically requires the uncertainty quantification problem to be solved many times, which reinforces the motivation to reduce the computational cost.

3.6 Projection Methods

3.6.1 Derivation using the Method of Weighted Residuals

Projection methods are derived from the method of weighted residuals by using the K Basis Functionals as the weighting functions.

For each $\vec{k}$

$$\begin{aligned} r_{\vec{k}}\left(\vec{\xi}\right) &= \Psi_{\vec{k}}\left(\vec{\xi}\right) \\ E_{\vec{\Xi}}\left[R\left(\mathbf{y},\vec{\xi}\right) r_{\vec{k}}\left(\vec{\xi}\right)\right] &= 0 \\ E_{\vec{\Xi}}\left[R\left(\mathbf{y},\vec{\xi}\right) \Psi_{\vec{k}}\left(\vec{\xi}\right)\right] &= 0 \\ \int\limits_{\Omega_{\vec{\Xi}}} R\left(\mathbf{y},\vec{\xi}\right) \Psi_{\vec{k}}\left(\vec{\xi}\right) d\vec{\xi} &= 0 \end{aligned}$$

Split up the Residual,

$$\begin{gathered} \int\limits_{\Omega_{\vec{\Xi}}} \left[\sum_{p=0}^{P} \sum_{\vec{j}:|\vec{j}|=p} y_{\vec{j}} \Psi_{\vec{j}}\left(\vec{\xi}\right)\right] \Psi_{\vec{k}}\left(\vec{\xi}\right) f_{\vec{\Xi}}\left(\vec{\xi}\right) d\vec{\xi} = \\ \int\limits_{\Omega_{\vec{\Xi}}} \mathcal{M}\left[\left\{\sum_{p=0}^{P_n} x_{np}\psi_p\left(\xi_n\right)\right\}_{n=1\ldots N}\right] \Psi_{\vec{k}}\left(\vec{\xi}\right) f_{\vec{\Xi}}\left(\vec{\xi}\right) d\vec{\xi} \end{gathered} \tag{3.40}$$

Because of orthogonality of the Basis Functionals, the left hand side of Equation 3.40 reduces:

$$\begin{aligned}
&\sum_{p=0}^{P} \sum_{\vec{j}:|\vec{j}|=p} \mathbf{y}_{\vec{j}} \int\limits_{\Omega_{\vec{\Xi}}} \Psi_{\vec{j}}\left(\vec{\xi}\right) \Psi_{\vec{k}}\left(\vec{\xi}\right) f_{\vec{\Xi}}\left(\vec{\xi}\right) d\vec{\xi} \\
&= \sum_{p=0}^{P} \sum_{\vec{j}:|\vec{j}|=p} \mathbf{y}_{\vec{j}} \left\langle \Psi_{\vec{j}} \Psi_{\vec{k}} \right\rangle \\
&= y_{\vec{k}} \left\langle \Psi_{\vec{k}} \Psi_{\vec{k}} \right\rangle
\end{aligned}$$

So Equation 3.40 becomes:

$$\begin{aligned}
y_{\vec{k}} \left\langle \Psi_{\vec{k}} \Psi_{\vec{k}} \right\rangle &= \int\limits_{\Omega_{\vec{\Xi}}} \mathcal{M} \left[\left\{ \sum_{p=0}^{P_n} x_{np} \psi_p \left(\xi_n\right) \right\}_{n=1\ldots N} \right] \Psi_{\vec{k}}\left(\vec{\xi}\right) f_{\vec{\Xi}}\left(\vec{\xi}\right) d\vec{\xi} \\
y_{\vec{k}} &= \frac{1}{\left\langle \Psi_{\vec{k}} \Psi_{\vec{k}} \right\rangle} \int\limits_{\Omega_{\vec{\Xi}}} \mathcal{M} \left[\left\{ \sum_{p=0}^{P_n} x_{np} \psi_p \left(\xi_n\right) \right\}_{n=1\ldots N} \right] \Psi_{\vec{k}}\left(\vec{\xi}\right) f_{\vec{\Xi}}\left(\vec{\xi}\right) d\vec{\xi} \qquad (3.41)
\end{aligned}$$

This results in a set of K equations, each of which determines one coefficient. They are decoupled, meaning that the value of each coefficient is independent of the others. This property is a result of the orthogonality of the Basis Functionals. The same equations can be derived using projection, which gives this class of methods its name.

3.6.2 Derivation using the projection approach

The same K equations can be derived by projecting the output Random Variable onto each of the K Basis Functionals by taking the inner product with respect to the Basis Random Variables.

$$\text{for each } \vec{k}$$

$$\begin{aligned}
&\left\langle Y(\omega) \Psi_{\vec{k}}\left(\vec{\xi}\right) \right\rangle \text{ w.r.t. } \vec{\xi} \\
&= \int\limits_{\Omega_{\vec{\Xi}}} Y(\omega) \Psi_{\vec{k}}\left(\vec{\xi}\right) f_{\vec{\xi}}\left(\vec{\xi}\right) d\vec{\xi}
\end{aligned}$$

Approximate the output using the Polynomial Chaos Expansion (indexed by $\vec{j}$ to avoid confusion) and simplify using orthogonality properties.

$$
\begin{aligned}
&\approx \int_{\Omega_{\vec{\Xi}}} \hat{Y}\left(\vec{\xi}\right) \Psi_{\vec{k}}\left(\vec{\xi}\right) f_{\vec{\Xi}}\left(\vec{\xi}\right) d\vec{\xi} \\
&= \int_{\Omega_{\vec{\Xi}}} \left(\sum_{p=0}^{P} \sum_{\vec{j}:|\vec{j}|=p} y_{\vec{j}} \Psi_{\vec{j}}\left(\vec{\xi}\right) \right) \Psi_{\vec{k}}\left(\vec{\xi}\right) f_{\vec{\Xi}}\left(\vec{\xi}\right) d\vec{\xi} \\
&= \int_{\Omega_{\vec{\Xi}}} y_{\vec{k}} \Psi_{\vec{k}}\left(\vec{\xi}\right) \Psi_{\vec{k}}\left(\vec{\xi}\right) f_{\vec{\Xi}}\left(\vec{\xi}\right) d\vec{\xi}
\end{aligned}
$$

Then the projection is approximately:

$$
\left\langle Y(\omega) \Psi_{\vec{k}}\left(\vec{\xi}\right) \right\rangle \approx y_{\vec{k}} \left\langle \Psi_{\vec{k}}\left(\vec{\xi}\right) \Psi_{\vec{k}}\left(\vec{\xi}\right) \right\rangle
$$

Solve for the coefficient:

$$
y_{\vec{k}} \approx \frac{\left\langle Y(\omega) \Psi_{\vec{k}} \right\rangle}{\left\langle \Psi_{\vec{k}} \Psi_{\vec{k}} \right\rangle} = \frac{1}{\left\langle \Psi_{\vec{k}} \Psi_{\vec{k}} \right\rangle} \int_{\Omega_{\vec{\Xi}}} Y(\omega) \Psi_{\vec{k}}\left(\vec{\xi}\right) f_{\vec{\Xi}}\left(\vec{\xi}\right) d\vec{\xi}
$$

Rewrite the output again in terms of the parameter Polynomial Chaos Expansion

$$
\begin{aligned}
y_{\vec{k}} &\approx \frac{1}{\left\langle \Psi_{\vec{k}} \Psi_{\vec{k}} \right\rangle} \int_{\Omega_{\vec{\Xi}}} \hat{Y}\left(\vec{\xi}\right) \Psi_{\vec{k}}\left(\vec{\xi}\right) f_{\vec{\Xi}}\left(\vec{\xi}\right) d\vec{\xi} \\
y_{\vec{k}} &\approx \frac{1}{\left\langle \Psi_{\vec{k}} \Psi_{\vec{k}} \right\rangle} \int_{\Omega_{\vec{\Xi}}} \mathcal{M} \left[\left\{ \sum_{p=0}^{P_n} x_{np} \psi_p\left(\xi_n\right) \right\}_{n=1\ldots N} \right] \Psi_{\vec{k}}\left(\vec{\xi}\right) f_{\vec{\Xi}}\left(\vec{\xi}\right) d\vec{\xi}
\end{aligned}
$$

3.6.3 Orthogonality and Inner Products

These K weighting functions produce a system of K independent equations. However, each equation has two N dimensional integrals over the Basis Random Variables $\vec{\xi}$. Because of orthogonality, the denominator $\left\langle \Psi_{\vec{k}} \Psi_{\vec{k}} \right\rangle$ can reduced to the product of 1-dimensional integral which is very easy to solve.

$$\begin{aligned}\left\langle \Psi_{\vec{k}} \Psi_{\vec{k}} \right\rangle &= \int\limits_{\Omega_{\vec{\Xi}}} \Psi_{\vec{k}}\left(\vec{\xi}\right) \Psi_{\vec{k}}\left(\vec{\xi}\right) f_{\vec{\Xi}}\left(\vec{\xi}\right) d\vec{\xi} \\ &= \int\limits_{\Omega_{\vec{\Xi}}} \left[\prod_{n=1}^{N} \psi_{\vec{k}(n)}\left(\xi_n\right)\right]^2 \prod_{n=1}^{N} f_{\Xi_n}\left(\xi_n\right) d\vec{\xi} \\ &= \int\limits_{\Omega_{\vec{\Xi}}} \prod_{n=1}^{N} \left[\psi_{\vec{k}(n)}\left(\xi_n\right)\right]^2 f_{\Xi_n}\left(\xi_n\right) d\vec{\xi} \\ &= \prod_{n=1}^{N} \int\limits_{\Omega_n} \left[\psi_{\vec{k}(n)}\left(\xi_n\right)\right]^2 f_{\Xi_n}\left(\xi_n\right) d\xi_n\end{aligned}$$

However, the model operator $\mathcal{M}$ disrupts the orthogonality of the Basis Functionals that comprise the parameter Polynomial Chaos Expansions, so the numerator integral is N-dimensional. This can be very expensive to compute when N is large.

3.6.4 Physical Intuition

Because the Basis Functionals are orthogonal, none of their 'coverage' will overlap in the probability space, and the sum of individual Basis Functional's coverage of Ω_Y will equal the coverage as a whole. This will minimize the mean-square error $R\left(\omega\right) = Y\left(\omega\right) - \hat{Y}\left(\omega\right)$ in the space $\hat{\Omega_Y}$. However, this does not necessarily provide the best functional approximation when the spaces Ω_Y and $\hat{\Omega_Y}$ are not equivalent.

3.6.5 Standard Galerkin Projection Methods

In the literature this has many names: Spectral projection, Stochastic Galerkin, Galerkin Projection, intrusive methods, etc. The major issue with the Galerkin Method is that computing the integral requires that the equations that make up operator $\mathcal{M}$ be known. These are then used to analytically solve for the Polynomial Chaos Expansion coefficients. This is not the case for many engineering applications where models are solved numerically. Therefore, in addition to the standard Galerkin Projection Method where the integrals are solved analytically, there are many variations that solve the integrals numerically.

Example 7e: Target Practice Uncertainty Quantification with Projection Methods

We will begin this work where Section 3.5.2 left off, with the formulation complete and the model in terms of the Basis Random Variables. This problem is solvable with the Galerkin Projection Method because the model is simple. The 2-dimensional model decomposes into the product of two 1-dimensional functions.

It is also useful to know the moments of the standard normal Random Variable Z, shown in Equation 3.24

The number of Basis Functionals and coefficients in the output Polynomial Chaos Expansion is $K = 10$. To solve for the unknown coefficients, the model shown in Equation 3.30 is projected onto each Basis Functional. For example the second Basis Functional:

$$
\begin{aligned}
d_{[1,0]} &= \frac{1}{\langle \xi_1, \xi_1 \rangle} \int\limits_{\Omega_{\vec{\Xi}}} \hat{D}\left(\vec{\xi}\right) \xi_1 f_{\vec{\Xi}}\left(\vec{\xi}\right) d\vec{\xi} \\
d_{[1,0]} &= \frac{1}{\int\limits_{\Omega_1}\int\limits_{\Omega_2} \xi_1 \xi_1 f_{\Xi_2}(\xi_2)\, d\xi_2 f_{\Xi_1}(\xi_1)\, d\xi_1} \int\limits_{\Omega_1}\int\limits_{\Omega_2} \hat{D}\left(\vec{\xi}\right) \xi_1 f_{\Xi_2}(\xi_2)\, d\xi_2 f_{\Xi_1}(\xi_1)\, d\xi_1 \\
d_{[1,0]} &= \frac{\int\limits_{\Omega_1}\int\limits_{\Omega_2} \frac{2}{g}(s_0 + s_1\xi_1)^2 \sin(t_0 + t_1\xi_2) \cos(t_0 + t_1\xi_2)\, \xi_1 f_{\Xi_2}(\xi_2)\, d\xi_2 f_{\Xi_1}(\xi_1)\, d\xi_1}{\int\limits_{\Omega_1}\int\limits_{\Omega_2} \xi_1 \xi_1 f_{\Xi_2}(\xi_2)\, d\xi_2 f_{\Xi_1}(\xi_1)\, d\xi_1}
\end{aligned}
$$

Both two-dimensional integrals can be separated into products of one-dimensional integrals.

$$
\begin{aligned}
d_{[1,0]} &= \frac{2}{g} \frac{\int\limits_{\Omega_1} \xi_1 (s_0 + s_1\xi_1)^2 f_{\Xi_1}(\xi_1)\, d\xi_1 \int\limits_{\Omega_2} \sin(t_0 + t_1\xi_2) \cos(t_0 + t_1\xi_2) f_{\Xi_2}(\xi_2)\, d\xi_2}{\int\limits_{\Omega_1} \xi_1 \xi_1 f_{\Xi_1}(\xi_1)\, d\xi_1 \int\limits_{\Omega_2} f_{\Xi_2}(\xi_2)\, d\xi_2} \\
d_{[1,0]} &= \frac{2}{g} \int\limits_{\Omega_1} \left({s_0}^2 \xi_1 + 2 s_0 s_1 {\xi_1}^2 + s_1^2 \xi_1^3\right) f_{\Xi_1}(\xi_1)\, d\xi_1 \ldots \\
&\quad \int\limits_{\Omega_2} \sin(t_0 + t_1\xi_2) \cos(t_0 + t_1\xi_2)\, d\xi_2
\end{aligned}
$$

$$
\begin{aligned}
u &= \cos\left(t_0 + t_1\xi_2\right) \\
\frac{du}{t_1} &= \sin\left(t_0 + t_1\xi_2\right) d\xi_2
\end{aligned}
$$

$$
\begin{aligned}
d_{[1,0]} =& \frac{2}{g}\left({s_0}^2 E\left[\xi_1\right] + 2 s_0 s_1 E\left[{\xi_1}^2\right] + s_1^2 E\left[\xi_1^3\right]\right) \int\limits_{\Omega_{\xi_2 \to u}} u \frac{1}{t_1} du \\
d_{[1,0]} =& \frac{2}{g} s_0 s_1 \frac{1}{t_1} \cos^2\left(t_0 + t_1\xi_2\right)\Big|_0^1 \\
d_{[1,0]} \approx& 6.0713
\end{aligned}
$$

This process must be done for each Basis Functional and coefficient.

3.6.6 Numerical Integration of Galerkin Projection Equations

Start with Equation 3.41

$$
y_{\vec{k}} \approx \frac{1}{\left\langle \Psi_{\vec{k}} \Psi_{\vec{k}} \right\rangle} \int\limits_{\hat{\Omega}} \mathcal{M}\left[\left\{\sum_{p=0}^{P_n} x_{np}\psi_p\left(\xi_n\right)\right\}_{n=1\ldots N}\right] \Psi_{\vec{k}}\left(\vec{\xi}\right) p_{\vec{\xi}} d\vec{\xi} \tag{3.41}
$$

Instead of solving the integral analytically as in the standard Galerkin Projection Method, calculate it numerically using quadrature.

Select N_Q quadrature nodes $\vec{\xi}^{(q)}$ for $q = 1 \ldots N_Q$

$$
\begin{aligned}
y_{\vec{k}} &\approx \frac{1}{\left\langle \Psi_{\vec{k}} \Psi_{\vec{k}} \right\rangle} \sum_{q=1}^{N_Q} \mathcal{M}\left[\hat{X}\left(\vec{\xi}^{(q)}\right)\right] \Psi_{\vec{k}}\left(\vec{\xi}^{(q)}\right) w_q \\
y_{\vec{k}} &\approx \frac{1}{\left\langle \Psi_{\vec{k}} \Psi_{\vec{k}} \right\rangle} \sum_{q=1}^{N_Q} \mathcal{M}\left[\left\{\sum_{p=0}^{P_n} x_{n,p}\psi_p\left(\xi_n^{(q)}\right)\right\}_{n=1\ldots N}\right] \Psi_{\vec{k}}\left(\vec{\xi}^{(q)}\right) w_q \\
y_{\vec{k}} &\approx \frac{1}{\left\langle \Psi_{\vec{k}} \Psi_{\vec{k}} \right\rangle} \sum_{q=1}^{N_Q} m_Q B_{\vec{k},q} w_q
\end{aligned}
$$

where

- Each quadrature node is an N-vector $\vec{\xi}^{(q)} = \left[\xi_n^{(q)}\right]_{n=1\ldots N}$ chosen from $\Omega_{\vec{\Xi}}$
- m_Q is the model evaluated at the parameters corresponding to the qth quadrature node

- w_Q is the quadrature weight of the quadrature node
- $B_{\vec{k},q}$ is the $\vec{k}$th Basis Functional in the output Polynomial Chaos Expansion, evaluated at the qth quadrature node. $B_{\vec{k},q} = \Psi_{\vec{k}}\left(\vec{\xi}^{(q)}\right)$. This forms a $K \times N_Q$ matrix $\mathbf{B}$.

Simplify this further to:

$$\mathbf{y} = \mathbf{PBWm} = \mathbf{A}^{(proj)}\mathbf{m} \tag{3.42}$$

where

- $\mathbf{y}$ are the coefficients - $K \times 1$
- $\mathbf{m}$ are the model outputs at the quadrature nodes $N_Q \times 1$
- $\mathbf{A}^{(proj)}$ is a $K \times N_Q$ matrix:
- $\mathbf{P}$ is a $K \times K$ matrix of Inner Products of Basis Functionals, called the normalizing factors $P_{\vec{k},\vec{k}} = \frac{1}{\langle\Psi_{\vec{k}}\Psi_{\vec{k}}\rangle}$ or $\mathbf{P} = diag\left(\frac{1}{\langle\Psi_{\vec{k}}\Psi_{\vec{k}}\rangle}\right)$. As shown in Section 3.6.3 these are easy to compute analytically because of orthogonality.
- $\mathbf{W}$ is a weighting matrix - $N_Q \times N_Q$ given by $\mathbf{W} = diag\left(w_q\right)$

The choice of quadrature nodes and weights is based on work for deterministic integrals [75]. These algorithms are discussed briefly below.

3.6.7 Projection Methods using Numerical Integration

Many techniques have been proposed for calculating coefficients of Polynomial Chaos Expansions using different numerical integration schemes. Among them are:

- Pseudo Spectral Polynomial Chaos - refers to any method that uses numerical integration
- Non-intrusive Galerkin - refers to any method that uses numerical integration
- Monte Carlo - calculates the integral with monte carlo
- Gaussian Quadrature - uses tensor products of one-dimensional Gaussian quadrature points
- Sparse Grids/ Cubature - uses sparse grids of one-dimensional Gaussian quadrature
- Stochastic Collocation - seems to refer to any method using numerical cubature. The name is unfortunate because the Basis Random Variables to not need to represent a stochastic quantity and it does not use a collocation method.

Two of these methods, Monte Carlo and Gaussian Quadrature, are demonstrated in Examples 7 and 7.

Example 7f: Projection Method using Monte Carlo

Monte Carlo is a very intuitive numerical integration method. The idea of sampling parameters is the same as the method described in Section 3.2 but instead of propagating uncertainty, Monte Carlo is used to evaluate an integral.

$$\int_a^b h(x)\, f_X(x)\, dx \approx \frac{1}{N_Q} \sum_{q=1}^{N_Q} h(x_q)$$

where x_q are sampled from the probability density function $f_X(x)$ and the bounds of x are $[a, b]$. This can be applied to each of the projection equations.

Example 7 has been solved previously in Section 3.3.2 and revisited in Section 3.6.5. Instead of sampling from the uncertain parameters, this version of Monte Carlo samples from the Basis Random Variables. These are then treated as quadrature nodes in the method from Section 3.6.6. Instead of propagating the samples through the model in order to characterize the output probability density function, this method uses the model outputs to compute the coefficients of the output Polynomial Chaos Expansion, which then approximates the model output.

For each $\vec{k}$, sample N_Q quadrature nodes from the probability density function of $\vec{\xi}$. Each of these samples has a weight of $w_q = \frac{1}{N_Q}$. For each sample calculate p_q and h_q:

$$\begin{aligned} p_q =& \Psi_{\vec{k}}\left(\vec{\xi}^{(q)}\right) \Psi_{\vec{k}}\left(\vec{\xi}^{(q)}\right) \\ h_q =& \mathcal{M}\left[\hat{X}\left(\vec{\xi}^{(q)}\right)\right] \Psi_{\vec{k}}\left(\vec{\xi}^{(q)}\right) \end{aligned}$$

Then the coefficient can be computed as:

$$y_{\vec{k}} \approx \frac{1}{\frac{1}{N_Q}\sum\limits_{q=1}^{N_Q} p_q} \frac{1}{N_Q} \sum_{q=1}^{N_Q} h_q$$

The result of a Monte Carlo simulation with 10^6 samples are displayed in Figure 3-18. The solution was verified by running an ensemble of 10^4 Monte Carlo simulations, and the results, standard deviations, and multi-indicies are:

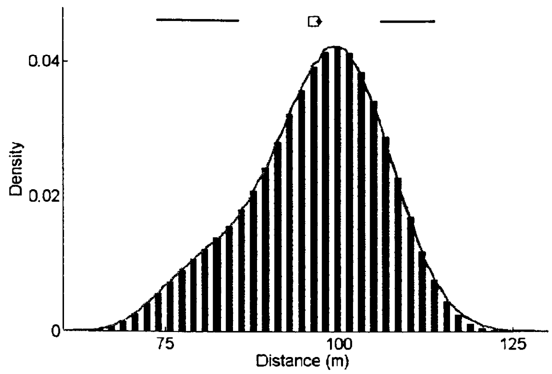


Figure 3-18: Example 7 – Uncertainty Quantification by Monte Carlo (—) and a Polynomial Chaos Expansion using the Projection method with Monte Carlo (bars)

$$\mathbf{y} = \begin{bmatrix} 96.45 \\ 6.023 \\ 8.559 \\ 0.09492 \\ -14.36 \\ 0.5346 \\ 2.229e-4 \\ -0.2440 \\ 5.593e-3 \\ -0.8971 \end{bmatrix}, \sigma_{\mathbf{y}} = \begin{bmatrix} 0.01004 \\ 0.09764 \\ 0.1571 \\ 0.06863 \\ 0.2166 \\ 0.1588 \\ 0.03999 \\ 0.2492 \\ 0.1125 \\ 0.2118 \end{bmatrix}, \vec{k} = \left\{ \begin{matrix} 0,0 \\ 1,0 \\ 0,1 \\ 1,1 \\ 2,0 \\ 0,2 \\ 1,2 \\ 3,0 \\ 0,3 \\ 2,1 \end{matrix} \right\}$$

The biggest concern is that the largest coefficients are calculated accurately. Higher order terms typically have less impact on the Polynomial Chaos Expansion than lower order terms, so a larger relative uncertainty in the last few coefficients can be tolerated. When the coefficients are computed with Monte Carlo, the same verification strategy can be used as in Section 3.3.4.

The Monte Carlo method gives good accuracy in the calculation of coefficients and is easy to verify, but it is inefficient as discussed in Section 3.2. The main reason we use Polynomial Chaos Expansions to represent the problem is to reduce the number of model evaluations required. Therefore other numerical integration schemes are more attractive.

Example 7g: Projection Method using Gaussian Quadrature

Gaussian quadrature is a one-dimensional integration method. In multiple dimensions, tensor products of the one-dimensional quadrature nodes are used as quadrature nodes. For this reason the method is also known as the Tensor Product method. The corresponding weights are the products of the one-dimensional weights.

The Algorithm

1. For each Basis Random Variable, indexed by n
   - Select the number of 1-dimensional quadrature nodes Q
   - Find the roots of the Qth order Basis Functional and denote these by: ${\xi_n}^{(i)}$ for $i = 1 \ldots Q$
   - For each root, calculate the weights according to given formulas.
2. Generate the N-dimensional quadrature nodes by selecting one 1-dimensional node from each of the N dimensions. There will be $N_Q = Q^N$ quadrature nodes.
3. Generate the weights for each N-dimensional quadrature node by taking the product of each of the respective 1-dimensional weights. Again the number is $N_Q = Q^N$.
4. Evaluate the model at each quadrature node
5. Evaluate each Basis Functional at each quadrature node
6. Proceed as described in Section 3.6.6

As shown in Section 3.3.2, the Target Practice uncertainty quantification example can be solved well using a third order output Polynomial Chaos Expansion. But Monte Carlo is too expensive. Here we will attempt to solve the same problem with Tensor Products. The Polynomial Chaos Expansion representation of this problem is given in Equations 3.29 and 3.30. There are 10 Galerkin Projection equations, like the one shown in Example 3.6.5. These will be solved with Gaussian quadrature.

The Basis Random Variables are Normal and Uniform, so the Basis Functionals are Hermite and Shifted Legendre Polynomials. For illustration, four nodes will be used in each dimension, giving a total of 16 collocation points. The relevant equations for the nodes are:

$$\hat{S}_4(\xi_1) = {\Xi_1}^4 - 6{\Xi_1}^2 + 3$$

$$\hat{\Theta}_4(\xi_2) = 70\xi_2^4 - 140\xi_2^3 + 90\xi_2^2 - 20\Xi_2 + 1$$

For roots of Hermite polynomials, the weights are:

$$w_H^{(i)} = \frac{Q!}{Q^2\left[H_{Q-1}\left({\xi_n}^{(i)}\right)\right]^2}$$

where $H_Q(x)$ is the Qth order Hermite Polynomial. For roots of Shifted Legendre Polynomials, the weights are:

$$w_L^{(i)} = \frac{2}{{\xi_n}^{(i)}\left(1-{\xi_n}^{(i)}\right)\left[L'_Q\left({\xi_n}^{(i)}\right)\right]^2}$$

where $L_Q(x)$ is the Qth order Shifted Legendre Polynomial. Using these equations, the roots and weights are:

$$\xi_1, w_H = \begin{cases} \pm 2.334, & 0.04588 \\ \pm 0.7420, & 0.4541 \end{cases} \text{ and } \xi_2, w_L = \begin{cases} \frac{1\pm\sqrt{\left(3-2\sqrt{6/5}\right)/7}}{2}, & \frac{18+\sqrt{30}}{36} \\ \frac{1\pm\sqrt{\left(3+2\sqrt{6/5}\right)/7}}{2}, & \frac{18-\sqrt{30}}{36} \end{cases}$$

and the 2-dimensional quadrature node are the 16 possible combinations of these roots, denoted by:

$$\left\{\vec{\xi}^{(q)}\right\}_{q=1\ldots16} = \left\{\begin{matrix} {\xi_1}^{(1)}, {\xi_2}^{(1)} \\ \vdots \\ {\xi_1}^{(4)}, {\xi_2}^{(4)} \end{matrix}\right\} \text{ and } \mathbf{W} = \begin{bmatrix} w_H^{(1)} w_L^{(1)} & & 0 \\ & \ddots & \\ 0 & & w_H^{(4)} w_L^{(4)} \end{bmatrix}.$$

At each node, the model and each Basis Functional are evaluated.

The model outputs are stored in the vector $\mathbf{m} = \begin{bmatrix} \mathcal{M}\left[\hat{X}\left(\vec{\xi}^{(1)}\right)\right] \\ \vdots \\ \mathcal{M}\left[\hat{X}\left(\vec{\xi}^{(N_Q)}\right)\right] \end{bmatrix}.$

The Basis Functional values are stored in $\mathbf{B} = \begin{bmatrix} \Psi_{[0,0]}\left(\vec{\xi}^{(1)}\right) & \cdots & \Psi_{[2,2]}\left(\vec{\xi}^{(1)}\right) \\ \vdots & \ddots & \\ \Psi_{[0,0]}\left(\vec{\xi}^{(16)}\right) & & \Psi_{[2,2]}\left(\vec{\xi}^{(16)}\right) \end{bmatrix}$

Finally, the coefficients can be calculated, following Equation 3.42

$$\mathbf{y} = \mathbf{PBWm} = \mathbf{A}^{(proj)}\mathbf{m} \tag{3.42}$$

The results are $\mathbf{y} = \begin{bmatrix} 96.45 \\ 6.024 \\ 8.559 \\ 0.5338 \\ 0.09434 \\ -14.36 \\ -0.8958 \\ 1.110e-16 \\ -0.2507 \\ 0.008343 \end{bmatrix}$, and there is very good agreement with the Polynomial Chaos Expansion solved with Monte Carlo. There were only 16 model evaluations for this solution, many orders of magnitude fewer than Monte Carlo. The quadrature nodes and output Polynomial Chaos Expansion are shown in Figure 3-19

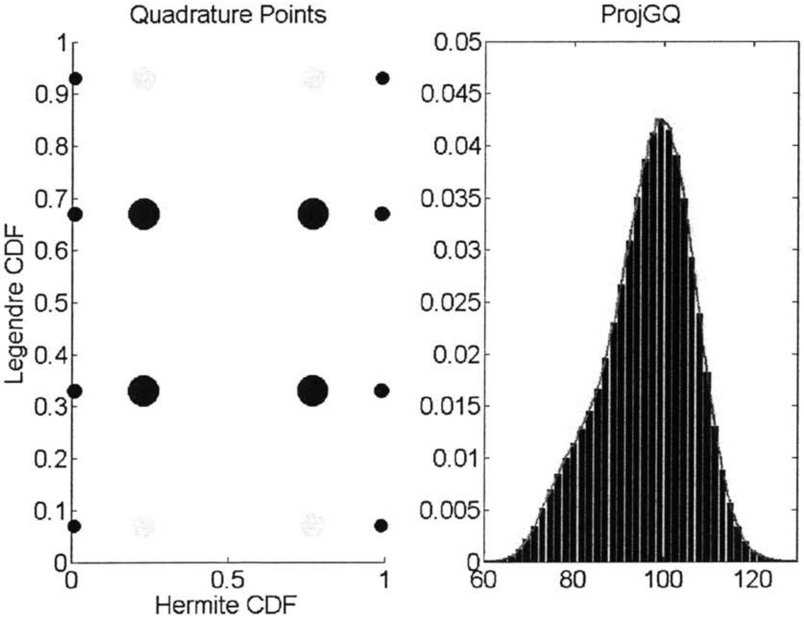


Figure 3-19: Example 7 – Gaussian quadrature points (weights indicated by point size) and the 4^{th} order output Polynomial Chaos Expansion

When the solution is repeated with 5-node quadrature, the coefficients are within one percent of the 4-node solution. This indicates that the coefficients are computed with acceptable accuracy. Next the order of the Polynomial Chaos Expansion is increased to $P = 4$. The coefficients are computed again with a 4-node quadrature scheme.

$$\mathbf{y} = \begin{bmatrix} 96.45 \\ 6.022 \\ 8.558 \\ 0.5354 \\ 0.09474 \\ -14.34 \\ -0.8974 \\ 1.332e-15 \\ -0.2504 \\ 0.008403 \\ 3.455e-15 \\ 2.335e-12 \\ -0.01566 \\ -0.01412 \\ -2.220e-15 \end{bmatrix} \text{ with multi-indices } \vec{k} = \begin{Bmatrix} 0,0 \\ 1,0 \\ 0,1 \\ 1,1 \\ 2,0 \\ 0,2 \\ 1,2 \\ 3,0 \\ 0,3 \\ 2,1 \\ 4,0 \\ 0,4 \\ 1,3 \\ 2,2 \\ 3,1 \end{Bmatrix}$$

The added coefficients are relatively small, so they have negligible impact on the Polynomial Chaos Expansion. In fact, the 3rd order coefficients are also small, meaning that the 2nd and 4th order expansions are essentially the same. Therefore we conclude that the expansion is converged, and the solution is verified. In this case, we also have the luxury of knowing the Monte Carlo solution, and we can compute the L^2 norm of the difference between the two solutions to be $O\left(10^{-6}\right)$ which is small relative to the magnitude of the probability density functions $O\left(10^{-2}\right)$.

Sparse Grid Cubature

This is an active research are for high-dimensional integration. It uses sparse grids across the space of the independent variables instead of a full tensor product to reduce the number of cubature nodes required. The formulas for nodes and weights are different but the concept is the same. The paper by Xiu and Hesthaven [89] has several references.

3.6.8 Verification with Projection Methods

Verification of Coefficients

Galerkin Method solves for the coefficients analytically so there should be no error in the coefficients and no verification is needed. The errors in the coefficients computed numerically are dependent on the accuracy of the quadrature. Quadrature rules typically provide a theoretical estimate of the errors but they can be difficult to implement in practice. As a practical solution, the quadrature order difference can be defined similarly to the expansion order difference in Equation 3.38, and the quadrature order can be increased until the quadrature order difference converges to zero.

Verification of Expansion Convergence

Because each coefficient is computed independently, when additional Basis Functionals are added the coefficients of the lower order Basis Functionals should not need to be computed. Also, model evaluations used in computing the existing coefficients can be reused.

3.7 Collocation Methods

The Collocation Method is often used to find solutions of integrals and differential equations. There are several references that discuss this deterministic problem [75]. The projection methods attempt to minimize the residual over the space $\hat{\Omega_Y}$, by using the Basis Functionals as weights. Collocation methods minimize the residual at specific points within the space $\hat{\Omega_Y}$. By doing so, the resulting output Polynomial Chaos Expansion is like an interpolating surface, anchored at those collocation points. This does not control the residual over the space $\hat{\Omega_Y}$, but it does minimize the error at the collocation points.

3.7.1 Derivation from the Method of Weighted Residuals

Similar to the derivation in Section 3.6.1, the idea is to set the expected value of the weighted residual to zero. Instead of using the Basis Functionals as the residual weights a delta function is used.

Choose N_c collocation points $\vec{\xi}^{(c)}$ for $c = 1 \ldots N_c$. Each collocation point is an N-vector $\vec{\xi}^{(c)} = \left[\xi_n^{(c)}\right]_{n=1 \ldots N}$ chosen from $\Omega_{\vec{\xi}}$

$$\begin{aligned}
&\text{for } c = 1 \ldots N_c \\
&E_{\vec{\xi}}\left[R\left(\mathbf{y}, \vec{\xi}\right) r_c\left(\vec{\xi}\right)\right] = 0 \\
&E_{\vec{\xi}}\left[R\left(\mathbf{y}, \vec{\xi}\right) \delta\left(\vec{\xi} = \vec{\xi}^{(c)}\right)\right] = 0 \\
&\int_{\Omega_{\vec{\xi}}} R\left(\mathbf{y}, \vec{\xi}\right) \delta\left(\vec{\xi} = \vec{\xi}^{(c)}\right) p_{\vec{\xi}} d\vec{\xi} = 0 \\
&R\left(\mathbf{y}, \vec{\xi}^{(c)}\right) = 0 \\
&\sum_{p=0}^{P} \sum_{\vec{k}:|\vec{k}|=p} y_{\vec{k}} \Psi_{\vec{k}}\left(\vec{\xi}^{(c)}\right) = \mathcal{M}\left[\left\{\sum_{p=0}^{P_n} x_{np} \psi_p\left(\xi_n^{(c)}\right)\right\}_{n=1 \ldots N}\right] \\
&\sum_{p=0}^{P} \sum_{\vec{k}:|\vec{k}|=p} y_{\vec{k}} B_{\vec{k}c} = m_c
\end{aligned} \tag{3.43}$$

Rewrite Equation 3.43 using the same notation as Equation 3.42

$$\mathbf{B}^T \mathbf{y} = \mathbf{m}$$

and solve

$$\mathbf{y} = \left(\mathbf{B}\mathbf{B}^T\right)^{-1} \mathbf{B}\mathbf{m} = \mathbf{B}^{T+}\mathbf{m} \tag{3.44}$$

3.7.2 Verification

Verifying the Computed Coefficients

These methods calculate the coefficients of the output Polynomial Chaos Expansion such that the residual is minimized at certain collocation points. The calculation is straightforward, using only a single matrix inversion, and the only verification required is a check of whether the matrix is ill-conditioned enough to cause numerical errors. If it is, then either alternate collocation points can be used or a singular value decomposition can be used to determine the significant coefficients and the others can be set to zero.

Verifying the Expansion Convergence

While the Interpolation Methods minimizes the residual at the collocation points, the effect on the residual away from the collocation points is unknown. Therefore it is important when testing for convergence that the collocation points change with increasing order. Otherwise, successive output Polynomial Chaos Expansions may falsely appear to be converged. The lack of an 'optimal' selection in coefficients makes the verification of Interpolation Methods more subjective and more difficult.

3.7.3 The Probabilistic Collocation Method

The Probabilistic Collocation Method was introduced by Tatang [76] as the Deterministically Equivalent Modeling Method. It has also been termed the Stochastic Response Surface Method [42, 3]. It falls under the broader category of Least Squares Estimation or Matrix Inversion methods. Unfortunately, the name has also been used to refer to unrelated methods [27, 50]. This method requires the minimum number of collocation points, making it the cheapest of all Polynomial Chaos Expansion methods for uncertainty quantification. It has been applied to a wide array of fields [67, 77, 49].

Compared to Projection Methods, the Probabilistic Collocation Method uses fewer collocation points to minimize the residual only at the collocation points. The original Probabilistic Collocation Method uses the minimum number of points required, equal to the number of unknown coefficients in the output Polynomial Chaos Expansion. In one-dimension, the Probabilistic Collocation Method is equivalent to the orthogonal projection, but in higher dimensions they produce different results. Instead of choosing collocation points and weights in order to compute an integral, these methods select collocation points and minimizes the residual at those points.

The difference between all the algorithms is the heuristic for choosing collocation points. Some are based on joint probability, others are based on design of experiments principles, others are simply random. These methods work well for output Random Variables that are smooth and similar to low-order polynomials, however, there are difficulties in accurately computing the coefficients if a higher order Polynomial Chaos Expansion is needed.

Algorithm of the Probabilistic Collocation Method

1. Let the order of the output Polynomial Chaos Expansion be P.
2. For each Basis Random Variable, indexed by n:
   - Find the roots of the $P+1$th order Basis Functional and denote these by ${\xi_n}^{(i)}$ for $i = 1 \dots P+1$.
   - For each root i, calculate the probability density $f_{\Xi_n}\left(\xi_n^{(i)}\right)$.
3. Generate the N-dimensional collocation points by taking every combination of the roots. There will be $N_c = (P+1)^N$.
4. Calculate the joint probability density of each collocation point as the product of each root's probability density. Again the number is $N_c = (P+1)^N$.
5. Rank the N_c collocation points according to highest joint probability density. If there are ties with joint probability density, map the points in the space of the cumulative distribution functions and choose the points closest to the center point. If ties remain, keep the 'center of mass' as close as possible to the center point. If ties, just pick one. Choose the K top ranked collocation points, but eliminate collocation points that would make the matrix **B** singular.
6. Evaluate the model and each Basis Functional at the K collocation points.
7. Proceed with the Collocation Method as described in Section 3.7.

Example 7h - Probabilistic Collocation Method

In Section 3.5.2, we finished the formulation of this uncertainty quantification problem using Polynomial Chaos Expansions. Equation 3.30 shows that result. Now we solve the problem using the Probabilistic Collocation Method. The residual from Equation 3.36 here is:

$$R\left(\mathbf{d},\vec{\xi}\right) = \frac{2}{g}(s_0 + s_1\Xi_1(\xi_1))^2 \sin(t_0 + t_1\Xi_2(\xi_2))\cos(t_0 + t_1\Xi_2(\xi_2)) - \hat{D}\left(\mathbf{d},\vec{\xi}\right)$$

There are only 10 coefficients in the output Polynomial Chaos Expansion, so the Probabilistic Collocation Method requires only 10 collocation points. The output Polynomial Chaos Expansion uses maximum order of 3, so we take roots of 4th-order 1D parameter Basis Functionals as the collocation points. The process is the same as in the 4-node Tensor Product method, however, only 10 collocation points are selected, compared with 16 quadrature nodes.

The weights are not required and are not meaningful because this method does not solve the Galerkin Projection equations. Instead, the coefficients of the Polynomial Chaos Expansions can be calculated setting the residual to zero at the collocation points.

$$\frac{2}{g}\left(s_0 + s_1\Xi_1\left(\vec{\xi}^{(c)}\right)\right)^2 \sin\left(t_0 + t_1\Xi_2\left(\vec{\xi}^{(c)}\right)\right)\cos\left(t_0 + t_1\Xi_2\left({\xi_2}^{(c)}\right)\right) - \hat{D}\left(\mathbf{d},\vec{\xi}^{(c)}\right) = 0$$

This can be simplified using matrices to

$$\hat{D}\left(\mathbf{d},\vec{\xi}^{(c)}\right) = \frac{2}{g}\left(s_0 + s_1\Xi_1\left(\vec{\xi}^{(c)}\right)\right)^2 \sin\left(t_0 + t_1\Xi_2\left(\vec{\xi}^{(c)}\right)\right)\cos\left(t_0 + t_1\Xi_2\left({\xi_2}^{(c)}\right)\right) \tag{3.45}$$

$$\hat{D}\left(\mathbf{d},\vec{\xi}^{(c)}\right) = m^{(c)} \tag{3.46}$$

$$\sum_{\vec{k}=[0,0]}^{[2,1]} d_{\vec{k}}\Psi_{\vec{k}}\left(\vec{\xi}^{(c)}\right) = m^{(c)} \tag{3.47}$$

$$\begin{bmatrix} \Psi_{[0,0]}\left(\vec{\xi}^{(1)}\right) & \Psi_{[1,0]}\left(\vec{\xi}^{(1)}\right) & \cdots & \Psi_{[2,1]}\left(\vec{\xi}^{(1)}\right) \\ \Psi_{[0,0]}\left(\vec{\xi}^{(2)}\right) & \ddots & & \vdots \\ \vdots & & \ddots & \Psi_{[2,1]}\left(\vec{\xi}^{(9)}\right) \\ \Psi_{[0,0]}\left(\vec{\xi}^{(10)}\right) & \cdots & \Psi_{[0,3]}\left(\vec{\xi}^{(10)}\right) & \Psi_{[2,1]}\left(\vec{\xi}^{(10)}\right) \end{bmatrix} \begin{bmatrix} d_{[0,0]} \\ d_{[1,0]} \\ \vdots \\ d_{[0,3]} \\ d_{[2,1]} \end{bmatrix} = \begin{bmatrix} m^{(1)} \\ m^{(2)} \\ \vdots \\ m^{(9)} \\ m^{(10)} \end{bmatrix} \tag{3.48}$$

$$\mathbf{Bd} = \mathbf{m}$$

$$\mathbf{d} = \left(\mathbf{B}^T\right)^{-1}\mathbf{m} \tag{3.44}$$

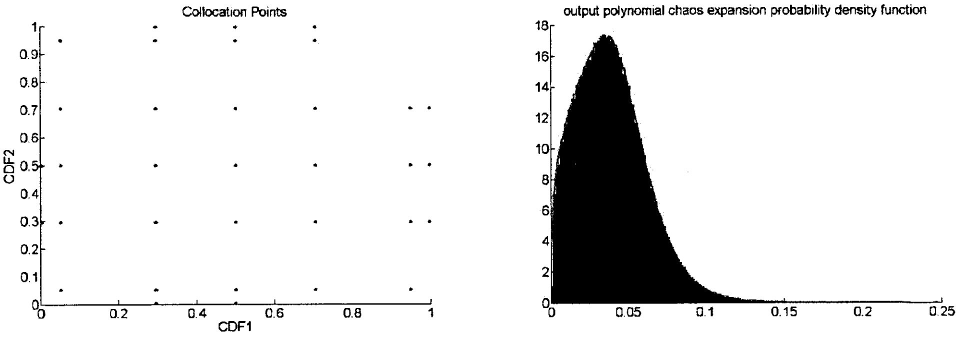
The coefficients are: $\mathbf{d} = \begin{bmatrix} 96.46 \\ 6.018 \\ 8.559 \\ 0.5533 \\ 0.09867 \\ -14.34 \\ -0.9083 \\ 1.260e-14 \\ -0.2503 \\ 0.008646 \end{bmatrix}$.

Increasing the order to $P = 4$ results in coefficients: $\mathbf{d} = \begin{bmatrix} 96.45 \\ 6.017 \\ 8.558 \\ 0.5480 \\ 0.09402 \\ -14.36 \\ -0.9150 \\ -2.696e-15 \\ -0.2489 \\ 0.008538 \\ -1.833e-15 \\ 0.1782 \\ 0.002060 \\ -0.01430 \\ -1.801e-16 \end{bmatrix}$.

The L^2 difference between 3rd and 4th order Polynomial Chaos Expansions is $O(-6)$, indicating that the expansion is converged. Even though this is considered converged, there are differences between corresponding coefficients in Polynomial Chaos Expansions of different order.

Example 8d - Solved with the Probabilistic Collocation Method

The formulation of this example was shown on page 94. Here it is solved using the Probabilistic Collocation Method. The proper collocation points are shown in Figure 3-20a for a 6^{th} order expansion. Then the output Polynomial Chaos Expansion and the Monte Carlo solution at $t = 1$ are shown in Figure 3-20b. The solution was verified at order 7 as well.



(a) Collocation Points

(b) Uncertain Model Output

Figure 3-20: Example 8 – Collocation Points for the Probabilistic Collocation Method and solution at time 1 s

To generate a time profile, this uncertainty quantification must be repeated for times ranging from 0 ~ 3s. The result is plotted in Figure 3-21. Figure 3-20b is a vertical cross-section of Figure 3-21. Note that the spread at initial time and long times is low. This is because it is known that the initial concentration of B is 0 and that eventually all B is converted to C.

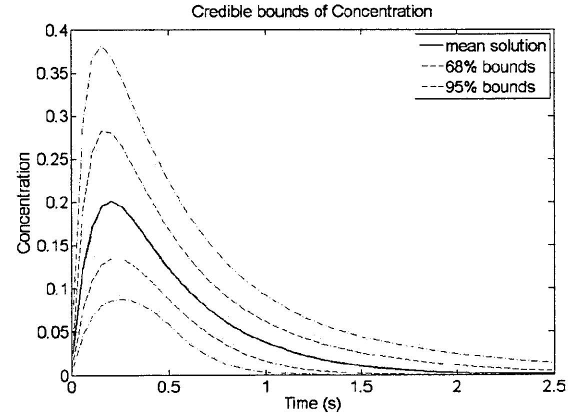


Figure 3-21: Example 8 – Uncertainty profile versus time computed for each time using a 6^{th} order expansion, showing the median value and two credible intervals

3.7.4 Least Squares Methods

The Probabilistic Collocation Method falls within the class of Least Squares Methods, which use different heuristics for selecting collocation points [21, 23, 22]. Here the number of collocation points is greater than the number of Basis Functionals, $N_c > K$. Therefore the $\mathbf{B}$ matrix is not square and the coefficients are calculated using least squares. Hosder et al. [39] reported that increasing the number of collocation points is necessary to make the Polynomial Chaos Expansion 'more stable'. In fact the problems they found in their work were a result of poor choice in collocation points. The Probabilistic Collocation Method was able to solve the problem with a minimum number of samples and a lower order expansion. The proper selection of collocation points is far more important than the number of collocation points. More analysis can be found in the work of Eldred et al. [23].

3.8 Connection between the Projection and Collocation Methods

- Projection method
  $\mathbf{y} = \mathbf{A}^{(proj)}\mathbf{m} = \mathbf{PBWm}$, see Equation 3.42
- Weighted Residuals
  $\mathbf{y} = \mathbf{A}^{(collo)}\mathbf{m} = \left(\mathbf{BB}^T\right)^{-1}\mathbf{Bm}$
- To be equivalent (with the same number of points, $N_Q = N_c$):

$$\begin{aligned}
\mathbf{A}^{(proj)} &= \mathbf{A}^{(collo)} \\
\mathbf{PBW} &= \left(\mathbf{BB}^T\right)^{-1}\mathbf{B} \\
\mathbf{BW} &= \mathbf{P}^{-1}\left(\mathbf{BB}^T\right)^{-1}\mathbf{B} \\
\mathbf{W} &= \left(\mathbf{B}^T\mathbf{B}\right)^{-1}\mathbf{B}^T\mathbf{P}^{-1}\left(\mathbf{BB}^T\right)^{-1}\mathbf{B} \\
\mathbf{W} &= \mathbf{B}^{+}\mathbf{P}^{-1}\mathbf{B}^{T+}
\end{aligned} \tag{3.49}$$

- If $\mathbf{B}$ is square

$$\mathbf{W} = \mathbf{B}^{-1}\mathbf{P}^{-1}\left(\mathbf{B}^{-1}\right)^T$$

The inner products of the Basis Functionals are the eigenvalues of the weighting matrix Even if extra collocation points are added to a collocation method, there are only K non-zero columns and rows in the weighting matrix. This makes sense because there are only K Basis Functionals to project onto. It is unclear how the accuracy of the integrals changes with the number of collocation points, because the MWR does not attempt to solve that problem.

- $\mathbf{P}$ is specified by the problem formulation, $\mathbf{B}$ depends on the collocation points, and $\mathbf{W}$ are the weights. In a projection collocation scheme, the weights remain the same for all Basis Functionals, however, the weighted residuals approach has different weights for each PCE coefficient.
- In general, the two approaches result in different weights and number of collocation points, with PCM using far fewer points.
- In the 1dimensional case, these approaches are equivalent.
- Typically the quadrature points and weights and collocation points are chosen by a heuristic, so each heuristic must be examined to see if the methods are equivalent.

Example 16: Uncertainty Quantification Methods Comparison in One-Dimension

In this example we compare two methods for solving Polynomial Chaos Expansion coefficients: Tensor Products of 1D Gaussian Quadrature rules and the Probabilistic Collocation Method. In one dimension, with a Pth order output Polynomial Chaos Expansion, the Probabilistic Collocation Method takes $P+1$ collocation points, so we will compare it to the $P+1$-node Gaussian quadrature.

Starting with Equation 3.49, if the two approaches are equivalent, then:

$$\begin{aligned}
\left(\mathbf{B}^{-1}\right)^T &= \mathbf{PBW} \qquad (3.49)\\
\mathbf{P}^{-1} &= \mathbf{BWB}^T\\
\left\langle \psi_k\left(\xi\right)\psi_k\left(\xi\right)\right\rangle &= \sum_{c=1}^{N_c} \psi_k\left(\xi^{(c)}\right) w_c \psi_k\left(\xi^{(c)}\right)\\
\int\limits_{\Omega_\xi} \psi_k(\xi)^2 f_\xi d\xi &= \sum_{c=1}^{N_c} \psi_k\left(\xi^{(c)}\right)^2 w_c
\end{aligned}$$

This integral is exact for polynomials of order $2\left(P+1\right)+1$. The polynomial order of $\psi_P(\xi)^2$ is

$2P$. Therefore, the 1D case of the projection method with Gaussian Quadrature is the same as the Probabilistic Collocation Method.

In the general case, the inner products will be products of one-dimensional integrals. The Tensor Products of Gaussian quadrature collocation points are not the same as those of the Probabilistic Collocation Method, and Equation 3.49 is not exact. The Probabilistic Collocation Method requires far fewer collocation points, but may take more work to verify.

3.9 Summary

Uncertainty quantification is critical in the decision theory framework that is used for design of experiments. The framework described in this chapters fits within decision theory and enables any uncertainty quantification method to be applied. The particular method used to solve the uncertainty quantification problems are not important. Each method has its benefits and drawbacks. Section 3.2.5 stated the reasons that Polynomial Chaos Expansions in particular are attractive. They can represent complex uncertainties but can still be computed efficiently. This will enable the application of Bayesian Design of Experiments to larger systems of interest to chemical engineers.

Chapter 4

Classical Design of Experiments

This chapter introduces the most common approach to Design of Experiments. It covers the development of the theory and the procedure. The focus is only on the selection of experiments. For the more general context of design of experiments within a larger study and principles of good experiments, see Appendix C and *Design and Analysis of Experiments* by Montgomery [58].

4.1 The Current State of Experimental Design

The current industry practice is to use the Classical Design of Experiments approach, specifically the response surface methodology, when designing experiments for engineering applications. The Classical approach is attractive because of its simplicity and history of application. These designs use predefined sets of experiments, instead of creating customized designs for the process at hand. These designs are configured to provide good data for polynomial regression models. In effect, the experimenter is using a design that was created for a different 'idealized' system, which does not match the system of interest and will often produce suboptimal results. If the experimenters want to modify the design, they must rely on their intuition and experience. At the same time, the Chemical Engineering industry has built a large repository of more representative models, either through knowledge of chemistry and physics or through empirical fits. These are examples of prior knowledge which could be used to design more informative experiments, however, these cannot be processed within the framework of the Classical approach.

4.2 History of Design of Experiments

Experimental Design theory was first formally studied in the 1940's with the development of Classical Design of Experiments. They were created for agricultural experiments to understand which factors had significant impacts on crop growth. These designs are ideal for identifying large scale trends especially when there are large, unexplained variabilities. Growing crops was a long term experiment and most factors could not be easily controlled. Therefore the best approach was to concentrate on identifying factors with large impacts and ignoring the less significant factors. In the 1950's, the concept of experimental designs for response surface maps was used for engineering applications. These were more applicable to engineering – they utilize more experiments and allow identification of trends and empirical modeling.

Classical Design of Experiments is a broad class of methods, including the full factorial designs used for crops, the response surface methods, and Taguchi methods. These are loosely categorized by their use of predefined designs and empirical models developed after collecting data. The second class consists of model based design of experiments methods. These are distinguished by their use of models to influence the experimental design. Chapter 5 is devote to these methods.

4.3 Classical Design Heuristics

Montgomery describes Classical Designs as heuristics to plan experiments. There is no unifying strategy or 'approach'. Each heuristic is meant to investigate a different feature of the system, such as identifying significant factors and interactions or developing an empirical model. The common goal of all the heuristics is to gather data which can be analyzed with classical statistical techniques. For the most part, this means the ANalysis Of VAriance technique or ANOVA. For details see Appendix A.3. The procedure is to choose a heuristic, collect data, and then use ANOVA to fit a linear model. The only complication is the choice of heuristics. This is typically done by experience and guesswork. There is no explicit reliance on a model or prior knowledge.

4.3.1 Full Factorial Designs

The most well known Classical Design heuristic is the Full Factorial design. Full Factorial means that every factor is varied at two values and an experiment is run at every possible combination of factor values. If there are F factors, this would use 2^F experiments. This design is used to test for first order dependencies and interactions between independent variables (also known as factors). In practice, the design works well at its goal. In fact the Full Factorial Design is the optimal design for parameter estimation on first-order, linear system models with Gaussian observation errors. That is to say, the result is the same as the D-Optimal design which will be explained in the next chapter.

The drawback of the Full Factorial design is that it uses a large number of experiments when many factors influence the system. Other heuristics have been developed to reduce the number of experiments while still allowing for good statistical results. These are not discussed here but Montgomery [58] is a good reference.

4.3.2 Central Composite Designs

Another common heuristic is the Central Composite Design. Here the goal is not just to identify large trends but to attempt to quantify the trends and predict system performance. For these studies, more detail about the performance is required, so additional experiments are used to fit a second-order, linear model. This is linear in the parameters, but quadratic with respect to the factors.

The full factorial and central composite designs exemplify the capabilities of classical design of experiments. An example of both designs is shown in Figure 9-9. The most important features are that the heuristics attempt to spread experiments across the design space and have a regular pattern. The designs are robust – they can be applied to any quantitative factor in any system without requiring much understanding of the system. They can also identify which factors cause significant changes in the system response and whether significant interactions are present. Unfortunately, the best case result from applying a heuristic design is a linear, empirical model. In many modern engineering studies the goal is to build a model based on first principles and for this purpose Classical Designs are less effective and less efficient.

4.3.3 Merits of Classical Designs

The notable advantages of Classical Designs are that they do not require much knowledge of the system and they are quite effective at identifying major trends. Unfortunately, in modern engineering experiments we seek much more information than major trends and therefore the Classical approach is lacking. They require large numbers of experiments and do not have the flexibility to deal with irregular design spaces that arise due to physical limits on the experimental equipment. Finally, they rely on classical statistics which cannot properly deal with uncertainty.

These drawbacks motivate the use of model based design of experiments, which includes the Bayesian approach. These are introduced in the next chapter.

Chapter 5

Model Based Design of Experiments

The design of experiments approaches in this chapter are distinguished from the Classical Designs because they depend explicitly on models. Models that are built from either first principles or empirical data are now everyday tools in modern engineering design. However, modeling the system is only one part of effective design work. Just as important is the ability to understand and model uncertainties. This is the feature that distinguishes the approaches with in the class of Model Based Design of Experiments.

This chapter describes the Optimal Design and Bayesian Design approaches using the decision theory framework. This emphasizes their similarities and differences, especially their treatment of uncertainty. First the decision theory framework is explained in the context of design of experiments, then some examples are shown in Section 5.2 to give an intuition of parametric uncertainty impacts model outputs and how understanding this helps to design experiments. Sections 5.3 through 5.4 detail the Optimal and Bayesian Design of Experiments approaches.

5.1 Decision Theory for Design of Experiments

Statistical decision theory was introduced in Section 2.4. This section shows the application to design of experiments by matching the experimental study process with components of decision theory.

5.1.1 Applying Decision Theory

Recall the necessary components to utilize the decision theory framework.

- Set of potential actions
- Uncertain parameters
- Known set of consequences
- Prior knowledge
- Observations
- Models of the system and observations of the system

The process of experimental design fits nicely into this framework. The actions are the choices of experimental designs. The uncertain parameters are the same as the model parameters, which have some prescribed prior information, and the consequences are the simulated datasets. The only difference is that there are typically no observations available before the design of experiments.

The three steps in applying decision theory are: enumerating the potential actions, correctly describing the uncertainties, and quantifying the utility of each consequence. In the application to design of experiments, the potential actions are represented by the design space, which is typically constrained for practical or physical reasons. These constraints must be assessed from knowledge of the system. The uncertainties that must be described are the uncertainties in the experimental system and our uncertain observations of the system. These correspond to uncertain parameters in the system model and the observation model. Finally, the consequences are the observations of data. The data obtained in a an experiment will have some utility, which can be computed by using the data for parameter estimation.

The methods for describing and quantifying uncertainties have already been shown. What remains are the theory and methods for describing utility.

5.1.2 Utility Functions

The most critical aspect of decision making is the metric used to determine which action is best. In decision theory, these are the utility function and risk metric. The role of the utility function is to objectively assess the usefulness of each consequence. The risk metric examines the probability and utility of all the possible consequences in order to make a decision. In this work, we use only the Expected Value as a risk metric, however, many different utility functions are discussed here.

All experiments are meant to gather information with some higher purpose in mind. Utility is a measure of how well the experiment did in achieve that purpose. An intuitive way to measure this is the compare the knowledge before and after the experiment. Since the focus is on the experiment, the collection of data will be considered the most important event. All steps before this are considered *prior*, and all steps after are consider post-event or *posterior* to the event.

The utility functions listed below are all functions of the prior and posterior parameter densities. They are based on the Shannon Information metrics and are suited for the Bayesian approach to Design of Experiments. The Optimal Designs approach has analogous concepts for each utility function but different implementation. Books by Pukelsheim [68], Atkinson [2], and Fedorov and Hackl [25] have much more detail on Optimal Designs.

D-Optimal Bayesian Designs: Improving Model Parameters

One of the most common goals of experimental studies is to improve estimates of the model parameters. This is incorporated into the design of experiments by using a utility function based on gain in parameter information. Continuing the discussion from Section 2.3.4, the utility function can either be the decrease in differential entropy from the prior to posterior parameter densities in Equation 2.3 or the Kullback-Leibler Divergence in Equation 2.2.

$$\begin{aligned}-\Delta h &= -\left(h\left[f_{\Theta|D}\left(\theta|\mathsf{d}\right)\right] - h\left[f_{\Theta}\left(\theta\right)\right]\right)\\ &= \int\limits_{\Omega_\Theta} f_{\Theta|D}\left(\theta|\mathsf{d}\right)\log f_{\Theta|D}\left(\theta|\mathsf{d}\right)d\theta - \int\limits_{\Omega_\Theta} f_{\Theta}\left(\theta\right)\log f_{\Theta}\left(\theta\right)d\theta\end{aligned}$$

The second term in Equation 2.3 depends only on the prior parameter density, and is therefore constant no matter which experiment is run. Therefore, this term has no impact on the maximum of $-\Delta h$ and can be neglected. Maximizing the decreasing in entropy from prior to posterior parameter densities is the same as minimizing the posterior entropy.

$$D_{KL}\left(f_{\Theta|D}\left(\theta|\mathsf{d}\right) \parallel f_{\Theta}\left(\theta\right)\right) = \int\limits_{\Omega_\Theta} f_{\Theta|D}\left(\theta|\mathsf{d}\right) \log \frac{f_{\Theta|D}\left(\theta|\mathsf{d}\right)}{f_{\Theta}\left(\theta\right)} d\theta$$

$$D_{KL}\left(f_{\Theta|D}\left(\theta|\mathsf{d}\right) \parallel f_{\Theta}\left(\theta\right)\right) = \int\limits_{\Omega_\Theta} f_{\Theta|D}\left(\theta|\mathsf{d}\right) \log f_{\Theta|D}\left(\theta|\mathsf{d}\right) d\theta - \int\limits_{\Omega_\Theta} f_{\Theta|D}\left(\theta|\mathsf{d}\right) \log f_{\Theta}\left(\theta\right) d\theta$$

The two utility functions in Equations 2.3 and 2.2 have slightly different meanings but they are closely related. When using the Expected Value as a risk metric, both utility functions give the same optimal design. For alternative risk metrics, these may not be equivalent and the effects of either utility function are unknown.

This is called a D-optimal design and generally improves the parameters with the highest prior differential entropy. This is an example of picking the lowest hanging fruit. Parameters with low prior differential entropy are already well known and it is difficult to learn more about them. Parameters with high prior differential entropy can be easily improved.

Partial D-Optimal Bayesian Designs: Improving a Subset of Parameters

The partial D-optimal utility function is very similar to the D-optimal utility function except that instead of improving all parameters, the partial version concentrates on a subset of parameters. Although the D-optimal designs are the most common, the partial D-optimal is more appropriate in most situations. Often we do not need to know the majority of the parameters and only wish to know a few key parameter values. In this case, a D-optimal design can be wasteful because it may select experiments that will greatly improve parameters we do not care about.

The theory is very simple - we split the parameters Θ into two subsets: those we wish to estimate Λ and the rest Γ. Then we only include Λ in the utility function. This can only be computed by integrating over Γ; because we are not interested in the value of Γ we will average over all its values.

$$
\begin{aligned}
-\Delta h &= -\left(h\left[f_{\Lambda|D}\left(\lambda|\mathbf{d}\right)\right] - h\left[f_{\Lambda}\left(\lambda\right)\right]\right) \\
-\Delta h &= -\left(h\left[\int\limits_{\Omega_\Gamma} f_{\Gamma,\Lambda|D}\left(\gamma,\lambda|\mathbf{d}\right)d\gamma\right] - h\left[\int\limits_{\Omega_\Gamma} f_{\Gamma,\Lambda}\left(\gamma,\lambda\right)d\gamma\right]\right) \\
&= \int\limits_{\Omega_\Lambda}\left[\int\limits_{\Omega_\Gamma} f_{\Gamma,\Lambda|D}\left(\gamma,\lambda|\mathbf{d}\right)d\gamma\right]\log\left[\int\limits_{\Omega_\Gamma} f_{\Gamma,\Lambda|D}\left(\gamma,\lambda|\mathbf{d}\right)d\gamma\right]d\lambda \\
&\quad - \int\limits_{\Omega_\Lambda}\left[\int\limits_{\Omega_\Gamma} f_{\Gamma,\Lambda}\left(\gamma,\lambda\right)d\gamma\right]\log\left[\int\limits_{\Omega_\Gamma} f_{\Gamma,\Lambda}\left(\gamma,\lambda\right)d\gamma\right]d\lambda
\end{aligned}
$$

and

$$
\begin{aligned}
D_{KL}\left(f_{\Lambda|D}\left(\lambda|\mathbf{d}\right) \parallel f_{\Lambda}\left(\lambda\right)\right) &= \int\limits_{\Omega_\Lambda} f_{\Lambda|D}\left(\lambda|\mathbf{d}\right)\log\frac{f_{\Lambda|D}\left(\lambda|\mathbf{d}\right)}{f_{\Lambda}\left(\lambda\right)}d\lambda \\
&= \int\limits_{\Omega_\Lambda}\left[\int\limits_{\Omega_\Gamma} f_{\Gamma,\Lambda|D}\left(\gamma,\lambda|\mathbf{d}\right)d\gamma\right]\log\frac{\int\limits_{\Omega_\Gamma} f_{\Gamma,\Lambda|D}\left(\gamma,\lambda|\mathbf{d}\right)d\gamma}{\int\limits_{\Omega_\Gamma} f_{\Gamma,\Lambda}\left(\gamma,\lambda\right)d\gamma}d\lambda \\
&= \int\limits_{\Omega_\Lambda}\left[\int\limits_{\Omega_\Gamma} f_{\Gamma,\Lambda|D}\left(\gamma,\lambda|\mathbf{d}\right)d\gamma\right]\log\left[\int\limits_{\Omega_\Gamma} f_{\Gamma,\Lambda|D}\left(\gamma,\lambda|\mathbf{d}\right)d\gamma\right]d\lambda \\
&\quad - \int\limits_{\Omega_\Lambda}\left[\int\limits_{\Omega_\Gamma} f_{\Gamma,\Lambda|D}\left(\gamma,\lambda|\mathbf{d}\right)d\gamma\right]\log\left[\int\limits_{\Omega_\Gamma} f_{\Gamma,\Lambda}\left(\gamma,\lambda\right)d\gamma\right]d\lambda
\end{aligned}
$$

Bayesian Designs for Discriminating Between Models

The last experimental studies goal discussed here is model discrimination. Often times, several models are proposed that can adequately explain the available data. The next step is to collect the data will allow some models to be eliminated. This idea has been proposed in many sources, with contributions from traditional optimal designs as well as Bayesian designs [10, 73, 66, 4, 78].

Most Bayesian methods are similar to the work of Bard [5]. They discriminate between models by computing the KullbackLeibler Divergence between all combinations of prior predictive densities. This is attractive because the prior predictive densities are easy to sample from. Unfortunately, there is no way to use the interactions between design points, because the Bayesian inference step is skipped. The common method to account for this is to design experiments sequentially. Leaving the computational aspects aside, this method is problematic because it is not realistic in the way models are discriminated. The algorithm is to simulate data and compare how well it matches the prior knowledge of each model.

A more natural approach for model discrimination experiments is to improve all the models conditioned on the data, and then compare how well the observed data matches the posterior knowledge. This follows the traditional Bayesian approach to experimental design and is more realistic to how experiments for model discrimination are carried out. The full derivation is in Section 6.4.2.

The concern with this method is whether the posterior knowledge will be overconfident, which occurs when a model does not fit well. For this reason, Gelman [33] advocates optimality conditions besides the KullbackLeibler divergence, however this was not investigated further.

The closest analogy for the Optimal Designs approach is the K-optimal design, but while this shares the goal of model discrimination the formulation is quite different.

Reducing Model Output Uncertainty

This utility function first computes the posterior parameter density and propagates it through the model to determine the posterior data-predictive density. Then the utility function is a metric on the posterior predictive densities, such as the Kullback-Leibler divergence from posterior to prior predictive densities. These are related to G-, I-, or V-optimal designs, however the Bayesian approach requires uncertainty quantification for the posterior parameters while the Optimal approach only requires model sensitivities. This is because the Optimal approach is only valid for linear models.

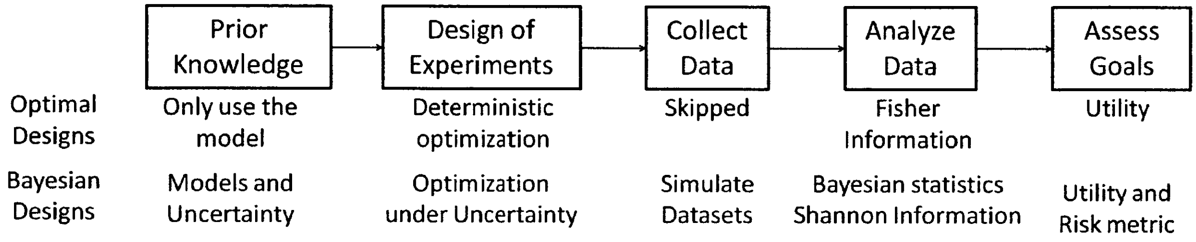
This utility function is illustrated using an example in Section 7.1.

Utility versus Loss

In general utility is a desirable thing, so the goal is to maximize utility. An analogous concept to the utility function is a loss function, which measures the negative aspects of a consequence. Here the term utility function is used but most optimization algorithms are based on loss functions (objective functions). For all the examples, the objective should be clear, whether we are minimizing loss or maximizing utility.

5.1.3 Motivation for the Decision Theory Framework

Framing the design of experiments approaches with decision theory serves to illustrate their similarities and highlight their differences. Figure 5-1 illustrates how Optimal Designs and Bayesian Designs fit into the decision theory framework.



	Prior Knowledge	Design of Experiments	Collect Data	Analyze Data	Assess Goals
Optimal Designs	Only use the model	Deterministic optimization	Skipped	Fisher Information	Utility
Bayesian Designs	Models and Uncertainty	Optimization under Uncertainty	Simulate Datasets	Bayesian statistics Shannon Information	Utility and Risk metric

Figure 5-1: Comparison of Model Based Approaches to Design of Experiments

Comparing the two approaches, we see that Bayesian Designs are much more general. This is because they use the full decision theory framework and do not make simplifying assumptions. The next two sections will further explain this figure, but the key message is that the Bayesian approach is more accurate because of its use of prior knowledge, probabilistic description of the consequences, and the ability to apply risk metrics.

5.2 Illustrative Examples of Model Uncertainty

Before showing the details of the Optimal and Bayesian approaches to design, some examples are shown to give an intuition of the effect of different models and prior information. First, two simple observation models are described. Then the several linear models and a nonlinear model are used to illustrate the impact of parametric and observation uncertainties on the model output and data predictive densities. The term 'linear model' refers to any model that is linear in the parameters so that the partial derivative with respect to the parameters does not depend on the parameters.

$$\frac{\partial}{\partial \theta_i} y \neq g(\theta)$$

for all parameters i.

5.2.1 Observation Models

The general form of an observation model is shown in Equation 5.1.

$$\varepsilon = \mathcal{M}^{\varepsilon}[\theta, x] \tag{5.1}$$

ε can be thought of as another uncertain parameter added to the the system model in order to create a data prediction model. Uncertainty quantification would then result in the data predictive density. Complicated observation models may change the structure of the model, but most observation models are much simpler. Examples include additive and proportional observation uncertainties.

$$\begin{aligned} D(\omega_\Theta, \omega_\mathrm{E}) =& Y(\omega_\Theta) + \varepsilon = \mathcal{M}[\theta; x] + \varepsilon \\ D(\omega_\Theta, \omega_\mathrm{E}) =& (1+\varepsilon)\mathcal{M}[\theta; x] \end{aligned}$$

where ε is the value of $\mathrm{E}(\omega_\mathrm{E})$, which represents the uncertain discrepancy caused by imperfect observation of the system. Throughout the remaining chapters, observation model is a generic term for Equation 5.1 which will specify the relation between $Y(\omega_\Theta)$ and $D(\omega_\Theta, \omega_\mathrm{E})$.

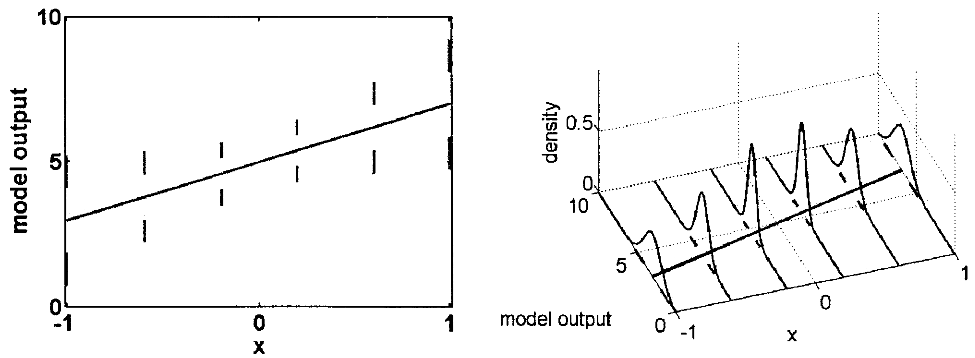
Example 17: First-order, Linear System with Gaussian Uncertainties

Start off with a first order linear model:

$$y = \theta_1 x + \theta_2 \text{ for } x = [-1, 1], \text{ with } \theta \sim \begin{cases} N(5,1) \\ N(2,1) \end{cases}$$

The source of uncertainty in the system is the parameters θ. This is represented by the prior parameter density $f_\Theta(\theta) = f_{\Theta_1,\Theta_2}(\theta_1, \theta_2)$. The model output uncertainty, $f_Y(\mathrm{y})$ is computed through uncertainty quantification and shown in Figure 5-2.

Next the observation model is included. This model is an additive Gaussian distribution, meaning that observations of each model output y_0 cannot be made perfectly and will be normally distributed around the model output. So data $d_0 = y_0 + N(0, \sigma^2)$. In this case, $\sigma = 0.5$. What



(a)

(b)

Figure 5-2: Example 17 – Linear system model with Gaussian parametric uncertainty, shown with credible intervals and probability density function

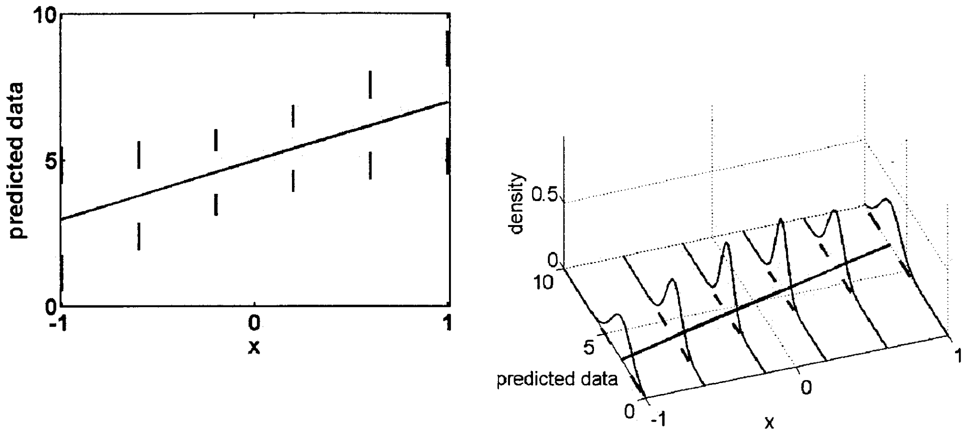
results is a distribution of simulated data, or the data that could potentially be observed for a single model output. Also, there is a distribution of model outputs for every design x_0. Because the Gaussian observation uncertainty is added to the model output uncertainty, the resulting data predictive density is a convolution of the two distributions, as shown in Figure 5-3.

The form of the model dictates the uncertainty profiles seen here. The standard deviation of the model output's density grows linearly with x and the model output uncertainty is minimized at $x = 0$. Figure 5-3 shows that the uncertainty in the data predictions are also largest at the bounds. This class of uncertainty profile is very standard for linear models and additive noise. Intuitively the best experiments are at the bounds of design variable $x = \{-1, 1\}$, but we would like a quantitative method to prove this. Clearly the best way to estimate the slope is by observing at the bounds but this does not directly gather information about the y-intercept. Conversely, observing data at the origin does not help estimate the slope.

It turns out the bounds are the best solution for estimating both parameters. The question that follows is whether the design is optimal because it corresponds with the highest uncertainty, or whether this is a coincidence. This is answered in Example 18.

Example 18: Offset, First-order, Linear System with Gaussian Uncertainties

Now say there is some context to the design variable x and the domain is not center around



(a) (b)

Figure 5-3: Example 17 – Data predictions with model from Figure 5-2 and an additive, Gaussian observation model

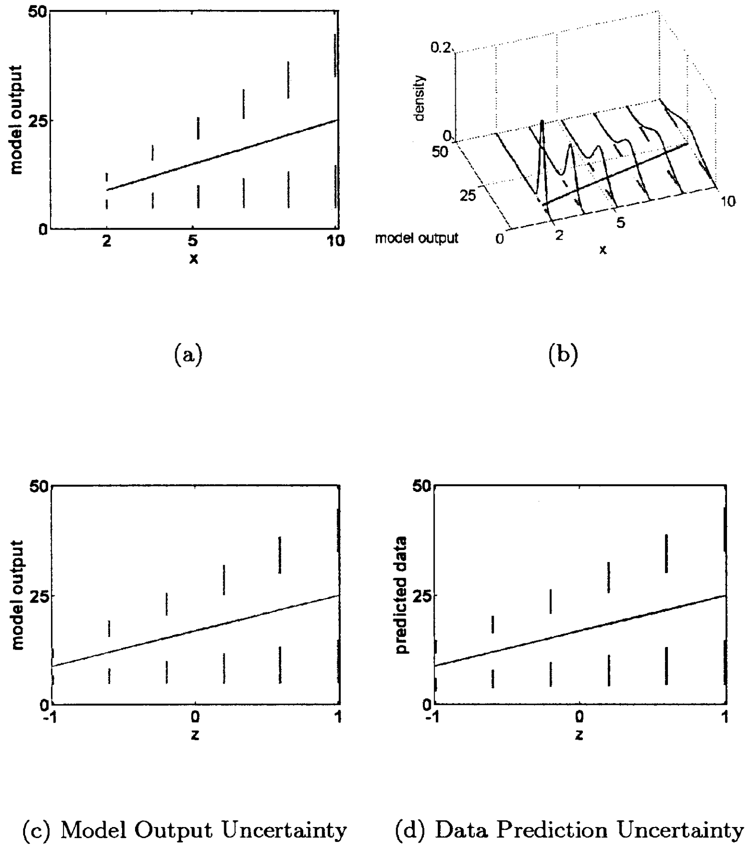
the origin. Scaling doesnt change the uncertainties. $y = \theta_1 x + \theta_2$. $x = [2, 10]$. Rescale to $z = \frac{x-7}{4}$. Then $y = 4\theta_1 z + 7\theta_1 + \theta_2$.

Figure 5-4c shows the same uncertainty profile, just shifted for the scaled design variable. Figure 5-4d shows the data predictions. Even though the uncertainty profiles of y vs. z are different than in the first example, the best design is still at the bounds, $z = -1, 1$ even though the best design is the same in this case, it is not where the data predictions uncertainties are greatest.

Example 19: Second-order, Linear System with Gaussian Uncertainties

Now we consider a second-order, linear system with Gaussian parametric uncertainties. This is shown in Figure 5-5.

As might be expected from seeing the first-order results, the second-order has a model output density that grows quadratically as x deviates from 0. The observation uncertainty is again additive Gaussian errors with $\sigma = 0.5$. The data predictions are shown in Figure 5-5b. Since there is a large difference in model output uncertainties across the domain, the observation uncertainty makes a noticeable difference in the uncertainty profile.



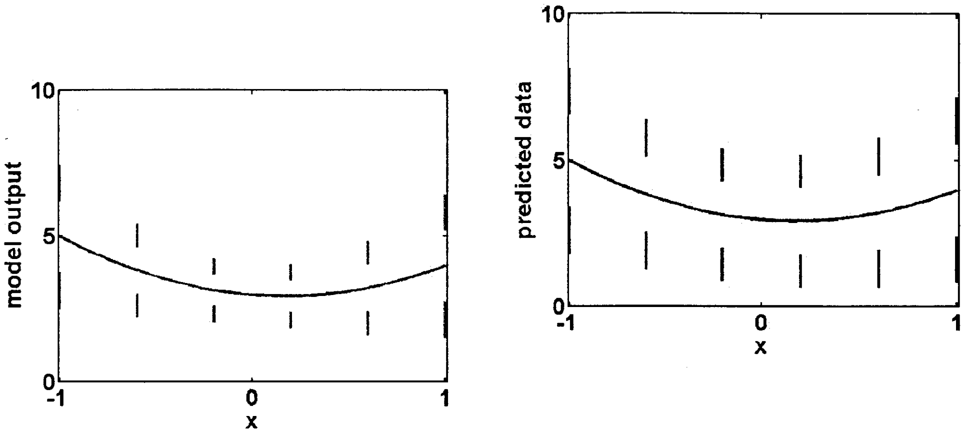
(a)

(b)

(c) Model Output Uncertainty

(d) Data Prediction Uncertainty

Figure 5-4: Example 18 – Offset linear model with Gaussian parametric uncertainty. Output uncertainty shown with credible intervals (a) and probability density function (b), then rescaled model output uncertainty (c) and data prediction uncertainty (d).



(a) Model output uncertainty

(b) Data prediction uncertainty

Figure 5-5: Example 19 – Second-order, Linear System with parametric uncertainty (a) and including observation uncertainty (b)

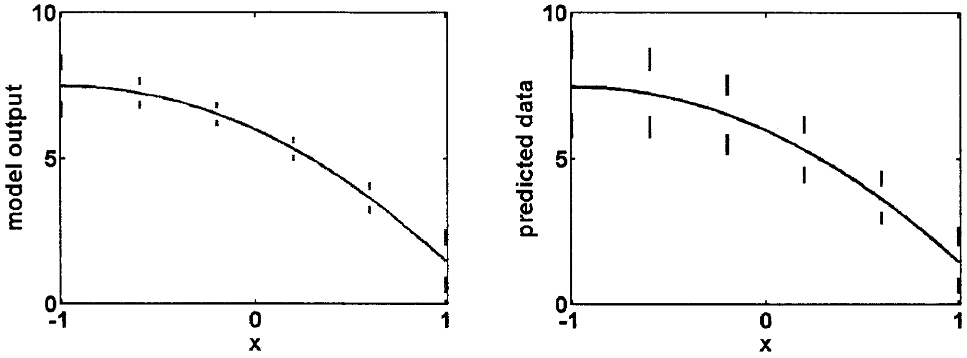
The best design is the bounds and the largest deviation from linear, which always occurs at the midpoint. At the midpoint of the domain however, the model output uncertainty and data prediction uncertainty can be smaller than at the bounds. Once again we see that the largest model output uncertainty is not a good measure of useful experiments.

These quick examples show that the naive approach of using the points with the worst data predictions are not the best. The correct approaches are shown in the following sections.

Example 20: Proportional Observation Uncertainty

In this example, we have a second order linear system, as in Example 19. In this case however, the observations are not additive but proportional. The larger the model output, the larger the observation uncertainty. This corresponds to $d_0 = (1+\epsilon)\, y_o$ where $\epsilon \sim N\left(0, \sigma^2\right)$ and $\sigma = 0.1$. This is interesting because the model output uncertainty increases with distance from the origin, however, the observation uncertainty is decreasing as x increases.

This observation model is representative of measurement tools that are uncertain because of calibration errors. Many instruments convert signals from the system to an electrical signal, which



(a) Model Output Uncertainty (b) Data Prediction Uncertainty

Figure 5-6: Example 20 – model output uncertainty (a) and data prediction uncertainty (b) indicating large differences in the observation uncertainty

must be calibrated to convert into physically meaningful observations. Uncertainty in calibration results in uncertainty that grows linearly with the electrical signal.

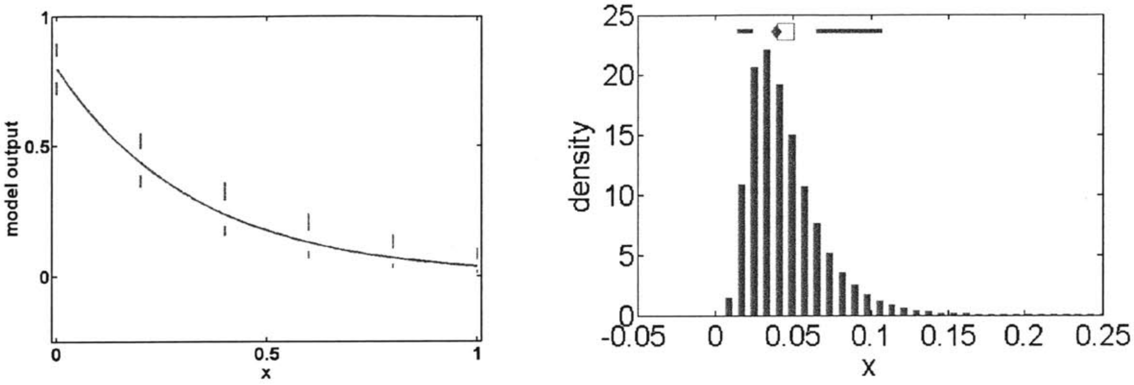
Example 21: Nonlinear System Model

Now we examine a model that is not linear in the parameters.

$$y = \alpha \exp(-\beta x)$$

This type of nonlinearity is very common in systems with mass action behavior. Assuming Gaussian parameter uncertainties and no observation uncertainty, the model output uncertainty is shown in Figure 5-7 along with a sample cross section.

If the system must be linearized, the uncertainty is very difficult to understand and characterize. First of all, the linearization occurs for a single value of parameters α^*, β^*. These parameters no longer have the same relationship within the model, so the prior knowledge of α, β may not propagate correctly through the model.



(a) Model Output Uncertainty

(b) Data Prediction Uncertainty

Figure 5-7: Example 21 – model output uncertainty (a) and data prediction uncertainty (b) for a Nonlinear system model with Gaussian parametric uncertainty and no observation error

A common choice for linearization parameter values is the mode (which coincides with the median and mean for Gaussian distributions).

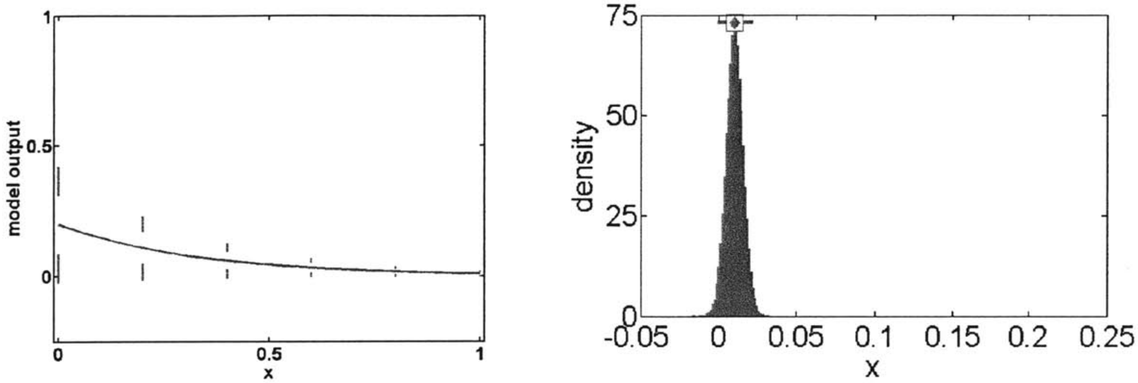
$$\begin{aligned} y^{lin} =& \alpha^* \exp\left(-\beta^* x\right) + \exp\left(-\beta^* x\right)\left(\alpha - \alpha^*\right) + \alpha^* \exp\left(-\beta^* x\right)\left(\beta - \beta^*\right) \\ y^{lin} =& \exp\left(-\beta^* x\right)\left[\alpha + \alpha^*\left(\beta - \beta^*\right)\right] \end{aligned}$$

The linearized version of the model is shown in Figure 5-8 using the same parameter uncertainties. Note that if this were treated as a Taylor Series with respect to the parameters and more terms were added, the approximation would become quite good. This Taylor Series would be equivalent to a Polynomial Chaos Expansion, as seen in Section 3.4.4. With only one term, however, the approximation is totally inadequate.

5.2.2 Conclusion

These examples illustrate how the form of the model and the parametric uncertainties shape the uncertainties in model outputs and data predictions. This is the knowledge that model based design of experiments methods use to select the best designs.

An interesting result is that all the design approaches, Classical, Optimal, and Bayesian, will give the same designs for first- and second-order, linear systems with Gaussian uncertainties [16, 2, 58]. In fact, Bayesian Designs simplify to Optimal Designs under the stated conditions. However, neither



(a) Model Output Uncertainty

(b) Data Prediction Uncertainty

Figure 5-8: Example 21 – model output uncertainty (a) and data prediction uncertainty (b) resulting from linearization of the model with respect to the parameters

Classical or Optimal Designs can handle more complex observation models or nonlinear models. Engineering model are becoming more and more sophisticated, which necessitates the use of the Bayesian approach to Design of Experiments.

5.3 Optimal Design of Experiments

Optimal Designs are the most commonly applied model based design of experiments approach. They remain an active research topic but their use in industry is not widespread. Adoption should grow in the future as very good results have been demonstrated on academic examples. The basics of the approach are described here in order to provide a contrast to the Bayesian approach.

5.3.1 Decision Theory Framework

Like all design of experiment approaches, Optimal Designs fit within the framework of statistical decision theory, as shown in Section 5.1. The key detail that distinguishes the approach are the use of classical statistical methods to compute the expected utility for each experimental design. The use of classical statistics impacts all the other steps. Because of these statistics methods, the prior information is not utilized, the information metric is simplified, and the optimization and parameter estimation aspects are straightforward. The main drawback is that certain assumptions about the system model and observational model must be satisfied.

5.3.2 Algorithm

Only an outline of the algorithm is shown here. We are more interested in the theory than the details, which have more to do with optimization. A good reference is the book by Atkinson and Donev [2].

Starting with a system model with some uncertain parameters θ:

$$Y\left(\omega_{\Theta}\right) = \mathcal{M}\left[\theta; x\right]$$

and an additive observation model:

$$D\left(\omega_{\Theta}, \omega_{\mathrm{E}}\right) = Y\left(\omega_{\Theta}\right) + \varepsilon = \mathcal{M}\left[\theta; x\right] + \varepsilon$$

where $\varepsilon \sim N\left(0, \sigma^2\right)$. The steps are:

1. Select a design x from the design space $\mathcal{X}$
2. Formulate the likelihood function as:

$$L\left(\Theta | Y\left(x\right)\right) = f_{Y|\Theta}\left(y | \theta, x\right) \tag{5.2}$$

3. For each design x
   (a) Compute the score
   $$V = \frac{\partial}{\partial \theta_i} \log L\left(Y | x, \theta\right)$$
   (b) Compute the Fisher Information Matrix as a function of the design
   $$\mathcal{I}\left(x, \theta\right) = E_{Y|\Theta}\left[V^2 | \theta\right]$$
   (c) Compute the utility as a function of the Fisher Information Matrix
   $$\phi\left(x, \theta\right) = g\left(\mathcal{I}\left(x, \theta\right)\right)$$
   (d) The objective function is equal to the utility $\Phi = \phi$
4. Maximize the utility while searching over all designs, $x \in \mathcal{X}$

The system and observation models are incorporated into the likelihood function $L(\Theta|Y(x)) = f_{Y|\Theta}(y|\theta, x)$. The algorithm is to select a design x from the design space to maximize the utility, which is a function of the Fisher Information Matrix. Fisher information is a metric related to Shannon information, and closely related to the variance of Gaussian Random Variables. Note that the Fisher information depends on θ, which is unknown. Therefore in order to compute the objective function, θ must be assigned a value, i.e., the model must be linearized.

5.3.3 The Fisher Information Matrix

By examining the experimental process, we see that some steps are missing. There is no incorporation of prior parameter knowledge, simulation of data, parameter estimation, or risk metric. The reason the steps above are unnecessary is because the parameter estimation or inference step is done implicitly by computing the Fisher Information Matrix. Doing this requires three large assumptions.

First, the model must be linear with respect to the parameters. Also, the observation model must be an unbiased Gaussian distribution added to the model output:

$$\mathrm{E}(\omega_{\mathrm{E}}) \sim N(0, \sigma^2)$$

Lastly, either the prior parameter information is 'noninformative' [43] or is simply ignored. If these assumptions are valid then Fisher Information Matrix completely describes the information content of the parameters without any need for data simulation or parameter estimation. This comes from the Cramer-Rao Bound, which is explained in Appendix B.5.

The summary is that when the assumptions are valid, the Least Squares estimator is guaranteed to give the best parameter estimates and the Fisher Information Matrix is inversely proportional to the parameter variance. Also, because the model is linear, any value of θ will result in the same utility. This means that data simulation and risk metrics are unnecessary - the only thing that matters is the Fisher Information Matrix, computed from the system model, and the observation model. The utility functions used for Optimal Designs are all functions of the Fisher Information Matrix. All of the objectives stated in Section 5.1.2 have formulations for Optimal Designs.

5.3.4 Optimization

Relying only on the system model, which is linear, and the observation model, which is Gaussian, has tremendous computational benefits. Unfortunately, the assumptions severely restrict the number of engineering models that Optimal design can be applied to. Nevertheless, these problems reduce to a deterministic optimization problem, which can be solved using global optimization algorithms.

5.3.5 Drawbacks of Optimal Designs

The source of Optimal Designs advantages is also the source of its weakness. Optimal designs restrict prior uncertainty knowledge, deal only with linear models, and only with Gaussian distributions. They are still superior to Classical designs, since they have at least incorporated a simplified understanding of the system performance into the design process. However, the simplifications do not allow a full description of uncertainties and so the solutions are based on incomplete information.

5.4 Bayesian Design of Experiments

The Bayesian Design of Experiments, as the name suggests, is a model based approach that utilizes Bayesian methods for treating uncertainty and performing inference [16]. The design approach is discussed here while many of the Bayesian methods are described in Chapter 6.

Like Optimal Designs, this approach addresses many concerns with Classical Design of Experiments since it implicitly relies on system models derived from the underlying physics and chemistry of the system. The advantage of utilizing system models is that they provide objective predictions of experimental results. This allows the design process to select the most useful experiments. This advantage increases with process complexity. An empirical fit or surface response map can be fairly accurate for a simple process with one or two units. Even if the system response is non-linear, the general trends can be captured. This is usually enough to provide a working knowledge of the system - enough to control the process. As the system becomes more complex, with many units with multiple interconnections, a simple empirical fit becomes inadequate. Not only is the fit unlikely to capture a complex response, it does not shed any light on the sources of uncertainty which are so critical to process understanding.

Bayesian Designs further improve upon Optimal Designs by allowing the use of any system model, using all prior knowledge, and treating complex uncertainties. Once again the theme of this work is that design of experiments hinges on the understanding of uncertainty. The assumptions that Optimal Designs take allow for efficient computation of the solutions, but no longer provide an accurate description of our uncertain system knowledge. Bayesian Designs properly address all these issues by dropping the assumptions. If the linear model and Gaussian distribution assumptions are kept, Bayesian Designs reduce to Model Based Experimental Designs [5] which are discussed in Appendix E.

5.4.1 Information Flow

The Bayesian approach to design of experiments process mimics the experimental study process. This process was previously illustrated in Figure 1-1, which is shown again in Figure 5-9 along with an information flowchart.

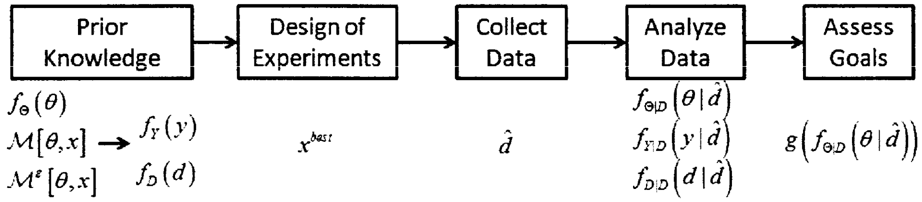


Figure 5-9: Information flowchart of the experimental process involving parameters, model outputs, data predictions, designs, and data

At each stage of the experimental process, different pieces of information are available. Here the flow of information is laid out explicitly to show how simulations can replace real experiments.

Prior Knowledge

Prior knowledge refers to all the information known before data is collected or simulated. This includes the system and observation models, and the knowledge of uncertain model parameters. The system model describes the response as it varies due to changes in the parameters and inputs. The additional complexity that arises from experiments is the fact that many important system properties cannot be measured or observed perfectly. Properties must be inferred from indirect measurements and the instruments are not perfectly accurate. Therefore, every observation of the system will have some error and an observation model is necessary to represent our knowledge of these errors. Because errors are never known, they are described by uncertain residuals or discrepancies from the model. We state that observations have uncertainty with a known relationship to the model output. This allows a prediction of what data will be observed from the system.

The implicit assumption with all model based designs is that the model is correct. The experimental designs that result are only optimal for a system that is accurately represented by the models. This can be problematic if the system behaves very differently from the model. On the other hand, if the experimental results are different from expected, then the experiment has successfully shown that the model is inadequate.

- System model: $y = \mathcal{M}[\theta; x]$
- Observation model
    - In general, the observation model must be written as a function of y, x, and/or θ. There is no simple formula for the observation errors density and it is expressed as the conditional probability $f_{\mathrm{E}|\mathrm{Y},\Theta;x}$.
    - In most cases, it can be assumed that the errors follow some parameterized distribution with parameters that depend on the values of y, θ, and x.
      $f_{\mathrm{E}|\mathrm{Y},\Theta;\mathrm{x}}(\varepsilon|\mathsf{y},\theta;x) = g(\varepsilon,\mathsf{y},\theta,x)$
    - A further simplification is to assume that all observation errors are additive and independent of y, θ, and x. Therefore:
      $\mathsf{d} = \mathsf{y} + \varepsilon$ and $f_{\mathrm{E}|\mathrm{Y},\Theta;\mathrm{x}}(\varepsilon|\mathsf{y},\theta;x) = f_{\mathrm{E}|\mathrm{Y},\Theta;\mathrm{x}} = g(\varepsilon)$
    - Then any distribution can be used to model the additive errors: $\mathrm{E}(\omega) \sim f_{\mathrm{E}}(\varepsilon)$
      Even though E is not directly dependent on Θ, it must be remembered that both d and y are dependent on Θ, and so E and Θ are conditionally dependent given D or Y.
- Independent variables: x
- Prior parameter densities: $f_{\Theta}(\theta)$

The structure of the models must be known exactly, with the only uncertainties represented by model parameters. There is no uncertainty in the independent variables since these are controlled by the experimenter. Any uncertainty in x, for example if the control is not precise, must be captured by adding parameters as discussed in Chapter 3. A discussion of how to characterize prior knowledge is included in Section 6.1.1.

Derived Distributions

Analysis of the prior knowledge and models results in much more insight into the system, including what system properties are expected (Model Outputs) and what observations could be expected from an experiment (Data Predictions).

- Prior model output density
  - $Y(\omega) = \mathcal{M}[\theta; x]$
  - $f_{Y;x}(\mathsf{y}; x) = \int\limits_{\Theta} f_{\Theta}(\theta)\, f_{Y|\Theta;x}(\mathsf{y}|\theta; x)\, d\theta$
- Prior data predictive density
  - The data predictions are based off the model outputs and the observation errors. Assuming an additive observation model,
    $D(\omega) = Y(\omega) + \varepsilon(\omega)$
  - $f_{D;x}(\mathsf{d}; x) = \int\limits_{Y} f_{Y;x}(\mathsf{y}; x)\, f_{D|Y;x}(\mathsf{d}|\mathsf{y}; x)\, d\mathsf{y}$
    or $f_{D;x}(\mathsf{d}; x) = \int\limits_{\Theta} f_{\Theta}(\theta) \int\limits_{Y} f_{Y|\Theta;x}(\mathsf{y}|\theta; x)\, f_{D|Y,\Theta;x}(\mathsf{d}|\mathsf{y}, \theta; x)\, d\mathsf{y} d\theta$
  - Note that when Θ and x are specified, so is Y. Therefore:

$$\int\limits_{Y} f_{Y|\Theta;x}(\mathsf{y}|\theta; x)\, f_{D|Y,\Theta;x}(\mathsf{d}|\mathsf{y}, \theta; x)\, d\mathsf{y} = f_{D|Y,\Theta;x}(\mathsf{d}|\mathsf{y} = \mathcal{M}[\theta; x], \theta; x)$$

  - Explicitly stating the additive observation model:
    $f_{D;x}(\mathsf{d}; x) = \int\limits_{\Theta} f_{\Theta}(\theta)\, f_{\mathrm{E}|\Theta;\mathrm{x}}(\varepsilon = \mathsf{d} - \mathcal{M}[\theta; x]) d\theta$

Experiments

At this step, an experiment is carried out or simulated. Simulating an experiment with a design x_0, is equivalent to drawing a sample from the prior predictive density, given $x = x_0$. Simulated datasets will be denoted with a hat: $\hat{d}$, to emphasize the difference between a real dataset.

$$\hat{d}_0 \sim f_{D;x}(\mathsf{d}; x_0)$$

Posterior Knowledge

After the simulation of a dataset additional knowledge can be inferred about the system, including improved parameter estimates and corresponding improved predictions about the system properties and future observations. For a generic simulated dataset $\hat{d}$, drawn using a generic design x:

- Posterior Parameter density
  - $f_{\Theta|\hat{d}}\left(\theta|\hat{d}\right) = \frac{f_{\Theta}(\theta) f_{D|\Theta;x}\left(\hat{d}|\theta;x\right)}{f_{D;x}\left(\hat{d};x\right)}$
  - Expanding for clarity:

$$\begin{aligned} f_{D|\Theta;x}\left(\hat{d}|\theta;x\right) &= f_{D|\Theta;x}\left(\hat{d}|\mathsf{y} = \mathcal{M}\left[\theta;x\right]\right) \\ &= f_{\mathrm{E}}\left(\varepsilon = \hat{d} - \mathcal{M}\left[\theta;x\right]\right) \\ f_{\Theta|\hat{d}}\left(\theta|\hat{d}\right) &= \frac{f_{\Theta}\left(\theta\right) f_{\mathrm{E}}\left(\varepsilon = \hat{d} - \mathcal{M}\left[\theta;x\right]\right)}{f_{D;x}\left(\hat{d};x\right)} \end{aligned}$$

- Posterior model output density
  - $f_{Y|\hat{d};x}\left(\mathsf{y}|\hat{d};x\right) = \int_{\Theta} f_{\Theta|\hat{d};x}\left(\theta|\hat{d};x\right) f_{Y|\Theta;x}\left(\mathsf{y}|\theta;x\right) d\theta$
  - $f_{Y|\hat{d};x}\left(\mathsf{y}|\hat{d};x\right) = \int_{\Theta} f_{\Theta|\hat{d};x}\left(\theta|\hat{d};x\right) f_{Y|\Theta;x}\left(\mathsf{y} = \mathcal{M}\left[\theta,x\right]\right) d\theta$
  - This is the same as the prior model output density, except that the uncertainty is propagated from the posterior knowledge of the parameters.
- Posterior predictive density
  - $f_{D|\hat{d};x}\left(\mathsf{d}|\hat{d};x\right) = \int_{\Theta} f_{\Theta|\hat{d}}\left(\theta|\hat{d}\right) f_{\mathrm{E}|\Theta;\mathrm{x}}(\varepsilon = \mathsf{d} - \mathcal{M}\left[\theta;x\right])d\theta$
  - Again this is the same as the prior predictive density, using the posterior parameter density.

5.4.2 Relation to Design of Experiments

The information flow described above is the same whether the experiments are real or simulated. The flow is naturally broken into two steps: predicting system performance (all steps prior to the observation of data) and the incorporation of new information (steps after or post-observation). As shown by the information flow, a simulated experiment can be used to estimate the usefulness of a real experiment. By using these estimates within the decision theory framework, we can make comparisons between hypothetical experiments and then choose the best ones.

5.4.3 The Bayesian Design Algorithm

The algorithm will be presented withing the decision theory framework. These translate directly to three computational loops.

1. Actions - Optimize

   The first step is the enumeration of actions or the selection of a design.

2. Uncertainties - Simulate Data

   The second step is to describe the uncertain consequences or compute the probability of each consequence. In the experimental process this corresponds to running the experiments and observing the uncertain results – datasets. In the Bayesian Design algorithm this is the middle loop in which many datasets are simulated. The uncertainty quantification methods described in Chapter 3 allow sample datasets $\hat{d}_j$ to be taken from the data prediction density for each design x_i.

3. Utility - Compute the objective function

   The final step is to assess the utility of each consequence. This is done using utility functions and risk metrics. Because the utility functions are integrals, they are typically computed with Monte Carlo methods which make up the third and inner most loop.

The algorithm is given using the most general approach.

1. Select $x_i \in \mathcal{X}$

   (a) Sample from the prior parameter density

   $\theta_j \sim f_\Theta(\theta)$

   (b) Evaluate the model

   $y_j = \mathcal{M}[\theta_j; x_i]$

   (c) Sample from the data predictive density

   $d_j \sim f_D(d; x_i)$

   For an additive observation model, this is equivalent to:

   $d_j \sim y_j + \varepsilon_j$ where $\varepsilon_j \sim f_{E|\Theta}(\varepsilon|\theta_j; x_i)$

        i. Sample from the posterior parameter density using Markov Chain Monte Carlo (See Section 6.3)

           $\theta_k \sim f_\Theta|D\left(\theta|d_j\right)$

        ii. Iterate $k = 1 \ldots K$

    (d) Compute the utility of dataset d_j for design x_i

        $\phi_i j = g\left(\theta_1 \ldots \theta_K, x\right)$

    (e) Iterate $j = 1 \ldots J$

2. Compute the objective function for design x_i

   $\Phi_i = \frac{1}{J} \sum\limits_{j=1}^{J} \phi_{ij}$

3. Iterate over i until optimum is found or search ends

5.4.4 Summary

By now all the theory has been presented to show how the Bayesian approach treats uncertainty and selects the best design. All that remains is the methodology that allows us to compute the utility functions. These are presented in the next chapter. One reason that the Bayesian approach is not widely used is the large computational cost. The reason for this cost is clear – three nested loops. In many cases, the benefits of proper treatment of uncertainty outweigh the increased computer time. Nevertheless, additional shortcut techniques are also presented to reduce computational cost while maintaining the rigorous characterization of uncertainty. These will aid in the adoption of the Bayesian approach in the engineering community.

Chapter 6

Bayesian Methods for Estimation and Design

This chapter describes all the methods required to implement the Bayesian Design algorithm described in Section 5.4. These methods are then demonstrated in Chapters 7 through 10.

6.1 Bayes' Theorem

A Bayesian method is any method that uses Bayes' Theorem. The theorem itself is deceptively simple. In terms of probabilities of events:

$$P(A|B) = \frac{P(A)\,P(B|A)}{P(B)} \tag{6.1}$$

where:

- $P(A)$ is the prior probability of event A, before event B occurs
- $P(B)$ is the prior probability of event B, which is a normalization constant
- $P(B|A)$ is the probability of event B occurring, given event A has occurred
- $P(A|B)$ is the posterior probability of A, after B has occurred

There are two equally valid perspectives of Bayes' Theorem. Either the prior knowledge about random event A is updated based on new new information that arrives from random event B, or the new information is observed from B and is then biased by the prior knowledge about A.

Using parameter estimation as an example, A would be a Random Variable Θ representing an uncertain parameter, and B would be a Random Variable $D(\omega)$ representing the observation of data. This is then written with probability density functions:

$$f_{\Theta|D}(\theta|\mathsf{d}) = \frac{f_\Theta(\theta)\, f_{D|\Theta}(\mathsf{d}|\theta)}{f_D(\mathsf{d})} = \frac{f_\Theta(\theta)\, L(\theta|\mathsf{d})}{f_D(\mathsf{d})}$$

- $f_\Theta(\theta)$ is the prior parameter probability density
- $f_D(\mathsf{d})$ is the prior predictive data density probability
- $f_{D|\Theta}(\mathsf{d}|\theta)$ or $L(\theta|\mathsf{d})$ is called the likelihood
- $f_{\Theta|D}(\theta|\mathsf{d})$ is the posterior parameter probability

The key pieces are the prior information, likelihood, and normalization.

6.1.1 Prior Information

The methods used here require the knowledge of model parameters to be represented by probability densities. When a density does not exist, it must be computed or assumed. In most cases, there is incomplete information - either summary statistics like mean, variance, minimum or maximum, or expert opinions about the parameter. When this happens the Maximum Entropy Principle is used to construct the probability density function.

Maximum Entropy

The Maximum Entropy Principle simply states that the best probability density to represent uncertain but incomplete information is the one having maximum differential entropy, while still being consistent with the available knowledge. This principle stems from the fact that constructing a probability density function requires adding information that is not supported by the current knowledge. Therefore, the amount of additional information should be minimized, which is equivalent to maximizing the entropy. This is formulated as an optimization problem:

$$\max_{f_\Theta(\theta)} - \int_{-\infty}^{\infty} f_\Theta(\theta) \log\left(f_\Theta(\theta)\right) d\theta$$

$$\text{s.t. Constraints from available knowledge}$$

Table 6.1: List of various conditions that will result in well know maximum entropy distributions. A good reference is *Entropy optimization principles with applications* by Kapur and Kesavan [44].

Available Information	Maximum Entropy Distribution
Upper and Lower Bounds	Uniform
Mean and Variance	Gaussian
Mean and One Bound	Shifted Exponential
Mean and Two Bounds	Truncated Exponential

As an example, say that we know or assume that an uncertain parameter has mean μ and variance σ^2. This problem is formulated as:

$$\min_{f_\Theta(\theta)} \int_\Theta f_\Theta(\theta) \log f_\Theta(\theta)\, d\theta$$

$$\text{s.t.}$$

$$\int_\Theta f_\Theta(\theta)\, d\theta = 1$$

$$\int_\Theta \theta f_\Theta(\theta)\, d\theta = \mu$$

$$\int_\Theta \theta^2 f_\Theta(\theta)\, d\theta = \sigma^2$$

Solving this problem is well beyond the scope of this thesis but it can be proved that the Gaussian distribution does minimize the negative entropy. Table 6.1 shows a few other conditions and the resulting Maximum Entropy distribution.

6.1.2 Likelihood Function

A likelihood function is a modified conditional probability where the independent variable is the second argument and the first argument is known.

$$L(\theta|\mathsf{d}) = f_{D|\Theta}(\mathsf{d}|\theta)$$

This is computed by evaluating the conditional probability on the right hand side, but likelihood function itself is not a probability density because it does not normalize over Σ_D to 1. Also, the likelihood is not equal to the similar conditional probability:

$$L(\theta|\mathsf{d}) \neq f_{\Theta|D}(\theta|\mathsf{d})$$

The likelihood describes the chances that the data d could be observed if θ is the true parameter value.

6.1.3 Normalization Factor

The denominator in Bayes' Theorem is a normalization factor:

$$f_D(\mathsf{d})$$

This is the prior data predictive density which describes all the possible data that could be observed. While it seems simple, this presents one of the biggest difficulties with Bayesian methods. Often times this probability density function is not known and must be derived by integrating:

$$f_D(\mathsf{d}) = \int\limits_{\Omega_\Theta} f_\Theta(\theta)\, f_{D|\Theta}(\mathsf{d}|\theta)\, d\theta$$

This integral is over the prior parameter density, which has dimension equal to the number of model parameters. This can be extremely difficult to compute.

Example 22: The Monte Hall Problem

In this game there are three closed doors. Behind two doors are goats; behind the third is a car (worth more than a goat). The rules allow you to choose a door and receive the prize behind it. First you pick one of the doors. Without revealing the prize behind the first door, the host has to open one of the remaining doors. He must choose a door with a goat, otherwise the game is over. After seeing his choice, you are given the chance to switch doors. Which door should you choose?

Using a Decision Tree

First lay out the prior knowledge. Let $p_D\left(\mathsf{d}\right)$ be the prior probability mass function that door d hides the car. Then,

$$p_D\left(\mathsf{d}\right) = \begin{cases} \frac{1}{3} & \mathsf{d} = 1,2,3 \end{cases} \tag{6.2}$$

Now there is a decision: choosing one of the three doors. At this point, the doors are all identical so this decision is irrelevant. Say we pick door 1. There are now three possible outcomes - that the car is behind door 1, 2, or 3. Next, the host reveals one of the doors, which must have a goat. If the car is behind door 1, then either 2 or 3 can be opened. However, in the other cases, only one option exists because the 2nd goat is behind door 1. Finally, we reach the third decision - whether to switch doors. This is illustrated by the decision tree in Figure 6-1.

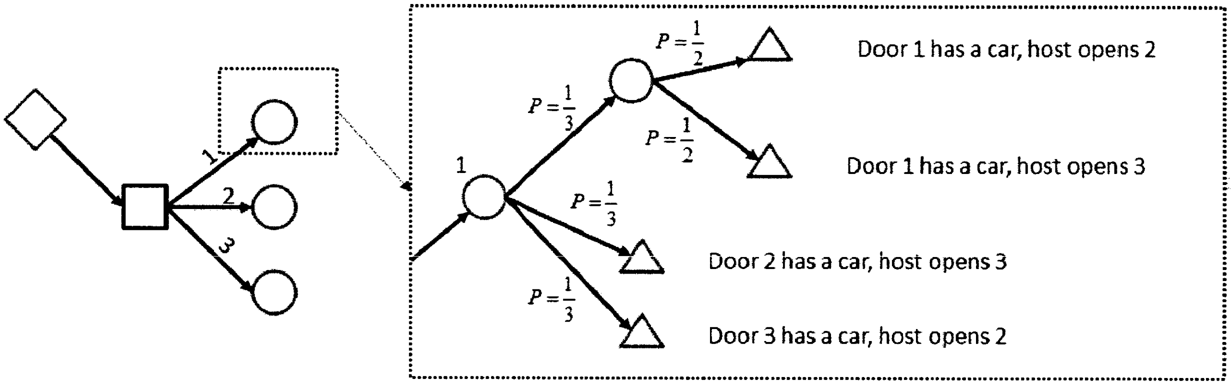

1
2
3
1
$P=\frac{1}{3}$
$P=\frac{1}{2}$
Door 1 has a car, host opens 2
$P=\frac{1}{2}$
Door 1 has a car, host opens 3
$P=\frac{1}{3}$
Door 2 has a car, host opens 3
$P=\frac{1}{3}$
Door 3 has a car, host opens 2

Figure 6-1: Example 22 – Monty Hall problem as a decision tree - given prior information (diamond) you must choose a door (circles). Each door is the same, so consequences (triangles) are only shown for Door 1

In order to fill in the probabilities on the decision tree, we start with the knowledge of the game rules. The initial decision is to choose between three doors, but again this is irrelevant so only one of these branches is shown. The uncertainty is over which door hides the car - each branch has probability of $\frac{1}{3}$. If the correct door was chosen, the host has two doors which can be opened, each with probability $\frac{1}{2}$. If a goat door was chosen, the host can only open one door, hiding the second goat. So, there are now four possible consequences (two pairs of identical outcomes): the player has selected the right door and by switching will win a goat, or the player has selected the wrong door and will win the car by switching doors. The probabilities are respectively: $\frac{1}{3}\frac{1}{2}\left(2\right) = \frac{1}{3}$ and $\frac{1}{3}\left(2\right) = \frac{2}{3}$. So switching doubles the odds of winning.

Using Bayes' Theorem

The decision tree from Figure 6-1 can be converted into conditional probabilities, which can then be computed using Bayes' Theorem. The prior knowledge, from the game rules, is that the probability of door c hiding the car is: $P(C=\mathsf{c}) = \frac{1}{3}$. Next, after the player Selects door s, the probabilities are unchanged: $P(C=\mathsf{c}|S=\mathsf{s}) = \frac{1}{3}$. Finally, we must compute the conditional probabilities after the host opens door h: $P(C=\mathsf{c}|S=\mathsf{s}, H=\mathsf{h})$.

To solve the problem, we will presume the player selects door 1. Then we are interested in the probabilities of the four consequences: $P(C=1|S=1, H=2)$ and $P(C=1|S=1, H=2)$ which are the probabilities of winning without switching doors – corresponding to the top two consequences in Figure 6-1 – and $P(C=2|S=1, H=3)$ and $P(C=3|S=1, H=2)$, which is the probability of winning by switching doors - corresponding to the bottom two consequences. To compute these we apply Bayes' Theorem.

$$P(C|S=1, H) = \frac{P(C|S=1)\,P(H|S=1, C)}{P(H|S=1)}$$

The necessary conditional probabilities can be read off of the decision tree, with the conditional statements correspond to branching points on the tree. For example, the probability that the car is behind door 1, given that you chose door 1 and the host opened door 2 would be:

$$\begin{aligned}P(C=1|S=1, H=2) &= \frac{P(C=1|S=1)\,P(H=2|S=1, C=1)}{\sum\limits_{c} P(C)\,P(H=2|S=1, C)}\\ &= \frac{P(C=1|S=1)\,P(H=2|S=1, C=1)}{P(C=1)\,P(H=2|S=1, C=1) + P(C=3)\,P(H=2|S=1, C=3)}\\ &= \frac{\frac{1}{3}\frac{1}{2}}{\frac{1}{3}\frac{1}{2} + \frac{1}{3}\frac{1}{2}}\end{aligned}$$

the probability that the car is behind door 2, given that you chose door 1 and the host opened door

3 would be:

$$\begin{aligned}P(C=2|S=1,H=3)=&\frac{P(C=2|S=1)\,P(H=3|S=1,C=2)}{\sum\limits_{c}P(C)\,P(H=3|S=1,C)}\\=&\frac{P(C=2|S=1)\,P(H=3|S=1,C=2)}{P(C=1)\,P(H=3|S=1,C=1)+P(C=2)\,P(H=3|S=1,C=2)}\\=&\frac{\frac{1}{3}1}{\frac{1}{3}\frac{1}{2}+\frac{1}{3}1}\end{aligned}$$

So, by symmetry, the probability of winning without switching is $\frac{1}{2}$ and the probability of winning by switching is $\frac{2}{3}$.

6.2 Bayesian Parameter Estimation

Bayesian Parameter Estimation is another application of Bayes' Theorem. In all parameter estimation problems, a model structure is assumed with uncertain but well defined parameters. Then a dataset is observed and the parameters must be inferred. In this context, the Bayesian approach is to use probability to describe the model parameters' uncertainty and the conditional event is the collection of data. The key concept is that the observed data contains information. The model establishes the relationship between the data and the parameters and therefore knowledge from the data can influence the knowledge of the parameters. Bayes' Theorem describes this inference step.

6.2.1 Details

As with all uses of Bayes' Theorem there are three components: the prior knowledge, the likelihood, and the normalization. Prior knowledge in the case of parameter uncertainty was discussed in Section 6.1.1. The likelihood function describes how likely it is that the observed data could be explained by the system and observation models. Since these models depend on the parameters, this establishes the connection from data to parameters. Finally, the normalization constant simply ensures that the posterior parameter density is properly normalized.

Example 23: A Biased Coin

Have a coin that seems to fall on heads more often than tails. Say that parameter θ represents the probability that the coin will fall on heads.

The prior parameter density is $\Theta \sim U(0.5, 1)$ and the likelihood is given by the binomial distribution: $P(heads|\theta) = \theta$ and $P(tails|\theta) = 1 - \theta$. Because each flip is assumed to be independent, the likelihood of multiple flips is the product of the likelihood of each individual flip.

Let's say we observe three heads and then one tail. There is no uncertainty in these observations. Then the likelihood of this event is $P\left(H^3T|\theta\right) = \theta^3(1-\theta)$. The Normalizing constant is:

$$\int\limits_{0.5}^{1} 2\theta^3(1-\theta)\,d\theta = 2\,\frac{1}{4}\theta^4 - \frac{1}{5}\theta^5\Big|_{0.5}^{1} = 2\left(\frac{1}{4}-\frac{1}{5}\right) - \frac{1}{2^4}\left(\frac{1}{4}-\frac{1}{2}\frac{1}{5}\right) = \frac{13}{160}$$

Bayes' Theorem gives us:

$$P\left(\theta|H^3T\right) = \begin{cases} \frac{320}{13}\theta^3(1-\theta) & 0.5 \le \theta \le 1 \\ 0 & else \end{cases}$$

The prior and posterior parameter densities are shown in Figure 6-2.

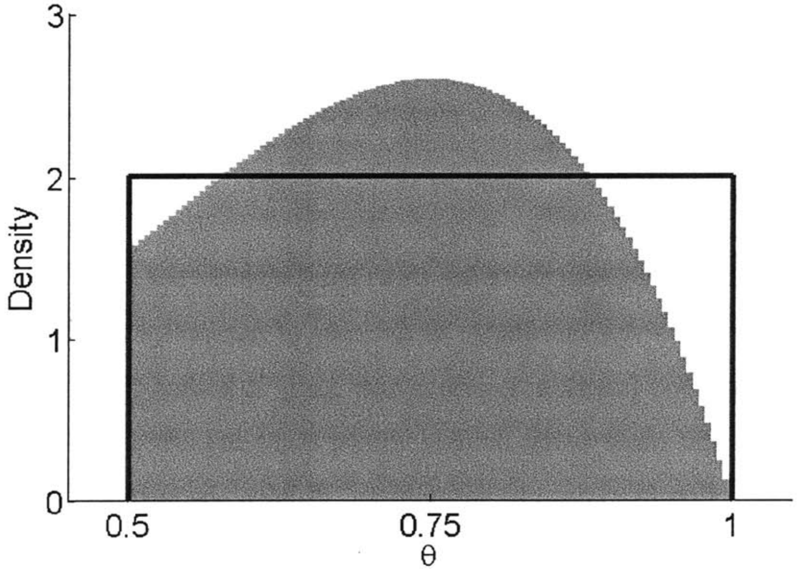


Figure 6-2: Example 23 – Prior (— Uniform density) and posterior (filled truncated Beta density) uncertainty of coin bias parameter θ after observing four flips: HHHT

6.2.2 Issues

Example 23 is very simple. It has no observation error and the system model can be easily manipulated which is rarely the case for engineering systems. The the main difficulty in most parameter estimation problems is not the system or observation model, but the normalization constant. In the coin example, there is a single parameter that must be normalized. Typical engineering systems have many parameters and so the computation of this normalization constant becomes a high dimensional integral. These are very difficult to compute numerically. This has motivated the development of numerical methods to compute posterior densities without requiring the normalization, such as Markov Chain Monte Carlo.

6.3 Markov Chain Monte Carlo

As previously mentioned, Markov Chain Monte Carlo is a numerical method for high dimensional problems. It is particularly beneficial for Bayesian methods because it does not need to compute the normalizing constant.

6.3.1 The Markov Chain Concept

Monte Carlo is a sampling technique where the values of the independent variables are selected randomly. Markov Chain Monte Carlo is a variation that uses a directed sampling heuristic instead of an entirely random sampling. A Markov Chain is a series of samples where the next state depends only on the current state. This means that the details of any previous state are totally irrelevant.

The chain can be thought of as a person walking around a field. From the current sampling point, a jump is taken and the new sampling point is subjected to an acceptance criteria. If the new sampling point acceptable, it is sampled (the person walks to the new point), otherwise the previous point is sampled again (the person stays put). Now imagine that the field is uneven with hills and valleys. At each time point, the person (often assumed to be drunk) surveys his surroundings and chooses another location to jump to. He then compares the height of the next location to his current location. If it is downhill, he is more likely to make the jump. If it is uphill, he is more likely to stay put. Over many time steps, this person will trace out a path across the field. The topography of the field can be determined from the amount of time the person spends in each spot. This is an attractive property if the topography cannot be directly measured, but relative heights between two points can be computed easily.

6.3.2 Application to Parameter Estimation

Markov Chains works especially well for computing probability density functions because the densities are all relative quantities. In parameter estimation problems we cannot compute the absolute density of the posterior parameter density (the normalization constant is unknown), but the relative densities are easily computed. Therefore, the posterior parameter density can be sampled by jumping around the parameter space in a Markov Chain. Once enough samples are taken, the full density can be approximated by normalizing the samples.

6.3.3 Algorithm

For the sample distribution matches the true posterior, the Markov Chain must apply the correct jump and acceptance criteria, collectively known as the 'sampler'. The Metropolis-Hastings algorithm [69] is commonly used to sample posterior probability distribution functions. A simplified version used in this thesis is shown below:

1. A starting point is chosen, θ_0 is a vector of parameters
2. At each step i, a random jump δ_i is chosen according to $\delta_i \sim N(0, \Sigma)$ and the proposed state is $\theta' = \theta_i + \delta_i$
3. The constant α is calculated as: $\alpha_i = \frac{f_\Theta(\theta')L(\theta'|\mathrm{d})}{f_\Theta(\theta_i)L(\theta_i|\mathrm{d})}$
   This is the ratio of the posterior parameter densities at the proposed state θ' versus the current state θ. Therefore, α is the probability that the new state should be chosen over the current state.
4. A random number x is generated from $[0, 1]$. If $\mathrm{x} < \alpha_i$ then the new state is chosen, $\theta_{i+1} = \theta'$. Otherwise, the current state is chosen, and $\theta_{i+1} = \theta_i$.
5. This is iterated until the chain converges or alloted computer resources are exhausted.

There are some key aspects here which are not discussed, namely the convergence and verification of the posterior density and the choice of Σ for the jumps.

6.3.4 Verification of a Markov Chain Monte Carlo Solution

Verification was not studied in this thesis, so the details are left for the literature [57, 70, 34]. Briefly put, verification of solutions is a big problem with Markov Chain Monte Carlo – even more so than with Monte Carlo as discussed in Section 3.3. A chain is considered to have found the correct solution when it is converged and the solution is no longer changing with additional samples. The chain jumps around in a random fashion, so exploration verification simply checks whether all areas of the parameter space have been sampled from. This is theoretically simple, but difficult to implement. Especially with large dimensional parameter spaces, the posterior parameter density will be vanishingly small in the majority of the design space. Therefore, the Markov Chain is not expected to sample these regions. It is difficult to determine whether the lack of sampling is due to a negligible density or failure to converge.

Many diagnostic tools exist to assess convergence, however, they are mostly heuristics and their reliability is questionable [34]. When Markov Chain Monte Carlo was used in this thesis, an analysis was performed to verify the solutions however, the posteriors computed with Markov Chain Monte Carlo should always be considered an approximation.

6.3.5 The Adaptive Metropolis Sampler

One way to improve the convergence rate is to change the jump and acceptance criteria. Intuitively, as the chain moves around the parameter space, it gathers information. The Adaptive Metropolis Sampler uses this information so that future jumps are more useful. It does this by modifying Σ while the chain is running. Details can be found in the works by Haario et al. [37, 36].

Example 23a - Markov Chain Monte Carlo Solution

The prior parameter density and likelihood function were already specified in Example 23 on page 159. Instead of solving for the posterior analytically, we will redo this example without computing the numerator. The standard Metropolis-Hastings algorithm is used.

The Markov Chain jumps around the space of the posterior parameters. As discussed by Geyer [34], the starting point is not important so we use the mean of the prior density, $\theta = 0.75$.

The variance of the jumps is taken to be the variance of the prior distribution, the burn in is 5000 jumps, and the thin rate is 4.

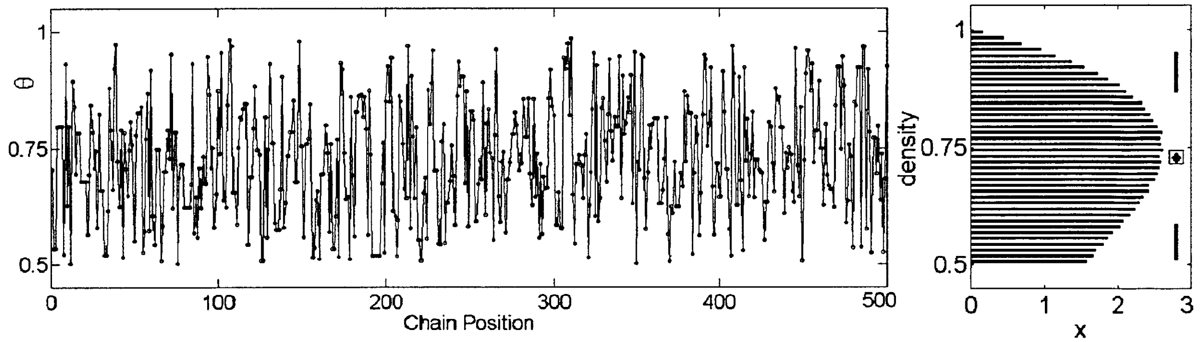
For every jump, from $\theta_i \to \theta'$ a new acceptance probability is computed.

$$\begin{aligned} a =& \frac{f_{\Theta|D}\left(\theta'|H^3T\right)}{f_{\Theta|D}\left(\theta_i|H^3T\right)} \\ =& \frac{f_\Theta\left(\theta'\right) f_{D|\Theta}\left(H^3T|\theta'\right)}{f_\Theta\left(\theta_i\right) f_{D|\Theta}\left(H^3T|\theta_i\right)} \end{aligned}$$

Because the prior is uniform, the first ratio cancels leaving:

$$a = \frac{f_{D|\Theta}\left(H^3T|\theta'\right)}{f_{D|\Theta}\left(H^3T|\theta_i\right)} = \frac{\theta'^3\left(1-\theta'\right)}{\theta_i{}^3\left(1-\theta_i\right)}$$

A section of the resulting Markov Chain (after thinning) is shown in Figure 6-3.



(a) First 500 jumps in the Markov chain

(b) Normalized chain values

Figure 6-3: Example 23 – A Markov chain of the posterior parameter along with normalized histogram of the density

The result of Markov Chain Monte Carlo is a set of samples from the posterior density. It does not compute an analytical probability density function. It does, however, enable a numerical approximation of the posterior density without the large cost and inaccuracy of computing the normalization constant. This method and its variants are used for the majority of Bayesian parameter estimation tasks.

6.4 Prior Sampling Formulation

The major difficulty with Bayesian Statistics is high-dimensional integration. The denominator of the Bayes theorem is in most cases a multi-dimensional integral over the prior knowledge. Also, determining statistics of a joint-probability density function is a multi-dimensional integral. Popular Bayesian methods are generally used because of their ability to circumvent the computation of these integrals. Markov Chain Monte Carlo is one example. By relying on a Markov Chain for parameter estimation, the normalization constant of the posterior parameter density does not need to be computed. Unfortunately, this method does not give the actual posterior parameter density function. It only produces samples. In many experimental designs, the utility function requires the posterior parameter density function to be known. In this case, the function must be approximated using kernel estimation or by assuming a functional form. This introduces errors into the utility function calculation.

One way to circumvent both the normalization problem and the probability density function approximation is the prior sampling formulation of information metrics for design of experiments. This is called the Prior Sampling Formulation for short. Instead of sampling and approximating a multi-dimensional probability density function and then its statistics, this approach computes the statistics directly in order to reduce the dimensionality of the problem.

This section shows the derivation of the Prior Sampling Formulations for the various utility metrics described in Section 5.1.2. The concept was originally applied by Ryan [71] for a particular utility function (the Kullback-Leibler Divergence between posterior and prior parameter densities) to emphasize overall parameter estimation quality. In his paper, Ryan did not propose a name for this method so keep in mind that this terminology is not universal. The original algorithm is described below. In addition, the same idea has now been applied for other utility functions for design of experiments, such as estimating a subset of parameters and discriminating between competing models. These are derived for the general probability density functions described in Section 5.4.1 so the asymptotic results are the same as Markov Chain Monte Carlo. The difference is that the explicit parameter estimation step, including sampling from the posterior parameter density, is skipped which reduces the numerical errors. For this reason, the Prior Sampling Formulations are typically more accurate for the same number of samples. For more details see the paper by Ryan [71].

For all objective functions and risk metrics, there is an implied dependence on the design x. Each probability density function for parameters, model outputs, and data predictions is dependent on the design, however, it is not shown in order to conserve space.

6.4.1 Estimating Parameters

A common goal of an experimental study is to improve model parameters. Despite the fact that not all parameters are equally important, it is typical to design experiments to estimate all the parameters. The standard Bayesian Design of Experiments algorithm was presented in Section 5.4.3. It uses Markov Chain Monte Carlo for parameter estimation and differential entropy of the posterior parameter density as the utility function. Ryan proposed a new formulation that allows the same utility function to be computed without Markov Chain Monte Carlo.

Starting from the outermost loop, the risk metric is the same average utility over all the possible datasets that might be observed. The risk metric is then:

$$\Phi = E_D\left[\phi\right] \tag{6.3}$$

The utility function is the Kullback-Leibler Divergence from the posterior parameter density to the prior parameter density.

$$\phi = -\int\limits_{\Omega_\Theta} f_{\Theta|D}\left(\theta|\mathsf{d}\right)\log\frac{f_{\Theta|D}\left(\theta|\mathsf{d}\right)}{f_\Theta\left(\theta\right)}d\theta \tag{D.1}$$

$$\phi = -\int\limits_{\Omega_\Theta} f_{\Theta|D}\left(\theta|\mathsf{d}\right)\log f_{\Theta|D}\left(\theta|\mathsf{d}\right)d\theta + \int\limits_{\Omega_\Theta} f_{\Theta|D}\left(\theta|\mathsf{d}\right)\log f_\Theta\left(\theta\right)d\theta$$

Substituting the utility function into Equation 6.3,

$$\Phi = \int\limits_{\Omega_D} f_D\left(\mathsf{d}\right)\int\limits_{\Omega_\Theta} f_{\Theta|D}\left(\theta|\mathsf{d}\right)\log\frac{f_{\Theta|D}\left(\theta|\mathsf{d}\right)}{f_\Theta\left(\theta\right)}d\theta d\mathsf{d} \tag{D.2}$$

The utility function depends on both the design x and the dataset d. Averaging over all datasets means that the risk metric Φ only depends on x. By applying Bayes' Theorem and consolidating terms, this reduces to:

$$\Phi = \int\limits_{\Omega_\Theta} f_\Theta(\theta) \int\limits_{\Omega_D} f_{D|\Theta}(\mathsf{d}|\theta) \log f_{D|\Theta}(\mathsf{d}|\theta)\, d\mathsf{d}d\theta - \int\limits_{\Omega_D} f_D(\mathsf{d}) \log f_D(\mathsf{d})\, d\mathsf{d} \tag{D.3}$$

The full derivation of this formulation as well as an alternate version is shown in Appendix D.1. The article by Ryan also discusses how sampling methods can introduce bias.

The integrals is Equation D.3 are evaluated with a Monte Carlo technique. The benefit is that each of the terms can be sampled without performing an explicit parameter estimation step. The prior parameters density is known, the prior predictive density is known as a function of the parameters and observation model, and the prior predictive density conditioned on the parameters is simply a function of the observation model. Although the integral over parameters Θ can be multi-dimensional, this formulation is much easier to compute the risk metric Φ than to estimate the parameters $f_{\Theta|D}(\theta|\mathsf{d})$.

The Algorithm

1. Sample parameter sets from the prior parameter density

   $$\theta_i \sim f_\Theta(\theta)$$

2. Compute the model output for each parameter set

   $$\mathsf{y}_i\ f_Y(\mathsf{y}) = \mathcal{M}[\theta_i]$$

   (a) Use the observation model to simulate datasets from each model output

   $$\mathsf{d}_{ik} \sim f_{D|\Theta}(\mathsf{d}|\theta_i)$$

   (b) Iterate $k = 1 \ldots N_{in}$

3. Iterate $i = 1 \ldots N_{out}$

4. The probability densities of all datasets are collected and used to compute the second term

$$-\int\limits_{\Omega_D} f_D(\mathsf{d}) \log f_D(\mathsf{d})\, d\mathsf{d} \approx -\frac{1}{N_{in}N_{out}} \sum_i \log \sum_k f_{D|\Theta}(\mathsf{d}_{ik}|\theta_i)$$

5. The first term is also a sum of sums:

$$\int\limits_{\Omega_\Theta} f_\Theta(\theta) \int\limits_{\Omega_D} f_{D|\Theta}(\mathsf{d}|\theta) \log f_{D|\Theta}(\mathsf{d}|\theta)\, d\mathsf{d}d\theta \approx \frac{1}{N_{in}N_{out}} \sum_i \sum_k \log f_{D|\Theta}(\mathsf{d}|\theta)$$

6. The prior predictive density conditioned on the parameters is just the observation model. For an additive error model, these two terms can be rewritten as:

$$-\frac{1}{N_{in}N_{out}} \sum_i \log \sum_k f_E(\mathsf{y}_i - \mathsf{d}_{ik})$$

and

$$\frac{1}{N_{in}N_{out}} \sum_i \sum_k f_E(\mathsf{y}_i - \mathsf{d}_{ik})$$

This requires sampling from the prior parameter density, evaluating the model, and sampling from the observation model. It also requires the probability density function of the observation model. For additive observation models where the probability density function is known, this is easy to do. Therefore this Prior Sampling Formulation is very efficient and because more samples can be taken, it is also accurate.

Estimating a Subset of Parameters

Estimating a subset of the uncertain parameters requires only a small change to marginalize out the parameters. The risk metric is shown in Equation D.5:

$$\begin{aligned}\Phi = &\int\limits_{\Omega_D}\int\limits_{\Omega_\Lambda} f_\Lambda(\lambda) \left[\int\limits_{\Omega_\Gamma} f_\Gamma(\gamma) f_{D|\Gamma,\Lambda}(\mathsf{d}|\gamma,\lambda)\, d\gamma\right] \left[\log \int\limits_{\Omega_\Gamma} f_\Gamma(\lambda) f_{D|\Gamma,\Lambda}(\mathsf{d}|\gamma,\lambda)\, d\gamma\right] d\gamma d\mathsf{d} \\ &- \int\limits_{\Omega_D} f_D(\mathsf{d}) \log f_D(\mathsf{d})\, d\mathsf{d}\end{aligned} \tag{D.5}$$

The derivation is shown in Appendix D.1.

The Algorithm

For every design x there are now three loops denoted inner (conditional data), middle (parameters Γ) and outer (parameters Λ).

1. Sample the parameters of interest from their prior parameter density

   $$\lambda_i \sim f_\Lambda(\lambda) = \int\limits_{\Omega_\Gamma} f_{\Gamma,\Lambda}(\gamma,\lambda)\, d\gamma$$

   (a) Sample the remaining parameters from the conditional prior parameter density

   $$\gamma_{ij} \sim f_{\Gamma|\Lambda}(\gamma|\lambda_i)$$

   (b) Compute the model output for each parameter set

   $$\mathsf{y}_{ij}\ f_Y(\mathsf{y}) = \mathcal{M}[\lambda_i, \gamma_{ij}, x]$$

   i. Use the observation model to simulate datasets from each model output

   $$\mathsf{d}_{ijk} \sim f_{D|\Lambda,}(\mathsf{d}|\lambda_i)$$

   ii. Iterate $k = 1 \ldots N_{in}$

   (c) Iterate $j = 1 \ldots N_{mid}$

2. Marginalize out the Γ parameters

   $$G_i \approx \frac{1}{N_{mid}N_{in}} \sum_j \sum_k f_{D|\Gamma,\Lambda}(\mathsf{d}_{ijk}|\gamma_j, \lambda_i)$$

3. Iterate $i = 1 \ldots N_{out}$

4. The probability densities of all datasets are collected and used to compute the second term

$$-\int\limits_{\Omega_D} f_D(\mathsf{d}) \log f_D(\mathsf{d})\, d\mathsf{d} \approx -\frac{1}{N_{in}N_{mid}N_{out}} \sum_i \sum_j \log \sum_k f_{D|\Lambda}(\mathsf{d}_{ijk}|\gamma_j, \lambda_i)$$

5. The data prediction component is now a triple nested sum:

$$\int_{\Omega_D}\int_{\Omega_\Lambda} f_\Lambda(\lambda)\left[\int_{\Omega_\Gamma} f_\Gamma(\gamma)\, f_{D|\Gamma,\Lambda}(\mathsf{d}|\gamma,\lambda)\, d\gamma\right]\left[\log\int_{\Omega_\Gamma} f_\Gamma(\lambda)\, f_{D|\Gamma,\Lambda}(\mathsf{d}|\gamma,\lambda)\, d\gamma\right] d\gamma d\mathsf{d}$$

$$\approx \frac{1}{N_{out}}\sum_i G_i \log G_i$$

6. The prior predictive density conditioned on the parameters is just the observation model. For an additive error model, these two terms can be rewritten as:

$$-\frac{1}{N_{in}N_{out}}\sum_i \log \sum_j f_E; x\left(\mathcal{M}\left[\lambda_i, x\right] - \mathsf{d}_j\right)$$

and

$$\frac{1}{N_{in}N_{out}}\sum_i \sum_j f_E; x\left(\mathcal{M}\left[\lambda_i, x\right] - \mathsf{d}_j\right)$$

This requires more loops than for estimating all parameters, but uses the same sampling strategy and has the same advantages.

6.4.2 Model Discrimination

In addition to the designs for information gain, or better predictions, we can also change the utility function in a Bayesian Design to choose experiments for Model Discrimination. In this case, we have two or more competing models which are consistent with our current knowledge of the system. We wish to run experiments in order choose a single model. There have been many proposed methods for this problem. Many methods are based on tests for checking model consistency with data. The goal then is to collect data which will support one model over the others. This novel method uses simulated data and posterior predictive densities in order to choose experiments that will improve all the model parameters in a way that will allow for discrimination.

This approach is more consistent with the experimental process and Bayesian Design of Experiments. In an actual study, experiments would be designed and run, and data collected. Then a common procedure is to estimate parameters with a portion of the data, and finally compare the models using the unused data. So, the current methods for designing experiments accomplish only the second half. This is remedied with this Hierarchical Bayesian approach.

To use this method, all the models are treated as a single hierarchical model. For each design x, a standard model produces a single output, with uncertainty. A hierarchical model is essentially a collection of submodels. Each submodel is given a weight, representing the confidence that the submodel's predictions are correct. Then the uncertain output of the hierarchical model is the weighted sum of all the submodel's uncertain outputs. This is illustrated in Section 7.2.

The hierarchical model is used when several competing models exist. Each is treated as a submodel of the larger hierarchical model and is given a probability of being correct. This will be called the prior submodel probability and is described by a probability mass function: $p_M(\mathsf{m})$, where the submodels are index by m. This uses mass instead of density because there are a finite number of discrete models. Discrete probability mass functions were not discussed in the background chapter, however a good reference is *Introduction to Probability* by Bertsekas [9]. The goal of model discrimination is to collect data such that the posterior submodel probability of one submodel is much higher than the others. This means that we have confidence in one model over the others.

To compute the posterior submodel probability we will use Bayes' Theorem, but first the likelihood must be chosen. The natural choice is the prior predictive density: $f_{D|M}(\mathsf{d}|\mathsf{m})$ - this asks how well the prior knowledge of each model could explain the data. This version is explored in Appendix D.2. Here however, we wish to ask: how well could the model explain the data, after the inference step? Now the likelihood is the posterior predictive density:

$$p_{M|D,\hat{d}}\left(\mathsf{m}|\mathsf{d},\hat{d}\right) = \frac{p_M(\mathsf{m})\, f_{D|M,\hat{d}}\left(\mathsf{d}|\mathsf{m},\hat{d}\right)}{f_{D,\hat{d}}\left(\mathsf{d},\hat{d}\right)} \tag{6.4}$$

Following the Bayesian Design of Experiments algorithm, we simulate data $\hat{d}$, then use the posterior predictive density as a likelihood function for model discrimination.

The Prior Sampling Algorithm

We have N_{mod} models, each with their own parameters. They share the same observation model, which is assumed to be additive (this can be easily generalized).

- For every design x

```
1. Sample a model from the prior model probability: $\mathsf{m}_j$
2. Sample parameters from the prior parameter density of model $\mathsf{m}_j$ : $\theta_j \sim f_{\Theta|M}(\theta|\mathsf{m}_j)$
3. Evaluate the model: $\mathsf{y}_j = \mathcal{M}[\mathsf{m}_j, \theta_j, x]$
4. Sample dataset $\mathsf{d}_j = \mathsf{y}_j + \varepsilon$ or $\mathsf{d}_j \sim f_{D|M,\Theta}(\mathsf{d}|\mathsf{m}_j, \theta_j)$
5. For each model $\mathsf{m}_i$
   (a) Sample parameters from the prior parameter density of model $\mathsf{m}_i$ :
       $\theta_k \sim f_{\Theta|M}(\theta|\mathsf{m}_i)$
       – Compute the parameter estimation likelihood of the data $\mathsf{d}_j$ being observed if
         model $\mathsf{m}_i$ is correct and the parameters are $\theta_k$ :
         $L(\mathsf{d}_j, \mathsf{m}_i, \theta_k) = f_{D|M,\Theta}(\mathsf{d}_j|\mathsf{m}_i, \theta_k)$ or
         $L(\mathsf{d}_j, \mathsf{m}_i, \theta_k) = f_{\mathrm{E}}(\mathsf{d}_j - \mathcal{M}[\mathsf{m}_i, \theta_k, x])$
   (b) Iterate over $k$ for $k = 1 \ldots N_k$
   (c) Evaluate the model discrimination likelihood (see Appendix D.2 for details):
       $\int\limits_{\Omega_\Theta} f_{\Theta|M}(\theta|\mathsf{m}_i) L(\mathsf{d}_j, \mathsf{m}_i, \theta_k) L(\mathsf{d}_j, \mathsf{m}_i, \theta_k)\, d\theta \approx \frac{1}{N_k} \sum\limits_k [L(\mathsf{d}_j, \mathsf{m}_i, \theta_k)]^2$
   (d) $Lnorm_{i,j} \approx \frac{1}{N_k} \sum\limits_k L(\mathsf{d}_j, \mathsf{m}_i, \theta_k)$
6. Iterate over $i$ for $i = 1 \ldots N_{mod}$
7. $Mnorm_j = \sum\limits_i \left[ \frac{p_M(\mathsf{m}_i)}{Lnorm_{i,j}} \frac{1}{N_k} \sum\limits_k [L(\mathsf{d}_j, \mathsf{m}_i, \theta_k)]^2 \right]$
• Iterate over $j$ for $j = 1 \ldots N_j$
• $\Phi \approx \frac{1}{N_j} \sum\limits_j \sum\limits_i \left[ \frac{p_M(\mathsf{m}_i) \frac{1}{N_k} \sum\limits_k [L(\mathsf{d}_j, \mathsf{m}_i, \theta_k)]^2}{Lnorm_{i,j} Mnorm_j} \log \frac{\frac{1}{N_k} \sum\limits_k [L(\mathsf{d}_j, \mathsf{m}_i, \theta_k)]^2}{Lnorm_{i,j} Mnorm_j} \right]$
```

The algorithm is demonstrated in Section 7.2

6.5 Polynomial Chaos Expansion Based Surrogate Models

Model complexity often becomes a problem for methods that require many model evaluations. Monte Carlo methods, for example, can require thousands of model evaluations. When each model evaluation takes a lot of resources, this can be prohibitively expensive. The purpose of model based design of experiments is to reduce the number of physical experiments by performing computer simulations, however, this is not an attractive solution when the computer simulations are also costly. For this reason, there are many strategies for reducing model complexity.

When choosing a strategy, it is important to remember our purpose of using models – understanding and describing uncertainty. We require a method that can efficiently describe the uncertainty of the model predictions while still retaining the physical meaning of the parameters. Using a polynomial response surface, for example would not meet either requirement. Some model order reduction methods [72] seek to retain the physical meaning of the models, but cannot properly treat uncertainties. Therefore, we have use Polynomial Chaos Expansions.

Chapter 3 discussed how Polynomial Chaos Expansions can be used to efficiently quantify complex sources of uncertainty and their impact on model outputs. From the perspective of uncertainty quantification, Polynomial Chaos Expansions cannot deal with independent variables. Mathematically, however, there is no difference between an uncertain parameter and an independent variable. Polynomial Chaos Expansions can in fact be used to simultaneously describe uncertainty in the parameters and changes in the model response due to the independent variable. This idea of a combination uncertainty and model response map has been applied to uncertainty quantification and parameter inference [41, 54, 53, 55]. In addition, Polynomial Chaos Expansion surrogates were recently used for design of experiments [40].

Scaled independent variables x with bounds $\mathcal{X} = [0,1]^N$ can be represented by a uniform Random Variable $X(\omega) \sim U[0,1]$. As discussed by Marzouk et al., the use of a uniform Random Variable is not required [53], but it is conceptually easiest. This way, a uniform Basis Random Variable Ξ_X can be substituted for $X(\omega)$ and the values match will both correspond to the independent variable $\xi_X \sim \mathrm{x} \sim x$. With all independent variables treated as Random Variables, the uncertainty quantification proceeds as in Chapter 3. The result is a polynomial approximation of the model output that also describes the uncertainty at every point in the domain $\mathcal{X}$.

The independent variables can still be controlled by setting the Basis Random Variable equal to the desired independent variable value, $\xi_X = x$, and then uncertainty at x can be easily quantified using Monte Carlo sampling from the Basis Random Variables corresponding to uncertain parameters.

The resulting Polynomial Chaos Expansion is called a surrogate model. Because it accurately describes parametric uncertainty and model responses, it is suitable for use in Bayesian Design of Experiments.

Chapter 7

Design of Experiments Examples

This chapter contains two examples which serve to illustrate the methods for Bayesian Design of Experiments for parameter estimation and for model discrimination.

7.1 Sequential Reactions in a Batch Reactor

This example shows the necessary methodology for Bayesian Design of Experiments using Markov Chain Monte Carlo for parameter estimation. The system is simple and well known to chemical engineers but the proposed design of experiments problem does not have an obvious solution. It demonstrates a utility function that depends on a derived system property, instead of the parameters themselves.

7.1.1 Setup

The system is the same as in Example 8 on page 56. We have two reactions in sequence $A \rightarrow B \rightarrow C$, where A, B, C are the dimensionless concentrations of three chemical species. There is uncertainty in the kinetic parameters and in this case we also add uncertainty in the initial concentration of Species A.

The experimental system is batch reactor in which two sequential reactions occur: $A \rightarrow B \rightarrow C$. The reactor has a fixed volume and temperature is held constant. At the beginning of the experiment, A is the only species present, with an uncertain initial condition represented by a random variable $A_0 \sim U\,[0.95, 1.05]$. The reactions both follow Arrhenius kinetics, with uncertain parameters α and β

$$
\begin{aligned}
k_1(\omega_{\alpha_1},\omega_{\beta_1}) =& \alpha_1(\omega_{\alpha_1})\, exp\left(\frac{-\beta_1(\omega_{\beta_1})}{T}\right)\\
k_2(\omega_{\alpha_2},\omega_{\beta_2}) =& \alpha_2(\omega_{\alpha_2})\, exp\left(\frac{-\beta_2(\omega_{\beta_2})}{T}\right)
\end{aligned}
\tag{7.1}
$$

The uncertain kinetic parameters are distributed as:

- $\alpha_1(\omega_{\alpha_1}) \sim N(4,0.5)$ s^{-1}
- $\alpha_2(\omega_{\alpha_2}) \sim N(2.1,0.5)$ s^{-1}
- $\beta_1(\omega_{\beta_1}) \sim N(700,20)$ K
- $\beta_2(\omega_{\beta_2}) \sim N(525,50)$ K

.

The species concentrations are then modeled by the following Ordinary Differential Equations:

$$
\begin{aligned}
\frac{dA}{dt} =& -k_1 A\\
\frac{dB}{dt} =& k_1 A - k_2 B\\
\frac{dC}{dt} =& k_2 B
\end{aligned}
$$

Only species A and B can be measured, with predicted observations distributed as additive Gaussian noise:

$$
\begin{aligned}
D^A =& \mathcal{M}[\alpha_1,\beta_1,t,T] + \varepsilon\\
D^B =& \mathcal{M}[\alpha_1,\beta_1,\alpha_2,\beta_2,t,T] + \varepsilon\\
\varepsilon \sim& N(0,0.01)
\end{aligned}
\tag{7.2}
$$

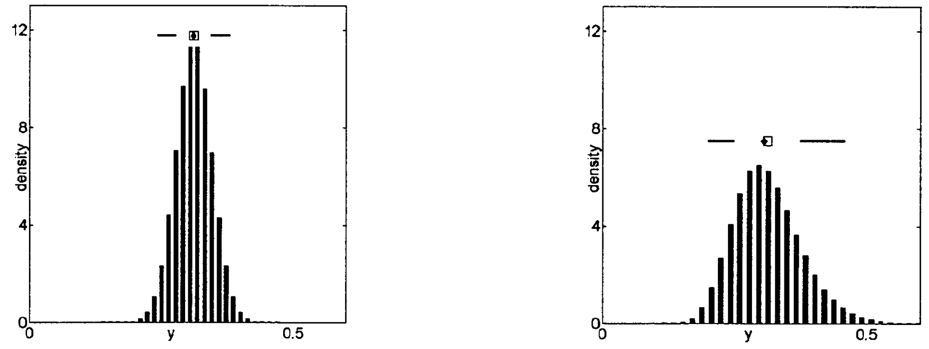
With the same dimensionless scaling as Species A and B. The (ω)'s are dropped but remember that the datasets, parameters, and errors are all Random Variables.

As is often the case in this textbook sequential reactions problem, we are interested in maximizing the production of Species B. The production will take place in a plug flow reactor, which behaves similarly to a batch reactor. Due to other process details, we are constrained to operate at 500 K but are free to set the residence time in the production reactor. The goal here is to determine the best residence time through experiments.

The design space is two dimensional with time and Temperature being selected: $t \in [1, 6]$ s and $T \in [300, 700]$ K.

7.1.2 Analysis of Prior Knowledge

Figure 3-3 shows the concentration profiles of all three species versus time, at specified temperature and parameter values. The formulation of this problem with uncertainty was described in Section 3.5.2 for output B, and the results of uncertainty quantification are shown in Figure 3-21. The example here is slightly more complicated because a fifth parameter A_0 has been added, but it does not appreciably change the figures so they are not reproduced here. The uncertain profiles are constructed using a series of model output densities. Each model output density is a vertical slice of the uncertain concentration profile, representing the uncertainty of Species B at a constant time and Temperature. Two examples are shown for specified times in Figure 7-1.



(a) *time* = 0.5 s and *Temp* = 500 K

(b) *time* = 2 s and *Temp* = 500 K

Figure 7-1: Prior Predictive Density of Species B

Although each of the three species has an uncertain concentration for each time and temperature, they are also related to each other through the mass balance. No matter what the true parameter values may be, we know that $A + B + C = 1$. Unfortunately, there is also uncertainty in the observations, so the total observed mass is also uncertain.

7.1.3 Bayesian Design of Experiments

The goal here is to select the time and temperature of the single best experiment. Best is defined later on. This is a small design with only two observations and a system with five unknown parameters, but for illustrative purposes optimizing two variables is easiest. Increasing the number of experiments only changes the size of the problem; the algorithm is exactly the same. The benefit of having only two design variables (time and temperature of the single experiment) is that a simple 2D grid search can be used instead of a true optimization algorithm.

At each grid point, the time and temperature are specified. 512 Monte Carlo samples (of five values) were taken from the joint prior parameter density and the 512 corresponding model outputs A and B were computed. Then 1024 samples (one for each observation of A and B) were taken from the observation model's error distribution, shown in Equation 7.2. This results in 512 datasets sampled from $D\left(\omega_D\right)$ - each with two observations, A and B.

For every dataset, Bayesian Parameter Estimation was run using Markov Chain Monte Carlo with 10^5 samples. For this method, we must be able to look up the prior probability density for each parameter and compute the dataset likelihood. The first is simple because the prior knowledge is represented with well known probability density functions. The second is also relatively simple because of the additive Gaussian observation model. For this type of observation model the only required information is the model-data discrepancy, i.e., the residual:

$$\varepsilon = \mathcal{M}\left[\alpha_1, \beta_1, \alpha_2, \beta_2, t, T\right] - D$$

Then, the likelihood is determined from the probability density function of this residual, given in Equation 7.2. Using the prior and the likelihood, the Markov Chain Monte Carlo algorithm samples from joint posterior parameter density, $f_{\Theta|D}\left(\theta|\mathsf{d}\right)$. At this point, any utility function can be applied.

The utility function must be related to the goals of the study - to determine the best residence time for production in a plug flow reactor. This corresponds to the time with maximum concentration of B, t_{max}. Without uncertainty, this is easy and the solutions are shown in Example 8. With uncertainty, the best reaction time cannot be determined exactly but can be described with the Random Variable, $T_{max}\left(\omega\right)$. The chosen utility function for this example is the differential entropy of the time of maximum concentration of Species B. If the uncertainty represented by $T_{max}(\omega)$ is reduced through experiments, then a reactor can be designed for optimal production of B.

Each Markov Chain is a set of parameter samples. For every sample, the kinetic coefficients k_1 and k_2 and the time of maximum B concentration can be determined, as in Equations 7.1 and 3.4 and shown in Figure 3-4b. Therefore, every Markov Chain can be used to create a distribution of maximum times $T_{max}(\omega)$. The utility function reduces each T_{max} density to a statistic: the negative differential entropy, for a total of 512 samples of utility. These are samples from the utility distribution. The risk metric is the expected value and the objective is to minimize the expected value.

To summarize – the goal of the study is to maximize the expected information gain about the optimal reaction time.

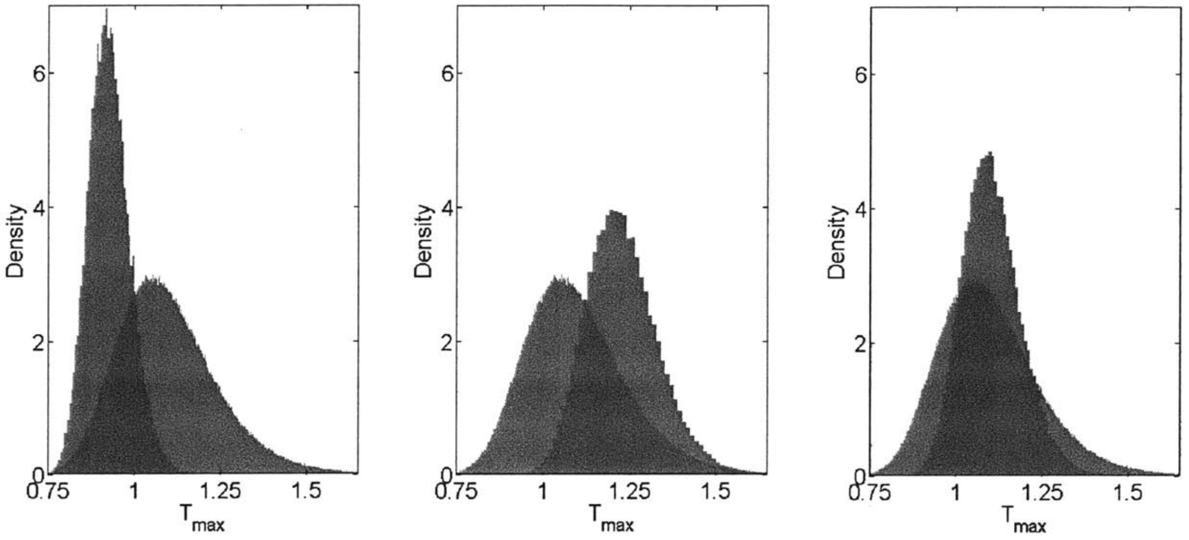
Visualization

For reference the prior information about T_{max} is shown in Figure 7-2 compared with three posteriors. The prior is constant for all simulated experiments and has a differential entropy of $h = -0.26$. All three posteriors result from simulating experiments at the same conditions ($t = 2\,\mathrm{s}$ and $T = 300\,\mathrm{K}$) but with different simulated datasets. Each results in a different utility. After many datasets and simulated, the posteriors are estimated, and utilities are computed. Then the average utility can is used as the risk metric.

7.1.4 Results and Discussion

Two grid searches were performed to find the best design. The first spanned the entire design space and used fewer simulated datasets. The second spanned a smaller region and used the values stated above. From these searches, we find that the best experiment for identifying optimal production conditions will be near 4.2 s and 350 K. The objective function is shown as a function of time and temperature in Figure 7-3. For consistency it is posed as a minimization problem. Instead of maximizing the utility, we are minimizing the posterior differential entropy.

While the objective function tells us the expected decrease in entropy, it is difficult to related this to a meaningful posterior T_{max} density. The objective function is an expectation over all possible posteriors. Although we cannot describe what the posterior T_{max} density will look like,



(a) $h = -3.55$

(b) $h = -0.74$

(c) $h = -1.54$

Figure 7-2: Lognormal prior T_{max} density ($h = -0.26$) compared with three posteriors from different datasets using the same experimental design - each has different entropy h

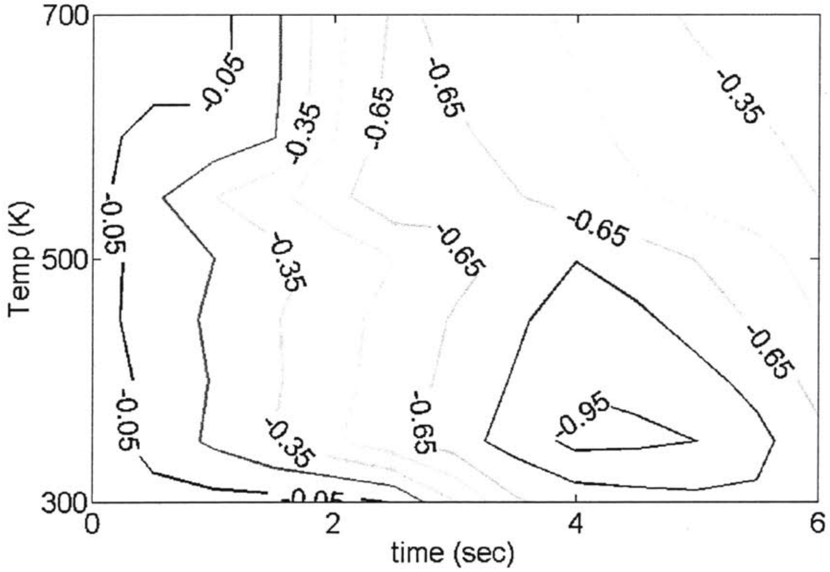


Figure 7-3: Objective function for the sequential reactions design of experiments problem

we expect that the optimal experiment will result in the largest reduction in entropy.

This example describes in detail how to perform each step in Bayesian Design of Experiments. It also demonstrates an interesting utility function that tailors the experiment to a practical goal. In this example, there are two issues which are not addressed: numerical accuracy and optimization.

First of all, the numbers of samples were chosen rather arbitrarily. The choices of how many samples to take from the Markov chain and the utility distribution are a compromise between computation time and accuracy. It is important to choose an appropriate number. This is discussed in more depth in Chapter 9. In addition, the choice of utility function was convenient because it depends only on the one-dimensional density of $T_{max}(\omega)$. The posterior parameter density is five-dimensional, and a utility function on this joint density would be difficult to compute with any accuracy. This motivates the use of a prior sampling formulation.

Secondly, the best design was found using a grid search. This quickly becomes infeasible as the number of design variables and experiments increases. This will be addressed in other examples.

7.2 Discrimination Between Linear Models

The second example demonstrates Bayesian Design of Experiments using the prior sampling formulation, rather than Markov Chain Monte Carlo. The application is model discrimination between three generic, linear models.

7.2.1 Three Competing Models

Three models have been proposed to explain some phenomenon. They are:

$$y = \alpha_1 + \alpha_2 x \tag{7.3}$$

$$y = \beta_1 + \beta_2 x \tag{7.4}$$

$$y = \theta_1 + \theta_2 x + \theta_3 x^2 \tag{7.5}$$

where $\alpha \sim \begin{bmatrix} N(1.1, 0.1^2) \\ N(-0.1, 0.05^2) \end{bmatrix}$, $\beta \sim \begin{bmatrix} N(7.5, 0.075^2) \\ N(0.1, 0.05^2) \end{bmatrix}$, and $\theta \sim \begin{bmatrix} N(0.75, 0.075^2) \\ N(0.25, 0.05^2) \\ N(0.05, 0.05^2) \end{bmatrix}$.

.

The models, α, β, and θ, are shown in Figure with all parameters at their mean values.

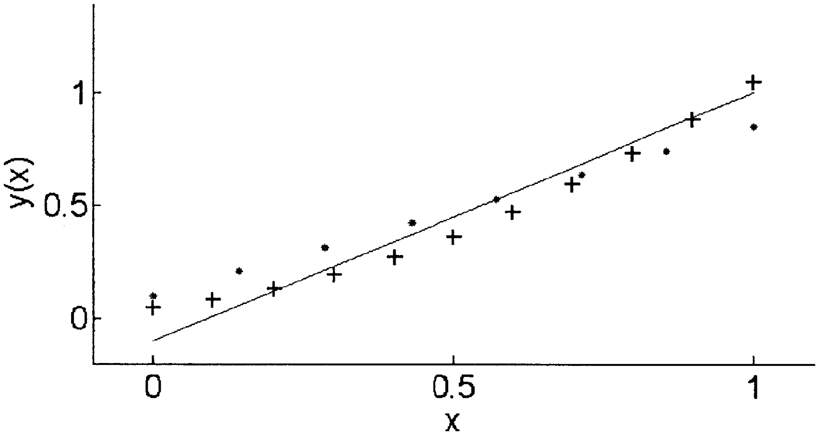


Figure 7-4: Three models: α —, β $\cdots$, and θ ++, with all parameters at their mean values

All three models are very similar to each other in the domain of interest. In addition, when the uncertain parameters are taken into account, the model output densities and data prediction densities largely overlap. This makes it difficult to distinguish the models. We would like to design a study to allow us to determine whether one model is superior to the others.

In this example there is only one design variable, x, and one uncertain output for each system model - $Y^{model}(\omega; x)$. The observation model is additive Gaussian errors:

$$\varepsilon \sim N\left(0, 0.05^2\right) \tag{7.6}$$

7.2.2 Analysis of Prior Knowledge

The prior knowledge of the parameters and observations can be analyzed as in the previous example. Each model is linear and so the uncertainty profiles will look very similar to the first order system in Example 17 and the second order system in Example 19. Unfortunately, it is difficult to develop an intuition for which experiments will give the best results. To illustrate the difficultly, all three model output densities are plotted together in Figure 7-5 for $x = 0.4$.

When all the models are treated as submodels and combined into a hierarchical model, they must be given prior submodel probabilities. Here they are all weighted the same: $\frac{1}{3}$. Then the hierarchical model's output uncertainty is described by the density in Figure 7-6. This is not a convolution; it is a normalized sum of weighted densities.

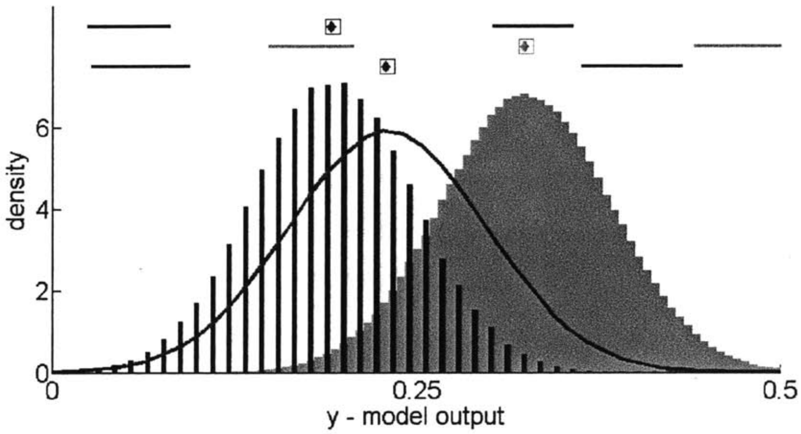


Figure 7-5: Three model output densities at $x = 0.4$, α —, β, shaded, θ bars

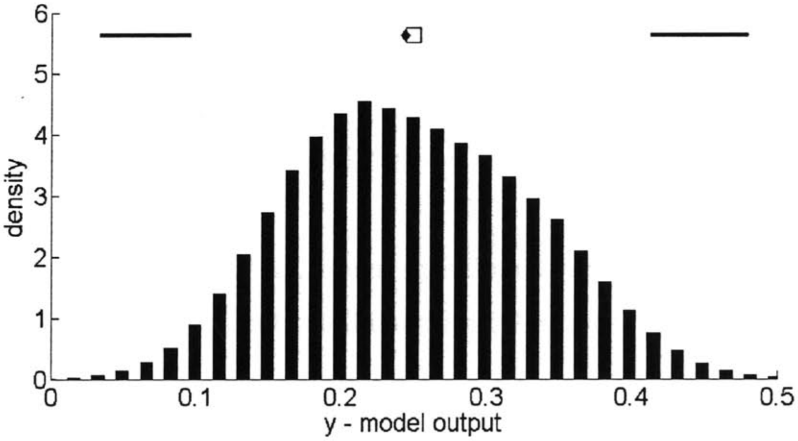


Figure 7-6: The model output uncertainty of hierarchical model, with uniform weighting of all submodels

7.2.3 Bayesian Design of Experiments

Because the submodels are simple and there is only one design point, several designs were created. The algorithm is shown in Section 6.4.2. More details are given here for a one-experiment design and the procedure was repeated for designs with up to six experiments. For the number of samples used (N_i, N_j, and N_k) the objective function was accurate enough to use optimization algorithms. The Implicit Filter algorithm [46, 11], version 1.0 was downloaded from the website of Kelley at North Carolina State University and was used with the default options. For designs with four to six experiments, the optimization problem was run several times with random starting points. This still does not guarantee a global optimum, but does increase confidence in the solution.

Within the optimization algorithm, $N_j = 5 \times 10^3$ datasets were simulated by sampling from the prior submodel probability mass function, sampling from the submodel's prior parameter density, then adding a sample from the error density. For the design $x = 0.4$, this corresponds to the samples from Figure 7-6, plus Gaussian error according to Equation 7.6.

For every model, we compute the likelihood of observing each dataset – assuming the model is correct. This must be done by averaging over the prior parameter density, so $N_i = 2 \times 10^4$ parameter samples were taken for each model, and the likelihood calculated as the probability density of the error model, with an error of:

$$\varepsilon = \mathsf{d}_j - \mathcal{M}\left[\mathsf{m}_i, \theta_k, x\right]$$

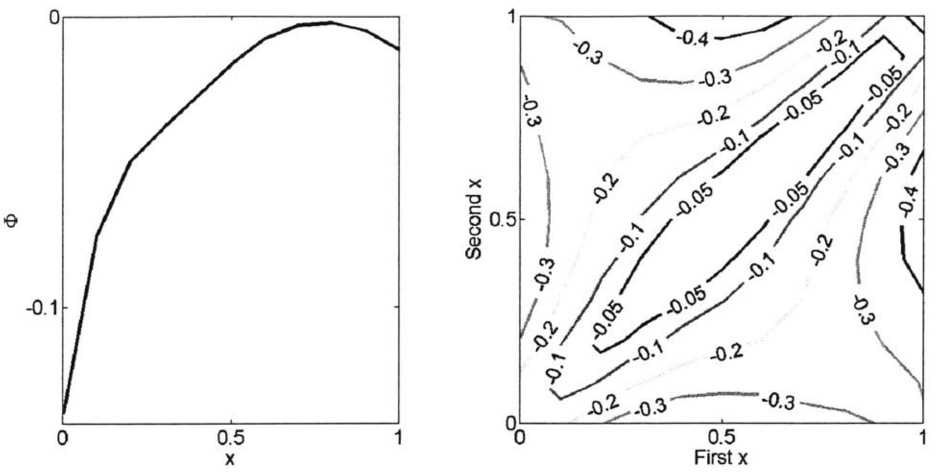
This is computed a total of $N_j N_k N_{mod} N_x = 3 \times 10^8$ times, where N_x is the number of experiments in design x.

All the required terms for the Bayesian Design algorithm can now be computed. Again for consistency, the objective function is formulated for minimization so it is the negative of the expected utility $-\Phi$.

7.2.4 Results and Discussion

The objective functions for one- and two-experiment designs are shown in Figure 7-7 as a function of the design variable x.

The shape of the one-dimensional objective function in Figure 7-7a is interesting because of its



(a) One Experiment

(b) Two Experiments

Figure 7-7: Objective Functions for discriminating between three linear models

maximum and local minima. The Kullback-Leibler divergence is near zero when the two distributions being compared are very similar and increases when the distributions are different. This indicates that an experiment at $x = 0.7$ would be of no benefit for discriminating between the three models since there is no change between the prior and posterior submodel probabilities. The minima are at the boundaries, with $x = 0$ being far superior to $x = 1$. These results are supported by Figure 7-4, where all three models have similar values at $x = 0.7$ but at the bounds at least one model is distinguishable from the other two. Without analyzing the uncertainties, this is an incomplete picture and cannot explain why experiments at $x = 0$ are better. Referring to the examples in Section 5.2, we see that for these linear models the uncertainty grows farther from $x = 0$. This explains why $x = 1$ is not as attractive an experiment as $x = 0$. With high data prediction uncertainty, like at $x = 1$, there is a higher probability that each model could explain the data collected from experiments. At $x = 0$, the uncertainties are smaller and therefore experimental data is more likely to distinguish between submodels.

The two-experiment objective function is shown in Figure 7-7b. The two axes represent the same variable x, so the objective function is symmetrical around the line from $[0, 0]$ to $[1, 1]$. Along this line, the same experiment is repeated twice and the objective function closely resembles the one-experiment objective function, with local minima at $[0, 0]$ and $[1, 1]$. The maximum is unclear in the plot but it occurs near $[0.7, 0.7]$.

The two-experiment design is interesting because it does *not* include $x = 0$. This is because the one-experiment design does not allow estimation of parameter correlations and so the submodels are not significantly improved by the experiment. The single point of data can only serve to distinguish the submodels. In a two-experiment design, experiments serve to improve the model parameters and distinguish between models. The combination of $x = 1$ and $x = 0.5$ is the best for performing both tasks. Also, see that the objective function for the optimal two-experiment design is about 3× that of the one-experiment design. The absolute numbers are not significant, however, this shows that the second experiment is very important for distinguishing between models.

Table 7.1: Bayesian optimal designs for discriminating between three linear models, also showing the information metric expected Kullback-Leibler Divergence from posterior to prior submodel probability mass functions

Design Points x	Φ
0	0.14
$0.5, 1$	0.47
$0, 0.5, 1$	0.77
$0, 0.5, 0.5, 1$	0.85
$0, 0, 0.5, 0.5, 1$	0.87
$0, 0, 0.5, 0.5, 0.5, 1$	0.91

Optimal designs with additional experiments are shown in Table 7.1. After seeing the examples in Section 5.2, an obvious approach might be to place experiments uniformly at $x = [0, 0.5, 1]$. This would be the result from the Classical or Optimal Design approaches. For model discrimination using Bayesian Design of Experiments, the results are different. Instead of running experiments at all three points uniformly, more experiments are needed at the middle point. This shows the importance of simulating the entire experimental process; the experiments in Table 7.1 are customized for this particular system and the given parametric uncertainties. Another interesting point is that while there is a large marginal gain in expected utility for the first three experiments, additional experiments do not yield the same benefits. This information helps to assess the value of running experiments.

7.3 Conclusion

The examples in this chapter show the details of the Bayesian Design of Experiments algorithm. The simple examples illustrate how characterizing uncertainty can reveal important features of the system and described how the results of the algorithm can be interpreted. In the following chapters, Bayesian Design of Experiments are applied to larger, more complex systems.

Chapter 8

Study 1: Air Mill and Classifier

This study provides a direct comparison of the Classical and Bayesian approaches to design of experiment. It explains and quantifies the real-world impact of incorporating knowledge of the physical process and uncertainties into the design phase. A previous study has been performed on the physical system using Classical Design of Experiments. The results are compared to a Bayesian approach using simulations. Both approaches were evaluated based on their ability to estimate system performance and their economic and time costs.

8.1 Background and Previous Work

An air mill classifier is a downstream unit in a pharmaceutical production process. Its role is to homogenize particles. The unit processes a continuous feed of drug product in the form of a coarse powder. The air mill classifier reduces the particle size and controls the size distribution of the exiting particles with the goal of producing a uniformly fine powder. The production scale unit used in this study was a Hosokawa Alpine AFB-400, while pilot scale studies were carried out on a Hosokawa Alpine AFB-100.

8.1.1 Process Description

An air mill classifier can be thought of as two distinct units: a fluidized bed where the milling occurs and a classifier which handles the separation. A diagram of the equipment is shown in Figure 8-1 along with a picture of the pilot scale unit. A process flowsheet is shown in Figure 8-2.

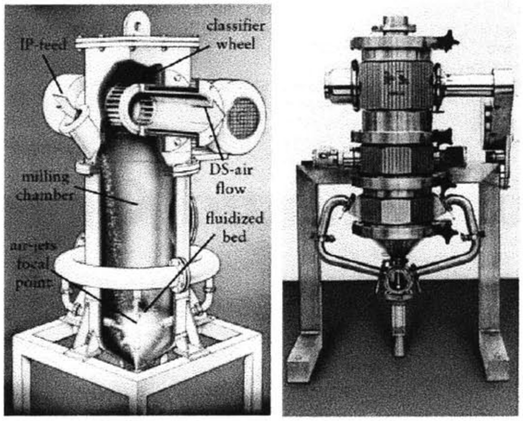


Figure 8-1: Diagram and Picture of a Hosokawa Alpine AFG-100 air mill and classifier

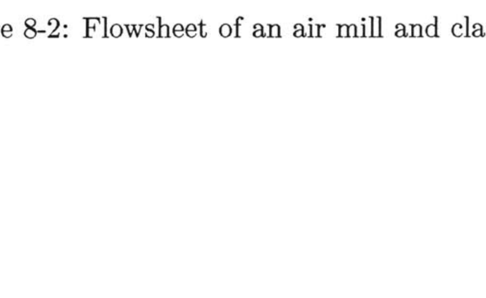


Figure 8-2: Flowsheet of an air mill and classifier

The fluidized bed is responsible for breaking down particles and transferring them to the classifier. The contents circulate throughout the bed and at the bottom supersonic air jets create high velocity streams of particles which impact each other and cause particle breakage. At the top of the bed is the entrance to the classifier section, where particles can be pushed into the classifier.

The classifier is responsible for controlling the particle size distribution of the product. This is accomplished by two opposing forces: drag and centrifugal 'force'. The classifier wheel forces particles to spin in a vortex down a horizontal tube. Particles smaller than a certain radius are forced toward the axis of the tube and released, while larger particles are forced radially outward and recycled to the fluidized bed.

8.1.2 Previous Classically Designed Studies

Over the course of a year, an industrial partner completed dozens of experiments on both pilot and production scale air mill classifiers in order to characterize their performance with a specific drug product. The company performed their previous experiments in a series of four studies: a high level assessment, screening and optimized studies at the pilot scale, and a final study at production scale. The purpose of these studies was to develop a production scale model to accurately predict two system responses: X_{50} and X_{90}, the 50^{th} and 90^{th} percentiles of product particles, by radius. The overarching goal of these studies was to determine the optimal operating conditions for large scale production. The two system responses give a quantitative measure of quality of the powder product and have strict regulatory targets that must be met. The product specifications for these were $X_{50} < 6\,\mu\text{m}$ and $X_{90} < 20\,\mu\text{m}$. More details about the previous studies are given in the following section.

8.1.3 Setup of the Current Study

This current study was intended to compare Classical and Bayesian Designs. We approached the problem as though we had no knowledge of the previous studies, relying instead on a baseline knowledge of the process and equipment. The study was built around a physics-based model which simulates the two units that make up the air mill classifier.

The model based approach essentially enables the experimenter to bypass the bulk of the scale-up process. The purpose of pilot scale experiments is to develop predictive power at the production scale. Rather than use experimental studies to develop a surface response map of the pilot scale system, a physics-based model was created without running any experiments. The model provides the predictive ability just like an empirical surface response map. More importantly, it allows the experimenter to quantify and understand the impact of uncertainties. Knowledge of the model uncertainties is crucial when performing extrapolations, especially scale up of processes. In this way, the model predictions can then replace and even improve upon the Classically Designed experimental studies, eliminating a large number of experiments and saving a great deal of time and money.

8.2 Classical Design of Experiments

As previously mentioned, four studies using Classical Designs were carried out in series. The initial assessment was used to narrow down the number of experimental factors from dozens to just two. Next, two studies were done at the pilot scale: a screening study to determine the suitable range of operating condition, and an optimized study in the suitable range. Finally, the two significant factors were tested again at the production scale with screening and optimized studies. At both scales, the data from optimized classical designs was used to develop an empirical fit with polynomial terms of the two significant factors.

8.2.1 Initial Assessment Study

The initial assessment identified 58 factors which could influence the system performance. Using lab scale tests and the input of industry experts, these were narrowed to four: the particle size distribution of the feed, the feedrate, the nozzle pressure in the milling section, and the rotation speed of the classifier.

8.2.2 Pilot Scale Studies

To determine the impact of these four factors, a screening study was carried out at pilot scale. This involved a 2-level Full Factorial design, plus four center point experiments for a total of $2^4 + 4 = 20$ experiments. Additional, undocumented experiments were also run for measurement calibration. Each experiment ran for an unknown time at a rate of $3\,\mathrm{kg\,h^{-1}}$, for economic modeling it was assumed that the test length was $4\,\mathrm{h}$.

The results indicated that while all factors had significant impact on the system performance, the classifier rotation speed and nozzle pressure were much more significant, and of these two the classifier rotation was a much larger influence. Therefore, future studies only varied these two factors.

After the first screening study was complete, the optimized study was run to create an empirical process model. The study used a Central Composite design with 22 experiments. The results were to two, seven-term empirical models for the two system responses of interest: X_{50} and X_{90}.

8.2.3 Production Scale Studies

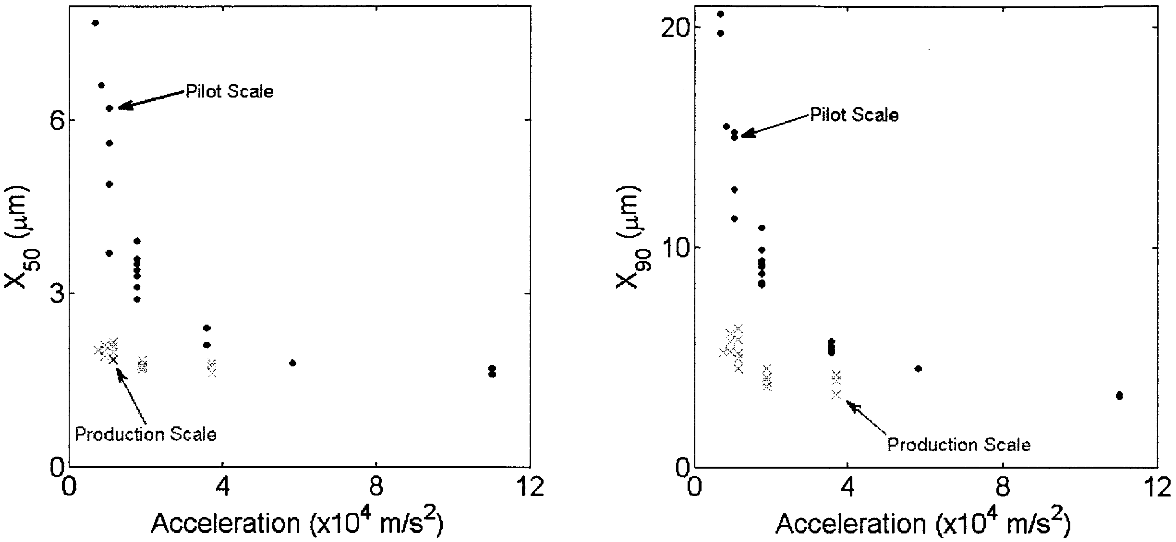
Using the model from the pilot scale, a fourth study was designed for the production scale using a Central Composite design with 16 experiments. This was run in order to develop production scale empirical models for particle size distribution. Each experiment ran for 4 to 8 h at a rate of 50 kg h^{-1}. The factors were the same and the ranges were determined using the pilot scale model and scale up techniques. The intention was to provide the same environment in the production scale classifier as the pilot scale classifier by decreasing the rotation speed to offset the effects of increased radius. In effect, the particles would experience the same centrifugal force.

8.2.4 Results and Discussion

The experiments at pilot and production scale varied both nozzle pressure and classifier rotation. Figure 8-3 shows the results from the optimized pilot scale study and the final production scale study. Here the x-axis shows the centrifugal acceleration felt by each particle due to the classifier rotation. All the data was collapsed onto the acceleration-particle size plane. The variation at each classifier speed is largely due to each data point having different nozzle pressure.

The first issue here is that the production scale performance was different than predicted by the pilot scale model. The particles in the product are much smaller than the corresponding product from pilot scale. This is a failure in scale up, caused by a lack of understanding the physics of the system.

Another issue with the empirical models is the lack of uncertainty analysis. Uncertainty in the parameters is not meaningful because they are not based on any physical principles, however quantifying the uncertainty in the model outputs would have provided useful insight into the discrepancy



(a) X_{50} data (b) X_{90} data

Figure 8-3: Particle size distribution versus centrifugal acceleration for pilot and production scales showing distinctly different trends although the scale up model predicted comparable performance

between model outputs and data.

8.3 Bayesian Designs – Building Models

Two models were used: a first principles model based only on dimensional analysis and a more rigorous model using first principles, empirical industry knowledge, and data from literature. The dimensional analysis model allowed us to identify the significant parameters and design variables that impact the particle size in the air mill and separations efficiency in the classifier. This information was then incorporated into the more rigorous model, which was used for the design of experiments. The model contains constants that adjust for pilot scale or production scale.

8.3.1 Dimensional Analysis Model

A separate dimensional analysis model was created for each of the units.

Fluidized Bed

The main particle interactions in the fluidized bed are collisions, which impart energy to break the particles. The dimensionless group of importance here is:

$$\frac{m_p {v_{fb}}^2}{W_p {r_p}^3} = \frac{\rho_p {v_{fb}}^2}{W_p}$$

where

- m_p is particle mass
- r_p is particle radius
- v_{fb} is air velocity in the fluidized bed impact zone
- W_p is the specific particle breakage energy (energy per volume)
- ρ_p is particle density

Particle interactions were analyzed based on the energy provided by the air mill and the energy required to break the particles. This provides a guideline for whether the particles will reduce in size when there is an impact.

Classifier

Separation is based on two competing forces: opposing drag force and centrifugal force. They are compared using a dimensionless group:

$$\frac{\frac{m}{r_p{}^3} v_{cl}{}^2 r_p{}^2}{m r_{cl} \omega^2} = \frac{v_{cl}{}^2}{r_p r_{cl} \omega^2}$$

where

- m is particle mass
- r_p is particle radius
- v_{cl} is the radial velocity of air entering the classifier
- r_{cl} is the radius of the classifier
- ω is the rotation speed of the classifier in radians

This dimensionless group gives an idea of the scales of the drag force and centrifugal force, but makes many assumptions which will not hold in the actual unit. For instance, the airflow will likely be turbulent but this assumes a stationary, spherical particle within laminar flowing air. Also, air velocities will vary with radial and axial position. Nevertheless, it is a fine starting point for our analysis.

Analysis

These two dimensionless groups were used to relate the opposing forces that drive system performance. These include:

- Air velocity in the fluidized bed impact zone
- Specific particle breakage energy
- Particle size
- Airflow through the classifier
- Classifier rotation

These terms were then related to the design variables available on the system.

Based on the ideal gas law, the exit velocity from nozzles in the fluidized bed should be a related to nozzle pressure drop. From this relationship,

$$
\begin{aligned}
v_{fb} =& A\Delta P + B \\
v_{cl} =& C v_{fb}
\end{aligned}
$$

where A, B, C are constants, and

$$\frac{r_p}{r_p^{initial}} = f\left(\frac{\rho_p(\Delta P + C_1)^2}{W}, \frac{(\Delta P + C_2)^2}{r_p r_{cl} \omega^2}\right)$$

From this simple analysis, the general trends of the system responses can be predicted – notably a dependence on the inverse square of classifier rotation, some polynomial dependence on the nozzle pressure drop, and some scaling with initial particle size distribution. There is nothing in the physical interactions of particles that relates feed rate to the responses.

8.3.2 Development of a Rigorous Model

The physical interactions described in the above analysis were modeled in detail to try to develop the functional form of the system response. The model structure was based upon the above dimensional analysis and literature models [45, 83, 82, 84].

Model Operation and Output

The purpose of the model was to take a known feed stream and process conditions and output the steady-state particle size distribution. In order to calculate this, the model simulated the process startup from zero holdup until the exit volume was equal the feed volume. The model represented the air mill classifier as the two unit system shown in Figure 8-2. There are two functions of the fluidized bed unit: particle breakage, and transfer to the classifier. The classifier unit calculates the fraction of particles that are released; the rest are recycled to the fluidized bed.

At each time point the operation of each unit is calculated and the population is balanced. The entire contents of the classifier turn over at every time point, either being released from the system or recycled back to the fluidized bed. At each time point, the fluidized bed receives a feed stream and recycle stream, then the population undergoes breakage and a fraction is transferred to the classifier. Once the exit stream has the same volume flowrate as the feed, the system is at steady state.

Particle Population

In order to simplify the system, the population was characterized using radius bins, which each track the number of particles with certain sizes. Bins were grouped into four collections: large, medium, small, and extra small. Within each collection, bins were delimited by linearly increasing particle volume. Particle density was assumed to be constant, so this was effectively the same as a mass balance.

The extra small collection bins ranged from volumes of $\frac{V_{\max}}{N^4} \sim \frac{V_{\max}}{N^3}$. The small collection bins were $\frac{V_{\max}}{N^3} \sim \frac{V_{\max}}{N^2}$, medium collection bins $\frac{V_{\max}}{N^2} \sim \frac{V_{\max}}{N}$, and large bins$\frac{V_{\max}}{N} \sim V_{\max}$. This method of linearly increasing volume allowed the use of collection index instead of volume for mass balances. For example, within a collection the volume of bin 25 is equal to the sum of volumes of bin 20 and bin 5. The use of multiple collections allows for better resolution of particles with small radii. This is important because the particle sizes range from $< 1\,\mu\text{m}$ to $> 250\,\mu\text{m}$, but most of the interest was in particles $< 25\,\mu\text{m}$.

8.3.3 Feed

The feed was a powder of intermediate drug product with a large spread of sizes. To model this, a lognormal distribution was used to approximate the particle size distribution of the feed. The lognormal parameters were fitted to available data, as shown in Figure 8-4.

To allow the feed distribution to be varied within the model, one of the independent variables *feedadjust* set the first parameter of the lognormal distribution in order to change the mean and variance. The lognormal distribution representing the feed was then used to determine the number of particles entering each bin of the rigorous model.

Fluidized Bed

There were four terms in the model's population balance: feed, breakage, transfer to the classifier, and recycle from the classifier. The recycle was determined by the classifier, the feed was specified, and the breakage and transfer depend on the holdup in the fluidized bed.

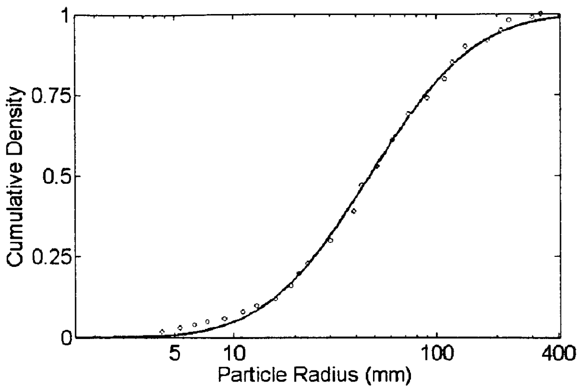


Figure 8-4: Particle Size Cumulative Distribution in the Feed Stream - Data (red circles) and fitted Lognormal distribution (black line)

The breakage was treated as a system of reactions between particles in the fluidized bed. In order to limit the number of possible reactions, the model used single particle breakage instead of two particle collisions. For example, a particle in bin Large 25 could be split into two smaller particles. This is represented by:

$$L_{25} \rightarrow L_{12} + L_{13}$$

Then the breakage reactions, at time step i would be:

$$\begin{aligned}\left[L_{25}^{(i)}\right] &= \left[L_{25}^{(i-1)}\right] - k_{25/12,13}\left[L_{25}^{(i-1)}\right]\\ \left[L_{12}^{(i)}\right] &= \left[L_{12}^{(i-1)}\right] + k_{25/12,13}\left[L_{25}^{(i-1)}\right]\\ \left[L_{13}^{(i)}\right] &= \left[L_{13}^{(i-1)}\right] + k_{25/12,13}\left[L_{25}^{(i-1)}\right]\end{aligned}$$

where k is similar to a rate constant. It describes the fraction of particles of size Large 25 that break into particles of size Large 12 and Large 13, at each timestep. This ensures that the total volume remains balanced. Unfortunately it was difficult to determine what the rate constants should be. An additional simplification is that the possible products are restricted. For each 'reactant' there is a minimum 'product' size. For example, particles from bin 200 can only split into particles in bins 100-199. The particles are distributed evenly into the eligible bins. This means that in one breakage step, the majority of the mass remains in larger particles, however, the smaller particles accumulate in number, which is reflected in relatively low values for the 50th and 90th percentile (less than $30\mu m$).

In this model, a fraction of each bin is fractured in each time step (the 'conversion'), which is determined by the function:

$$Conv = 1 - \exp\left(\left(\frac{-ax^{\alpha}}{1+\left(\frac{x}{\mu}\right)^{\beta}} - \gamma\right)\Delta t\right)$$

This depends on five fitting parameters, $a, \alpha, \beta, \mu, \gamma$ and the time step. Literature indicated that nozzle pressure and an intrinsic particle strength parameter should control particle breakage, however, no specific correlation was found in the literature so this empirical equation was used. Once the conversion is calculated, the rate constants of the allowable reactions are set so that they sum to the correct conversion. The second function of the fluidized bed is the transfer of the contents to the classifier. The first step was to calculate fluidization velocity for each particle radius. This assumes each particle is a constant density sphere, and calculates the velocity at which upward drag force balances with gravity. The drag equation and a correlation for drag coefficient were used:

- $F_{drag} = \frac{1}{2}\rho_{air}v^2 C_f A$,
- ρ_{air} is the air density
- v is the air velocity
- $C_f = \frac{1}{3}\left(\sqrt{\frac{72}{\mathrm{Re}}} + 1\right)^2$ is the drag coefficient, for Re in transition region $\left(10^2, 10^3\right)$
- A is the projected area $\left(\pi {r_p}^2\right)$
- $F_{gravity} = \rho_{particle}\left(\frac{4}{3}\pi {r_p}^3\right) g$

At each time point, the air velocity through the fluidized bed was calculated, and the compared to the fluidization velocities for each particle radius. The air velocity depended on the nozzle pressure and the amount of material in the fluidized bed, which decreased the void spaces where air could flow. If the air velocity was greater than the fluidization velocity, all particles of that radius were transferred to the classifier. Otherwise, a smaller fraction of particles were transferred depending on the ratio of air velocity to fluidization velocity.

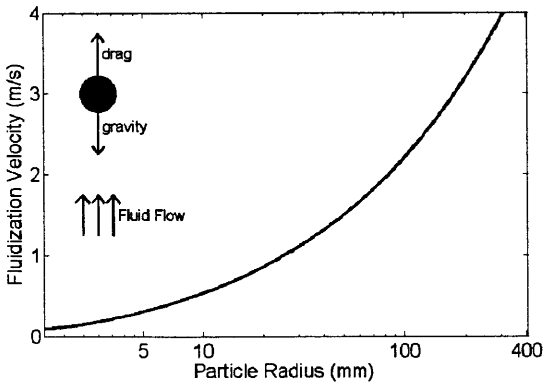


Figure 8-5: Fluidization velocities of various particle sizes based on empirical drag correlations versus gravity

Classifier

The classifier is responsible for controlling the particle size distribution of the product. The classifier is composed of a deflector wheel and a tube. The wheel has radius r_0 and is closed at the front axial end and open at the back end. Instead of walls, the deflector wheel has a set of rectangular fins as seen in Figure 8-1. The tube section is just a cylinder open at the front end, where it meets the wheel, and the back end is separated into two parts, an inner opening of radius r_{crit}, and the outer opening, of $r_{crit} < r < r_d$.

Air entering the cylinder continues to vortex and creates a centrifugal force which carries particles radially outward – orthogonal to the axial motion. The vortexing air and particles travel down the tube, giving the particles time to separate based on size. The physical separation takes place at the axial end of the tube in which smaller particles are released inner opening at the back end, and larger particles are rejected back into the fluidized bed through the outer opening.

The high pressure in the fluidized bed forces air into the classifier radially inward, through the deflector wheel. The deflector wheel spins about the cylinder's axis, creating a cyclone. The only outlet is at the back end of the cylinder, which forces the air in the axial direction into the tube. The vortexing air and particles travel down the tube, giving the particles time to separate based on size. The physical separation takes place at the axial end of the tube in which smaller particles are released inner opening at the back end, and larger particles are rejected back into the fluidized bed through the outer opening. This is shown in Figure 8-6.

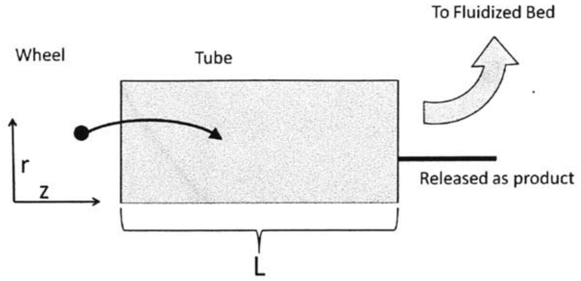


Figure 8-6: The classifier unit - particles enter axially and are vortexed - those that reach the axial end with radial position $r < r_{crit}$ are released, while all others are rejected into the fluidized bed

The model used a finite element method to determine whether a particle would be released. It simplified the particle motion into two dimensions, radial and axial. The axial motion was assumed to be constant. Particles entered the classifier at $z = 0$ and radius r_0. Once there entrance position is set, the residence time in the classifier is determine by the axial velocity. If the particle is in the exit tube $r < r_1$ after one residence time, it is released. Next, the radial position is modeled. The initial velocity is set by the overall airflow through the fluidized bed, which in turn depends on the nozzle pressure drop. The open area along the outer radius of the deflector wheel is known, so the initial velocity is: $v_r^{(0)} = v_{air} = -\frac{\text{airflow}[m^3s^{-1}]}{\text{Deflectorwheelsurfacearea}[m^2]}v_{factor}$ where v_{factor} is a scaling factor

The governing equations within the classifier are:

- $Accel = \frac{F_{cent} - F_{drag}}{mass}$
  - $F_{drag} = \frac{1}{2}\rho_{air}v^2C_fA$
    - $C_f = \frac{1}{3}\left(\sqrt{\frac{72}{\text{Re}}} + 1\right)^2$
    - $\text{Re} = \frac{\rho\left(v_{air} - v_r^{(i-1)}\right)2r_p}{\mu_{air}}$
    - $v_r = v_{r,outer}\left(\frac{r_{outer}}{r^{(i-1)}}\right)^m$
  - $F_{cent} = \rho\left(\frac{4}{3}\pi {r_p}^3\right)\omega r$
- $v_r^{(i)} = v_r^{(i-1)} + Accel^{(i-1)}\Delta t$
- $r^{(i)} = r^{(i-1)} + {v_r}^{(i-1)}\Delta t$

This model is used to determine what fraction of each bin is released, depending on the process conditions and physical parameters. Once these release fractions are determined, they are used for the entire simulation, as opposed to the breakage function, which is updated at each time point. The release function calculates how much of the classifier contents is released, and all the rest is recycled.

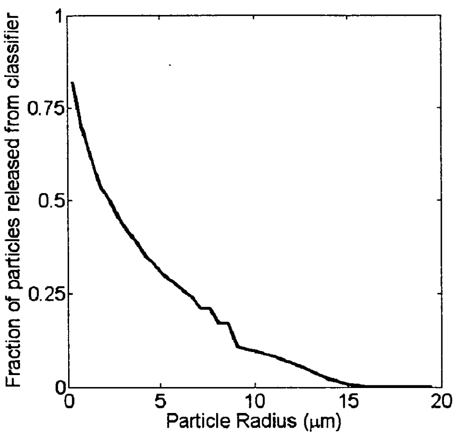


Figure 8-7: The release profile of particles from the classifier - this varies with process conditions

The sharp drop in particle release rate occurs around the critical particle radius, for which the drag force and centrifugal force are balanced. This will depend strongly on process variables. Unfortunately, no process data was available so these results could not be validated.

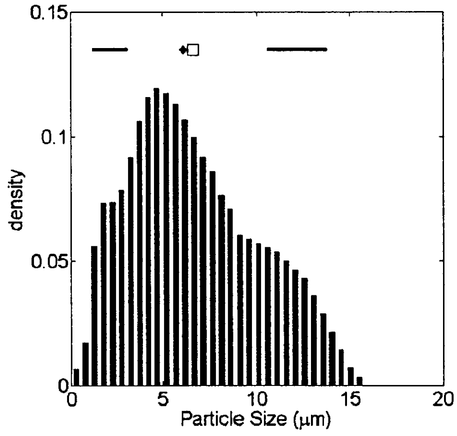


Figure 8-8: A sample distribution of product from the air mill classifier - this varies with process conditions, X_{50} is shown as a diamond

This is the final result of the model, which takes release rate multiplied by the classifier contents at each time point until a steady state is reached. Again, this will distribution with change depending on the process conditions. The model responses X_{50} and X_{90} were calculated from the size distribution of particles released from the classifier.

Table 8.1: Approximating Gaussian Distributions of the Model Parameters

	Mean	Standard Deviation
Vortex Coefficient	0.65	0.1
Velocity Factor	1	0.15
a	5.00E-05	1.00E-05
α	1.2	0.25
β	1.5	0.25
μ	150	25
γ	0.05	0.01

Table 8.2: Initial Ranges of Process Variables

	Min	Max	Mean	Standard Deviation
Nozzle Pressure (bar)	3	6	4	0.4
Classifier RPM (1/min)	4000	20000	16000	1600
Feed Adjust	0.5	1.5	1	0.1

8.3.4 Prior Information

The inputs that were expected to be significant were the design variables: nozzle pressure drop (ΔP), classifier rpm (ω), and feedadjust, as well as physical parameters: vortex flow coefficient (m) initial velocity factor (v_{factor}) and breakage parameters ($a, \alpha, \beta, \mu, \gamma$). The initial values and feasible ranges of the parameters and variables were taken from literature values and equipment specifications. The model parameters are assumed to be uniformly distributed, their means and standard deviations are given in Table 8.1.

The bounds of the process variables for model evaluation were set as in Table 8.2. These were taken from equipment limits suggested by the previous studies.

8.3.5 Sensitivity Analyses

Using these nominal parameter values, a local sensitivity analysis was done on the three process variables. The variables were varied one at a time to determine their impact on the two model responses. The Feed Adjust variable was found not to have significant impact on the responses, so it was changed to a constant.

The same procedure was repeated for parameter sensitivities. These calculations replaced the screening studies done in the pilot scale. Parameters were varied one at a time from their mean to two standard deviations. After this analysis, the parameter uncertainties were changed to those in

Table 8.3: Prior distributions of model parameters used for parameter estimation

	Lower Bound	Upper Bound
Vortex Coefficient	0.5	1
Velocity Factor	0.75	1.25
a	$1e-4$	0.01
α	1	1
β	0	0
μ	1	1
γ	0	0.25

Table 8.3. Three of the parameters were changed to constants.

8.4 Bayesian Designs – Procedure

The design approach used here was not the full Bayesian Design described in Section 5.4. Instead a simplified version, called Model Based Experimental Designs [5] was used. This approach falls within the Bayesian Designs and decision theory framework, but linearizes the models and uses Gaussian uncertainties. A complete description is included in Appendix E.

The goal for this study was to improve the parameter estimates. The design variables were the nozzle pressure and classifier rotation speed. The ranges for the design variables were taken from equipment operating ranges used in previous studies. The observations were of the two system responses X_{50} and X_{90}. This was dictated by the data we had available. We had assumed for this problem that data from previous studies was not available, and this included the prior knowledge of the parameters. Without any data, we had to assume probability density functions for each of the parameters using literature values and best guesses. Due to the use of a simplified algorithm, all uncertainties were Gaussian.

8.4.1 Software Implementation

The study calculations were all done using Matlab and Excel. The study utilized simple parallel processing to speed up calculations. A three level structure was used: the Master, Excel, and Server levels. The Master level was a function which organized the DEMM trials and submitted them to an Excel spreadsheet. The Excel spreadsheet was used to keep track of the trials status and results, and facilitate communication between levels. Finally, the server level was another Matlab function that took inputs from the Excel spreadsheet and ran the classifier model function. This structure allowed the use of multiple processors.

8.4.2 Model Based Experimental Design

The Model Based Experimental Design uses the same modeling assumptions as Optimal Designs. Therefore, it also does not need an explicit dataset simulation or parameter estimation step. Instead the model was linearized at the mean parameter values using numerical approximation of the gradient.

This method was applied sequentially. A one-experiment design was selected using prior information, then data was collected, and parameters updated. Then a second design was selected to compliment the first. In both cases, optimization was done using a grid search. The study was small enough (two process variables) that the entire design space could be mapped out and searched for the maximum.

Once an optimal experiment was selected, data was collected. Because physical experiments were not feasible for this project, a similar dataset was taken from the previous study's data. This was then treated as data 'collected' for the current study. Using the first dataset, Markov Chain Monte Carlo was used for Bayesian parameter estimation. Seven MCMC chains were used to test for convergence, with roughly $1e4$ samples in each. The updated model was then used to select the next design point and the process repeated.

The intention was to iterate the process at pilot scale until the uncertainty at production scale was deemed to be acceptable. Unfortunately, this could not be completed because the proper data was unavailable. For this reason, an incomplete pilot scale model was scaled-up to production scale and two more data points collected. After one final round of parameter estimation, the quality of the physics based model was compared to the empirical model produced from previous studies.

8.5 Results

8.5.1 Selection of the First Experiment

The Model Based Experimental Design algorithm was used to selected a single experiment. The result indicated that experiments should be run at slow classifier rotations and high nozzle pressure. Two data points were ‘collected’ which best matched this criterion. These points are shown in Figure 8-9.

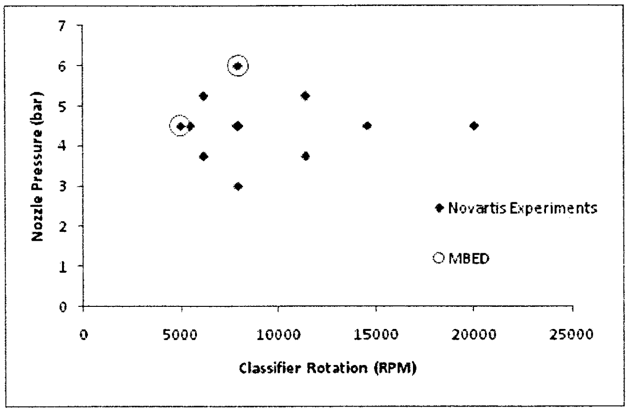


Figure 8-9: Classical Screen Study compared with Model Based Experimental Design

8.5.2 Bayesian Parameter Estimation - Iteration 1

The four significant parameters were estimated using Markov Chain Monte Carlo. The prior distributions for each parameter were Gaussian as shown in Table 8.1.

The results from the parameter estimation are shown in Figure 8-10. The collected data is shown in the top histograms, while the bottom row shows a kernel density estimate of the probability density and a fitted Gaussian density.

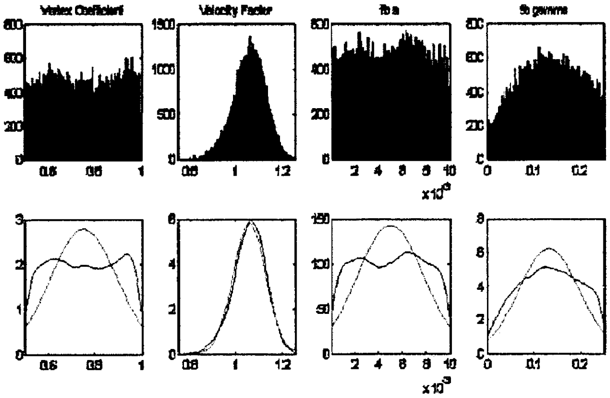
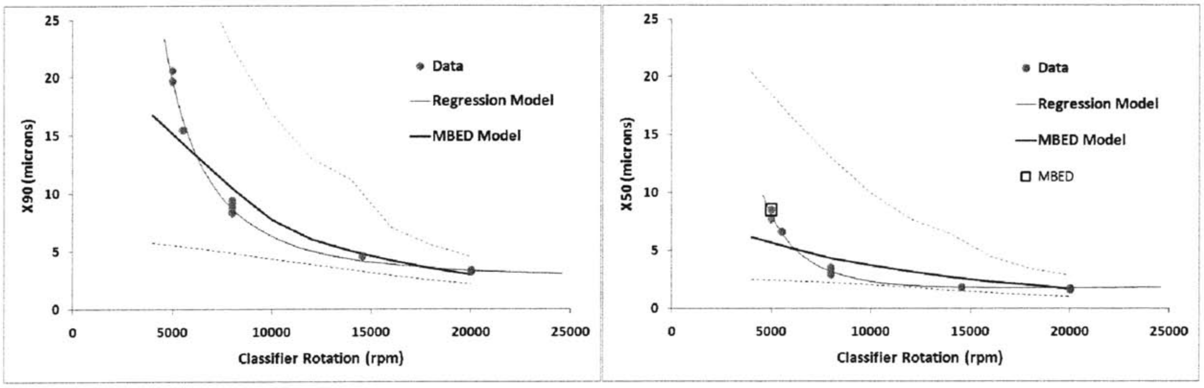


Figure 8-10: Posterior Distributions from Bayesian Parameter Estimation

These Gaussian fits to the posterior distributions were selected as the parameter estimates of the updated model. The uncertainty in the model output was then quantified. Cross sections at constant Nozzle Pressure are shown in Figure 8-11. The dotted lines represent the 95% confidence bounds on the model predictions. These are shown compared to the data from the previous studies. During the study, only the data points in squares were available (only one is shown on these charts). The other data points were not available.



(a) Uncertain X_50 Predictions

(b) Uncertain X_90 Predictions

Figure 8-11: Pilot scale data and empirical model (dots and line) compared with the model predictions and 95% confidence bounds (— and - - -) at Nozzle Pressure of 4.5 bar

Also shown in Figure 8-11 is the empirical fit created from the previous studies. It uses two scaled variables which represent the nozzle pressure drop and classifier rotation.

$$X_1 = \left[\left(\frac{1}{\omega} - \frac{1}{8000}\right)\frac{2}{\frac{1}{8000} - \frac{1}{5000}}\right]^{-1}$$

where ω is classifier rotation in rotations per minute.

$$X_2 = \frac{(\Delta P - 4.5)}{0.75}$$

where ΔP is the nozzle pressure drop

The updated model has very large uncertainties but it does match the experimental data throughout the entire design space. If this study were able to continue, the best course would be to collect more data, refine the model, and improve the parameters. Unfortunately, the algorithm once again suggested running experiments at slow classifier rotations and high nozzle pressure where the data had already been used. Therefore, it was decided to end the study and assess the production scale model.

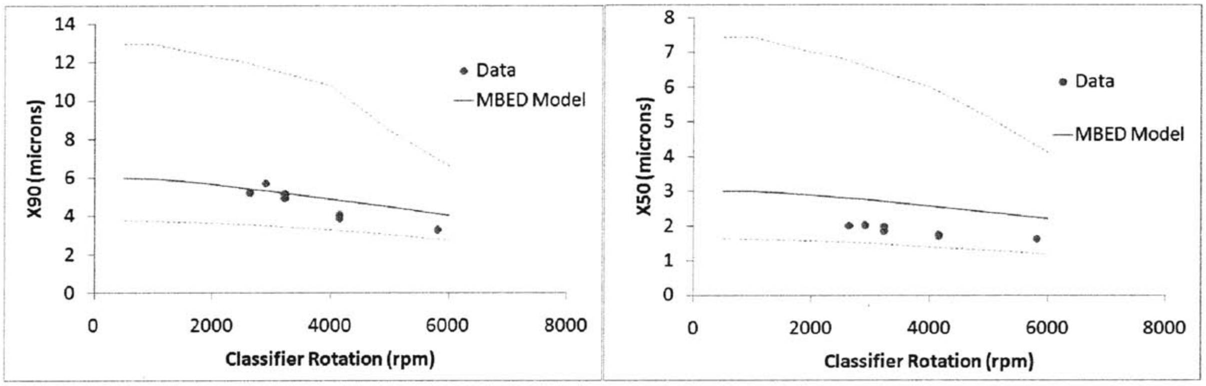
Table 8.4: Parameters Used for Scale Up of Equipment

Parameter	Pilot scale	Production Scale
Classifier radius (m)	0.025	0.11
Vane size (m)	$2.5e^{-3}$	$1.1e^{-3}$
Classifier exit (m)	$8e^{-3}$	$1.5e^{-2}$
Classifier length (m)	0.1	0.2
Volume (m^3)	$2.0e^{-4}$	$2.2e^{-3}$

8.5.3 Production Scale Predictions

To assess the quality of the model, it was adjusted to represent a production scale air mill classifier. This was accomplished by changing various equipment constants in the model. Unfortunately, details of the equipment were not available from Hosokawa Alpine, so best guesses were used from images of the equipment. The feed rate to the production scale equipment was known to be roughly twelve times larger than the pilot scale, so the equipment volume was made roughly twelve times larger. To keep the parameters somewhat consistent, the fraction of classifier surface area open for airflow was kept the same. The remaining surface area was covered by the vanes of the classifier wheel. A comparison of parameters is shown in Table 8.4.

The uncertainty profiles of the two final model outputs are shown compared to previous study data, in Figure 8-12. Once again, the uncertainties are quite large but they do match the data.



(a) Uncertain X_50 Predictions

(b) Uncertain X_90 Predictions

Figure 8-12: Production scale data (dots) compared with the model predictions and 95% confidence bounds (— and - - -) at Nozzle Pressure of 4.5 bar

In addition, simulations were done using the mean parameter values to predict the valid range of process conditions – those which would release product on specification. One of these analyses is shown in Figure 8-13. The model predicted that any combination of nozzle pressure and classifier rotation (up to 10 bar and down to 500 rpm) would produce on spec product. In fact, from the uncertainty profile in Figure 8-12, all of the model predictions would predict that the X_{90} <20 µm specification. The X_{50} <6 µm is predicted to meet specification with 95% confidence when the classifier rotation is at least 4000rpm. This was consistent with the findings from the previous study at production scale in which all experiments resulted in system responses well below the specifications.

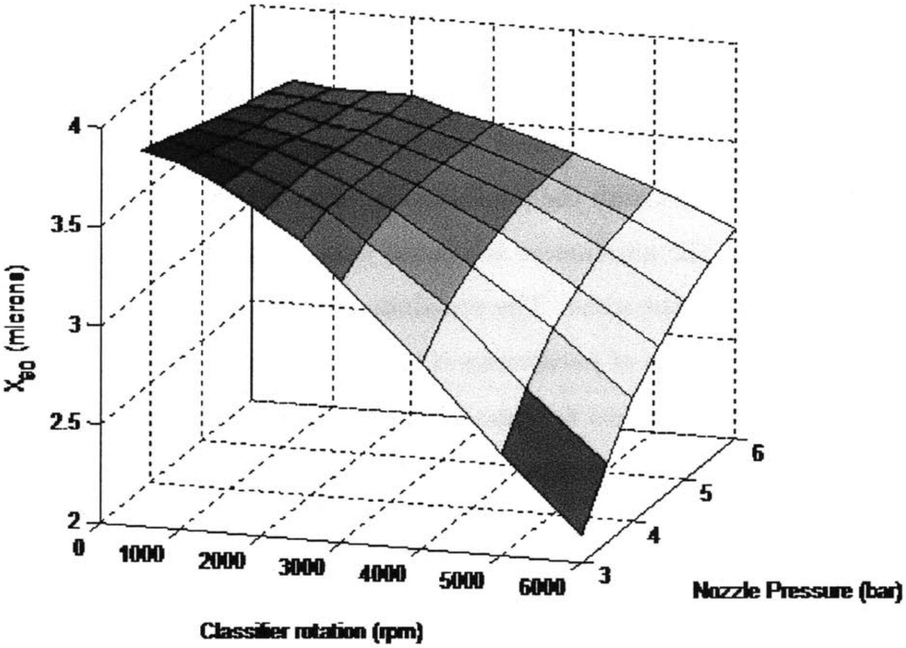


Figure 8-13: Production Scale Model Output X_{90}

8.6 Discussion

The goal of this study was to contrast the Classical and Bayesian Design approaches. The Classical Design is requires many more experiments, however, the empirical models fit data very well. Unfortunately, the do not extrapolate at all. The Bayesian approach required much more modeling effort but considerably fewer experiments. In addition, it provides physical insight into the system and an assessment of uncertainty that is valuable when making decisions.

As an example, we notice that the shape of the model output uncertainty curves at production

Table 8.5: Variable Costs of Two Experimental Studies

Resources & Costs	Classical Design	Model Based Experimental Design
Time (Worker-Days)	40	2
Intermediate Product (kg)	5000	230
Labor (k$)	14.8	0.5
Intermediate Product (k$)	1000	46
Total (k$)	1,015	47

scale differs from that at pilot scale. Focusing on the classifier rotation, the pilot scale response follows a trend $X_{90} \propto \omega^{-2}$ which indicates that the centrifugal force from the classifier is dominating. At the production scale, the response is flattened, which may indicate that the breakage in the fluidized bed or the drag force from higher airflow rates is dominating. One hypothesis is that the empirical model failed to account for the fact that nozzle pressure also determines the axial velocity of air into the classifier. A classifier with larger radius was predicted to have superior separations capabilities, but would also have reduced axial velocity. These effects would compounds each other, leading to a much smaller product particle size distribution. No matter the physical cause, there clearly was a regime change that altered the dominant driving force in the system. This explains why the Classical scale-up efforts failed. By basing the scale up effort on a correlation, the extrapolated values could not account for the regime change and the predictions were very inaccurate.

8.6.1 Economic Analysis

A simple economic model was created to compare two experimental campaigns. The Classical Design of Experiments side is estimated based on descriptions of previous studies.

Total time estimates were less than one month for the Model Based Experimental Design approach including time to develop the modeling and experimental design techniques, versus 9+ months for the Classical approach. Forty days are required for experiments alone. In addition, the classical experiments required 7.5 tons (estimated) of intermediate product as feed to the air mill classifier. The model based approach would require a tiny fraction of that, less than 30kg. The variable costs of the two experimental studies are compared in Table 8.5.

Here we assume the value of Intermediate Product to be \$200kg^{-1} and labor to be \$240d^{-1}.

8.7 Conclusions

The results of this study show the strength and weaknesses of both Classical Designs approach and Model Based Experimental Design. Classical Designs were applied to pilot and production scale classifiers. Roughly 75 experiments were conducted to make empirical models with very good fits but there was not an accurate way to relate the two systems. The Model Based Approach used two experiments at pilot scale to predict the same trends and one production scale experiment to confirm, over 95% fewer experiments and a 95% reduction in variable costs for this phase of the study. Also the time frame for experiments could have been compressed by eight months. As previously mentioned, being the first product to market is a huge advantage, so the time savings are the most impressive of all.

The major difference between the two approaches is the incorporation of physical knowledge into the design and decision process. The Classical study determined the production scale design space by understanding the pilot scale and then screening the design space by experiments. The model based approach takes the physics of particle classification into account and can extrapolate a range of suitable process conditions. This advantage is illustrated by the ability to accurately predict performance at the production scale – a process that took 17 experiments in the unnamed pharmaceutical company screening study.

The stated goal of this study was to develop a model with the same predictive power as the empirical model while greatly reducing the number of experiments required. The empirical models were highly accurate but had no physical meaning and no uncertainty estimates. Although the physics-based model was highly uncertain, the predictions were good enough to make an assessment of production scale performance because the uncertainty was quantified.

While the Model Based Experimental Designs used in this study compared favorably with Classical Designs, they also have limitations. In particular, they utilize the same assumptions as Optimal Designs – linearization of the model and Gaussian uncertainties. This means that the computed uncertainties are only approximations. The studies shown in Chapters 9 and 10 illustrate the advantages of the full Bayesian Designs approach.

Chapter 9

Study 2: Reaction with Uncertain Stoichiometry

This case study shows the application of the full Bayesian approach to a system of chemical reactions. Four different objective functions are demonstrated which illustrates the flexibility of the approach. In each instance, two Classical Designs are shown for comparison.

9.1 Background

The system is hypothetical process involving a single reactant: A. Under the conditions of interest A decomposes into species B and C, as well as some unknown byproducts - all together denoted X. In addition, both products further decompose into unknown byproducts, denoted Y and Z. A plausible, physical explanation for this scenario is a fission reaction or the breakup of a biological molecule, in which A splits into many parts, the main species being B and C. These products then decay away or are consumed.

9.1.1 Description of Experimental Studies

The experimental setup is a small batch reactor, which we can charge with some initial amount of species A. The reactor is isobaric and the reaction is assumed to keep a constant volume. The small reactor allows precise control of the temperature, to initiate and quench the reaction quickly relative to the reaction time. After the reaction is stopped, an analytical technique can be applied to the reactor contents. Species B and C can be measured fairly accurately but the reactant A and all the byproducts X, Y, Z cannot be detected or distinguished from each other.

A reactant samples were prepared using a well established protocol, but because of measurement uncertainty and human errors, there is some uncertainty in the initial concentration of A. Additional tests could be run but there is only enough material for a small number of experiments. The goal of this exercise is to design experiments to extract information about the model parameters. Several scenarios are present to demonstrate the ability to tailor experiments to the goals of the study.

9.1.2 Design Variables

Due to limitations with the analytical techniques, the reaction must be stopped to take measurements. This means that only one sample can be taken per experiment. The design variables are the time,t, or length of reaction and the temperature, T. We restrict our experiments to the ranges $t = [1, 5]$ s and $T = [300, 700]$ K.

9.2 Models

9.2.1 System Model

Equation 9.1 shows the mechanistic model of the system.

$$\begin{aligned} &A \underset{k_1}{\rightarrow} pB + (1-p)\, C + X \\ &B \underset{k_2}{\rightarrow} Y \\ &C \underset{k_3}{\rightarrow} Z \end{aligned} \tag{9.1}$$

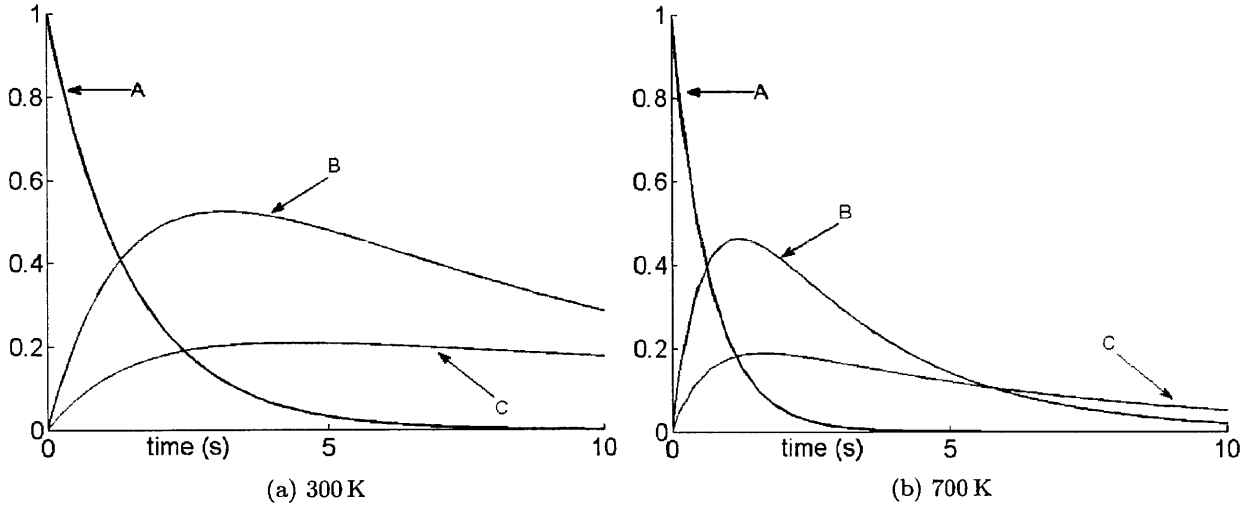
Assuming isothermal, isochoric, mass-action kinetics, this system can be described by the following mathematical model:

$$\begin{aligned} A =& A_0 \exp\left(-k_1 t\right) \\ B =& \frac{A_0 p}{k_2 - k_1}\left[\exp\left(-k_1 t\right) - \exp\left(-k_2 t\right)\right] \\ C =& \frac{A_0 \left(1-p\right)}{k_3 - k_1}\left[\exp\left(-k_1 t\right) - \exp\left(-k_3 t\right)\right] \end{aligned} \tag{9.2}$$

$$k_i = \alpha_i \exp\left(\frac{\beta_i}{T}\right) \text{ for } i = 1 \ldots 3 \tag{9.3}$$

For simplicity, all the species concentrations are scaled and dimensionless. The rate constants and parameters have units: k_i, α_i [=] s^{-1} and β_i [=] K^{-1}.

The system is shown for nominal parameter values in Figure 9-1.



(a) 300 K

(b) 700 K

Figure 9-1: Concentration profiles using the mean values of the uncertainty parameters at two temperatures

9.2.2 Model Parameters

The model uses Arrhenius kinetics and has eight parameters: A_0, p, α_1, β_1, α_2, β_2, α_3, β_3.

- Uncertainty in A_0 is due to difficulty in working with species A. Only a small sample was prepared and there is not enough to measure the concentration accurately. Therefore, the initial concentration of A in these experiments is not known exactly, but is the same for all experiments in this study. Because the uncertainty is the result of many small errors, none of which should be biased, A_0 is assumed to be normally distributed around the target dimensionless value of 1.

$$A_0 \sim \mathcal{A}_0\left(\omega_{\mathcal{A}_0}\right) \sim N\left(1, 0.05\right)$$

- A few experiments have already been run on this same system and various literature sources give estimates of p ranging from 0.6 and 0.8 with no discernible pattern. Therefore we assume a uniform prior density:

  $$p \sim \mathcal{P}\left(\omega_{\mathcal{P}}\right) \sim U\left(0.55, 0.85\right)$$

  where some extra margin is given just to ensure that the true value lies within the range.
- Previous experiments have also estimated the kinetic parameters of the decomposition reaction of A. The estimates appear to be centered around a mode, but are positively skewed. Therefore a shifted lognormal distribution is fitted to the data.

  $$\alpha_1 \sim \mathrm{A}_1\left(\omega_{\mathrm{A}_1}\right) \sim \log N\left(0.584, 0.5\right) + 1$$

  Similarly, the activation energy is distributed as

  $$\beta_1 \sim \mathrm{B}_1\left(\omega_{\mathrm{B}_1}\right) \sim \log N\left(3.3566, 0.466\right) + 368$$

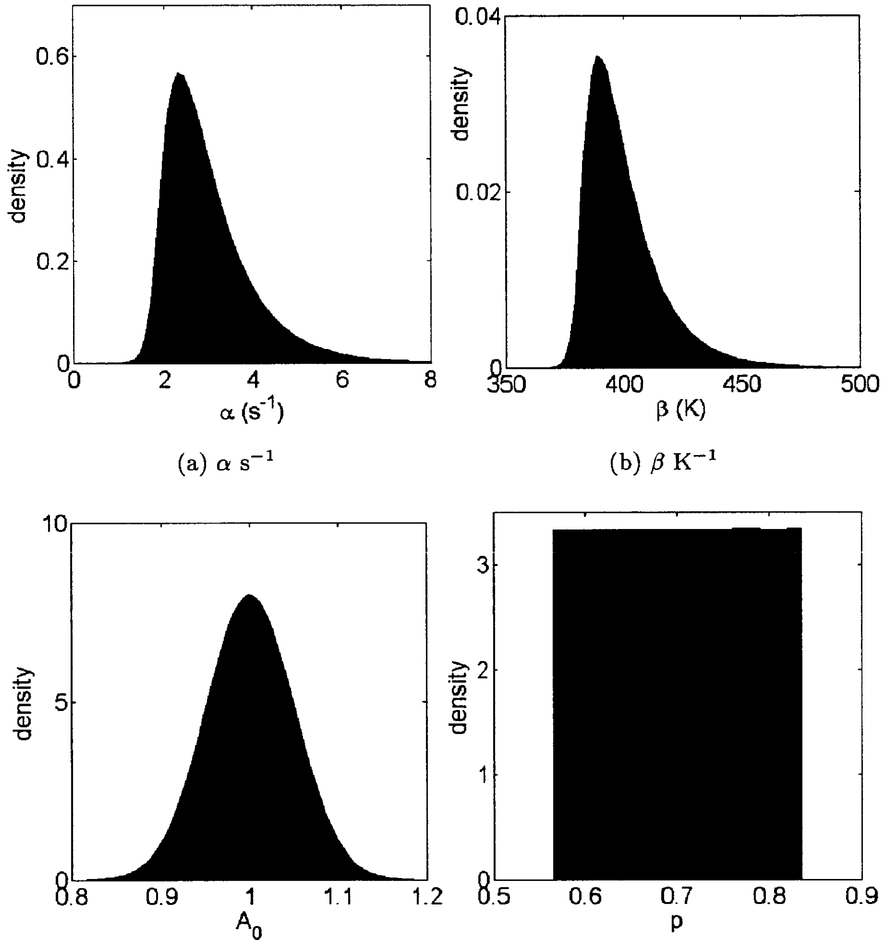
  The lognormal distribution is also attractive for modeling parameters that have a fixed lower bound. This is common with physical parameters which are known to be positive.
- The remaining kinetic parameters $\alpha_2, \beta_2, \alpha_3, \beta_3$ have been well determined in other experimental studies and have little uncertainty. The uncertainty from these parameters are small enough that they do not have significant impact on the model output density, as determined by a global sensitivity analysis. Therefore, these kinetic parameters are treated as constants.

  $$\alpha_2 = 1\ \mathrm{s}^{-1},\ \beta_2 = 650\ \mathrm{K}^{-1},\ \alpha_3 = 0.5\ \mathrm{s}^{-1},\ \beta_3 = 750\ \mathrm{K}^{-1}$$

The prior parameter densities are shown in Figure 9-2.

Observation Model

The existing equipment is only capable of taking offline measurements and of the many products, only species B and C can be measured reliably. Through calibrations and testing, the equipment has



(a) α s^{-1}

(b) β K^{-1}

(c) A_0

(d) p

Figure 9-2: Probability densities representing prior parameter knowledge

been shown to take unbiased measurements with some noise. The noise is modeled as $\varepsilon \sim N(0, 0.02)$ with the same dimensionless scale as the species concentrations.

9.3 Study Goals

Estimating All Model Parameters

The goal of the first study is to determine the best experimental design for estimating all the uncertain parameters. This is referred to as a D-optimal Bayesian Design. The procedure for this was described in Sections 5.1.2 and 6.4.1.

Parameter p

For the second study, we are primarily interested in the ratio of species B and C, represented by the parameter p. Estimating the kinetic constants is not important. Therefore our objective is to minimize the differential entropy of the posterior probability density of parameter p or maximize the Kullback-Leibler divergence between the posterior marginal distribution of p and the prior marginal distribution of p. This idea was described in Section 5.1.2 and D.1.2.

Kinetic Parameters α_1 and β_a

The third study is focused on the decomposition kinetics of species A and neglects the other parameters. Conceptually, it is the same as the second study and uses the Kullback-Leibler divergence between the posterior joint distribution $f_{\mathrm{A}_1,\mathrm{B}_1|D}(\alpha_1,\beta_1|\mathsf{d})$ and the prior marginal distribution of $f_{\mathrm{A}_1,\mathrm{B}_1}(\alpha_1,\beta_1)$.

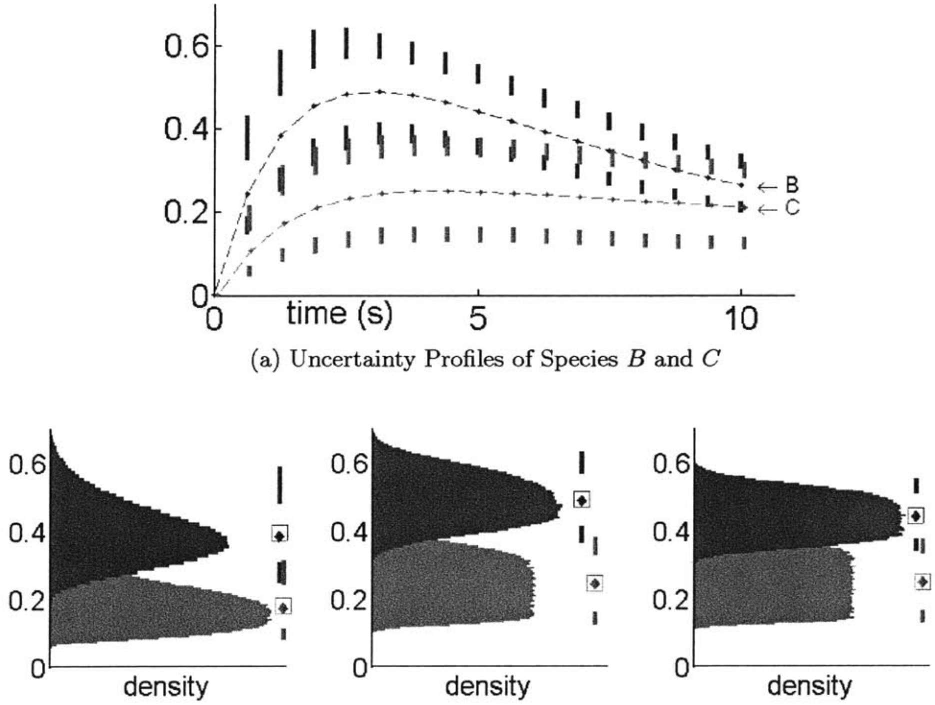
Initial Condition A_0

The goal of the fourth study is to estimate the initial concentration of species A. Again, it uses the same methods as the second and third studies.

9.4 Analysis of Prior Knowledge

9.4.1 Model Output Uncertainty

Using the system model and methods from Chapter 3, we can propagate the prior parameter uncertainty through to determine the uncertainty of each model output. This is illustrated in Figures 9-3 and 9-4, which show the model outputs of species B and C. All the concentration plots represent the uncertain species concentrations for a specified temperature, while varying time. The data was generated by propagating parameter uncertainty through the model using Polynomial Chaos Expansions.



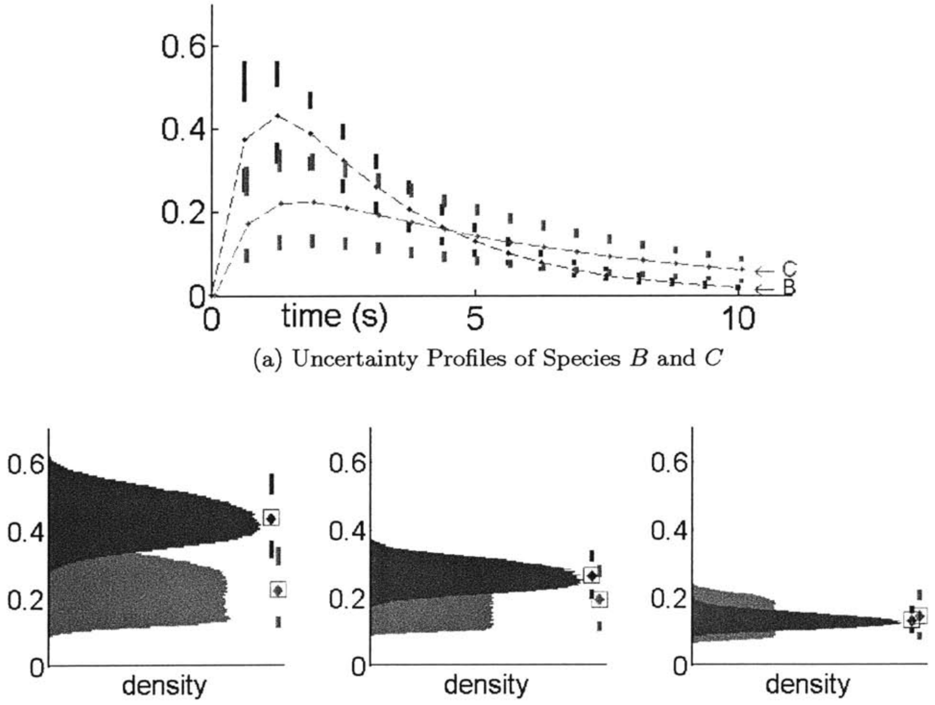
(a) Uncertainty Profiles of Species B and C

(b) Cross sections at times 1.25, 2.5, and 5 sec

Figure 9-3: Uncertainty Analysis of Uncertain Stoichiometry System Model at 300 K

9.4.2 Model Output Sensitivity to Parameters

In addition to the model output uncertainty in B and C, we compute the global sensitivities of the model output to each parameter, as shown in Figures 9-5 and 9-6. These are shown as normalized bar charts, indicating the relative contributions to the total uncertainty from each of



(a) Uncertainty Profiles of Species B and C

(b) Cross sections at times 1.25, 2.5, and 5 sec

Figure 9-4: Uncertainty Analysis of Uncertain Stoichiometry System Model at 700 K

the four uncertain parameters.

9.4.3 Data Prediction Uncertainty and Sensitivity

Because the measurements are relatively accurate, the model output uncertainty and data predictive uncertainties are not appreciably different. However, the sensitivities are impacted by the additional source of uncertainty. As time of reaction increases, the concentrations of B and C fall, however, the observation uncertainty remains the same. Therefore, the observation uncertainty because more significant as time goes on. See Figures 9-7 and 9-8.

9.4.4 Interpreting the Prior Knowledge

The uncertainty and sensitivity analyses provide valuable insight into the uncertainty stoichiometry model. It describes how each parameter influences the model output and how the influence changes over the design space. This information is used by the algorithm to select the best experiments. The figures in this section give an intuition for why certain experiments are more useful

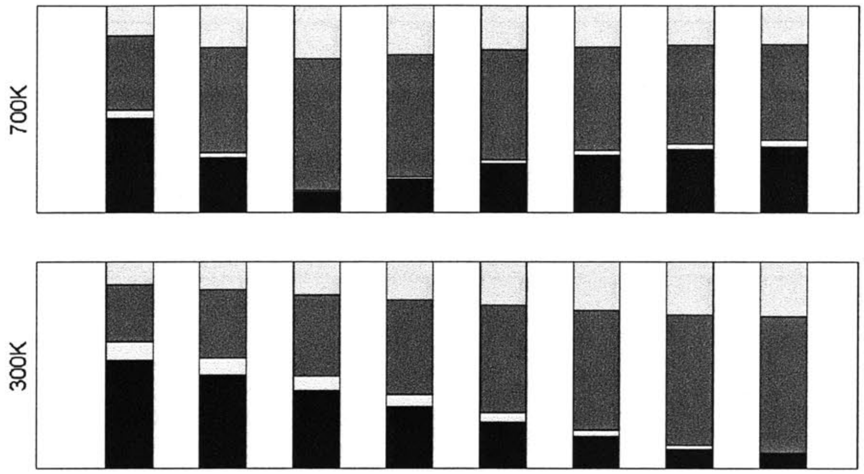


Figure 9-5: Global Sensitivities of Model Output B, from top to bottom, to A_0, p, β_1, and α_1

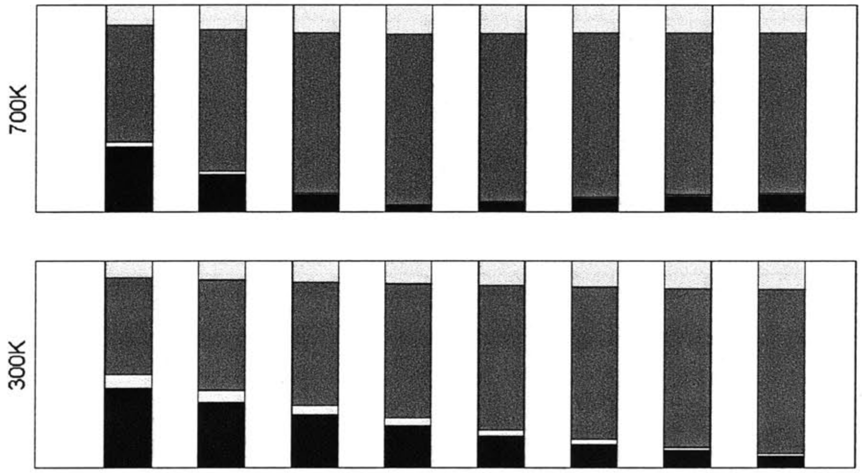


Figure 9-6: Global Sensitivities of Model Output C, from top to bottom, to A_0, p, β_1, and α_1

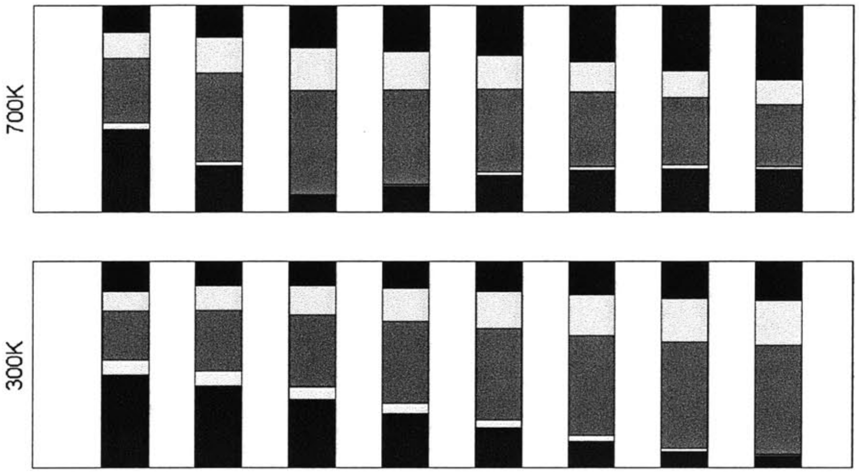


Figure 9-7: Global Sensitivities of Data Predictions B, from top to bottom, to ε, A_0, p, β_1, and α_1

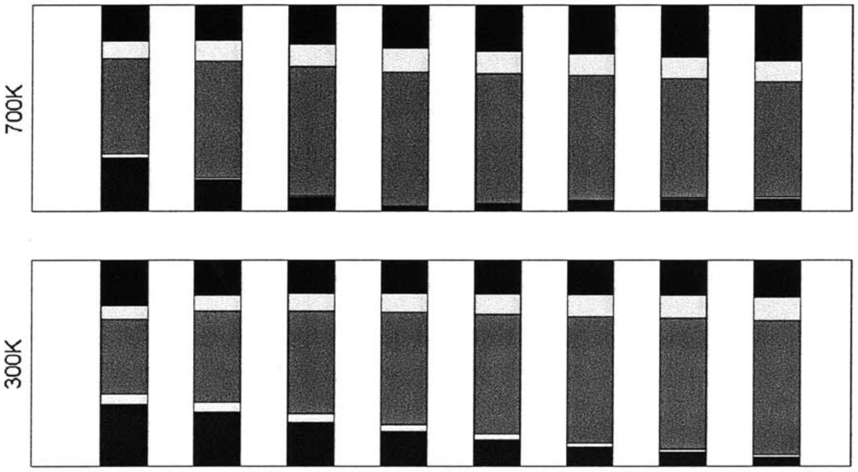


Figure 9-8: Global Sensitivities of Data Predictions C, from top to bottom, to ε, A_0, p, β_1, and α_1

than others. For instance, experimental conditions with large prior model output uncertainties indicate that an experiment would result in a large overall information gain. However, if we are only interested in learning about one particular parameter it may be best to choose experiments where the model output is most sensitive to that parameter.

9.5 Methods

9.5.1 Evaluation Criterion

For every study introduced in Section 9.3 the Bayesian approach was compared to two Classical designs – the Full Factorial and Central Composite designs. To quantify the effectiveness of the designs, they were compared using the objective function for Bayesian Designs, which is a study-specific measure of information gain.

This is not a fair comparison, because Bayesian methods are being applied to a Classical design. A more accurate implementation of Classical Designs would have used Least Squares estimation to determine the model parameters, but this would make it difficult to compare results. The assumptions used to estimate parameter error from Least Squares are clearly violated. Therefore, the expected information gain from using Classical Designs was computed with Bayesian methods. This in fact overstates the effectiveness of Classical Designs.

9.5.2 Classical Designs

Full Factorial and Central Composite designs were introduced in Chapter 4. There are many variations on these designs which might have been chosen by an expert, however, these were chosen as representative designs. With two design variables, the full factorial consists of four design points and the central composite requires eight. The points are listed in Table 9.1 and shown in Figure 9-9

As shown in two dimensions, the classical designs attempt to maximize coverage of the design space because this allows for the best identification of large trends. Qualitatively, this makes them robust because the odds are that at least some of the design points will provide useful information. On the other hand, robustness is gained at the expense of efficiency.

Table 9.1: Classical Design Points (time in sec, Temp in K)

Full Factorial	Central Composite
$(1, 300)$ $(5, 300)$ $(1, 700)$ $(5, 700)$	$(1, 500)$ $(5, 500)$ $(3, 300)$ $(3, 500)$ $(3, 700)$ $(1.5858, 359)$ $(1.5858, 641)$ $(4.4142, 359)$ $(4.4142, 641)$

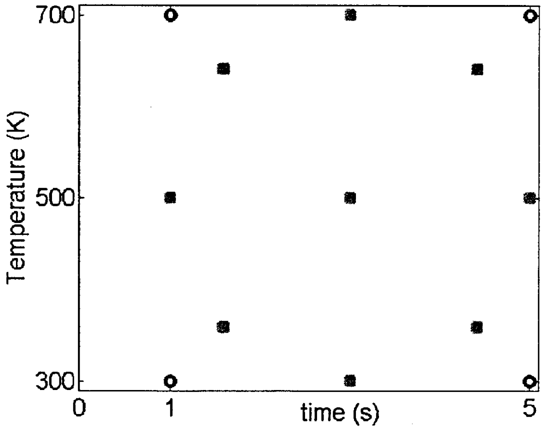


Figure 9-9: Classical Designs for Uncertain Stoichiometry Study - Full factorial in circles and central composite in squares

9.5.3 Bayesian Design using Markov Chain Monte Carlo

The original Bayesian Design algorithm was only applied to the second study, for the estimation of parameter p. Three designs were tested with one, two, and four design points. The algorithm from Section 5.4 was used for each scenario. The optimization step was the same as the posterior statistics and is discussed below. For every design x being evaluated, between 128 and 1024 datasets were simulated, and for every dataset d_j, Markov Chain Monte Carlo with adaptive Metropolis sampling was used for Bayesian parameter estimation. Delayed rejection was not used here because the posterior parameter spaces were easy to sample, so the additional expense of the delayed rejection was unnecessary.

The utility metric used was posterior entropy of p. This was calculated by taking the Markov Chain Monte Carlo samples from the joint posterior parameter density and keeping only the samples of parameter p. Then a Gaussian kernel density estimator (Matlab function ksdensity) was applied to approximate the marginal posterior density $f_{P|D}(\mathsf{p}|\mathsf{d})$. The utility function is given by:

$$\phi(\mathsf{d}_j) = h(P|D) = \int_{-\infty}^{\infty} f_{P|D}(\mathsf{p}|\mathsf{d}_j) \log\left[f_{P|D}(\mathsf{p}|\mathsf{d}_j)\right] d\mathsf{p} \tag{9.4}$$

The objective function for each design was the expected utility over all datasets:

$$\Phi = \sum_j \phi(\mathsf{d}_j) \tag{9.5}$$

9.5.4 Bayesian Design of Experiments

For each study, the Bayesian approach was used to select designs with between one and six design points. This serves to show the marginal gain in information for each experiment. The appropriate objective function was evaluated using the Prior Sampling Formulation. For problems of this size, it was found to be faster than Markov Chain Monte Carlo for the same degree of accuracy.

9.5.5 Optimization

For this problem, both the implicit filter algorithm and a genetic algorithm were employed while using a greedy strategy. All optimization problems were limited to four dimensions, or two-experiments. So for the five-experiment design was created by running a two-experiment optimization problem and including the optimal three-experiment design. This provided a small measure of redundancy, because it can be compared to the four-experiment design. Nevertheless, all the results are only lower bounds of the global maximum.

9.6 Results

9.6.1 Estimating All Model Parameters

The results of the study are shown in Table 9.2. All the selected experiments lie at low temperature and early times or high temperature and later times, with a greater emphasis on the latter. From our analysis of the prior information, we can see that these experiments likely reveal the most information about the kinetic parameters. The high temperature experiments are repeated more because they observations are obscured to a greater degree by the observation uncertainty.

Figure 9-10 shows the gain in parameter information, measured by Kullback-Leibler Divergence, for each Bayesian designed experimental study compared with the classical designs.

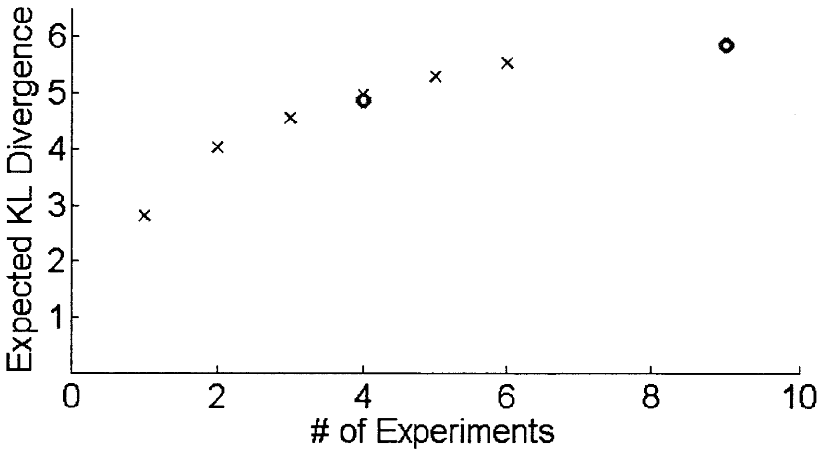


Figure 9-10: Comparison of Classical (circles) and Bayesian (x's) Designs for Estimating All Parameters

9.6.2 Parameter p

In this study wé were only concerned with estimating the parameter p. As in the previous study, designs with one to six experiments were constructed. All the designs were similar with

Table 9.2: Bayesian optimal designs for estimating all parameters, also showing the information metric expected Kullback-Leibler Divergence between posterior to prior entropy - absolute and per experiment

Design Points (time,Temp)	$E_D\left[D_{KL}\left(f_{\theta\|Y} \parallel f_\theta\right)\right]$	per Expt
Bayesian designs		
(1.9, 300)	2.81	2.81
(1.4, 300) (5, 700)	4.03	2.01
(1.2, 300) (4.3, 700) (4.9, 700)	4.55	1.52
(1.2, 300) (1.9, 300) (4.6, 700) (4.8, 700)	4.96	1.24
(1.3, 300) (1.3, 300) (3.7, 700) (4.8, 700) (4.9, 700)	5.25	1.05
(1.3, 300) (1.6, 300) (3.8, 700) (4.0, 700) (4.8, 700) (4.9, 700)	5.52	0.92
Classical designs		
Full Factorial	4.87	1.22
Central Composite	5.86	0.65

experiemnts at the highest temperature, 700 K, and times ranging from 3 s to 5 s. Again, looking at the analysis of prior information in Section 9.4 gives some intuition for why these experiments were selected. The model outputs and data predictions are most sensitive to parameter p at high temperatures and later times. However, if experiments were only run at late times, there would be no information gained about the kinetics or initial conditions. Therefore when multiple experiments are run, the times are spread out.

Figure 9-11 shows the gain in information for each Bayesian designed experimental study compared with the classical designs. Note that when one particular region of the design space is clearly more informative than the rest, the Bayesian approach is superior to the Classical approach, with only half the number of experiments required for the same expected information gain.

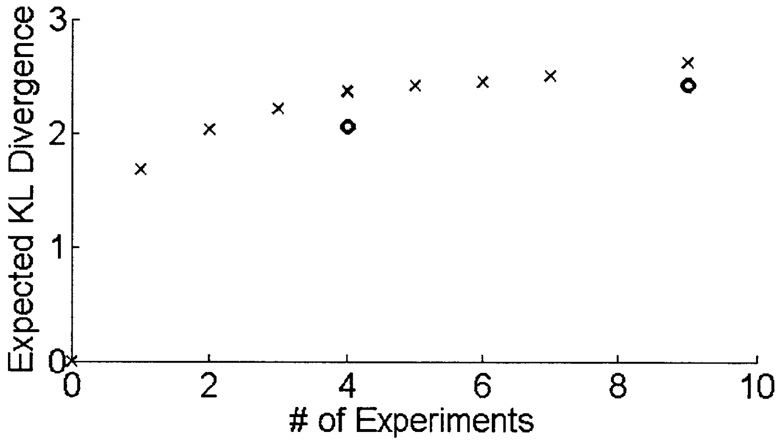


Figure 9-11: Comparison of Classical (circles) and Bayesian (x's) Designs For Estimating p

Insight from Markov Chain Monte Carlo

Representative results from an individual chain from the two-experiment design serve to illustrate the advantage of a Bayesian design. The posterior parameter densities for the four uncertain parameters are shown in Figure 9-12 for a two-experiment Bayesian Design.

Interesting features are that no matter what experiments are done, the uncertainty in activation energy is not reduced. It doesn't make sense to run experiments at different temperatures because we are not interested in reducing the uncertainty about activation energy.

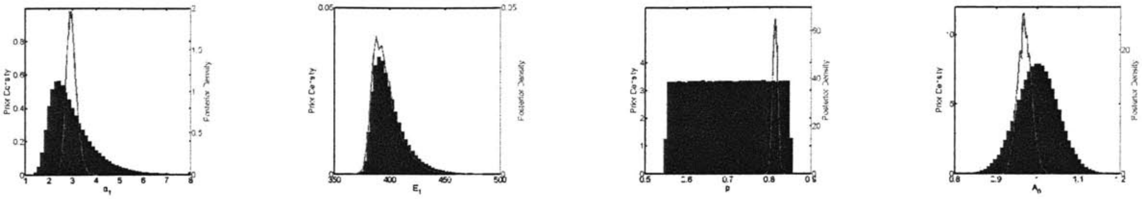


Figure 9-12: Representative Prior and Posterior Marginal Parameter Densities – Bayesian Design

9.6.3 Kinetic Parameters A and β_a

The results of the third study are shown in Table 9.3. For this example, the times were restricted to multiples of 0.25 s, which is why only rounded numbers appear. The designs here are similar to the designs for estimating all parameters, but the lower temperature/ earlier time experiments are more important.

Figure 9-13 shows the gain in information for each Bayesian designed experimental study compared with the classical designs. A decreasing in Expected Kullback-Leibler Divergence is not possible, due to the Gibbs Inequality, so the decrease between the five- and six-experiment designs is likely due to the optimization algorithm finding a local minimum instead of the global minimum.

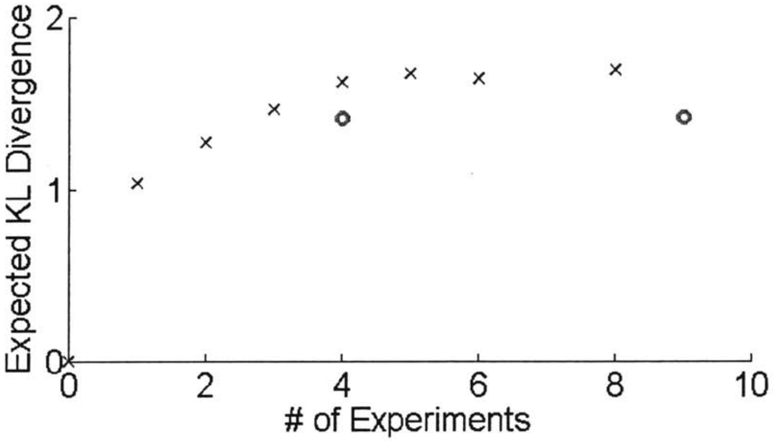


Figure 9-13: Comparison of Classical (circles) and Bayesian (x's) Designs for Estimating A and β_a

9.6.4 Initial Condition A_0

The final study is the least intuitive. It would make sense that all the experiments should be at the earliest times and lowest temperatures, because this is the closest possible approximation

Table 9.3: Bayesian optimal designs for estimating A and β_a, also showing the information metric expected Kullback-Leibler Divergence between posterior to prior entropy - absolute and per experiment

Design Points (time,Temp)	$E_D\left[D_{KL}\left(f_{\theta\|Y} \parallel f_\theta\right)\right]$	per Expt
Bayesian designs		
$(1, 300)$	1.03	1.03
$(1, 300)$ $(5, 700)$	1.27	0.64
$(1, 300)$ $(1, 300)$ $(5, 700)$	1.48	0.49
$(1, 300)$ $(1, 300)$ $(1, 300)$ $(5, 700)$	1.56	0.39
$(1, 300)$ $(1, 300)$ $(1.5, 300)$ $(5, 700)$ $(5, 700)$	1.67	0.31
$(1, 300)$ $(1, 300)$ $(1, 300)$ $(1.5, 300)$ $(5, 700)$ $(5, 700)$	1.83	0.30
Classical designs		
Full Factorial	1.41	0.35
Central Composite	1.42	0.16

of the initial conditions. Unfortunately, it is not possible to take measurements immediately after starting the experiment, as the design space is limited to times above 1 s. Instead, all the points are at the latest time and highest temperature. Again this can be explained by the sensitivities of the model outputs and data predictions to the parameter A_0. At earlier times, there is much higher uncertainty caused by the other parameters, which prevents good inference on A_0.

Figure 9-14 shows the gain in information for each Bayesian designed experimental study compared with the classical designs. As before, when only one region of the design space is useful, the Classical approach is very inefficient.

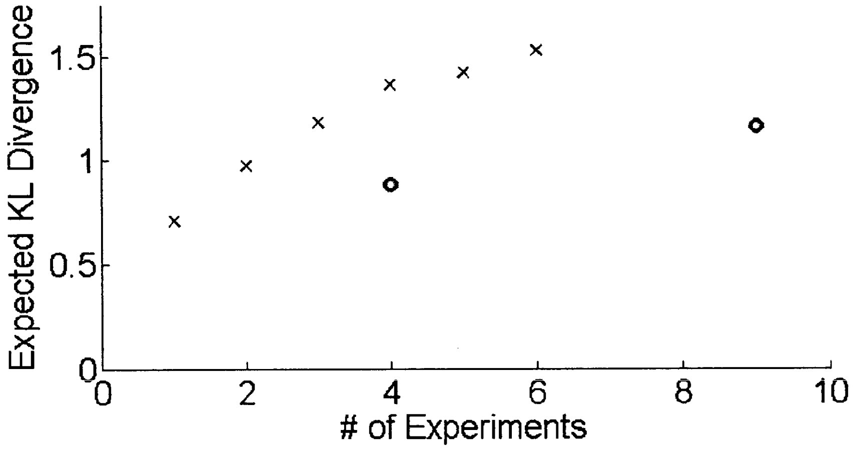


Figure 9-14: Comparison of Classical (circles) and Bayesian (x's) Designs for Estimating A_0

9.7 Conclusion and Discussion

The Bayesian Design gives a much better expected gain in information per experiment. This does not mean that the results of the Bayesian designed experiments will be better, just that the expected result given all our prior knowledge is better.

One useful feature of this simple system is the recognizable connection between the results and the prior information and models. Every one of the results from the four studies can be connected to the dynamics of the system model. In particular, the first study – a combined estimate of all parameters, is a mixture of the following three studies. Estimating the kinetic parameters required experiments at early times/ low temperatures and late times/ high temperatures at a 2:1 ratio, while the other two studies only required late times/ high temperatures. The combined parameters study attempts to balance information gain for the four parameters.

Although the results of the Bayesian approach can be explained from knowledge of the system, it is quite difficult to compose a design relying solely on intuition. As seen in the fourth study to estimate A_0, this can lead to very incorrect results. The Bayesian Design of Experiments algorithm allows the designer to use modeling for a quantitative assessment of information gain, instead of relying on intuition.

The comparison with Classical Designs highlights some interesting properties of the Bayesian Designs. Because there are several competing objectives in this study, the best design is spread across the design space. This is similar to the strategy employed by the two Classical Designs. In fact the Classical Designs are quite competitive with the Bayesian Designs. When the study goals are more specific, with only one or two parameters being estimated, the Bayesian Designs are superior. The same information gain was achieved using half the experiments.

Chapter 10

Study 3: Biomass-to-Liquids Process

The third major case study is an analysis of a Biomass-to-Liquids process. This demonstrates the feasibility of the Bayesian approach to design of experiments to relatively large and complex chemical engineering process simulations. It also showcases the importance of predicting the information gain from experiments and how this allows the experimenter to budget the appropriate number of experiments.

This study was also carried out with an industrial partner, Eni S.p.A. The goal was to design experiments for a pilot-scale Biomass-to-Liquids plant. The model was created in a process simulator which was treated as a black box model.

The first analysis done on the Fisher-Tropsch flowsheet was an uncertainty analysis. For this exercise, four parameters were given prior uncertainties and the output uncertainty was quantified. The second exercise was a design of experiments using additional inputs, treating four as design variables and five as an uncertain parameters.

10.1 The Biomass-to-Liquids Process

The goal of this process is to make usable fuel from biomass, i.e. wood chips, municipal solid waste, or algae. The biomass is gasified to produce hydrogen and carbon monoxide, which are then converted longer hydrocarbon chains using the Fischer-Tropsch process [20].

A large simulation effort has produced a life-cycle analysis model of the Biomass-to-Liquids process. This spans the upstream activities of gathering and preparing the biomass and downstream activities of gasification, processing, Fischer-Tropsch, and separation into various fuel grades.

10.1.1 Process Simulations

The Bayesian approach to design of experiments was applied to the Fischer-Tropsch and separations sections of the Biomass-to-Liquids process. The two sections were modeled using the CheOpe, the proprietary process simulation software used by Eni. The software is very similar to ASPEN Plus and PRO/II process simulators. CheOpe is used to simulate steady-state processes by computing mass and energy balances. It follows the sequential modeling paradigm, as opposed to the equation oriented modeling paradigm. This means that each unit operation is computed sequentially, depending only on its state and feed. An important disadvantage of this type of software is that the equations that describe each unit are not accessible to the user. Due to the nature of sequential modeling software, process models are treated here as black boxes. This dictates the types of methods that can be used to analyze or optimize the models.

To avoid the difficulty of learning the CheOpe software, which was written in Italian, a software interface was written in Excel. This connects the Bayesian design algorithm running Matlab with the process simulation running in CheOpe. Again, the important detail is only the inputs and outputs from CheOpe are required. Many Windows-based softwares such as ASPEN Plus can also be linked to Matlab in the same fashion.

10.2 Fischer-Tropsch Flowsheet

The flowsheet of the Fischer-Tropsch process and downstream separations units is shown in Figure 10-1.

The flowsheet has a detailed simulation of the Fischer-Tropsch reactions with a proprietary catalyst and then a series of separations units. The flowsheet has dozens of units and streams and includes a recycle loop, so the input-output relationships are quite nonlinear. However, there are no discontinuities in the outputs over the given ranges of inputs. The CheOpe simulation was designed as a black-box system and the underlying equations were not available.

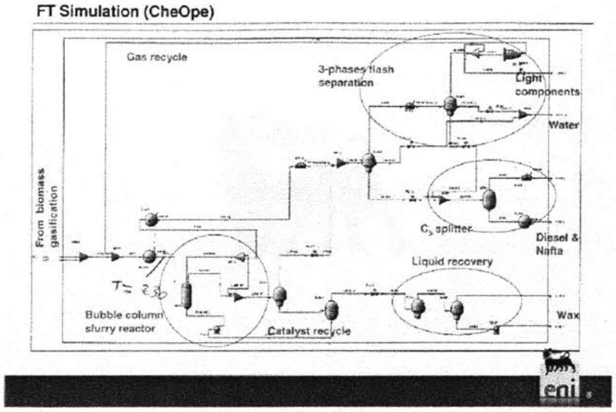


Figure 10-1: Flowsheet of the Fischer-Tropsch process and product separations

10.2.1 Inputs and Outputs

A total of nine flowsheet inputs were varied. They included temperatures, stream compositions, recycle ratios, and properties of the Fischer-Tropsch reactions. In order to carry out a design of experiments, different scenarios were presented, changing which inputs were treated as uncertain parameters and which were treated as controllable system properties.

A multitude of outputs were available from CheOpe simulations. These included unit and stream properties, as well as economic indicators. For these examples, the observable outputs were the flowrates of the product streams, two economic indicators, and composition of the gas product.

- Product Stream Volumetric Flowrates
  - Gases (Stream 19)
  - Diesel fuel (Stream 40)
  - Waxes (Stream 52)
- Gas Product Properties
  - Mass fraction of H_2
  - Mass fraction of CO_2
  - Mass fraction of CH_4
- Earnings from fuel production
- Earnings from electricity production

10.3 Uncertainty Analysis with Polynomial Chaos Expansions

10.3.1 Prior Parameter Knowledge

The prior knowledge of the parameters was described in terms of a mean and upper and lower bounds. The parameters are referenced by the equation number in the CheOpe simulation:

- Parameter 1: (Eqn 2) Temperature of Unit D303
  Domain is $[90, 110]$ °C. Mean is 100 °C.

- Parameter 2: (Eqn 3) Temperature of Unit D304
  Domain is $[45, 65]$°C. Mean is 55 °C.

- Parameter 3: (Eqn 9) H_2/CO_2 ratio (molar) in Stream 4 as controlled by a makeup stream of Hydrogen
  Domain is $[1.85, 2.1]$. Mean is 2, which is dimensionless.

- Parameter 4: (Eqn 4) Gas recycle ratio (volumetric) in Streams 24 and 4, the parameter is the negative of the actual recycle ratio
  Domain is $[-1, -0.1]$. Mean is -0.75, which is dimensionless.

From this data, a prior probability density function was be created for each parameter. The principle of maximum entropy was used. The first two parameters have uniform distributions, while the second two have truncated exponential distributions. The densities are shown in Figure 10-2.

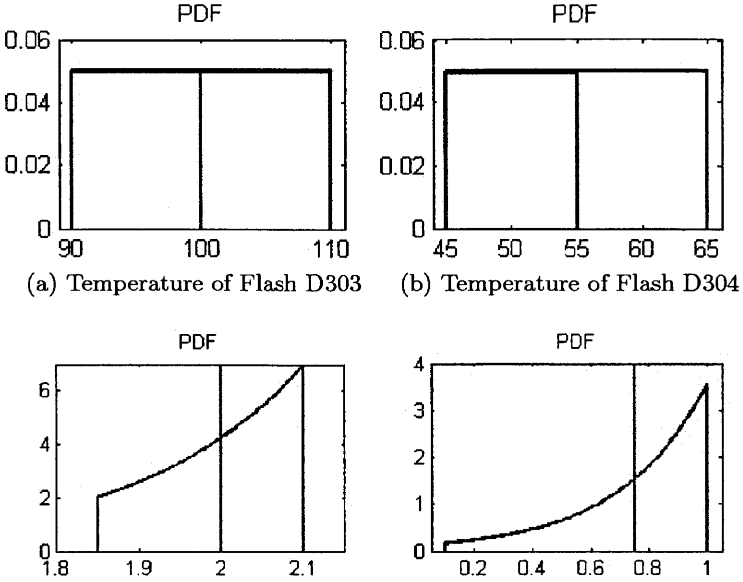
10.3.2 Outputs to Analyze

The model outputs of interest were a two economic metrics and three product flowrates.

- Production Earnings
- Electrical Earnings
- The flowrates of the product streams: 19, 40, 52

10.3.3 Uncertainty Quantification Results

Uncertainty Quantification was completed with Monte Carlo and Polynomial Chaos Expansions. 2500 Monte Carlo samples were taken. These samples were then used to solve for the coefficients of



(a) Temperature of Flash D303 (b) Temperature of Flash D304

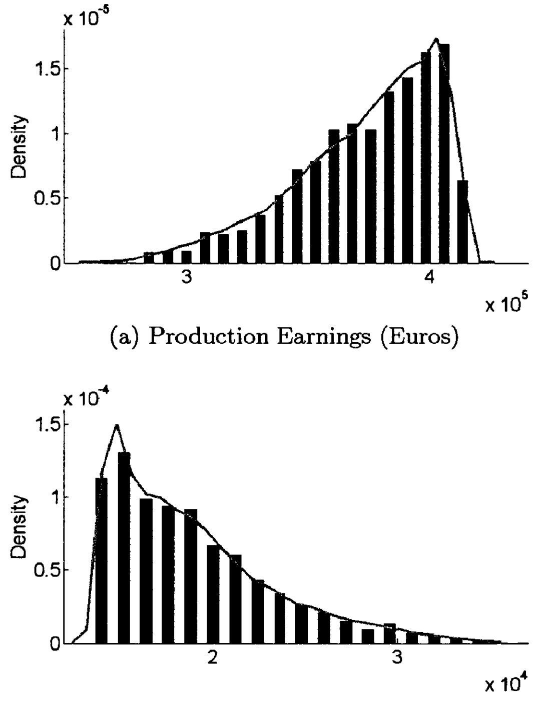
(c) Feed ratio $H_2 : CO$ (d) Gas Recycle Ratio

Figure 10-2: Prior parameter densities and their means

a $10^t h$ order Polynomial Chaos Expansion with 291 terms. The coefficients were computed using the Collocation Method with Monte Carlo. Not all the interaction terms were included – those with small coefficients were dropped. Samples for which the model did not converge were not used but were replaced with feasible new collocation points and solutions.

The results for each of the economic metrics are shown in Figure 10-3 product streams are shown in Figure 10-4.

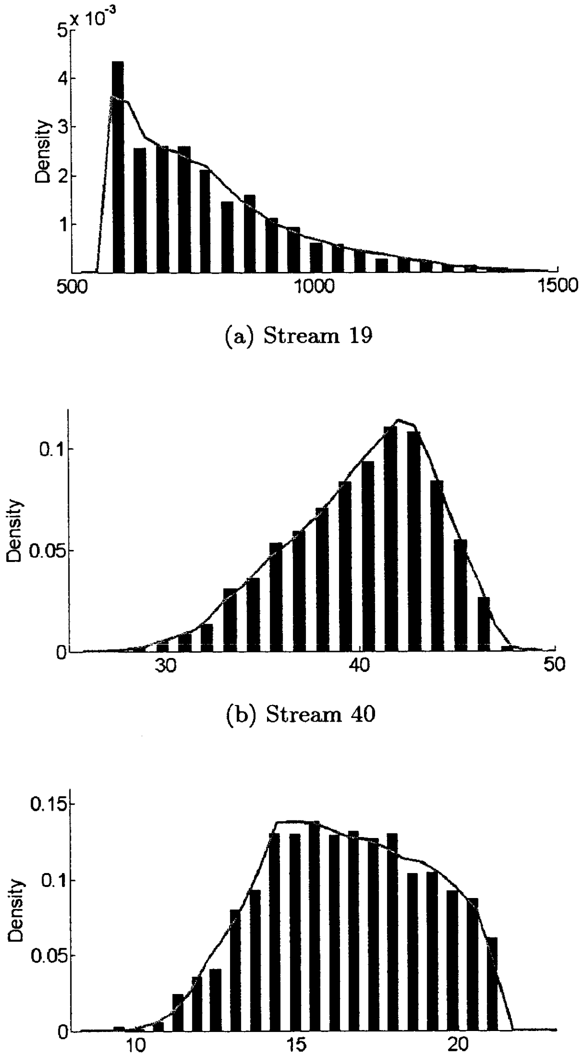
With enough terms, the Polynomial Chaos Expansions can accurately describe the uncertainty in the model outputs, even though the model is highly nonlinear and the input uncertainties are non-Gaussian.



(a) Production Earnings (Euros)

(b) Electricity Earnings (Euros)

Figure 10-3: Economic uncertainties due to parametric uncertainty as quantified by Monte Carlo (bars) and Polynomial Chaos Expansions (—)



(a) Stream 19

(b) Stream 40

(c) Stream 52

Figure 10-4: Product volumetric flowrates ($\mathrm{L\,h^{-1}}$) uncertainties due to parametric uncertainty as quantified by Monte Carlo (bars) and Polynomial Chaos Expansions (—)

10.4 Design of Experiments

The second exercise using the Fisher-Tropsch flowsheet was a design of experiments. Future studies are currently being planned to estimate parameters of interest, using different catalysts and process settings. This exercise demonstrated the benefits of Bayesian Design of Experiments for such a study.

The details for the experimental study were different from the uncertainty quantification problem. The same four inputs were used as design variables instead of uncertain parameters: the Temperatures of two flash units in the separations process (Flash D303 and Flash D304), the feed ratio of $H_2 : CO$ and the gas recycle ratio. These all had bounds on them from operational limits as described in Section 10.3, which defined the design space.

In addition to the four design variables, five additional inputs were treated as uncertain parameters. They included a process temperature, catalyst selectivities, the mass fraction of olefins in the products, and a parameter related to the distribution of products. All were given uniform distributions, as with the uncertain temperatures in Section 10.3.

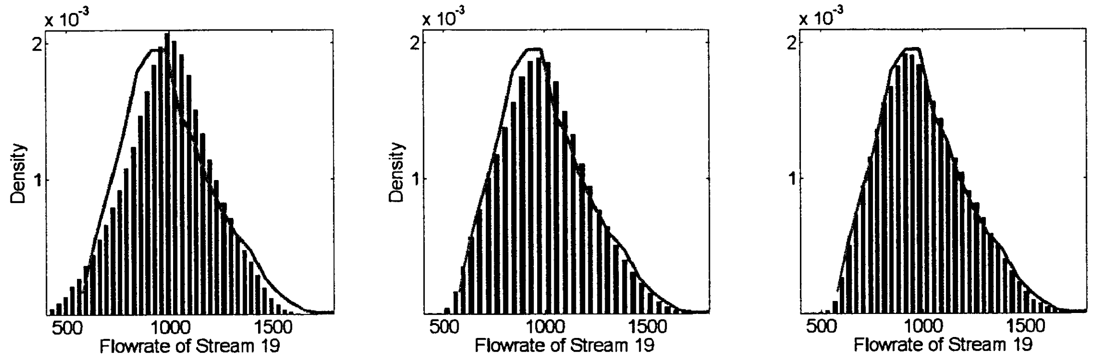
10.4.1 Observation Model

From the many outputs of the model, we chose the flowrates of three product streams to be our observable variables. These were the gas, diesel, and wax streams. In addition, the major components in the gas product stream could be observed. The observation uncertainty was proportional to the predicted flowrate $\dot{V}$ and is normally distributed as $\varepsilon \sim N\left(0, 0.05^2\dot{V}\right)$ $\mathrm{kg\,h^{-1}}$. This means that a stream with higher flowrate had higher uncertainty as well.

10.4.2 Polynomial Chaos Expansion Based Surrogate Model

The entire flowsheet takes between $20 \sim 90$ seconds to converge, based on the input values and the starting guess. It would be infeasible to do the required number of flowsheet runs in an appropriate time frame. Therefore Polynomial Chaos Expansion based surrogate models were used to approximate the flowsheet outputs. The process is exactly the same as uncertainty quantification, except that design variables are modeled as Uniform Random Variables. Using a Uniform Basis Random Variable, the input is simply a first order Polynomial Chaos Expansion. The outputs are approximated using nine-dimensional Polynomial Chaos Expansions. Each surrogate model was 10^{th} order, with only a subset of possible interaction terms included. Each Polynomial Chaos Expansion used 981 terms. As discussed in Chapter 3, the final order was determined by comparing a sequence of surrogate models with increasing order, until convergence was reached.

Figure 10-5 shows the convergence of the surrogate models for Stream 19 flowrate. The 10^{th}-order Polynomial Chaos Expansion was deemed to be converged. Note the difference from Figure 10-4a is due to different uncertain parameters.



(a) 2^{nd} order

(b) 6^{th} order

(c) 10^{th} order

Figure 10-5: Convergence of the Polynomial Chaos Expansion used as a surrogate model for Stream 19 flowrate of the Fisher-Tropsch model

Although surrogate models were constructed for many outputs, it was decided to only observe three outputs in order to simplify the problem. The selected outputs were: the flowrate of Stream 19, and the H_2 and CO_2 mass fractions in Stream 19.

10.4.3 Experimental Study

The goal of the study is to provide the best estimates of the five uncertain parameters. The utility function used is the Kullback-Leibler divergence from the joint posterior parameter density to the joint prior parameter density. This was computed using the Prior Sampling Formulation because the joint densities are five-dimensional.

Because each design point is four-dimensional, it is difficult to perform the comprehensive analysis of prior knowledge used in Chapter 9. However, some physical intuition is available from the flowsheet. Both flash units affect the split between the gas and diesel products. Flash D303 is the first in line with the light components going to Flash D304 and the heavy components going to the diesel. Flash D304 sends the heavy components to diesel product and the light components to gas products and recycle. The recycle stream is mixed with the feed in order to improve the process yield. This means that higher recycle will increase the flowrates through all internal streams and increase the conversion of H_2 and CO_2. Each of the design variables can impact the flowrate and composition of Stream 19, however the impact on design of experiments is difficult to assess.

10.4.4 Classical Designs

Full factorial and central composite Classical Designs were used for comparison. There are many variations on these designs which might have been chosen by an expert, however, these were chosen as representative designs. With three design variables, the full factorial consists of eight design points and the central composite requires 15. The spread of design points is very similar to those in Figure 9-9 but are in three dimensions. The full factorial covers all the vertices of the design space, while the central composite has points on the inscribed sphere.

10.4.5 Bayesian Designs

Bayesian Designs were constructed using up to 14 experiments. Since three data points are observed for each experiments, this is up to 42 observations of the system. The prior sampling formulation was used on the Polynomial Chaos Expansion surrogate model following the algorithm from Section 6.4.1. Because of the four-dimensional design space, a greedy optimization strategy was used, with each successive $N+1$-experiment design using the previous N-experiments. This means that each optimization was only over four dimensions.

10.5 Results

The results from Bayesian Design of Experiments are shown in Table 10.1. Because a greedy optimization strategy was used, only the newest experiment is shown for each sucessive design.

Figure 10-6 shows the gain in information for each Bayesian designed experimental study compared with the classical designs.

In this study, the Full Factorial Classical Design is nearly as effective as the Bayesian approach. This is because all of the optimal experiments are at vertices of the design space. However, the Bayesian Design repeats a small number of vertices and shows a small improvement over the

Table 10.1: Bayesian optimal designs for parameter estimation, also showing the information metric expected Kullback-Leibler Divergence between posterior to prior entropy

Number of Experiments (Temp D303, Temp D304, Feed Ratio, Gas Recycle Ratio)	Additional Design Point $E_D\,[\phi]$	
Bayesian designs		
1	(110, 65, 2.1, 0.1)	1.66
2	(90, 65, 1.85, 1)	2.30
3	(90, 65, 2.1, 0.1)	2.79
4	(110, 45, 2.1, 0.1)	3.12
5	(90, 45, 1.85, 0.1)	3.41
6	(90, 65, 1.85, 1)	3.67
7	(110, 65, 2.1, 0.1)	3.89
8	(90, 65, 2.1, 0.1)	4.08
9	(90, 65, 1.85, 1)	4.25
10	(90, 45, 1.85, 0.1)	4.41
11	(90, 65, 2.1, 0.1)	4.55
12	(110, 65, 2.1, 0.1)	4.68
13	(90, 45, 1.85, 0.1)	4.81
14	(110, 45, 2.1, 0.1)	4.87
Classical designs		
Full Factorial	4.77	
Central Composite	4.41	

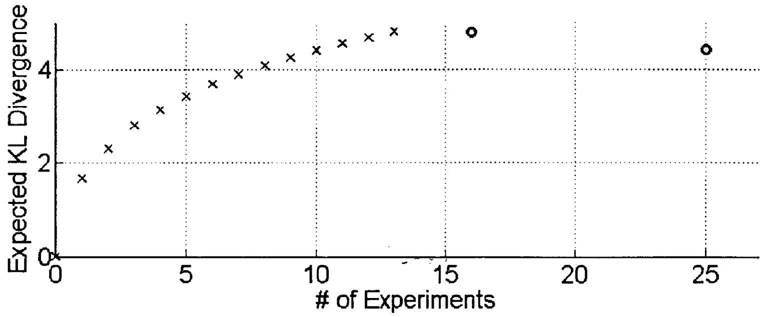


Figure 10-6: Comparison of Classical (circles) and Bayesian (x's) Designs for estimating the uncertain parameters)

Classical Design. Interestingly, the Central Composite design, which has more experiments, fares worse than the Full Factorial. This is because none of its experiments are at vertices.

10.6 Conclusion and Discussion

At first glance, there does not appear to be a large benefit from applying Bayesian Designs to this flowsheet. The Full Factorial Design is able to provide nearly the same expected information gain. Remember though, that this comparison is not realistic because it uses Bayesian methods for both the Bayesian and Classical Designs. In practice, this analysis would be used to identify the best Bayesian Designs – the sequence of x's in Figure 10-6, while a Classical Design would provide no prediction of information gain at all. This highlights two advantages to Bayesian Design of Experiments. First, this approach provides better performance than Classical Design of Experiments. Second, by incorporating all the knowledge of the process and uncertainties the Bayesian approach is able to predict the value of future experiments. This is invaluable when budgeting time and resources for an experimental study because it allows the designer to determine the best stopping point of a study – when future experiments will not yield enough information to be worth their cost.

The Fischer-Tropsch process represents a highly nonlinear and complex system that is typical of Chemical Engineering models. This study shows that the Bayesian approach can be effectively applied to these models and gives insight into the experimental process that cannot be gained using other Design of Experiments approaches.

Chapter 11

Conclusions

The central theme of this work is that all the available knowledge should be used to make design decisions. This includes the creation of system and observation models and just as importantly an understanding of how valid the model predictions are. The modeling aspect has been a part of chemical engineering for decades. Models are used for designing and improving processes, studying economics, and even comparing technologies. What is missing from most current design work – especially for design of experiments – is a rigorous treatment of the uncertainties that influence the design choices. This is a problem because design and investment decisions are naturally made by considering uncertainties.

11.1 A Focus on Uncertainty

This thesis reformulates the design of experiments as a problem of quantifying and managing uncertainties. The uncertainty focus changes many facets of the problem, including the way it is posed. The tangible result of experiments is the collection of data, but the true purpose of experiments is to improve predictive power in order to make some decisions. In the Bayesian approach, this can be made explicit using the decision theory framework.

In order to take advantage of the decision theory framework certain modeling tools are required: a model of the system, a model of the observations taken from the system, and an understanding of the sources of uncertainty that impact the system. Computational methods are required to characterize the uncertainties (probability and information theory), analyze their impacts (uncertainty quantification), and incorporate new information (parameter estimation). Every one of these methods must be able to deal with nonlinear models and complex descriptions of uncertainty, so Bayesian methods for every step.

Finally, the models and the methods are integrated within the decision theory framework to determine the experiments that will reduce the important uncertainties. This requires the ideas of utility and risk, as well as optimization methods.

11.2 Case Studies

The benefits of the uncertainty-focused approach were demonstrated on three systems: an air mill classifier, a series of chemical reactions, and a process simulation of a biomass-to-liquids plant. Each of these case studies highlights different advantages of the Bayesian Design of Experiments approach.

The first case was carried out with the help of an industrial partner, which allowed a direct comparison between the proposed methods and current industrial practice. In addition to the technical comparison, an economic and logistical analysis revealed the benefits of reducing the number of experiments. Both the time invested and money spent were drastically reduced, while achieving the same end goal of predicting system performance. This is a reflection of how conservative the current industry approaches are, as well as the improved efficiency of the model based approach.

The second case illustrated the impact and importance of designing experiments with a clear, quantifiable goal. Several studies were designed for a system with sequential chemical reactions. Each study had a different objective, such as estimating kinetic constants, initial conditions, or other reaction properties. Therefore the experimental designs for each study were different, each tailored to its specific objective. This is in direct contrast to the classical approach to design of experiments, in which the objective does not impact the design at all.

Finally, the third case study was a design of experiments on a process simulation. This showcased the use of Polynomial Chaos Expansions to characterize the uncertainty in a system. This method is ideally suited for chemical engineering applications since they are very efficient – requiring many orders of magnitude fewer model evaluations than standard Monte Carlo methods, and can also be applied to black box, nonlinear systems. This is important because many chemical engineering models are built in process simulators or other legacy codes that are not written to permit uncertainty analysis. The methods used in this study demonstrate the feasibility and effectiveness of Bayesian Design of Experiments for a wide range of chemical engineering models. An additional benefit over classical designs is the ability to estimate the information gain from future experiments. This allows the designer to budget the appropriate number of experiments to achieve the study goals.

11.3 Using Appropriate Methods

Design work relies on the modeling of systems and any related uncertainties. It is important to balance the level of detail used in all aspects of the design process. It does not make sense to create highly detailed system models for a process that is not well understood. The uncertainties will far outweigh the predictive abilities of the model. In the same vein, it does not make sense to apply complicated design methods to all models.

All model based design of experiments approaches assume that the supplied models are correct. Distinctive features of the Bayesian Design approach are: a strong dependence on the models and utility functions and a large gain in information for a small number of experiments with quickly diminishing returns. The study in Chapter 9 especially illustrates the advantages over the Classical Approach, with large information gains from certain experiments and no information gain from others. This indicates that the design is highly customized and effective, however, it is quite unclear how useful the design will be if the system behavior is different than the system model. This does not mean that the model based design of experiments are not useful or too risky. It is a caution that the appropriate levels of modeling detail must be employed with the appropriate treatment of uncertainty.

The Bayesian Design of Experiments is meant for systems that are well understood and also have a clear study purpose. For this combination of circumstances, reliable (but uncertain) models can be created and a descriptive utility function can be chosen. On systems with large uncertainties and little understanding of the underlying driving forces, it is not appropriate to create highly detailed models and use the Bayesian approach.

11.4 Advantages of Bayesian Designs

The rigorous treatment of uncertainty requires additional modeling and computational efforts, but yields additional benefits. Although the idea of risk informed decision making was not fully developed in this thesis, the point was made that real world decisions are made based on consequences, uncertainties, and risk. Using this framework allows this assessment to be made naturally and the benefit is not just the quantifiable increase in information gain, but also the improved ability to make decisions.

Focusing the entire process around uncertainty exposes the weaknesses in the other approaches to design of experiments. The assumptions taken by the Classical and Optimal Design approaches are unable to treat complex uncertainties. They deal only with point estimates of parameters, Gaussian observation uncertainties, and linear models. This is akin to making decisions based on assumptions instead of analysis. The Bayesian approach incorporates all the available knowledge, allowing decisions to be made using a complete assessment of uncertainties and risk.

Chapter 12

Thesis Contributions

This thesis provides a framing of design of experiments that is substantially different than the current approaches used within the chemical engineering community. The focus is on uncertainty and decision making rather than optimization and classical statistics. This perspective follows from the realization that the primary role of engineering experiments is not only gaining knowledge but to gather the necessary information to make an informed design decision. By properly dealing with uncertainty, Bayesian Design of Experiments gives a clearer picture of the important factors that affect the system and how experiments will impact future decisions.

12.1 Synthesis of Methods

The main contribution is the synthesis and demonstration of the ideas and methodology for model based design of experiments. In particular, the Bayesian perspective of representing uncertainty with probability theory is commonly used in chemical engineering, but is not well understood. For this reason, most design work uses classical statistics and Bayesian methods are uncommon. This thesis has integrated a large number of Bayesian ideas and methods into the decision theory framework and applied them to the design of experiments problem. These include: Bayesian probabilistic modeling of uncertainty, use of prior knowledge, use of non-Gaussian distributions and corresponding Information Theory metrics, Markov Chain Monte Carlo methods, and Polynomial Chaos Expansions. Many of these techniques have not previously been applied to the design of experiments. In particular, the use of Polynomial Chaos Expansions to efficiently characterize the uncertainty of systems across the entire design space enables the practical application of the Bayesian approach to a much wider range of models. This has been used for parameter inference but not the design of experiments.

The novel methods used in this thesis are extensions of the work by Ryan [71], which are designated as prior sampling formulation of information metrics used for design of experiments. These are only applicable to design of experiments because they take expected values over data predictive densities, which only occurs when simulation experiments. The paper by Ryan proposed the first such formulation for the case of D-optimal Bayesian design of experiments. This work has modified the method for optimal estimation of a subset of parameters, as well as a novel approach to Bayesian model discrimination.

12.2 Demonstration

The second contribution of this thesis is the application of the above methodology to Chemical Engineering systems including: chemical kinetics models, a mechanical separations unit, and a process flowsheet. Literature searches reveal many attempts to apply a Bayesian approach to chemical engineering systems [29, 28, 62, 63, 14, 15, 60, 61]. However, these invariably make approximations to simplify the system and reduce the computational burden. The models used in this thesis they exhibit highly nonlinear responses and utilize non-Gaussian distributions. These are proof-of-concept demonstrations which show that the methods can easily be extended for larger and more complex models. The demonstration of the Bayesian approach to design of experiments had not previously been demonstrated on systems of this level complexity.

By applying these methods to practical chemical engineering systems the benefits of the Bayesian Design of Experiments approach over the commonly used Classical Designs approach are clearly established.

Chapter 13

Directions for Future Work

13.1 Sensitivity of Results

Each of the studies in this thesis assumed that the inputs to the Bayesian Design of Experiments algorithm are well defined. "Inputs" refers to the priors knowledge, system models, observation models, utility functions, and risk metrics. Once these components are defined, the Bayesian approach will objectively determine the best experiments. However, the modeling of these components has considerable room for interpretation. Many system models can be proposed using different levels of sophistication or different physical understanding of the same system. Prior parameter knowledge can be quite subjective depending on interpretation of the available data. Very little is known about observation models and little effort has been made to study them. Utility functions and risk metrics must be correctly tailored to the purpose of the study. Many choices must be made in formulating the design of experiments problem. Perhaps the largest barrier to the adoption of Bayesian methods is the subjectivity of these and the unknown impact these choices have on the results.

There are many interesting opportunities in examining the effects of changing the inputs on the resulting optimal experimental design. Unfortunately, the sensitivity to changes in the mathematical modeling of the inputs will have to be evaluated for each study. A robust and consistent method to compute the sensitivity of the results to the subjective inputs is necessary for two reasons. First of all, such a method would allow an assessment of how accurately the inputs must be described. Also, this analysis could help understand the impact of designing experiments using an incorrect model.

13.2 Risk Informed Decision Making

The only risk metric used in this work was the expected utility. Given that the utility function and risk metric have a large impact on the result, more work should be done to formulate metrics that accurately represent the goals of the study. The current metric assumes that the designer is interested in the best expected result. However there may be other constraints that make the problem more interesting such as ensuring the probability of obtaining high value results is greater than a threshold.

Incorporating these types of constraints into the risk metric is critical in the context of designing experiments for making project decisions. Large scale projects are often carried out in a stage-gate iterative process, illustrated in Figure 13-1. Research and development is carried out in each stage, and the project must meet certain criteria before passing the next gate.

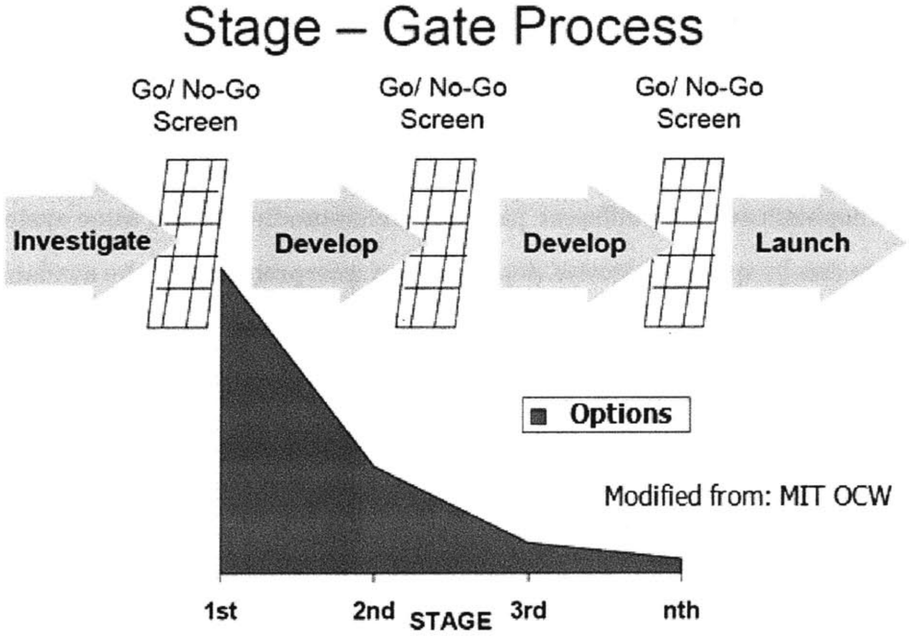


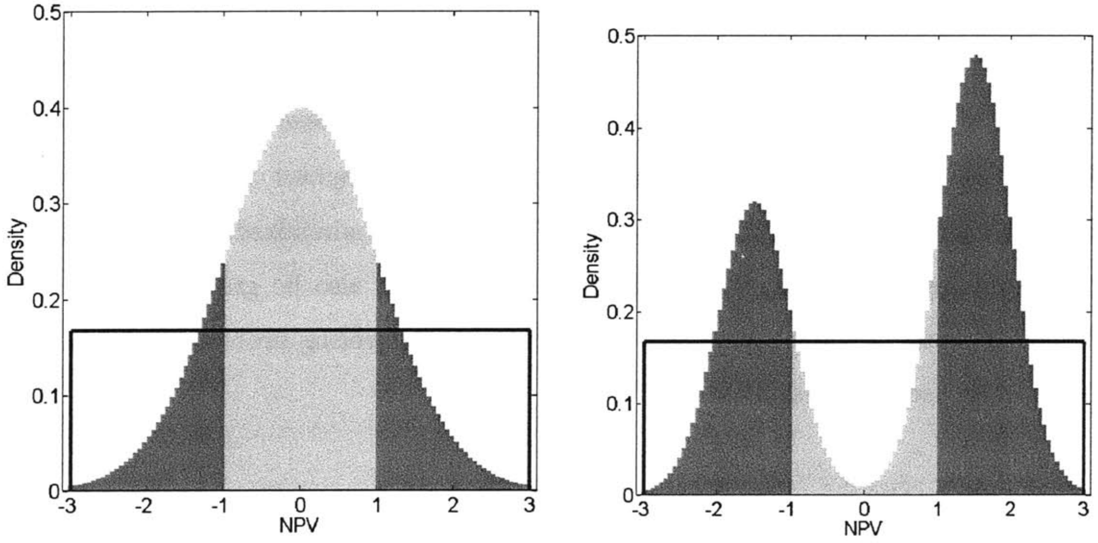
Figure 13-1: The Stage-Gate model for decision making in large projects

In these situations, the success of an experiment is not measured in information gained but in the ability to advance the gate decision process. The risk metric should be formulated to ensure a high probability of useful results. To give a simple example, say that we are interested in a new technology and are currently in the research phase - Stage 2. The Gate 2 decision will be based on the economic analysis of Net Present Value (NPV). If the NPV is greater than V_0, the project will move to Stage 3, otherwise it will be dropped. With uncertain conditions however, this criteria becomes difficult to assess. Clearly, if $NPV >> V_0$ or $NPV << V_0$, the decision will be easier to make than if $NPV \approx V_0$.

For this decision, a utility function based on information gain about the net present value does not accurately capture the purpose of this experiment. A better solution is to base the utility on the discrepancy D_{util} between the model predictions of Net Present Value and V_0, ie:

$$D_{util} = P\left(\left|\frac{NPV - V_0}{V_0}\right| > 1\right) \tag{13.1}$$

and the risk metric is maximize the minimum D_{util} over all possible datasets (a common robust optimization metric [6]). Each experimental design will be judged on its expected ability to allow a comparison of NPV to V_0. This is illustrated by showing the results for two simulated datasets in Figure 13-2.



(a) Less Useful Dataset

(b) Very Useful Dataset

Figure 13-2: Illustrating utility and risk metrics of two datasets for making Stage-Gate project decisions, where the target value is $V_0 = 0$

Figure 13-2a shows results from a dataset that reduces parametric uncertainty such that the uncertain NPV values are clustered around V_0. Even with this additional information it is difficult to determine whether the project will meet the required threshold value. An information gain utility would be high, but the discrepancy utility is low. Figure 13-2b shows results from another dataset in which the parametric uncertainty results in a bi-modal NPV distribution. Neither mode is near V_0, indicating that if this dataset were observed the Gate 2 decision would be relatively easy to make. In this case the discrepancy utility would be high.

Finally, the risk metric is applied. A robust risk metric would select an experiment that can guarantee a minimum improvement in decision making ability. There are obvious problems with these choices in utility function and risk metric. The utility function does not reward information gain, and the robust risk metric is often too conservative. A more appropriate utility might be multi-objective - assessing both discrepancy and information gain simultaneously.

Developing appropriate utility functions and risk metrics is an active research topic that will greatly benefit any design work using the decision theory framework. Use of these functions will allow design work to be reframed as a risk informed investment decision.

13.3 Direct Comparison to Other Model Based Approaches

This has been compared with the Classical Design approach but not Optimal Design approach. The advantages of the Bayesian approach compared with Optimal Designs were discussed but not demonstrated. We do know that the Optimal Design approach is a subset of the Bayesian approach, when the correct assumptions are applied and prior knowledge is minimized. Because the Bayesian approach is more general, the expected information gain must also be greater than or equal to that of Optimal Designs. Quantifying this on a chemical engineering problem would be valuable motivation for the use of Bayesian Designs.

13.4 Better Optimization Methods

There were two drawbacks to the optimization tools used in this thesis: inefficiency and a lack of global optimum guarantees. Genetic algorithms in particular are very inefficient and the global optimum issue has been discussed in Section 2.5. The Implicit Filter algorithm was found to work well when the objective function was computed accurately, however when larger and more complex systems are used, the signal-to-noise ratio in the objective function will decrease. Other methods have been developed to deal with these types of optimization problems by gradient approximation from noisy data. The idea is that the gradient will be highly uncertain and so multiple objective function evaluations are needed form an approximate gradient. Perhaps the most promising are works by Spall [74] and Kraft [80, 79].

Appendix A

Probability, Statistics, and Estimation

A.1 Full Definition of Random Variables

A Real-valued Random Variable $X(\omega)$ describes a set of outcomes and the probability of each one occurring. Outcomes are the smallest distinguishable results of a random occurrence and are unique and mutually exclusive, while events are collections of outcomes. Here we are interested only in continuous, real-valued Random Variables which have outcomes that can be related to real-valued numbers in $\Re$. $X(\omega)$ is defined by a probability space (Ω, Σ, P), where

- Ω is the outcome space - the set of all possible outcomes ω of the Random Variable
- Σ is the sigma algebra - the set of all the events of interest
- P is the probability measure - the function which assigns a real valued probability to every possible event.

For a continuous, real-valued Random Variable the outcome space contains an infinite number of outcomes, and each one can be uniquely mapped to a value in $\Re$. In this way, the abstract notion of outcomes can be associated with comparable quantities. The dependence on ω is an indicator that the value x of the Random Variable depends on the outcome that occurs. Here the probability spaces will be distinguished by subscripts on the outcome space, while the sigma algebra and probability measures should be obvious by association.

Example 24: Measurement of Weight - Events and Probability Distribution

Functions

Say we wish to estimate the weight of a rock so we place it on a scale. Due to natural fluctuations in the environment and our scale, the observed weight will vary by a small amount. These varying observations can be modeled with a Random Variable $W(\omega_W)$. The outcomes and events associated with each measurement need to be defined carefully so that the values are physically intuitive.

We define $W(\omega_W)$ so that it describes the probability of measuring particular values of weight. The outcomes are that the measurement, $W(\omega_W) = \mathsf{w}$ for $0\,\mathrm{kg} < \mathsf{w} < \infty\mathrm{kg}$. The events are collections of outcomes such as $\{\omega_W : W(\omega_W) < \mathsf{w}\}$. We will call this particular type of event *intrinsic*. Each value has a corresponding intrinsic event. We can then assign a probability to each intrinsic event/value. The first is easy: measuring $W < 0\,\mathrm{kg}$ is impossible so it has probability $P(W(\omega_W) < 0\,\mathrm{kg}) = 0$. Also, the probability that the weight is less than ∞kg is $P(W(\omega_W) < \infty\mathrm{kg}) = 1$. The rest of the P vs $W(\omega_W)$ curve is the interesting part. The only restrictions are that it starts at 0, ends at 1, and is monotonic. This is called the cumulative distribution function of the Random Variable.

$$F_W(\mathsf{w}) = P(W(\omega_W) < \mathsf{w}) \tag{A.1}$$

The cumulative distribution function describes the relationship between outcomes, intrinsic events, values, and probability. The probability density function is another way of describing the probability space. It is derived from the cumulative distribution function and is only defined for continuous Random Variables.

$$f_W(\mathsf{w}) = \frac{dF_W(\mathsf{w})}{d\mathsf{w}} \tag{A.2}$$

The probability density is not an absolute measurement of probability; it is relative. In fact, there is no absolute measurement because all individual outcomes have zero probability. For continuous Random Variables, it only makes sense to talk about relative probability of outcomes calculated as ratios of the probability density functions, or the probability of events. In addition to intrinsic events such as $\{\omega_W : W(\omega_W) < 1\,\mathrm{kg}\}$ we are sometimes interested in *derived* events which are derived from intrinsic events: $\{\omega_W : 1\,\mathrm{kg} < W(\omega_W) < 2\,\mathrm{kg}\}$. This intrinsic vs derived convention is not a universal terminology, but is used here to simplify the concepts of outcomes and events.

A.2 Important Details about Probability Theory

A.2.1 Functionals of Random Variables

A functional of a Random Variable is also a Random Variable, with extra context. Say that you wish to sell the rock from Example 24. A very simple model could be: $m = \frac{\$10}{kg} w$ where w is the weight and m is the sale price. Since the weight is uncertain, the price is also uncertain. Therefore the price can be represented with a Random Variable as $M(\omega_M) = \frac{\$10}{kg} W(\omega_W)$. However, $M(\omega_M)$ can also be thought of as $M(\omega_W)$, because every outcome in Ω_M can be mapped back to an outcome in Ω_W. We say that $W(\omega_W)$ is the underlying probability space of $M(\omega_W)$. When a Random Variable has a known underlying probability space, we will show that dependency with the argument of the Random Variable.

A.2.2 Representation of Probability

In this tutorial we use two approaches for manipulating and presenting an uncertain quantity. The first is a direct representation, in which an uncertain quantity is represented by a Random Variable. The other is indirect representation, in which the uncertain quantity is expressed in terms of the Random Variable's value. For example, direct representations are are Random Variables or functionals of Random Variables. Examples of indirect representations are the probability density function or cumulative distribution function. Direct representation enables underlying probability spaces to be explicitly shown, and also allows different kinds of mathematical manipulations of uncertain quantities. For example, integration over a probability space is conceptually easier with indirect representation, whereas nonlinear transformations of a probability space are easier with direct representation.

A.2.3 Uncertain and Stochastic Quantities

One distinction that is important to make is stochastic versus uncertain quantities. Stochastic refers to a quantity that is changing in an unpredictable way, as with reactions that are caused by random collisions in a population of molecules. Properties of stochastic quantities can be characterized by random variables and probability theory. For example a random walk is modeled with a Gaussian distribution.

Uncertain quantities refer to those that are not changing and have a fixed, true value. However, due to our limited knowledge, the true value is not well known. In this case, the probability describes how credible the value is or how likely it is that the value equal to the true value.

Therefore random variables can be used in two ways, illustrated with examples.

1. stochastic: a quantity such as concentration is determined by an unpredictable driving force, and so throughout the system of interest (let's say it is random with respect to time), the concentration will vary according to a distribution. So repeated measurements will result in a distribution. The goal is often to find some correlation with other stochastic processes in order to infer a mechanism.

2. uncertainty: the concentration is constant, but it is unknown because it cannot be measured accurately. Although it is constant, repeated measurements will vary according to a distribution. The goal is often to reduce the variance of the distribution to better infer the 'true' value.

In reality, both uncertainty and stochasticity are present and are impossible to separate. The uncertainty quantification methods from Chapter 3 deal with uncertainty, but stochasticity can also fit into the same framework. For instance, if there is a stochastic process, like a time varying concentration. We may be interested in modeling the uncertainty, at a specific time point. At a single time point, the concentration can be regarded as an uncertain, fixed quantity, with a given distribution, and we may be able to propagate that uncertainty through a time forward model, to predict the concentration at a future time.

L^2 Convergence

In an abstract sense, the uncertainty quantification problem is one of comparing two probability density functions, or even more abstractly - function approximation. When comparing two functions with the same independent variables, the typical metric is the L^2 norm of their difference:

$$\|f(t) - g(t)\| = \left[\int_T [f(t) - g(t)]^2 dt \right]^{\frac{1}{2}} \tag{A.3}$$

This can be extended to comparing two Random Variables that share the same underlying probability space. These are represented here as two functionals of the same Random Variable, $X(\omega)$.

$$\|A(X(\omega)) - B(X(\omega))\| = \left[\int_{\Omega_X} [f_A(x) - f_B(x)]^2 f_X(x)\, dx \right]^{\frac{1}{2}} \tag{A.4}$$

The L^2 norm is a measure of discrepancy between the two Random Variables, averaged over their shared outcome space of Ω_X.

A.3 Estimation Theory

A.3.1 ANOVA

Analysis of Variance, or ANOVA, is a technique used to make statistical inferences about the parameters in a regression models. The technique is based on classical hypothesis testing. In the model development or parameter estimation context, ANOVA is used to accept or reject the hypotheses that individual terms in a regression model are different from zero (whether they are significant or not). The assumptions of the ANOVA process enable the selection of a statistically significant model, whose parameters can be fitted using least squares estimation.

The necessary conditions for ANOVA are:

1. All errors are independent
2. All errors are normally distributed around zero (the measured responses are normally distributed around the true value)
3. All error distributions have the same variance
4. The model is linear with respect to the parameters

If these conditions apply, the variance in the observations can be decomposed into contributions from changes in the independent variables, and contributions that cannot be explained by the independent variables. This tells you which independent variables are significant - meaning which ones correlate with changes in the observations.

Unfortunately, the assumptions ANOVA requires are invalid for most engineering systems.

If these conditions apply, then each observation of response y can be written:

$$y_{ij} = \mu + \tau_i + \varepsilon_{ij} \tag{A.5}$$

for factor i, and observation ij. In this case, μ is the 'true' overall mean of the response for all factors, and τ_i are adjustments to the overall mean due to changes in the independent variables i. Note that $\sum\limits_{i=1}^{I} \tau_i = 0$. The unexplained errors ε_{ij} are randomly distributed as $\varepsilon_{ij} \sim N\left(0, \sigma^2\right)$. These are assumed to be from measurement error.

ANOVA states that the total variance of collected data can be partitioned into components. This partitioning then forms the basis of the following hypothesis testing.

1. Say an experimental design has
   (a) one factor, I levels → I treatments
   (b) J observations at each treatment
   (c) $I \cdot J - 1$ degrees of freedom
2. The total sum of square errors can be split into the sum of square errors due to different factor levels, and the sum of square errors due to measurement errors.
   $SS_{Total} = SS_{Treatments} + SS_{Measurements}$ or equivalently:
   $\sum\limits_{i=1}^{I} \sum\limits_{j=1}^{J} (y_{ij} - \bar{y}_{..})^2 = J \sum\limits_{i=1}^{I} (\bar{y}_{i.} - \bar{y}_{..})^2 + \sum\limits_{i=1}^{I} \sum\limits_{j=1}^{J} (y_{ij} - \bar{y}_{i.})^2$
   - $\bar{y}_{..}$ is the average of all observations an estimate of μ
   - $\bar{y}_{i.}$ is the average of all observations at treatment i - an estimate of $\mu + \tau_i$
   - y_{ij} is observation j at level i. There are a total of IJ observations.
3. Also, define the mean square of errors, which is the sum of square errors divided by the degrees of freedom:
   $MSE = \frac{SS_E}{a(n-1)}$.
4. ANOVA is used to test the hypothesis that multiple populations have the same mean. Above, each treatment can be regarded as a separate population and the means can be compared using the F-test. The same concept can be used to determine goodness of fit, except instead of multiple populations, one population (represented by the collected data) is compared to the model.

The coefficient of determination, a measure of goodness of fit, is calculated as $R^2 = 1 - \frac{SS_{\text{Residuals}}}{SS_{Total}}$, where $SS_{Total} = \sum_{i=1}^{I} (y_i - \bar{y})^2$, $SS_{\text{Regression}} = \sum_{i=1}^{I} (f_i - \bar{f})^2$, and $SS_{\text{Residuals}} = \sum_{i=1}^{I} (y_i - f_i)^2$.

1. The responses are $y_i = g(x_i) + \varepsilon_i$ where $g(x)$ is some unknown function, x are the independent variables, and ε are the measurement errors.
2. The model is $f_i = f(x_i)$.
3. $\bar{y}$ is the average of all measurements
4. $\bar{f}$ is the average of all model predictions
5. Under certain conditions (linear model),

$$SS_{\text{Total}} = SS_{\text{Regression}} + SS_{\text{Residuals}} \tag{A.6}$$

or

$$\sum_{i=1}^{I} (y_i - \bar{y})^2 = \sum_{i=1}^{I} (f_i - \bar{f})^2 + \sum_{i=1}^{I} (y_i - f_i)^2 \tag{A.7}$$

6. The closer R^2 is to one, the better the model fits the data.

Appendix B

Information Theory

B.1 Fisher Information

The Fisher Information is a similar concept to Shannon Information discussed in Section 2.3. Fisher information measures the amount of information that an observable random variable Y carries about unknown parameters represented by the random variable Θ, when they are related by the likelihood function, $L(\theta|y) = f_{Y|\Theta}(y|\theta)$,

Setup

- Given an observable continuous random variable Y, with values y from support $\mathcal{Y}$
- Given unknown parameters represented by random variable Θ, with values θ from the set $\mathcal{T}$
- Y and Θ are related by the likelihood function $L(\theta|y) = f_{Y|\Theta}(y|\theta)$.
  The likelihood describes how much the data y is supported by the choice of parameter θ.
- In the design of experiments context, the y's are the future observations of data from your experiment, and Θ's are the model parameters you need to estimate

The Score

- The score V is the normalized sensitivity of the log likelihood to the parameters. Multiple parameters → multiple scores

$$V(\theta, y) = \frac{\partial}{\partial \theta} \log L(\theta|y) = \frac{1}{L(\theta|y)} \frac{\partial}{\partial \theta} L(\theta|y) \tag{B.1}$$

- $E_{Y|\Theta}[V] = 0$, the expectation with respect to Y given Θ is zero

Fisher Information

- The Fisher Information is the (co)variance of the score, given Θ.
  $\mathcal{I}(\theta) = E_{Y|\Theta}\left[V^2\right] - E_{Y|\Theta}[V]^2 = E_{Y|\theta}\left[V^2\right]$
- $\mathcal{I}(\theta) = E_{Y|\Theta}\left[\left\{\frac{\partial}{\partial\theta}\log f_{Y|\Theta}(y|\theta)\right\}^2\right]$
  The expectation is taken over the conditional density $Y|\Theta$. θ is the independent variable, treated as constant in the calculation.
- $\mathcal{I}(\theta) = \int\limits_{y\in\mathcal{Y}} \left\{\frac{\partial}{\partial\theta}\log f_{Y|\Theta}(y|\theta)\right\}^2 f_{Y|\Theta}(y|\theta)\,dy$
- $\mathcal{I}(\theta) = \int\limits_{y\in\mathcal{Y}} \left\{\frac{1}{f_{Y|\Theta}(y|\theta)}\frac{\partial}{\partial\theta} f_{Y|\Theta}(y|\theta)\right\}^2 f_{Y|\Theta}(y|\theta)\,dy$
- For Multiple Parameters, this becomes a matrix. Each element in the matrix is calculated as:
  $\mathcal{I}_{ij}(\theta) = \int\limits_{y\in\mathcal{Y}} \frac{\partial}{\partial\theta_i}\log f(y|\theta)\,\frac{\partial}{\partial\theta_j}\log f(y|\theta)\,f(y|\theta)\,dy$

$$\mathcal{I}_{ij}(\theta) = E_{Y|\theta}\left[\frac{\partial}{\partial\theta_i}\log f(y|\theta)\,\frac{\partial}{\partial\theta_j}\log f(y|\theta)\right] \tag{B.2}$$

Properties of the Fisher Information

B.2 Connections

B.3 Interpretation

- The Fisher Information can be thought of as a property of the likelihood function.
- Measure of the information that the random variable Y carries about the parameter value θ
- 'Sharpness' of the peak of the density near the maximum likelihood estimate of θ. Sharper = more information.
- When the data matches parameter value well, the likelihood function has a peak.
- This means that the sensitivity of the likelihood function (the score) goes from positive to negative vs θ, but how does it vary with Y?
- Variance of the score wrt Y indicates how much the data impacts the sensitivity of the likelihood function. High impact on sensitivity means that the data is informative.

- Entropy is a property of a density   Related to the volume of the typical set
- Fisher Information is a property of a family of distributions   Related to the surface area of the typical set

B.4 Cramer-Rao Bound

The Cramer-Rao Bound places a limit on how well parameters can be estimated by any unbiased estimator, based on the form of the likelihood function.

Derivation

- Start with the error formula, for parameter estimate $\widehat{\theta}$, true parameter value θ^{true}, and data y

  $e(y) = \widehat{\theta}(y) - \theta^{true}$

  Then $\text{var}(e(y)) = \text{var}\left(\widehat{\theta}(y)\right)$
- Assume estimator is mean-unbiased

  $E_Y[e(y)] = E_Y\left[\widehat{\theta}(y)\right] - \theta^{true} = 0$
- Let $g(y) = \frac{\partial}{\partial\theta}\log p_Y(y;\theta)$

  $E_Y[g(y)] = E_Y\left[\frac{1}{p_Y}\frac{\partial}{\partial\theta}p_Y\right] = \int\limits_{\mathcal{Y}} \frac{1}{p_Y}\left(\frac{\partial}{\partial\theta}p_Y\right)p_Y dy$
- Assuming some regularity conditions, like differentiability

  $= \frac{\partial}{\partial\theta}\int\limits_{\mathcal{Y}} p_Y dy = \frac{\partial}{\partial\theta}1 = 0$
- Using the Cauchy Schwartz inequality

  $[\text{cov}(e(y), g(y))]^2 \leq \text{var}(e(y))\,\text{var}(g(y))$
- LHS is 1^2

  $\text{cov}(e(y), g(y)) = E_Y[e(y)\,g(y)]$

  $= \int\limits_{\mathcal{Y}}\left(\widehat{\theta}(y) - \theta\right)\left(\frac{\partial}{\partial\theta}\log p_Y\right)p_Y dy = \int\limits_{\mathcal{Y}}\widehat{\theta}(y)\left(\frac{\partial}{\partial\theta}p_Y\right)dy - 0$

  Assuming the estimator $\widehat{\theta}$ is valid, it has no dependence on θ. Since it is unbiased, its expected value is θ

  $= \frac{\partial}{\partial\theta}\int\limits_{\mathcal{Y}}\widehat{\theta}(y)(p_Y)\,dy$

  $= \frac{\partial}{\partial\theta}\theta = 1$
- Back to Cauchy Schwartz,

$1 \leq \text{var}(e(y))\,\text{var}(g(y))$

- By definition of Fisher Info
  $\text{var}(e(y)) = \text{var}\left(\widehat{\theta}(y)\right) \geq [\text{var}(g(y))]^{=1} = \mathcal{I}^{-1}(y)$

Notes and Properties

- Let $T(Y)$ be some estimator of the parameter θ based on the data Y
- In the multiple parameter case, $\mathbf{T}(Y)$ is a column vector of estimators of a vector of parameters
- The matrix inequality $\text{var}(\mathbf{T}(Y)) \geq \mathcal{I}^{-1}$ in this case would mean that the matrix $\text{var}(\mathbf{T}(Y)) - \mathcal{I}^{-1}$ is positive semi-definite
- Bound is achieved then the unbiased estimator is called "efficient"
  Also achieves the lowest possible mean squared error of any unbiased estimator
- IF an efficient estimator exists, it is the Maximum Likelihood Estimator. Another way to say this is that the MLE is asymptotically efficient, if any estimator is efficient. Therefore, in the case where another efficient estimator exists, the MLE is also efficient (and equivalent).
- Cramer-Rao Bound varies with θ and $\mathbf{x}$
- If no efficient estimator exists, then Cramer-Rao Bound is not tight, and the tightness is NOT KNOWN. The Fisher Information matrix only gives a LOWER BOUND on the variance of the estimator.
- The bound can be violated by an unbiased estimator.

B.5 Optimal Designs, Fisher Information and the Cramer-Rao Bound

B.5.1 Examples

The FIM works well w/ the assumptions of Optimal DoE. Linearized (in the parameters) model → math is really easy

Example 1

- model $y = \beta_0 + \beta_1 x$
- use three design points then the model equations are $Y = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ 1 & x_3 \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix}$
- FIM= $\begin{bmatrix} 3 & x_1 + x_2 + x_3 \\ x_1 + x_2 + x_3 & x_1^2 + x_2^2 + x_3^2 \end{bmatrix} = \begin{bmatrix} 1 & x_1 \\ x_1 & x_1^2 \end{bmatrix} + \begin{bmatrix} 1 & x_2 \\ x_2 & x_2^2 \end{bmatrix} + \begin{bmatrix} 1 & x_3 \\ x_3 & x_3^2 \end{bmatrix}$
- Not a function of theta (because it is linear)
- Optimal DoE objective is some function of FIM.

In the case where the model is not linear and the errors are not Gaussian, the Cramer-Rao bound is not guaranteed to be tight and there is no estimate of how inaccurate the bound will be. Therefore, the validity of the Fisher Information as a metric is totally unknown.

There are plenty of information theory reasons, geometric interpretations, etc.

1. Parameter uncertainty

   Max FIM   min posterior variance

2. Geometric

   Maximize parameter uncertainty ellipse

   Eigenvalues/ vectors of FIM → ellipse in p-space

   Vectors might not be orthogonal

3. Model uncertainty → Standardized variance ≈ info per experiment

   FOSM uncertainty propagation → Model prediction uncertainty linearized around the design point (as objective for a minmax)

   $d(x, \xi) = N \frac{\text{var}\{\widehat{y}(x)\}}{\sigma^2} = N f^T(x) \left(F^T F\right)^{-1} f(x)$

B.5.2 Decision Theory Framework

We will see that the decision theory framework gives the same results, while making all the steps explicit.

As discussed in Section 5.1.1, the actions are the choice of designs, and are searched using some optimization algorithm.

1. Assume a linear model:

$$y = \mathbf{F}(x)\,\theta \tag{B.3}$$

where $\mathbf{F}(x) = \begin{bmatrix} f_1(x_1) & f_2(x_1) & \cdots & f_P(x_1) \\ f_1(x_2) & f_2(x_2) & \cdots & f_P(x_2) \\ \vdots & & & \\ f_1(x_N) & f_2(x_N) & \cdots & f_P(x_N) \end{bmatrix}$ are all the terms that depend only on the design variables x. Here, x is an N-point design, and the model has P parameters.

2. Assume the true parameter values fall within some known set

$$\theta \in \Theta \tag{B.4}$$

3. Assume the observations are Gaussian

$$d = y + \varepsilon \tag{B.5}$$

where $\varepsilon \sim N\left(0, {\sigma_\varepsilon}^2\right)$

4. Use maximum information gain or minimum posterior variance as the utility metric

5. For every design x

   (a) Use uncertainty quantification to identify all possible datasets (consequences). The prior knowledge of parameter uncertainty, from Equation B.4, is propagated through the model and the observation model is added on to produce all the possible datasets. In classical statistics there is no concept of probability associated with these consequences, so there is no prior predictive density, however, this doesn't matter because the probabilities are ignored by the estimation method anyways.

   (b) For each simulated dataset, estimate the parameters using Maximum Likelihood Estimation. Least Squares estimation is gives the same results in this case. Because of the linear form of the model, this can be done analytically

$$\begin{aligned} d =& \mathbf{F}(x)\,\theta \\ \theta =& \left(\mathbf{F}^T\mathbf{F}\right)^{-1}\mathbf{F}^T d \end{aligned} \tag{B.3}$$

The uncertainty in the parameters is simply proportional to the uncertainty in the data points and is independent of the actual value of the data point. Therefore the utility can be computed without even predicting the data points. So the utilities for all datasets under the same design x will be identical. In addition, because the uncertainty is only a linear combination of the uncertainty in the data points, the Gaussian observation model will result in Gaussian uncertainty for the parameters.

It is critical to note that this requires the observation model to be independent of the value of the model output and design variables. This limits Optimal designs to the simplest of observation models - independent, identically distributed, and additive observation errors.

(c) Compute the expected utility - gain in information, reduction in entropy, and reduction in variance are all equivalent in this case

discuss the failure of the CRB.

These are the steps, working backwards:

1. In order to predict the posterior parameter variance, Optimal designs use the Cramer-Rao Bound. This places a lower bound on the variance of all unbiased estimators.

   Unfortunately, this only provides a bound. It is only met with equality under certain circumstances.

   - Case 1: The system is linear in the parameters, and the errors are distributed normally. Then the Least Squares Estimator DOES meet the bound with equality (most restrictive).
   - Case 2: If the system is not linear or has non-Gaussian errors, the assumption is that there exists an efficient estimator (which would be the Maximum Likelihood Estimator) and that this estimator will be used for the parameter inference. This can be tested, in theory. For the combination of system model, error model, and estimator, the tightness of the bound is unknown.

2. If the system satisfies the terms of the Gauss-Markov Theorem, then the Least Squares Estimator is the Best Linear Unbaised Estimator, meaning it has the minimum variance of all possible unbiased estimators. It does not mean that the Cramer-Rao Bound is met.

   This is the case for linear systems with non-Gaussian errors.

3. Assuming the Cramer-Rao Bound holds with equality, the predicted covariance of the posterior parameters is equal to the inverse of the Fisher Information Matrix, which depends on both x and θ through the likelihood function. The issue here is that the value of θ is unknown, so a point estimate is used.

   The solution to the optimal design problem is sensitive to the choice of linearization point (for the parameters). The effects of this linearization will depend on the model.

Note that these assumptions involve the selection of an estimator (an unknown estimator that is efficient) that is able to achieve the Cramer-Rao Bound. These assumptions are fine for some engineering applications, however, as nonlinear models, non-Gaussian errors, or even errors with non-uniform variance come into use, these will surely be invalid.

Using the Cramer-Rao Bound, Optimal DoE basically replaces estimator variance with inverse Fisher Information matrix. Because of the reciprocity of estimator-variance and Fisher information, minimizing the variance *loosely* corresponds to maximizing the information, but the correspondence is not completely rigorous.

I ran some case studies:

- Baseline - Linear system, Gaussian errors, Least Squares Estimator
- Case 1 - Nonlinear system, Gaussian errors, Least Squares Estimator
- Case 2 - Linear system, lognormal errors, Least Squares Estimator
- Case 3 - Linear system, uniform errors, Least Square Estimator

The Least Squares Estimator is used in all cases, because it is the most common. In many cases, the parameter variance cannot be accurately estimated using LSE. The algorithm for these tests is:

1. Assume some values for the model parameters, and measurement error distribution (likelihood function)
2. Calculate the FIM, assuming that the current parameter estimates are the 'true' parameter values
3. Given a set of design points, calculate the Cramer Rao Bound
4. Generate a large number of estimates, using the same design points. From this, get the covariance of the estimator and compare to Cramer-Rao Bound.

B.5.3 Baseline - Linear system, Gaussian Errors, Least Squares Estimator

- System - $f(x;\theta) = \theta_1 + \theta_2 x$
- True Parameters- θ^*
- Data - $Y \sim N\left(\theta_1 + \theta_2 x, \sigma_e^2\right)$, realizations y
- Likelihood Function

  $L(Y|x,\theta) = \frac{1}{(2\pi\sigma_e^2)} \exp\left(-\frac{[y-f(x;\theta)]^2}{2\sigma_e^2}\right)$

  The likelihood is Gaussian, centered around the data with variance given by σ_e^2, which is known. This is the likelihood that these parameters can explain the data.

  $l(Y|x,\theta) = \log L(Y|x,\theta) = C - \frac{[y-(\theta_1+\theta_2 x)]^2}{2\sigma_e^2}$

The Score

$$\frac{\partial}{\partial\theta_i} l(Y|x,\theta) = \frac{y-(\theta_1+\theta_2 x)}{\sigma_e^2} \begin{bmatrix} 1 \\ x \end{bmatrix}$$

Fisher Information Matrix

$$\mathcal{I}(x) = E_{Y|\theta}\begin{bmatrix} \left(\frac{y-(\theta_1+\theta_2 x)}{\sigma_e^2}\right)^2 & \left(\frac{y-(\theta_1+\theta_2 x)}{\sigma_e^2}\right)\left(x\frac{y-(\theta_1+\theta_2 x)}{\sigma_e^2}\right) \\ \left(\frac{y-(\theta_1+\theta_2 x)}{\sigma_e^2}\right)\left(x\frac{y-(\theta_1+\theta_2 x)}{\sigma_e^2}\right) & \left(x\frac{y-(\theta_1+\theta_2 x)}{\sigma_e^2}\right)^2 \end{bmatrix}$$

$$\mathcal{I}(x) = \frac{1}{\sigma_e^4}\begin{bmatrix} 1 & x \\ x & x^2 \end{bmatrix} E_{Y|\theta}\left[(y-(\theta_1+\theta_2 x))^2\right] = \frac{1}{\sigma_e^4}\begin{bmatrix} 1 & x \\ x & x^2 \end{bmatrix} E_{Y|\theta}\left[(\theta_1+\theta_2 x)^2 - 2y(\theta_1+\theta_2 x) + y^2\right]$$

$$\mathcal{I}(x) = \frac{1}{\sigma_e^4}\begin{bmatrix} 1 & x \\ x & x^2 \end{bmatrix}\left((\theta_1+\theta_2 x)^2 - 2(\theta_1+\theta_2 x)E[y] + \mathrm{var}(y) + E[y]^2\right)$$

$$\mathcal{I}(x) = \frac{1}{\sigma_e^4}\begin{bmatrix} 1 & x \\ x & x^2 \end{bmatrix}\left((\theta_1+\theta_2 x)^2 - 2(\theta_1+\theta_2 x)^2 + \sigma_e^2 + (\theta_1+\theta_2 x)^2\right) = \frac{1}{\sigma_e^2}\begin{bmatrix} 1 & x \\ x & x^2 \end{bmatrix}$$

Result

- Compare the Cramer Rao Bound to the distribution of parameter estimates - realizations of the Estimator for different values of the data
- The covariance of the estimators is exactly equal to the Cramer Rao Bound - the Least Squares Estimator is *efficient* for linear Gaussian systems
- This is a special case - all models that are linear in the parameters (can be nonlinear in the

independent variables) and have Gaussian errors will meet the Cramer Rao Bound exactly, for all x.

Case 1 - Nonlinear system, Gaussian Errors, Least Squares Estimator

- Model - $f(x;\theta) = \theta_1 + x^{\theta_2}$
- Data - $Y \sim N\left(f(x;\theta^*), \sigma_e^2\right)$, realizations y
- Parameters
  - True - θ^*
  - Estimates - $\widehat{\theta}$
- Likelihood Function
  $L(Y|x,\theta) = \frac{1}{(2\pi\sigma_e^2)} \exp\left(-\frac{[y-f(x;\theta)]^2}{2\sigma_e^2}\right)$
  The likelihood is Gaussian, centered around the data with variance given by σ_e^2, which is known. This is the likelihood that these parameters can explain the data.
  $l(Y|x,\theta) = \log L(Y|x,\theta) = C - \frac{[y-(\theta_1+x^{\theta_2})]^2}{2\sigma_e^2}$

The Score

$$\frac{\partial}{\partial\theta_i} l(Y|x,\theta) = \frac{y-(\theta_1+x^{\theta_2})}{\sigma_e^2}\begin{bmatrix} 1 \\ x^{\theta_2}\log x \end{bmatrix}$$

Fisher Information Matrix

$$\mathcal{I}(x) = \frac{1}{\sigma_e^4}\begin{bmatrix} 1 & x^{\theta_2}\log x \\ x^{\theta_2}\log x & \left(x^{\theta_2}\log x\right)^2 \end{bmatrix} E_{Y|\theta}\left[\left(\frac{y-(\theta_1+x^{\theta_2})}{\sigma_e^2}\right)^2\right]$$

$$= \frac{1}{\sigma_e^4}\begin{bmatrix} 1 & x^{\theta_2}\log x \\ x^{\theta_2}\log x & \left(x^{\theta_2}\log x\right)^2 \end{bmatrix} E_{Y|\theta}\left[\left(\theta_1 + x^{\theta_2}\right)^2 - 2y\left(\theta_1 + x^{\theta_2}\right) + y^2\right]$$

$$= \frac{1}{\sigma_e^4}\begin{bmatrix} 1 & x^{\theta_2}\log x \\ x^{\theta_2}\log x & \left(x^{\theta_2}\log x\right)^2 \end{bmatrix} \left(\left(\theta_1 + x^{\theta_2}\right)^2 - 2\left(\theta_1 + x^{\theta_2}\right)^2 + \sigma_e^2 + \left(\theta_1 + x^{\theta_2}\right)^2\right)$$

$$\mathcal{I}(x) = \frac{1}{\sigma_e^2}\left.\begin{bmatrix} 1 & x^{\theta_2}\log x \\ x^{\theta_2}\log x & \left(x^{\theta_2}\log x\right)^2 \end{bmatrix}\right|_{\theta=\widehat{\theta}}$$

Result

- The resulting estimator has a covariance matrix that violates the Cramer Rao Bound... Perhaps I made a mistake.

  I used the true parameters to calculate the Fisher Information Matrix, with some value for the measurement error variance.
- I'm pretty sure this isn't correct
  - only 1000 samples were used, however, I ran it a couple times with consistent results
  - Or is the error introduced by linearizing the FIM responsible?
  - did the math wrong?

Case 2 - Linear system, Lognormal Errors, Least Square Estimator

- Model - $f(x;\theta) = \theta_1 + \theta_2 x$
- Independent var - x
- Data - $Y \sim \log\mathcal{N}\left(\log\left(\frac{\mu^2}{\sqrt{\sigma_e^2+\mu^2}}\right), \sqrt{\log\left(1+\frac{\sigma_e^2}{\mu^2}\right)}\right) - \mu + \theta_1 + \theta_2 x$

  Lognormal errors centered (mean unbiased, not median) on the model, variance σ_e^2. Has realizations y
- Define new Random Variable $K = Y + \mu - (\theta_1 + \theta_2 x)$ and constants m and s

  $K \sim \log\mathcal{N}\left(\log\left(\frac{\mu^2}{\sqrt{\sigma_e^2+\mu^2}}\right), \sqrt{\log\left(1+\frac{\sigma_e^2}{\mu^2}\right)}\right)$

  $K \sim \log\mathcal{N}\left(m, s^2\right)$
- Parameters
- True - θ^*
- Estimates - $\widehat{\theta}$
- Likelihood Function

  $L(K|x,\theta) = \frac{1}{k\sqrt{2\pi s^2}} \exp\left(-\frac{\{\log(k)-m\}^2}{2s^2}\right)$

  $l(K|x,\theta) = C - \log k - \frac{\{\log(k)-m\}^2}{2s^2}$

The Score

$$\frac{\partial}{\partial\theta_i} l\left(K|x,\theta\right) = -\frac{1}{k}\frac{dk}{d\theta_i} - \frac{\{\log(k)-m\}}{s^2}\frac{1}{k}\frac{dk}{d\theta_i}$$

$$\frac{\partial}{\partial\theta_i} l\left(K|x,\theta\right) = \left\{\frac{1}{k} + \frac{\{\log(k)-m\}}{s^2}\frac{1}{k}\right\}\begin{bmatrix}1\\x\end{bmatrix}$$

Fisher Information Matrix

$$\mathcal{I}\left(x\right) = \begin{bmatrix}1 & x\\x & x^2\end{bmatrix} E_{K|\theta}\left[\left(\frac{1}{k} + \frac{\log(k)-m}{ks^2}\right)^2\right]$$

$$\mathcal{I}\left(x\right) = \begin{bmatrix}1 & x\\x & x^2\end{bmatrix} E_{K|\theta}\left[\left\{\frac{1}{k} + \frac{\log(k)-\log\left(\frac{\mu^2}{\sqrt{\sigma_e^2+\mu^2}}\right)}{k\log\left(1+\frac{\sigma_e^2}{\mu^2}\right)}\right\}^2\right]$$

err... we'll just calculate that numerically.

In terms of Y

$$L\left(Y|x,\theta\right) = \frac{1}{(y+\mu-(\theta_1+\theta_2x))\sqrt{2\pi s^2}}\exp\left(-\frac{\{\log(y+\mu-(\theta_1+\theta_2x))-m\}^2}{2s^2}\right)$$

$$l\left(Y|x,\theta\right) = C - \log\left(y+\mu-(\theta_1+\theta_2x)\right) - \frac{\{\log(y+\mu-(\theta_1+\theta_2x))-m\}^2}{2s^2}$$

$$\frac{\partial}{\partial\theta_i} l\left(Y|x,\theta\right) = \frac{1}{y+\mu-(\theta_1+\theta_2x)}\left(1+\frac{1}{s^2}\left[\log\left\{y+\mu-(\theta_1+\theta_2x)\right\}-m\right]\right)\begin{bmatrix}1\\x\end{bmatrix}$$

$$\mathcal{I}\left(x\right) = \begin{bmatrix}1 & x\\x & x^2\end{bmatrix} E_{Y|\theta}\left[\left\{\frac{1}{y+\mu-(\theta_1+\theta_2x)}\left(1+\frac{1}{s^2}\left[\log\left\{y+\mu-(\theta_1+\theta_2x)\right\}-m\right]\right)\right\}^2\right]$$

Result

- In this case, for several different values of μ and σ_e, the Cramer-Rao Bound is NOT met.
- I'm not sure how to measure or 'rank' the discrepancy between the estimator covariance and the Cramer-Rao Bound, because it has matrix form.
- As the number of data points increases, the Cramer-Rao Bound decreases, the estimator variance decreases, and the discrepancy decreases.
- As the magnitude of the design point increases, the discrepancy decreases (makes sense because MPU increases quadratically with distance from $x = 0$)
- The Gauss-Markov theorem states that the Ordinary Least Squares Estimator is the Best Linear Unbiased Estimator, so no other estimator can give better estimators. Therefore,

there is no efficient estimator.

Appendix C

Design of Experiments Theory

C.1 General Principles of Classical Approaches

The four basic criteria for classical design of experiments are: randomization, replication, blocking, and orthogonality. The common theme is to eliminate bias and isolate the effect of each factor, in order to obtain the best possible statistics. Each of these is meant to address the concern of identifying large trends in the presence of large uncertainties. These are quite important and effective for studies where the uncertainties are too large to even predict system performance. However, this is not the case in most engineering studies.

Randomization is used to minimize the effect of all potential biases in experiments. It refers to randomizing all factors which are not explicit set in the experimental design, such as: the ordering of the trials, the operator, the time of day, etc. Each experiment should be as independent as possible. This is necessary for the assumption that all errors are random, independent, and unbiased - and the expected value of the errors is zero. This way, even if unknown, uncontrollable biases are present, they will be randomly spread throughout all trials, which minimizes their ability to impact the conclusions.

Replication refers to experimental repeats, with the purpose of achieving more accurate statistics. To clarify, this refers to repeats of entire experiments, under the same conditions and procedures in addition to repeated measurements during a single experiment. Measurements of a system response are only estimates of the true value, and replicates allow for more accurate estimates, as long as the estimates are unbiased.

Blocking is a technique to isolate the effect of a particular factor. For example, to determine the effect of different operators on product quality: one block of experiments, containing all the treatments, should be completed by technician A and the second by technician B. The difference between the two blocks - and thus the effect of the factor in question, will be evident in the resulting statistics. This is commonly done to eliminate the effect of nuisance factors - factors that have significant impact on the response, but do not concern the objective of the experiment.

Lastly, orthogonality refers to the use of factors that are uncorrelated, meaning that the response to a factor is unaffected by the value of other factors. Again, this is related to isolating the effect of each factor, independently from all others.

Ideally, if all four of the basic principles could be applied to a system, a full characterization would be possible by analyzing each factor separately. Unfortunately, it is not always possible to make systems conform to these principles. Bias and factor interaction are inevitable, so additional heuristics exist to guide the design.

Limitations

1. Large number of experiments: As previously stated, the number of experiments increases exponentially with the number of factors. Even then, the factor interactions are not always captured without further experiments.
2. Dependence on fixed design forms, no formal method of adapting to different situations
3. Assumption of independent and identically normally distributed errors: This is almost never true, which means that the ANOVA results will be incorrect.
4. While many random processes can be well approximated as normal, the systems response to those random variations is often not normally distributed.
5. Errors are very often correlated, meaning that the shape of the confidence region is not a hypercube.
6. Correlation, not causation. There is little insight given by the form of experiments. Without further experiments, only an empirical model can be created from classical designs. Note that this limitation is true for all experimental designs.
7. Focus is on isolating the effect of each factor, there is no use of prior knowledge, which could capitalize on the known interactions and system behavior
8. The domain of the factors may not be entirely feasible, meaning that some design points of

a 2^f design may be impossible to run.

There are a number of limitations to this heuristic. The full factorial design does not always suggest the best design points for systems with nonlinear responses or irregular design spaces. Only interactions and significant factors can be identified using this heuristic. Not only would additional experiments would be required to determine the functional form of the system response, but the statistical approaches would not apply to the results? Imagine that x_1 were volumetric flowrate and x_2 were concentration of species A, and the system required that the minimum molar flowrate of A was c. Then a constraint, $x_1 x_2 \geq c$, is placed on the design space. While Model Based Experimental Designs can calculate alternate optimal design points, classical experimental designs must rely on the designers experience to choose alternate design points.

C.2 Summary

These are great if there is so much uncertainty that there is not good model for the system or the measurements. But this should not be the case for engineering applications. In the case where we do have useful prior knowledge about the system we should look for other approaches.

C.3 The Role of Experimental Design

The evolving role... before focused on the experiments and the results, now that part is done. need to focus on making it more efficient. focus on the design. Why the Chemical Engineering is Ready for a Change

Much better modeling, much better computational capabilities, economic environment, acceptance of models as reflections of reality.

C.4 Organizing Experimental Studies

C.4.1 The Study Formulation Process

In this work we define: System, study, experiment,rounds of experiments, A study can be broken into the following steps:

1. Problem statement

   Define the goals of the study

2. Select the response variables of interest

   Choose the variables that the model should predict, and devise a metric to judge the quality of the model

3. Assess and quantify prior knowledge

   Identify factors/control variables

   Use either existing information or run preliminary experiments to determine significant factors, reasonable ranges to study

4. Experimental design

   Determine the best design points at which to carry out experiments

5. Run experiments

6. Iterate steps 3-5 until the goals have been achieved

The Design of Experiments Process

Steps 2 and 3 could both require an experimental design. Prior knowledge of the system must be used to design the preliminary experiments which tend to be very broad and do not require high accuracy. To save costs and time, the number and range of factors should be minimized, while still achieving the study objective. In this step, the goal is to find any correlation between the factors and the responses, and determine which factors have a significant effect. After this, a classical design heuristic is often used to design experiments for model development, which are more focused and accurate. In the following sections, only the design of experiments for model development is discussed. Classical designs follow four basic criteria and then apply one of many possible heuristics. These are described below, along with a summary of analysis of variance techniques.

Assess Study Goals

Incorporate new information, and iterate until done.

Lastly the new model parameters are estimated, resulting in an improved model representing the *posterior* knowledge, and the model is used to address the goal of the study.

C.4.2 The Design of Experiments Scenario

Seems like a lot of work! So when should you use these computationally expensive DoE approaches?

when doing DoE will substantially reduce the cost of achieving the experimental goal. If experiments are cheaper than doing computation, then why bother? but most cases you should do DoE. example: if you want to find out what happens when you do a specific thing, then just do that thing.

C.5 Prior Knowledge

C.5.1 What do you know???

- Know the system
  -
  - Process driven by physic and chemistry
  - Kinetics, thermodynamics, etc.
  - Interaction with environment
  - Key variables, quantities
  - How system varies
- Know the experimental equipment
  -
  - How the system will be examined
  - Measurement accuracy, precision
  - Know what information you want to learn

C.5.2 How can you use this?

Formalize by organizing the prior knowledge into 5 types

- System models Based on the physics and chemistry
- Parametric uncertainties

  True values are unknown, but can represent what IS known using probability density functions

- Model uncertainties

  Characterize inaccuracies or variability in the model

  Can be parameterized

  While parametric uncertainties do not depend on the state of the system, model uncertainty most likely will. ie; you know that the model is accurate for the low Reynold's number regime, but not the high Re regime.

  ie: The model doesn't account for heat losses, so inaccuracy will grow with time.

- Equipment uncertainties

  Errors in measurements, etc that occur outside the true system, and thus should not be in the process model, but do cause extra variability.

- Objective Function quantify the goals of the study

System Model

In order to utilize this prior knowledge, there must be a model of the system. This system model,

$$y = f\left(x, z, \theta, \eta\right)$$

will be a function of the design variables, x, which can be controlled in an experiment, the nuisance variables, z, which cannot be controlled but are considered known, the unknown parameters, θ, and the known parameters, η, which are considered constants. These functions can be in the form of a simple polynomial response surface or a process simulation with dozens of units.

Parametric Uncertainty

Parameters are values that are constant across the design space. In the Bayesian approach, uncertain parameters can be represented with random variables. The probability density of each outcome (parameter value) can be interpreted as a degree of belief that the outcome is equal to the true parameter value.

Model Uncertainty

Sometimes the model is known to be inadequate to explain the true system. This may be from missing terms or simplifications. This is recognized as model uncertainty, and can also be captured as a random variable. These terms can be treated similarly to parametric uncertainty, except they will likely vary with the design space.

Equipment Uncertainty

measurements, etc. often has bias. most people assume these are normal. but usually they are not. there is bias, there combined effects. if you can do better, you should.

C.6 Overview

For this project, model based experimental design refers only to the selection of design points for future experiments (the operating conditions at which to collect data). The remainder of the study formulation (problem statement, selection of response variables, etc.) is the same as in classically design studies and is described in the Appendix. Model based experimental designs replace the design heuristics of classical designs with a framework that selects design points by using both knowledge of prior system behavior and model prediction of future behavior. In general the Model Based Experimental Design framework can be broken down into seven steps as shown in Figure 3-4 and described in the following sections. Keep in mind that this framework is preceded by the prior steps in the overall study formulation, as discussed in the Appendix section on Classical Design of Experiments. The general strategy is to run through many iterations of the framework to develop successively better models, until the study objective has been met.

1. Model Development
2. Estimate measurement uncertainty (variance of the residuals)
3. Parameter Estimation
4. Uncertainty Quantification
5. Optimization
6. Experiments
7. Data Analysis

The specific formulation for model based experimental design can vary; the framework integrates each of the procedures described above: parameter estimation, uncertainty propagation, and optimization. As previously discussed, there are a variety of techniques available for each of these procedures and these techniques can be classified as either Bayesian or non-Bayesian. This can be confusing because the three procedures in the Model Based Experimental Design framework are fairly independent - a Bayesian parameter estimation technique can be used with a non-Bayesian uncertainty propagation technique. Model Based Experimental Design refers to the framework. Bayesian Experimental Design will be used to refer to a Model Based Experimental Design which incorporates a Bayes theorem in the utility function for optimization, even if non-Bayesian techniques are used for the other steps.

C.6.1 Comments on Optimal- and Model Based- Experimental Designs

In a D-optimal design, the objective is:

$$\text{minimize } \det \tilde{V} \text{ where } \tilde{V} = \left[\mathrm{B}^T \Pi^{-1} \mathrm{B}\right]^{-1}$$

Compared to Model Based Experimental Designs:

$$\text{minimize } \det \tilde{V} \text{ where } \tilde{V} = \left[\mathrm{B}^T \Pi^{-1} \mathrm{B} + V_0^{-1}\right]^{-1}$$

The only difference between approaches is the inclusion of the prior parameter variance, which arises from the Bayesian use of prior information. Interestingly, the Optimal Design is derived from Fisher Information, whereas the Model Based Experimental Design is derived from Shannon Information. The common assumptions reduce the two approaches to the same results. To make use of complex models and uncertainty representations, those assumptions must be removed. This leads to the Bayesian Design Approach.

C.6.2 Example

Consider a dryer as an example system. The framework is described, then two scenarios are presented: a) the first iteration of the model based experimental design, and b) any future iterations. Following the framework description is a comparison of model based and classical designs. 1. A model is built to represent the system by using first principles such as heat and mass balances or transport phenomena, as well as prior data. It is also possible to use Process Simulation software such as AspenPlus to simplify this step. a. The model has multiple responses (exit temperature, water content, etc.), independent variables (flowrate, composition, etc.), and parameters (mass and heat transfer properties, etc.) identified from first principles. b. Based on prior data, the structure of the model is modified to more accurately represent the system. The list of independent variables and parameters is expanded or reduced to include only terms with significant impact on the responses.

2. The measurement uncertainty (variance of the residuals) is estimated. a. This might come from knowledge of measurement capabilities or prior experiences. b. Uncertainty can be calculated directly from the data

3. The parameter values are estimated and their uncertainty is characterized. a. This can come from previous studies which might list upper and lower bounds and distributions, or from educated guesses. b. Nonlinear parameter estimation techniques are applied to the data

Steps 4-7 are the same for both scenarios

4. The model representation allows the calculation of model uncertainty using uncertainty propagation techniques. This allows analysis of the contribution of each parameter to the uncertainty in the model response.

5. Estimates are available for 1) the uncertainty of the measurements (the variance of the residuals), 2) the uncertainty in the parameter estimates, and 3) predictions of model uncertainty for different values of the independent variables. Using a combination of these three uncertainty estimates, a measure of uncertainty in the model predictions is used to formulate an objective function. In step five, an optimization method is used to maximize the 'model prediction uncertainty' over the domain of the independent variables (the design space), to determine the best design point.

6. Running experiments at the best design point will yield the most useful information as determined by the chosen objective function.

7. The data is analyzed to determine whether the model can acceptably explain the data. If not, another iteration is required to determine another model structure or decrease the parameter uncertainty.

In a classical design, a set of experiments are carried out and then a model is built from the data. Then a new design is selected and the process iterates. The major difference with model based designs is how the design is selected. Classical designs must rely on intuition or predefined sets of experiments, while model based designs use algorithms. The objective function used in Step 5 ensures that the experiments reveal information about the most significant parameters, while ignoring the least significant parameters.

Appendix D

Utility Functions and the Prior Sampling Formulation

This appendix contains derivations and detailed explanations of the Prior Sampling Formulations of several utility functions. These were introduced in Section 6.4 and used extensively in the examples and studies. These were derived because some probability density functions are not known, but can be easily sampled. The standard Bayesian Design of Experiments algorithm does not take advantage of this, and requires the probability density function to be approximated. The Prior Sampling Formulation allows this step to be skipped and computes the objective function using only samples, which increases the accuracy.

D.1 Prior Sampling Formulation of Information Metrics for Design of Experiments

This section shows the derivation of several Prior Sampling Formulation for different information metrics. All are based on the work of Ryan [71]. These formulations are very attractive because they do not need to generate the posterior parameter density before computing the information metrics. This means that one fewer high-dimensional integration step is reqiured.

D.1.1 Formulation for Estimating Parameters

Throughout this derivation, every density, utility value, and objective function is computed for a specific design x. This dependence is not shown to conserve space.

- The utility function is the Kullback-Leibler divergence from posterior parameter density to prior parameter density:

$$\phi = -\int\limits_{\Theta} f_{\Theta|D}\left(\theta|\mathsf{d}\right) \log \frac{f_{\Theta|D}\left(\theta|\mathsf{d}\right)}{f_{\Theta}\left(\theta\right)} d\theta \tag{D.1}$$

$$\phi = -\int\limits_{\Theta} f_{\Theta|D}\left(\theta|\mathsf{d}\right) \log f_{\Theta|D}\left(\theta|\mathsf{d}\right) d\theta + \int\limits_{\Theta} f_{\Theta|D}\left(\theta|\mathsf{d}\right) \log f_{\Theta}\left(\theta\right) d\theta$$

- The risk metric is the expected value of the utilities, averaged over all possible datasets, at the design x.

$$\Phi = E_D\left[\phi\right] \tag{6.3}$$

- This gives us the risk metric shown in Equation D.2:

$$\Phi = \int\limits_{\Omega_D} f_D\left(\mathsf{d}\right) \int\limits_{\Omega_\Theta} f_{\Theta|D}\left(\theta|\mathsf{d}\right) \log \frac{f_{\Theta|D}\left(\theta|\mathsf{d}\right)}{f_{\Theta}\left(\theta\right)} d\theta d\mathsf{d} \tag{D.2}$$

- Bayes' Theorem is applied to the posterior parameter density

$$\Phi = \int\limits_{\Omega_D} f_D\left(\mathsf{d}\right) \int\limits_{\Omega_\Theta} \frac{f_{\Theta}\left(\theta\right) f_{D|\Theta}\left(\mathsf{d}|\theta\right)}{f_D\left(\mathsf{d}\right)} \log \frac{f_{\Theta}\left(\theta\right) f_{D|\Theta}\left(\mathsf{d}|\theta\right)}{f_D\left(\mathsf{d}\right) f_{\Theta}\left(\theta\right)} d\theta d\mathsf{d}$$

$$\Phi = \int\limits_{\Omega_D} f_D\left(\mathsf{d}\right) \int\limits_{\Omega_\Theta} \frac{f_{\Theta}\left(\theta\right) f_{D|\Theta}\left(\mathsf{d}|\theta\right)}{f_D\left(\mathsf{d}\right)} \log \frac{f_{D|\Theta}\left(\mathsf{d}|\theta\right)}{f_D\left(\mathsf{d}\right)} d\theta d\mathsf{d}$$

- The prior predictive density is not a function of θ, so it can be brought inside of the integral and canceled

$$\Phi = \int_{\Omega_D}\int_{\Omega_\Theta} f_D\left(\mathsf{d}\right)\frac{f_\Theta\left(\theta\right)f_{D|\Theta}\left(\mathsf{d}|\theta\right)}{f_D\left(\mathsf{d}\right)}\log\frac{f_{D|\Theta}\left(\mathsf{d}|\theta\right)}{f_D\left(\mathsf{d}\right)}d\theta d\mathsf{d}$$

$$\Phi = \int_{\Omega_D}\int_{\Omega_\Theta} f_\Theta\left(\theta\right)f_{D|\Theta}\left(\mathsf{d}|\theta\right)\log\frac{f_{D|\Theta}\left(\mathsf{d}|\theta\right)}{f_D\left(\mathsf{d}\right)}d\theta d\mathsf{d}$$

- The logarithm is expanded

$$\Phi = \int_{\Omega_D}\int_{\Omega_\Theta} f_\Theta\left(\theta\right)f_{D|\Theta}\left(\mathsf{d}|\theta\right)\log f_{D|\Theta}\left(\mathsf{d}|\theta\right)d\theta d\mathsf{d} - \int_{\Omega_D}\int_{\Omega_\Theta} f_\Theta\left(\theta\right)f_{D|\Theta}\left(\mathsf{d}|\theta\right)\log f_D\left(\mathsf{d}\right)d\theta d\mathsf{d}$$

- The second term can be reorganized and the parameter density marginalized out:

$$\Phi = \int_{\Omega_D}\int_{\Omega_\Theta} f_\Theta\left(\theta\right)f_{D|\Theta}\left(\mathsf{d}|\theta\right)\log f_{D|\Theta}\left(\mathsf{d}|\theta\right)d\theta d\mathsf{d} - \int_{\Omega_D}\int_{\Omega_\Theta} f_\Theta\left(\theta\right)f_{D|\Theta}\left(\mathsf{d}|\theta\right)d\theta\log f_D\left(\mathsf{d}\right)d\mathsf{d}$$

$$\Phi = \int_{\Omega_D}\int_{\Omega_\Theta} f_\Theta\left(\theta\right)f_{D|\Theta}\left(\mathsf{d}|\theta\right)\log f_{D|\Theta}\left(\mathsf{d}|\theta\right)d\theta d\mathsf{d} - \int_{\Omega_D} f_D\left(\mathsf{d}\right)\log f_D\left(\mathsf{d}\right)d\mathsf{d} \qquad \text{(D.3)}$$

The details of how to sample from this risk metric are shown in Section 6.4.1.

D.1.2 Estimating a Subset of Parameters

For this information metric, only a subset of the parameters need to be estimated. Therefore a new formulation is required. The parameters Θ are broken into Λ which we wish to estimate and Γ which we do not. The utility function is the Kullback-Leibler Divergence of the posterior to prior densities of Λ.

$$D_{KL}\left(f_{\Lambda|D}\left(\lambda|\mathbf{d}\right) \parallel f_{\Lambda}\left(\lambda\right)\right) = \int\limits_{\Omega_\Lambda} f_{\Lambda|D}\left(\lambda|\mathbf{d}\right) \log \frac{f_{\Lambda|D}\left(\lambda|\mathbf{d}\right)}{f_{\Lambda}\left(\lambda\right)} d\lambda$$

Starting with Equation D.2

- The equation is modified for the new utility function:

$$\Phi = \int\limits_{\Omega_D} f_D\left(\mathbf{d}\right) \int\limits_{\Omega_\Lambda} f_{\Lambda|D}\left(\lambda|\mathbf{d}\right) \log \frac{f_{\Lambda|D}\left(\lambda|\mathbf{d}\right)}{f_{\Lambda}\left(\lambda\right)} d\lambda d\mathbf{d} \tag{D.4}$$

- The derivation is the same as for the full set of parameters. The equivalent formulation to Equation D.3 is:

$$\Phi = \int\limits_{\Omega_D}\int\limits_{\Omega_\Lambda} f_{\Lambda}\left(\lambda\right) f_{D|\Lambda}\left(\mathbf{d}|\lambda\right) \log f_{D|\Lambda}\left(\mathbf{d}|\lambda\right) d\lambda d\mathbf{d} - \int\limits_{\Omega_D} f_D\left(\mathbf{d}\right) \log f_D\left(\mathbf{d}\right) d\mathbf{d}$$

- Now substitute in

$$f_{\Lambda}\left(\lambda\right) = \int\limits_{\Omega_\Gamma} f_{\Gamma,\Lambda}\left(\gamma,\lambda\right) d\gamma$$

$$f_{D|\Lambda}\left(\mathbf{d}|\lambda\right) = \int\limits_{\Omega_\Gamma} f_{\Gamma}\left(\gamma\right) f_{D|\Gamma,\Lambda}\left(\mathbf{d}|\gamma,\lambda\right) d\gamma$$

- The final risk metric is:

$$\begin{aligned} \Phi = & \int\limits_{\Omega_D}\int\limits_{\Omega_\Lambda} f_{\Lambda}\left(\lambda\right) \left[\int\limits_{\Omega_\Gamma} f_{\Gamma}\left(\gamma\right) f_{D|\Gamma,\Lambda}\left(\mathbf{d}|\gamma,\lambda\right) d\gamma\right] \left[\log \int\limits_{\Omega_\Gamma} f_{\Gamma}\left(\lambda\right) f_{D|\Gamma,\Lambda}\left(\mathbf{d}|\gamma,\lambda\right) d\gamma\right] d\gamma d\mathbf{d} \\ & - \int\limits_{\Omega_D} f_D\left(\mathbf{d}\right) \log f_D\left(\mathbf{d}\right) d\mathbf{d} \end{aligned} \tag{D.5}$$

- The algorithm used to evaluate the risk metric, using Monte Carlo sampling is shown in Section 6.4.1.

D.1.3 Improving Model Outputs

Often times we are interested in running experiments in order to improve the model outputs. Given some prior parameter knowledge, the model output to be improved is then:

$$f_G(g) = \mathcal{M}[\theta; x]$$

The Random Variable $G(\omega)$ is used instead of $Y(\omega)$ because the model output of interest may not represent the system property that is being measured. In this case, data is simulated using $Y(\omega)$ and $D(\omega)$, in order to reduce the uncertainty of $G(\omega)$. We will call $G(\omega)$ the target Random Variable.

One potential utility function is the Kullback-Leibler divergence from the posterior to the prior density of $G(\omega)$:

$$\phi = D_{KL}\left(f_{G|D}(\mathsf{g}|\mathsf{d})\,||f_G(\mathsf{g})\right)$$

The Prior Sampling Formulation is more complicated than for estimating parameters. Starting with the expected value of the utility, and substituting in the Kullback-Leibler divergence:

$$\Phi = \int\limits_{\Omega_D} f_D(\mathsf{d}) \int\limits_{\Omega_G} f_{G|D}(\mathsf{g}|\mathsf{d}) \log \frac{f_{G|D}(\mathsf{g}|\mathsf{d})}{f_G(\mathsf{g})} d\mathsf{g}d\mathsf{d}$$

As before, Bayes' Theorem is used to compute the posterior target density.

$$\Phi = \int\limits_{\Omega_D} f_D(\mathsf{d}) \int\limits_{\Omega_G} \frac{f_G(\mathsf{g})\, f_{D|G}(\mathsf{d}|\mathsf{g})}{f_D(\mathsf{d})} \log \frac{f_G(\mathsf{g})\, f_{D|G}(\mathsf{d}|\mathsf{g})}{f_D(\mathsf{d})\, f_G(\mathsf{g})} d\mathsf{g}d\mathsf{d}$$

Then the prior predictive density can be brought inside the integral and canceled.

$$\Phi = \int\limits_{\Omega_D}\int\limits_{\Omega_G} f_G(\mathsf{g})\, f_{D|G}(\mathsf{d}|\mathsf{g}) \log \frac{f_{D|G}(\mathsf{d}|\mathsf{g})}{f_D(\mathsf{d})} d\mathsf{g}d\mathsf{d} \tag{D.6}$$

The difference between Equations D.6 and D.3 is the density on which the data predictive density is conditioned. When the data predictive density is conditioned on the parameters, it simply reduces to some function of the system and observation models. Often these observation models are not even dependent on the parameters. This makes the conditional probability easy to compute.

When the data predictive density is conditioned on the target density, they must be related through the model parameters. Focusing on the difficult conditional probability:

$$\begin{aligned}
f_{D|G}(\mathsf{d}|\mathsf{g}) &= \frac{f_{D,G}(\mathsf{d},\mathsf{g})}{f_G(\mathsf{g})} \\
&= \frac{\int\limits_{\Omega_\Theta} f_\Theta(\theta) f_{D,G|\Theta}(\mathsf{d},\mathsf{g}|\theta) d\theta}{f_G(\mathsf{g})} \\
&= \frac{\int\limits_{\Omega_\Theta} f_\Theta(\theta) f_{G|\Theta}(\mathsf{g}|\theta)\, f_{D|\Theta}(\mathsf{d}|\theta) d\theta}{f_G(\mathsf{g})} \\
&= \frac{\int\limits_{\Omega_\Theta} f_\Theta(\theta) f_{G|\Theta}(\mathsf{g}|\theta)\, f_{D|\Theta}(\mathsf{d}|\theta) d\theta}{\int\limits_{\Omega_\Theta} f_\Theta(\theta) f_{G|\Theta}(\mathsf{g}|\theta)\, d\theta}
\end{aligned}$$

The trick is that the target Random Variable and the Data Prediction Random Variable are conditionally independent, given the parameters. The full risk metric is then:

$$\Phi = \int\limits_{\Omega_D}\int\limits_{\Omega_G} f_G(\mathsf{g}) \frac{\int\limits_{\Omega_\Theta} f_\Theta(\theta) f_{G|\Theta}(\mathsf{g}|\theta)\, f_{D|\Theta}(\mathsf{d}|\theta) d\theta}{f_G(\mathsf{g})} \log \frac{\int\limits_{\Omega_\Theta} f_\Theta(\theta) f_{G|\Theta}(\mathsf{g}|\theta)\, f_{D|\Theta}(\mathsf{d}|\theta)\, d\theta}{f_D(\mathsf{d}) \int\limits_{\Omega_\Theta} f_\Theta(\theta)\, f_{G|\Theta}(\mathsf{g}|\theta) d\theta} d\mathsf{g}d\mathsf{d}$$

$$\Phi = \int\limits_{\Omega_D}\int\limits_{\Omega_G}\int\limits_{\Omega_\Theta} f_\Theta(\theta) f_{G|\Theta}(\mathsf{g}|\theta)\, f_{D|\Theta}(\mathsf{d}|\theta) d\theta \log \frac{\int\limits_{\Omega_\Theta} f_\Theta(\theta) f_{G|\Theta}(\mathsf{g}|\theta)\, f_{D|\Theta}(\mathsf{d}|\theta)\, d\theta}{\int\limits_{\Omega_\Theta} f_\Theta(\theta)\, f_{D|\Theta}(\mathsf{d}|\theta) d\theta \int\limits_{\Omega_\Theta} f_\Theta(\theta)\, f_{G|\Theta}(\mathsf{g}|\theta) d\theta} d\mathsf{g}d\mathsf{d} \tag{D.7}$$

This is the best possible for Prior Sampling Formulation. None of the densities in Equation D.7 require sampling from the posterior parameter density, however, this is still not easy to evaluate. Unlike the observation model which is known and relatively simple, the target random variable does not have a known conditional density function $f_{G|\Theta}(\mathsf{g}|\theta)\, d\theta$. In order to use this formulation, this needs to be approximated for every value of θ. This greatly increases the computational work required and adds to the error in computing the risk metric.

D.2 Utility Functions for Model Discrimination

Here we discuss four Bayesian utility functions used for model discrimination. The first was proposed by Bard [5] and is derived for the Model Based Experimental Design approach. The other two use hierarchical modeling and metrics of submodel probabilities.

D.2.1 Bard's Utility Function

This utility function was created for Model Based Experimental designs, which assumed a linear model and Gaussian probability densities. This assumption is relaxed here and the utility function is shown for general probability densities. There is no inference step and no use of prior model probability. The dependence on the independent variable is not shown because every density is dependent on a particular value of the independent variable.

- Start with the prior predictive densities for each model

$$f_{D|M}\left(\mathsf{d}|\mathsf{m}\right) = \int\limits_{\Omega_\Theta} f_{D,\Theta|M}\left(\mathsf{d},\theta|\mathsf{m}\right) d\theta = \int\limits_{\Omega_\Theta} f_{\Theta|M}\left(\theta|\mathsf{m}\right) f_{D|\Theta,M}\left(\mathsf{d}|\theta,\mathsf{m}\right) d\theta$$

- Compute the Kullback-Leibler divergence from each prior predictive density to all others and sum them up. It is assumed that all the predictive densities have the same support.

$$\Phi = \sum_i \sum_{j \neq i} D_{KL}\left(f_{D|M}\left(\mathsf{d}|\mathsf{m}_i\right) || f_{D|M}\left(\mathsf{d}|\mathsf{m}_j\right)\right)$$

$$\Phi = \sum_i \sum_{j \neq i} \int\limits_{\Omega_D} f_{D|M}\left(\mathsf{d}|\mathsf{m}_i\right) \log \frac{f_{D|M}\left(\mathsf{d}|\mathsf{m}_i\right)}{f_{D|M}\left(\mathsf{d}|\mathsf{m}_j\right)} d\mathsf{d} \tag{D.8}$$

$$\Phi = \sum_i \sum_{j \neq i} \int\limits_{D} f_{D|M}\left(\mathsf{d}|\mathsf{m}_i\right) \log f_{D|M}\left(\mathsf{d}|\mathsf{m}_i\right) - f_{D|M}\left(\mathsf{d}|\mathsf{m}_i\right) \log f_{D|M}\left(\mathsf{d}|\mathsf{m}_j\right) d\mathsf{d}$$

$$\Phi = \sum_i \sum_{j \neq i} \left\{ \begin{array}{l} \int\limits_D f_{D|M}\left(\mathsf{d}|\mathsf{m}_i\right) \log f_{D|M}\left(\mathsf{d}|\mathsf{m}_i\right) d\mathsf{d} \\ -\int\limits_D f_{D|M}\left(\mathsf{d}|\mathsf{m}_i\right) \log f_{D|M}\left(\mathsf{d}|\mathsf{m}_j\right) d\mathsf{d} \end{array} \right\}$$

$$\Phi = \sum_{i} \left\{ \begin{array}{l} (N_{mod} - 1) \int\limits_{D} f_{D|M}\left(\mathsf{d}|\mathsf{m}_i\right) \log f_{D|M}\left(\mathsf{d}|\mathsf{m}_i\right) d\mathsf{d} \\ - \sum\limits_{j \neq i} \int\limits_{D} f_{D|M}\left(\mathsf{d}|\mathsf{m}_i\right) \log f_{D|M}\left(\mathsf{d}|\mathsf{m}_j\right) d\mathsf{d} \end{array} \right\}$$

$$\Phi = \left\{ \begin{array}{l} (N_{mod} - 1) \sum\limits_{i} \int\limits_{D} f_{D|M}\left(\mathsf{d}|\mathsf{m}_i\right) \log f_{D|M}\left(\mathsf{d}|\mathsf{m}_i\right) d\mathsf{d} \\ - \sum\limits_{i} \sum\limits_{j \neq i} \int\limits_{D} f_{D|M}\left(\mathsf{d}|\mathsf{m}_i\right) \log f_{D|M}\left(\mathsf{d}|\mathsf{m}_j\right) d\mathsf{d} \end{array} \right\}$$

- sum over all possible model combinations, of the weighted log ratio of likelihoods of the data being predicted by the current model, averaged over all possible model predictions with certain parameter values, averaged over all possible parameter values.

D.2.2 Hierarchical approach 1

Instead of comparing models discretely as in Bard's approach, this uses a single hierarchical model with each submodel having some probability of being correct. Because this uses only prior predictive densities, there is no inference step required. The information metric is Kullback-Leibler divergence from prior to posterior model probabilities.

- Simulate data from the prior predictive density over all models, parameters, and observations. This is a mixture of all the prior predictive densities of the submodels, with each weighted by their prior model probabilities.

$$\mathsf{d} \sim f_D\left(\mathsf{d}\right) \sim \sum_{\mathsf{m} \in \Omega_M} p_M\left(\mathsf{m}\right) f_{D|M}\left(\mathsf{d}|\mathsf{m}\right)$$

- Compute posterior model probabilities using the prior predictive densities as likelihood

$$p_{M|D}\left(\mathsf{m}|\mathsf{d}\right) = \frac{p_M\left(\mathsf{m}\right) f_{D|M}\left(\mathsf{d}|\mathsf{m}\right)}{f_D\left(\mathsf{d}\right)}$$

- Objective - maximize the expected gain in information, measured by Kullback-Leibler divergence from prior to posterior model probabilities, averaged over all possible data observations.

$$\Phi = \int\limits_{\Omega_D} f_D\left(\mathsf{d}\right) D\left\{p_M\left(\mathsf{m}\right) || p_{M|D}\left(\mathsf{m}|\mathsf{d}\right)\right\} d\mathsf{d}$$

$$\Phi = \int\limits_{\Omega_D} f_D(\mathsf{d}) \sum_i p_M(_i) \log \frac{p_M(\mathsf{m}_i)}{p_{M|D}(\mathsf{m}_i|\mathsf{d})} d\mathsf{d}$$

$$\Phi = \int\limits_{\Omega_D} f_D(\mathsf{d}) \sum_i p_M(\mathsf{m}_i) \log p_M(\mathsf{m}_i) - p_M(\mathsf{m}_i) \log p_{M|D}(\mathsf{m}_i|\mathsf{d}) d\mathsf{d}$$

- Because the prior is unchanged during the problem, no matter what data was observed or the design that was used, $p_M(\mathsf{m}_i) \log p_M(\mathsf{m}_i)$ will always be the same. So, we can formulate a new variable which will always have the same maximum as Φ

$$\Phi_2 = -\int\limits_{\Omega_D} f_D(\mathsf{d}) \sum_i p_M(\mathsf{m}_i) \log p_{M|\mathsf{d}}(\mathsf{m}_i|\mathsf{d}) d\mathsf{d}$$

- Expand

$$\Phi_2 = -\int\limits_{\Omega_D} f_D(\mathsf{d}) \sum_i p_M(\mathsf{m}_i) \log \frac{p_M(\mathsf{m}_i) f_{D|M}(\mathsf{d}|\mathsf{m}_i)}{f_D(\mathsf{d})} d\mathsf{d} \tag{D.9}$$

In words

- Sample a model from the prior model probability
- Simulate data from that model
- Using the prior predictive densities for each model as likelihoods, compute the posterior model probabilities for each model
- Compute the Kullback-Leibler divergence between the prior and posterior model probabilities
- Repeat for many datasets

D.2.3 Hierarchical approach 2

This approach uses the same theory as Hierarchical approach 1, however the information gain is reversed, to become Kullback-Leibler divergence from posterior to prior model probabilities.

- Simulate data from the joint prior predictive density over all models, parameters, and observations

$$\mathsf{d} \sim f_D(\mathsf{d}) \sim \sum_m p_M(\mathsf{m}) f_{D,M}(\mathsf{d}, \mathsf{m})$$

- compute posterior model probabilities

$$p_{M|\mathsf{d}}\left(\mathsf{m}|\mathsf{d}\right) = \frac{p_M\left(\mathsf{m}\right) f_{D|M}\left(\mathsf{d}|\mathsf{m}\right)}{f_D\left(\mathsf{d}\right)}$$

- Objective - maximize the gain in information, measured by Kullback-Leibler divergence from posterior to prior model probabilities. Averaged over all possible data

$$\Phi = \int\limits_{\Omega_D} f_D\left(\mathsf{d}\right) D\left\{p_{M|D}\left(\mathsf{m}|\mathsf{d}\right)||p_M\left(\mathsf{m}\right)\right\} d\mathsf{d}$$

$$\Phi = \int\limits_{\Omega_D} f_D\left(\mathsf{d}\right) \sum_i p_{M|D}\left(\mathsf{m}|\mathsf{d}\right) \log \frac{p_{M|D}\left(\mathsf{m}|\mathsf{d}\right)}{p_M\left(\mathsf{m}_i\right)} d\mathsf{d}$$

- Expand

$$\Phi = \int\limits_{\Omega_D} f_D\left(\mathsf{d}\right) \sum_i \frac{p_M\left(\mathsf{m}_i\right) f_{D|M}\left(\mathsf{d}|\mathsf{m}_i\right)}{f_D\left(\mathsf{d}\right)} \log \frac{f_{D|M}\left(\mathsf{d}|\mathsf{m}_i\right)}{f_D\left(\mathsf{d}\right)} d\mathsf{d}$$

$$\Phi = \int\limits_{\Omega_D} \sum_i p_M\left(\mathsf{m}_i\right) f_{D|M}\left(\mathsf{d}|\mathsf{m}_i\right) \log \frac{f_{D|M}\left(\mathsf{d}|\mathsf{m}_i\right)}{f_D\left(\mathsf{d}\right)} d\mathsf{d} \tag{D.10}$$

D.2.4 Comparison Between Utility Functions

The first three utility functions are all very similar, involving comparisons of log likelihoods between models. Here the three are analyzed to determine if Bard is the same as Hierarchical approach 1 assuming a uniform prior model probability mass function. Let $F_i = f_{D|M}\left(\mathsf{d}|\mathsf{m}_i\right)$.

Bard

$$\Phi = \sum_i \sum_{j \neq i} \int\limits_{\Omega_D} f_{D|M}\left(\mathsf{d}|\mathsf{m}_i\right) \log \frac{f_{D|M}\left(\mathsf{d}|\mathsf{m}_i\right)}{f_{D|M}\left(\mathsf{d}|\mathsf{m}_j\right)} d\mathsf{d} \tag{D.8}$$

$$\Phi = \sum_i \sum_{j \neq i} \int_{\Omega_D} F_i \log \frac{F_i}{F_j} d\mathsf{d} = \int_{\Omega_D} \sum_i \sum_{j \neq i} F_i \log \frac{F_i}{F_j} d\mathsf{d}$$

$$\Phi = \int_{\Omega_D} \sum_i F_i \log \prod_{j \neq i} \frac{F_i}{F_j} d\mathsf{d} = \int_{\Omega_D} \sum_i F_i \log \prod_j \frac{F_i}{F_j} d\mathsf{d}$$

$$\Phi = \int_{\Omega_D} N_{mod} \sum_i F_i \log \frac{F_i}{\left[\prod_j F_j\right]^{\frac{1}{N_{mod}}}} d\mathsf{d}$$

Hierarchical 1

$$\Phi_2 = -\int_{\Omega_D} f_D(\mathsf{d}) \sum_i p_M(\mathsf{m}_i) \log \frac{p_M(\mathsf{m}_i) f_{D|M}(\mathsf{d}|\mathsf{m}_i)}{f_D(\mathsf{d})} d\mathsf{d} \tag{D.9}$$

$$\Phi_2 = -\int_{\Omega_D} \left[\sum_j p_M(\mathsf{m}_j) F_j\right] \sum_i p_M(\mathsf{m}_i) \log \frac{p_M(\mathsf{m}_i) F_i}{\sum_j p_M(\mathsf{m}_j) F_j} d\mathsf{d}$$

$$\Phi_2 = -\int_{\Omega_D} \left[\sum_j \frac{1}{N_{mod}} F_j\right] \sum_i \frac{1}{N_{mod}} \log \frac{F_i}{\sum_j F_j} d\mathsf{d}$$

$$\Phi_2 = -\frac{1}{N_{mod}^2} \int_{\Omega_D} \left[\sum_j F_j\right] \log \prod_i \frac{F_i}{\sum_j F_j} d\mathsf{d}$$

$$\Phi_2 = -\frac{1}{N_{mod}} \int_{\Omega_D} \left[\sum_j F_j\right] \log \frac{\left[\prod_i F_i\right]^{\frac{1}{N_{mod}}}}{\sum_j F_j} d\mathsf{d}$$

$$\Phi_2 = \frac{1}{N_{mod}} \int_{\Omega_D} \left[\sum_j F_j\right] \log \frac{\sum_j F_j}{\left[\prod_i F_i\right]^{\frac{1}{N_{mod}}}} d\mathsf{d}$$

Hierarchical 2

$$\Phi = \int_{\Omega_D} \sum_i p_M(\mathsf{m}_i) f_{D|M}(\mathsf{d}|\mathsf{m}_i) \log \frac{f_{D|M}(\mathsf{d}|\mathsf{m}_i)}{f_D(\mathsf{d})} d\mathsf{d} \tag{D.10}$$

$$\Phi = \int_{\Omega_D} \sum_i p_M(\mathsf{m}_i) F_i \log \frac{F_i}{\sum_j p_M(\mathsf{m}_j) F_j} d\mathsf{d}$$

$$\Phi = \int_{\Omega_D} \frac{1}{N_{mod}} \sum_i F_i \log \left[N_{mod} \frac{F_i}{\sum_j F_j} \right] d\mathsf{d}$$

Discussion

The Bard algorithm turns out to be a sum of the log of the ratio of each prior predictive density to the geometric mean of all densities, weighted by each prior predictive density. The Hierarchical approach using Kullback-Leibler divergence from posterior to prior is the sum of the log of the ratio of each prior predictive density to the arithmetic mean, weighted by the prior predictive density. The Hierarchical approach using Kullback-Leibler divergence from prior to posterior is the sum of the log of the ratio of prior predictive density over all models to the geometric mean of individual prior predictive densities weighted by the prior predictive density over all models.

It is not clear what the physical interpretation should be or what impact this has on the results. It is expected that the results should be the same, but that the absolute values of Φ may be scaled differently.

D.2.5 Derivation of Posterior Evidence Model Discrimination approach

The fourth and final utility function differs from the others because it relies on posterior evidence using the posterior predictive density instead of prior predictive density. While the first three utility functions focus on finding datasets that will be useful for model discrimination alone, this utility function attempts to estimate the parameters of all models such that the posterior knowledge will allow for the best model discrimination. This fits better with the idea of collecting new information.

The Algorithm

- Simulate data from the joint prior predictive density over all models, parameters, and observations

$$\hat{d} \sim f_D(\mathsf{d}) \sim \sum_m p_M(m)\, f_{D,M}(\mathsf{d}, m)$$

- compute posterior model probabilities using the posterior predictive density as a likelihood

$$p_{M|\{D|\hat{d}\}}\left(\mathsf{m}|\left\{\mathsf{d}|\hat{d}\right\}\right) = \frac{p_M(\mathsf{m})\, f_{D|M,\hat{d}}\left(\mathsf{d}|\mathsf{m}, \hat{d}\right)}{f_{D|\hat{d}}\left(\mathsf{d}|\hat{d}\right)} \tag{6.4}$$

  This is the prior model probability times the likelihood of the model with posterior parameter density $\hat{\Theta}$ after inference with dataset $\hat{d}$, being able to explain dataset d, normalized by the sum over all models with their posterior parameter densities

$$p_{M|D}(\mathsf{m}|\mathsf{d}) = \frac{p_M(\mathsf{m}) \int\limits_{\Omega_\Theta} f_{\hat{\Theta}|M}\left(\hat{\theta}|\mathsf{m}\right) f_{D|M,\hat{\Theta}}\left(\mathsf{d}|\mathsf{m}, \hat{\theta}\right) d\hat{\theta}}{\sum\limits_i p_M(\mathsf{m}_i) \int\limits_{\Omega_\Theta} f_{\hat{\Theta}|M}\left(\hat{\theta}|\mathsf{m}_i\right) f_{D|M,\hat{\Theta}}\left(\mathsf{d}|\mathsf{m}_i, \hat{\theta}\right) d\hat{\theta}}$$

- Objective – maximize the gain in information, measured by Kullback-Leibler divergence from posterior to prior model probabilities. This is repeated for many data sets, with parameters inferred from the same dataset.

$$\Phi = \int\limits_{\Omega_D} f_D(\mathsf{d})\, D\left\{p_{M|D}(\mathsf{m}|\mathsf{d})\, ||p_M(\mathsf{m})\right\} d\mathsf{d}$$

$$\Phi = \int\limits_{\Omega_D} f_D\left(\mathsf{d}\right) \sum_i p_{M|D}\left(\mathsf{m}_i|\mathsf{d}\right) \log \frac{p_{M|D}\left(\mathsf{m}_i|\mathsf{d}\right)}{p_M\left(\mathsf{m}_i\right)} d\mathsf{d}$$

- Expand

$$\Phi = \int\limits_{\Omega_D} f_D\left(\mathsf{d}\right) \sum_i \left[\begin{array}{c} \dfrac{p_M(\mathsf{m}_i) \int\limits_{\Omega_\Theta} f_{\hat{\Theta}|M}\left(\hat{\theta}|\mathsf{m}_i\right) f_{D|M,\hat{\Theta}}\left(\mathsf{d}|\mathsf{m}_i,\hat{\theta}\right) d\hat{\theta}}{\sum_j p_M(\mathsf{m}_j) \int\limits_{\Omega_\Theta} f_{\hat{\Theta}|M}\left(\hat{\theta}|\mathsf{m}_j\right) f_{D|M,\hat{\Theta}}\left(\mathsf{d}|\mathsf{m}_j,\hat{\theta}\right) d\hat{\theta}} \cdots \\ \cdots \log \dfrac{\int\limits_{\Omega_\Theta} f_{\hat{\Theta}|M}\left(\hat{\theta}|\mathsf{m}_i\right) f_{D|M,\hat{\Theta}}\left(\mathsf{d}|\mathsf{m}_i,\hat{\theta}\right) d\hat{\theta}}{\sum_j p_M(\mathsf{m}_j) \int\limits_{\Omega_\Theta} f_{\hat{\Theta}|M}\left(\hat{\theta}|\mathsf{m}_j\right) f_{D|M,\hat{\Theta}}\left(\mathsf{d}|\mathsf{m}_j,\hat{\theta}\right) d\hat{\theta}} \end{array} \right] d\mathsf{d}$$

- The density we need to compute is a predictive density given a particular dataset $\hat{d}$. This can be broken down into a parameter density and a conditional density (the observation model):

$$f_{D|M}\left(\mathsf{d}|\mathsf{m}\right) = \int\limits_{\Omega_\Theta} f_{\hat{\Theta}|M}\left(\hat{\theta}|\mathsf{m}\right) f_{D|M,\hat{\Theta}}\left(\mathsf{d}|\mathsf{m},\hat{\theta}\right) d\hat{\theta} \tag{D.11}$$

- The problem is that we need the posterior parameter density, having observed dataset $\hat{d}$. This is where this derivation changes from that of the Hierarchical Bayesian approach 2.

$$f_{\hat{\Theta}|M}\left(\hat{\theta}|\mathsf{m}\right) = f_{\Theta|M,\hat{D}}\left(\theta|\mathsf{m},\hat{d}\right) = \frac{f_{\Theta|M}\left(\theta|\mathsf{m}\right) f_{\hat{D}|M,\Theta}\left(\hat{d}|\mathsf{m},\theta\right)}{f_{\hat{D}|M}\left(\hat{d}|\mathsf{m}\right)}$$

- Substituting this back in:

$$\Phi = \int\limits_{\Omega_D} f_D\left(\mathsf{d}\right) \sum_i \left[\begin{array}{c} \dfrac{p_M(\mathsf{m}_i) \int\limits_{\Omega_\Theta} \frac{f_{\Theta|M}(\theta|\mathsf{m}_i) f_{\hat{D}|M,\Theta}(\hat{d}|\mathsf{m}_i,\theta)}{f_{\hat{D}|M}(\hat{d}|\mathsf{m}_i)} f_{D|M,\Theta}(\mathsf{d}|\mathsf{m}_i,\theta) d\theta}{\sum_j p_M(\mathsf{m}_j) \int\limits_{\Omega_\Theta} \frac{f_{\Theta|M}\left(\theta|\mathsf{m}_j\right) f_{\hat{D}|M,\Theta}\left(\hat{d}|\mathsf{m}_j,\theta\right)}{f_{\hat{D}|M}\left(\hat{d}|\mathsf{m}_j\right)} f_{D|M,\Theta}(\mathsf{d}|\mathsf{m}_j,\theta) d\theta} \cdots \\ \cdots \log \dfrac{\int\limits_{\Omega_\Theta} \frac{f_{\Theta|M}(\theta|\mathsf{m}_i) f_{\hat{D}|M,\Theta}(\hat{d}|\mathsf{m}_i,\theta)}{f_{\hat{D}|M}(\hat{d}|\mathsf{m}_i)} f_{D|M,\Theta}(\mathsf{d}|\mathsf{m}_i,\theta) d\theta}{\sum_j p_M(\mathsf{m}_j) \int\limits_{\Omega_\Theta} \frac{f_{\Theta|M}\left(\theta|\mathsf{m}_j\right) f_{\hat{D}|M,\Theta}\left(\hat{d}|\mathsf{m}_j,\theta\right)}{f_{\hat{D}|M}\left(\hat{d}|\mathsf{m}_j\right)} f_{D|M,\Theta}(\mathsf{d}|\mathsf{m}_j,\theta) d\theta} \end{array} \right] d\mathsf{d}$$

- And then use the same dataset for both inference and computing the likelihood...

$$\Phi = \int\limits_{\Omega_D} f_D(\mathsf{d}) \sum_i \left[\frac{p_M(\mathsf{m}_i) \int\limits_{\Omega_\Theta} \frac{f_{\Theta|M}(\theta|\mathsf{m}_i) f_{\hat{D}|M,\Theta}(\hat{d}=\mathsf{d}|\mathsf{m}_i,\theta)}{f_{\hat{D}|M}(\hat{d}=\mathsf{d}|\mathsf{m}_i)} f_{D|M,\Theta}(\mathsf{d}|\mathsf{m}_i,\theta) d\theta}{\sum\limits_j p_M(\mathsf{m}_j) \int\limits_{\Omega_\Theta} \frac{f_{\Theta|M}\left(\theta|\mathsf{m}_j\right) f_{\hat{D}|M,\Theta}\left(\hat{d}=\mathsf{d}|\mathsf{m}_j,\theta\right)}{f_{\hat{D}|M}\left(\hat{d}=\mathsf{d}|\mathsf{m}_j\right)} f_{D|M,\Theta}(\mathsf{d}|\mathsf{m}_j,\theta) d\theta} \cdots \atop \cdots \log \frac{\int\limits_{\Omega_\Theta} \frac{f_{\Theta|M}(\theta|\mathsf{m}_i) f_{\hat{D}|M,\Theta}(\hat{d}=\mathsf{d}|\mathsf{m}_i,\theta)}{f_{\hat{D}|M}(\hat{d}=\mathsf{d}|\mathsf{m}_i)} f_{D|M,\Theta}(\mathsf{d}|\mathsf{m}_i,\theta) d\theta}{\sum\limits_j p_M(\mathsf{m}_j) \int\limits_{\Omega_\Theta} \frac{f_{\Theta|M}\left(\theta|\mathsf{m}_j\right) f_{\hat{D}|M,\Theta}\left(\hat{d}=\mathsf{d}|\mathsf{m}_j,\theta\right)}{f_{\hat{D}|M}\left(\hat{d}=\mathsf{d}|\mathsf{m}_j\right)} f_{D|M,\Theta}(\mathsf{d}|\mathsf{m}_j,\theta) d\theta} \right] d\mathsf{d}$$

- Break this up

$$\begin{aligned} F_i &= f_{D|M,\Theta}\left(\mathsf{d}|\mathsf{m}_i,\theta\right) \\ f_{\hat{D}|M}\left(\hat{d}=\mathsf{d}|m\right) &= \int\limits_{\Omega_\Theta} f_{\Theta|M}\left(\theta|m\right) f_{\hat{D}|M,\Theta}\left(\hat{d}=\mathsf{d}|m,\theta\right) d\theta \end{aligned}$$

$$\Phi = \int\limits_{\Omega_D} f_D\left(\mathsf{d}\right) \sum_i \frac{p_M\left(\mathsf{m}_i\right) \int\limits_{\Omega_\Theta} \frac{f_{\Theta|M}(\theta|\mathsf{m}_i)F_i}{\int\limits_{\Omega_\Theta} f_{\Theta|M}(\theta|\mathsf{m}_i)F_i d\theta} F_i d\theta}{\sum\limits_j p_M\left(\mathsf{m}_j\right) \int\limits_{\Omega_\Theta} \frac{f_{\Theta|M}(\theta|\mathsf{m}_j)F_j}{\int\limits_{\Omega_\Theta} f_{\Theta|M}(\theta|\mathsf{m}_j)F_j d\theta} F_j d\theta} \log \frac{\int\limits_{\Omega_\Theta} \frac{f_{\Theta|M}(\theta|\mathsf{m}_i)F_i}{\int\limits_{\Omega_\Theta} f_{\Theta|M}(\theta|\mathsf{m}_i)F_i d\theta} F_i d\theta}{\sum\limits_j p_M\left(\mathsf{m}_j\right) \int\limits_{\Omega_\Theta} \frac{f_{\Theta|M}(\theta|\mathsf{m}_j)F_j}{\int\limits_{\Omega_\Theta} f_{\Theta|M}(\theta|\mathsf{m}_j)F_j d\theta} F_j d\theta} d\mathsf{d}$$

$$\Phi = \int\limits_{\Omega_D} f_D\left(\mathsf{d}\right) \sum_i \frac{p_M\left(\mathsf{m}_i\right) \frac{\int\limits_{\Omega_\Theta} f_{\Theta|M}(\theta|\mathsf{m}_i)F_iF_i d\theta}{\int\limits_{\Omega_\Theta} f_{\Theta|M}(\theta|\mathsf{m}_i)F_i d\theta}}{\sum\limits_j p_M\left(\mathsf{m}_j\right) \frac{\int\limits_{\Omega_\Theta} f_{\Theta|M}(\theta|\mathsf{m}_j)F_jF_j d\theta}{\int\limits_{\Omega_\Theta} f_{\Theta|M}(\theta|\mathsf{m}_j)F_j d\theta}} \log \frac{\frac{\int\limits_{\Omega_\Theta} f_{\Theta|M}(\theta|\mathsf{m}_i)F_iF_i d\theta}{\int\limits_{\Omega_\Theta} f_{\Theta|M}(\theta|\mathsf{m}_i)F_i d\theta}}{\sum\limits_j p_M\left(\mathsf{m}_j\right) \frac{\int\limits_{\Omega_\Theta} f_{\Theta|M}(\theta|\mathsf{m}_j)F_jF_j d\theta}{\int\limits_{\Omega_\Theta} f_{\Theta|M}(\theta|\mathsf{m}_j)F_j d\theta}} d\mathsf{d}$$

$$\Phi = \int\limits_{\Omega_D} f_D\left(\mathsf{d}\right) \sum_i \left[\frac{p_M\left(\mathsf{m}_i\right) \int\limits_{\Omega_\Theta} f_{\Theta|M}\left(\theta|\mathsf{m}_i\right) F_iF_i d\theta}{F_i^{norm} M^{norm}} \log \frac{\int\limits_{\Omega_\Theta} f_{\Theta|M}\left(\theta|\mathsf{m}_i\right) F_iF_i d\theta}{F_i^{norm} M^{norm}} \right] d\mathsf{d}$$

- There are two normalization constants

$$\begin{aligned} M^{norm} &= \sum_j \frac{p_M\left(\mathsf{m}_j\right)}{F_j^{norm}} \int\limits_{\Omega_\Theta} f_{\Theta|M}\left(\theta|\mathsf{m}_j\right) F_jF_j d\theta \\ F_i^{norm} &= \int\limits_{\Omega_\Theta} f_{\Theta|M}\left(\theta|\mathsf{m}_i\right) F_i d\theta \\ F_i &= f_{D|M,\Theta}\left(\mathsf{d}|\mathsf{m}_i,\theta\right) \end{aligned}$$

Now hidden in here are several integrals over the prior parameter density given a model. But the posterior parameter density is never explicitly generated. The posterior predictive density is sampled using a Prior Sampling Formulation. This formulation is significantly faster than computing posterior parameter densities with Markov Chain Monte Carlo. Also, the Kullback-Leibler divergence does not need to be computed between any continuous probability densities, like the parameter densities or predictive densities.

D.3 Comparing Utility Functions for Parameter Estimation

In this section we compare three ways to compare probability density functions. Differential entropy is a statistic from information theory that is used to quantify information content in a probability density function. The Kullback-Leibler divergence is a weighted comparison, very much related to differential entropy. The third, mutual information, is another way to relate probability density functions.

D.3.1 Differential Entropy and Kullback-Leibler Divergence

Here we will show that two utility functions for experimental designs produce the same result, as long as the risk metric is expected utility. The objective function for Kullback-Leibler divergence from the posterior parameter density to the prior parameter density was derived in Section D.1. Here the objective function is derived for the utility function of negative differential entropy.

$$\Phi = \int\limits_{\Omega_D} f_D(\mathsf{d}) \int\limits_{\Omega_\Theta} f_{\Theta|D}(\theta|\mathsf{d}) \log f_{\Theta|D}(\theta|\mathsf{d})\, d\theta d\mathsf{d}$$

$$\Phi = \int\limits_{\Omega_D}\int\limits_{\Omega_\Theta} f_D(\mathsf{d}) \frac{f_\Theta(\theta) f_{D|\Theta}(\mathsf{d}|\theta)}{f_D(\mathsf{d})} \log f_{\Theta|D}(\theta|\mathsf{d})\, d\theta d\mathsf{d}$$

$$\Phi = \int\limits_{\Omega_D}\int\limits_{\Omega_\Theta} f_\Theta(\theta) f_{D|\Theta}(\mathsf{d}|\theta) \log f_{\Theta|D}(\theta|\mathsf{d})\, d\theta d\mathsf{d}$$

$$\Phi = \int\limits_{\Omega_D}\int\limits_{\Omega_\Theta} f_\Theta(\theta) f_{D|\Theta}(\mathsf{d}|\theta) \log \frac{f_\Theta(\theta) f_{D|\Theta}(\mathsf{d}|\theta)}{f_D(\mathsf{d})} d\theta d\mathsf{d}$$

$$
\begin{aligned}
\Phi = & \int\limits_{\Omega_D}\int\limits_{\Omega_\Theta} f_\Theta(\theta)\, f_{D|\Theta}(\mathsf{d}|\theta) \log f_\Theta(\theta)\, d\theta d\mathsf{d} \\
& + \int\limits_{\Omega_D}\int\limits_{\Omega_\Theta} f_\Theta(\theta)\, f_{D|\Theta}(\mathsf{d}|\theta) \log f_{D|\Theta}(\mathsf{d}|\theta)\, d\theta d\mathsf{d} \\
& - \int\limits_{\Omega_D}\int\limits_{\Omega_\Theta} f_\Theta(\theta)\, f_{D|\Theta}(\mathsf{d}|\theta) \log f_D(\mathsf{d})\, d\theta d\mathsf{d}
\end{aligned}
$$

$$
\begin{aligned}
\Phi = & \int\limits_{\Omega_\Theta} f_\Theta(\theta) \log f_\Theta(\theta)\, d\theta \\
& + \int\limits_{\Omega_D}\int\limits_{\Omega_\Theta} f_\Theta(\theta)\, f_{D|\Theta}(\mathsf{d}|\theta) \log f_{D|\Theta}(\mathsf{d}|\theta)\, d\theta d\mathsf{d} \\
& - \int\limits_{\Omega_D} f_D(\mathsf{d}) \log f_D(\mathsf{d})\, d\mathsf{d}
\end{aligned}
\tag{D.12}
$$

When Equation D.12 is compared with Equation D.3

$$
\Phi = \int\limits_{\Omega_D}\int\limits_{\Omega_\Theta} f_\Theta(\theta)\, f_{D|\Theta}(\mathsf{d}|\theta) \log f_{D|\Theta}(\mathsf{d}|\theta)\, d\theta d\mathsf{d} - \int\limits_{\Omega_D} f_D(\mathsf{d}) \log f_D(\mathsf{d})\, d\mathsf{d} \tag{D.3}
$$

the only difference is the prior parameter differential entropy term. This is constant for all designs x so these two utility functions result in the same optimal design, if the risk metric is expected value of utility.

D.3.2 Mutual Information

Mutual information is a measure of how much two Random Variables depend on each other. It is conceptually similar to covariance but is more general and more descriptive. For example, two Random Variables can be dependent but have covariance of 0. The mutual information is only 0 if the Random Variables are independent. The formula is:

$$
I(\Theta(\omega_\Theta); D(\omega_D)) = \int\limits_{\Omega_D}\int\limits_{\Omega_\Theta} f_{D,\Theta}(\mathsf{d},\theta) \log \frac{f_{D,\Theta}(\mathsf{d},\theta)}{f_D(\mathsf{d})\, f_\Theta(\theta)} d\theta d\mathsf{d} \tag{D.13}
$$

This is the Kullback-Leibler Divergence from the joint density function to the product of the marginal density functions. A simple manipulation shows that this is the same as the objective function in Equation D.2.

$$I\left(\Theta\left(\omega_{\Theta}\right);D\left(\omega_{D}\right)\right)=\int\limits_{\Omega_D}\int\limits_{\Omega_\Theta} f_{D}\left(\mathsf{d}\right)f_{\Theta|D}\left(\theta|\mathsf{d}\right)\log\frac{f_{D}\left(\mathsf{d}\right)f_{\Theta|D}\left(\theta|\mathsf{d}\right)}{f_{D}\left(\mathsf{d}\right)f_{\Theta}\left(\theta\right)}d\theta d\mathsf{d}$$

$$I\left(\Theta\left(\omega_{\Theta}\right);D\left(\omega_{D}\right)\right)=\int\limits_{\Omega_D}\int\limits_{\Omega_\Theta} f_{D}\left(\mathsf{d}\right)f_{\Theta|D}\left(\theta|\mathsf{d}\right)\log\frac{f_{\Theta|D}\left(\theta|\mathsf{d}\right)}{f_{\Theta}\left(\theta\right)}d\theta d\mathsf{d}$$

D.3.3 Summary

Maximizing the mutual information between the data predictive density and the prior parameter density is equivalent to maximizing the expected Kullback-Leibler divergence from posterior parameter density to prior parameter density. Maximizing these objective functions give the same result as minimizing the differential entropy of the posterior parameter density.

Appendix E

Model Based Experimental Designs

Model Based Experimental Designs were discussed in Bard [5], and have been mentioned in literature[56, 31]. Within the Chemical Engineering discipline, the Macchietto group at Imperial has done studies with this approach [1, 28, 29, 30, 32, 24]. Simply put, Model Based Experimental Designs are the Bayesian equivalent of Optimal Designs. Although they are a subset of Bayesian Designs, they will be discussed separately because they are a bridge between Optimal and Bayesian Designs.

Instead of developing a metric related to the Fisher Information and simulated, posterior parameter density, the Bayesian metric is based on the change in information between a prior density and a posterior density. Technically, they are simply a subset of Bayesian Designs. Because of computational limitations during the time period when Model Based Experimental Designs were suggested, various assumptions were made to simplify the Bayesian Designs:

1. linearization of the model (to get local sensitivities)
2. use of parameter point estimates, for linearization
3. assumption of normal error distributions and prior/posterior parameter distributions

Note that these assumptions match those of Optimal Designs, and therefore the results are essentially the same as for Optimal Designs. The only remaining Bayesian influence is the appearance of the variance of the prior parameter density in the objective function.

E.0.4 Algorithm

Start with a system model and prior knowledge of the parameters being normally distributed with known mean and variance. Assume that the errors are normally distributed with a known variance. Linearize the model at the mean of the parameters. Analytically calculate the prior predictive density (which will be normal). Lastly, approximate the posterior parameter density with a normal distribution. Then the gain in Shannon Information can be calculated analytically as: $\tilde{V} = \left[\mathrm{B}^T \Pi^{-1} \mathrm{B} + V_0^{-1}\right]^{-1}$.

The Model Based Experimental Design approach, described by Bard, is shown below. Similar to the Optimal Designs, there are many possible formulations of this approach, depending on the goal of the experiment. The objective of this particular formulation is to minimize the variance of the posterior parameter density.

1. Assume a model $\hat{y} = f(x, \theta)$, where f is a vector of M functions, depending on Jindependent variables x and L parameters θ. Calculate the best conditions for a set of N experiments.
2. Assume the data to be centered on the model prediction with some distribution of residuals described by the variances V_m, $m = 1...M$, and arrange these values onto the diagonals of an $M \times M$matrix V, where M is the number of responses. This means that each response can have different variances associated with its measurement, but each measurement of one response is expected to follow the same distribution. The matrix Π is formed by repeating V along the diagonal, into a $MN \times MN$ matrix.
3. Using previous data, fit the data to the model using least squares regression to determine θ_0: an L-vector of parameter values
4. Determine the formula of the sensitivities of each of Mresponses, $B_{ml,n} \equiv \left.\frac{\partial f_m}{\partial \theta_l}\right|_{x=x_n, \theta=\tilde{\theta}}$. This results in B_n, an $M \times L$matrix of sensitivities that is a function of the nth set of independent variables x_n. B_n is calculated for each of the N new design points and stacked up, to produce B, a $MN \times L$ matrix

   Assume that the prior distribution of parameters, θ_0, is normally distributed. Estimate the variance of each of the L parameters using the D previously collected data points, by linearization of the model $\hat{y} = f(x, \theta)$ into $\hat{y} = X\theta$, where X is a $D \times L$ matrix of partial derivatives of f. Then the matrix of prior parameter variances for is:

$$V_0 = \left(X^T X\right)^{-1} \sigma^2$$

whereσ^2 is the variance of the residuals.. Each model response will give different values for V_0, which was not addressed in Bard's description of this problem.

5. The sum $\Pi + BV_0B^T$ is the model prediction uncertainty under Bard's method. The matrix Π represents the uncertainty from measurements, and $BV_0B^T$is the uncertainty in the model (derived from the moment method). Each row of the matrix corresponds to one of M responses, and one of N sets of independent variables.

   Vary the N sets of independent variables x_n, to maximize det $\left(\Pi + BV_0B^T\right)$ to obtain the N new design points at which to carry out experiments.
6. Run experiments and collect data
7. Data analysis

E.0.5 Comparing different optimality criteria

Section 5.1.2 describes several utility functions that can be used to tailor experiments to the goals of a study. Each of these has a formulation for Model Based Experimental Designs with simplifications due to the linearity and Gaussian distributions.

Criteria for Parameter Estimation

The objective function for estimating parameters is Ψ = det $\left(\Pi + BV_0B^T\right)$, which takes into account the measurement errors for each response, and the propagation of parameter uncertainty through the model for each response. The relevant optimization is:.

$\max\limits_{x_n;\, n=1\ldots N} \Psi$ where x_n is a design point (a set of the J independent variables) and N is the number of design points. Note that this is equivalent to: $\min\limits_{x_n;\, n=1\ldots N} \Psi' = \det\left(\left[B^T\Pi^{-1}B + V_0{}^{-1}\right]^{-1}\right)$

Criteria for Prediction

Use of parameters for prediction refers to the situation where an experiment is done on one system (modeled by $\hat{y} = f\left(x, \theta\right)$), to determine equivalent parameters in a different system (modeled by $\eta = \phi(\xi, \theta)$ where ξ are the independent variables). For instance, the volume of an object was measured to estimate its mass, and that mass were to be used to predict the force the object exerts in another situation. In this case, the relevant variance is the variance in the predictions of the model ϕ, which can be expressed as

$$V_p = \left(\frac{\partial \phi}{\partial \xi}\right) V_\xi \left(\frac{\partial \phi}{\partial \xi}\right)^T + \left(\frac{\partial \phi}{\partial \theta}\right) V_\theta \left(\frac{\partial \phi}{\partial \theta}\right)^T + V_\eta$$

Similarly to the Parameter Estimation Criteria, this includes variance of the residuals, V_η, and propagated variance from the parameters, $V_\theta = \left[B^T \Pi^{-1} B + V_0{}^{-1}\right]^{-1}$. It also adds in propagated variance from uncertainty in the independent variables, V_ξ, which is often assumed to be zero.

Criteria for Model Differentiation

As part of the model development process, it is often necessary to compare two competing models. Each of these models has a different structure and parameters, but uses the same independent variables. The concept is to search for the design points at which the models have the greatest difference in predicted values, while also taking into account the uncertainty in the model predictions and measurements. Experiments at this point will be best able to differentiate between the two models. The general idea is to calculate the probability of the data supporting one model over the other, over the domain of the independent variables. Then select the set of independent variables where the probability of discriminating between models is highest.

The formulation is shown below:

1. Start with two models:

   $$\hat{y}^{(1)} = f^{(1)}\left(x, \hat{\theta}^{(1)}\right) \text{ and } \hat{y}^{(2)} = f^{(2)}\left(x, \hat{\theta}^{(2)}\right)$$

   with variance of measurement errors $V^{(i)}$, $i = 1, 2$.

2. Define $a_{ij}(x) = \log\left(\frac{p^{(i)}(y|x)}{p^{(j)}(y|x)}\right)$ which measures how much the observations y support model i over model j, and $E^{(i)}\left[a_{ij}(x)\right] = \int p^{(i)}(y|x) \log\left(\frac{p^{(i)}(y|x)}{p^{(j)}(y|x)}\right) dy$, which is interpreted as how superior model i will appear to be given the data x, assuming that model i is correct. Note, to calculate this, a distribution of residuals (the likelihood distribution) must be assumed.

3. At each value of the independent variable, the expected preference for one model or the other can be calculated. These are maximized over the design space:
   $\max_{x} J_{1,2}(x) = E^{(1)}[a_{12}(x)] + E^{(2)}[a_{21}(x)]$
   The objective function can be calculated using Monte Carlo sampling methods, or by making assumptions that allow for simple solutions. For example, models prediction errors are assumed to be normal and with variances given by $V_y^{(i)} = \left(V^{(i)} + BV_\theta B^T\right)$ the measurement variance plus the propagated parameter variance.

4. This is effectively maximizing the probability of measuring data that will allow for model discrimination. The dependence on the design x comes from the models $y^{(i)}$. $J_{1,2}$ must be maximized over the design space to obtain the most effective experimental conditions.

E.0.6 Discussion

The Model Based Experimental Design approach is attractive when compared with the Classical Approach, because it is able to incorporate knowledge of the system and the significant uncertainties. The drawbacks are similar to Optimal Designs. Both the Model Based Experimental Design and Optimal Design approaches consider only linearized models and Gaussian uncertainties. This is equivalent to a first order approximation in both model response and parametric uncertainties. Most systems of interest have much more complex behavior, and so these approaches will give suboptimal results.

Bibliography

[1] S.P. Asprey and S. Macchietto. Designing robust optimal dynamic experiments. *Journal of Process Control*, 12(4):545–556, 2002.

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

[3] S. Balakrishnan, A. Roy, M.G. Ierapetritou, G.P. Flach, and P.G. Georgopoulos. A comparative assessment of efficient uncertainty analysis techniques for environmental fate and transport models: application to the fact model. *Journal of Hydrology*, 307(1-4):204–218, 2005.

[4] M. Barbieri and J.O. Berger. Optimal predictive model selection. *Annals of Statistics*, 32(3):870–897, 2004.

[5] Y. Bard. *Nonlinear Parameter Estimation.* Academic Press, New York, 1974.

[6] A. Ben-Tal, L. El Ghaoui, and A.S. Nemirovskii. *Robust optimization.* Princeton University Press, Princeton, 2009.

[7] J.O. Berger. *Statistical decision theory and Bayesian analysis.* Springer series in statistics. Springer-Verlag, New York, 1985.

[8] D.P. Bertsekas, A. Nedic, and A.E. Ozdaglar. *Convex analysis and optimization.* Athena Scientific, Belmont, Mass., 2003.

[9] D.P. Bertsekas and J.N. Tsitsiklis. *Introduction to probability.* Athena Scientific, Belmont, Mass., 2002.

[10] D.R. Bingham and H.A. Chipman. Incorporating prior information in optimal design for model selection. *Technometrics*, 49(2):155–163, 2007.

[11] D.M. Bortz and C.T. Kelley. The simplex gradient and noisy optimization problems. In *Computational Methods in Optimal Design and Control*, pages 77–90. Birkhuser, 1998.

[12] S.P. Boyd and L. Vandenberghe. *Convex optimization.* Cambridge University Press, Cambridge, UK; New York, 2004.

[13] 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.

[14] K. Chaloner. A note on optimal bayesian design for nonlinear problems. *Journal of Statistical Planning and Inference*, 37(2):229–235, 1993.

[15] K. Chaloner and K. Larntz. Optimal bayesian design applied to logistic regression experiments. *Journal of Statistical Planning and Inference*, 21(2):191–208, 1989.

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

[17] P.A.P. de Man. *Applications of the Bayesian Approach for Experimentation and Estimation.* PhD thesis, Massachusetts Institute of Technology, 2006.

[18] E. de Rocquigny. Quantifying uncertainty in an industrial approach : an emerging consensus in an old epistemological debate. *S.A.P.I.EN.S*, 2(1), 2009.

[19] D. Draper. Assessment and propagation of model uncertainty. *Journal of the Royal Statistical Society. Series B (Methodological)*, 57(1):45–97, 1995.

[20] M.E. Dry. The fischer-tropsch process: 1950-2000. *Catalysis Today*, 71(3-4):227–241, 2002.

[21] M.S. Eldred. Recent advances in non-intrusive polynomial chaos and stochastic collocation methods for uncertainty analysis and design, 4 - 7 May, 2009 2009.

[22] M.S. Eldred and J. Burkardt. Comparison of non-intrusive polynomial chaos and stochastic collocation methods for uncertainty quantification. In *47th AIAA Aerospace Sciences Meeting*, Orlando, FL, 2009. AIAA.

[23] Webster C.G. Eldred, M.S. and P. Constantine. Evaluation of non-intrusive approaches for wiener-askey generalized polynomial chaos. In *49th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference (10th AIAA Non-Deterministic Approaches Conference)*, Schaumburg, IL, 2008. AIAA.

[24] G. Federico, B. Massimiliano, B. Fabrizio, and M. Sandro. A backoff strategy for model-based experiment design under parametric uncertainty. *AIChE Journal*, 9999(9999):NA, 2009.

[25] V.V. Fedorov and P. Hackl. *Model-oriented design of experiments.* Springer, New York, 1997.

[26] B.A. Finlayson. *The method of weighted residuals and variational principles, with application in fluid mechanics, heat and mass transfer.* Mathematics in science and engineering, v. 87. Academic Press, New York, 1972.

[27] J. Foo, X. Wan, and G.E. Karniadakis. The multi-element probabilistic collocation method (me-pcm): Error analysis and applications. *Journal of Computational Physics*, 227(22):9572–9595, 2008.

[28] G. Franceschini and S. Macchietto. Validation of a model for biodiesel production through model-based experiment design. *Industrial & Engineering Chemistry Research*, 46(1):220–232, 2006.

[29] G. Franceschini and S. Macchietto. Model-based design of experiments for parameter precision: State of the art. *Chemical Engineering Science*, 63(19):4846–4872, 2008.

[30] G. Franceschini and S. Macchietto. Novel anticorrelation criteria for model-based experiment design: Theory and formulations. *AIChE*, 54(4):1009–1024, 2008.

[31] F. Galvanin, A. Boschiero, M. Barolo, and F. Bezzo. Model-based design of experiments in the presence of continuous measurement systems. *Industrial & Engineering Chemistry Research*, pages null–null, 2011.

[32] F. Galvanin, S. Macchietto, and F. Bezzo. Model-based design of parallel experiments. *Industrial & Engineering Chemistry Research*, 46(3):871–882, 2007.

[33] A. Gelman. *Bayesian data analysis.* Texts in statistical science. Chapman & Hall/CRC, Boca Raton, Fla., 2004.

[34] C. Geyer. Charlie geyer's personal home page. http://www.stat.umn.edu/ charlie/, October 25 2010.

[35] R. Ghanem and P. Spanos. *Stochastic Finite Elements: A Spectral Approach.* Springer-Verlag, 1991.

[36] H. Haario, M. Laine, A. Mira, and E. Saksman. Dram: Efficient adaptive mcmc. *Statistics and Computing*, 16(4):339–354, 2006.

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

[38] R. Horst and T. Hoang. *Global optimization : deterministic approaches.* Springer-Verlag, Berlin; New York, 1990.

[39] S. Hosder, R.W. Walters, and B. M. Efficient sampling for non-intrusive polynomial chaos applications with multiple uncertain input variables, 2007.

[40] X. Huan. Accelerated bayesian experimental design for chemical kinetic models. Master's thesis, Massachusetts Institute of Technology, 2010.

[41] S. Isukapalli. Stochastic response surface method (srsm), automatic differentiation, and bayesian approaches for uncertainty propagation and parameter estimation. 2005.

[42] S. Isukapalli, S. Balakrishnan, and P. Georgopoulos. Computationally efficient uncertainty propagation and reduction using the stochastic response surface method. *43rd IEEE Conference on Decision and Control*, 2004.

[43] E.T. Jaynes and G.L. Bretthorst. *Probability theory : the logic of science.* Cambridge University Press, Cambridge, UK; New York, NY, 2003.

[44] J.N. Kapur and H.K. Kesavan. *Entropy optimization principles with applications.* Academic Press, Boston, 1992.

[45] L. Karunakumari, C. Eswaraiah, S. Jayanti, and S.S. Narayanan. Experimental and numerical study of a rotating wheel air classifier, 2005.

[46] C.T. Kelley. *Iterative methods for optimization.* Frontiers in applied mathematics. SIAM, Philadelphia, 1999.

[47] 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.

[48] O.P. Le Matre and O.M. Knio. *Spectral methods for uncertainty quantification with applications to computational fluid dynamics.* Springer, Dordrecht; New York, 2010.

[49] H. Li and D. Zhang. Probabilistic collocation method for flow in porous media: Comparisons with other stochastic methods. *WATER RESOURCES RESEARCH*, 2007.

[50] G. Lin and A.M. Tartakovsky. An efficient, high-order probabilistic collocation method on sparse grids for three-dimensional flow and solute transport in randomly heterogeneous porous media. *Advances in Water Resources*, 32(5):712–722, 2009.

[51] D.V. Lindley. *Understanding uncertainty.* Wiley, Hoboken, N.J., 2006.

[52] D.J.C. MacKay. *Information theory, inference, and learning algorithms.* Cambridge University Press, Cambridge, UK; New York, 2003.

[53] 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, 2009.

[54] 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, 2007.

[55] Y.M. Marzouk and D. Xiu. A stochastic collocation approach to bayesian inference in inverse problems. *COMMUNICATIONS IN COMPUTATIONAL PHYSICS*, 6(4):826–847, 2009.

[56] B. Massimiliano. An application of information theory to the problem of the scientific experiment. *Synthese*, 140(3):355–389, 2004.

[57] Robert C. P. Mengersen, K.L. and C Guihenneuc-Jouyaux. Mcmc convergence diagnostics: a "reviewww". *Bayesian Statistics*, 6:412–440, 1999.

[58] D.C. Montgomery. *Design and Analysis of Experiments.* Wiley, 5 edition, 2000.

[59] R.H. Moss and S.H. Schneider. Uncertainties in the ipcc tar: Recommendations to lead authors for more consistent assessment and reporting. Technical report, Intergovernmental Panel on Climate Change, 2000.

[60] P. Muller, D.A. Berry, A.P. Grieve, M. Smith, and M. Krams. Simulation-based sequential bayesian design. *Journal of Statistical Planning and Inference*, 137(10):3140–3150, 2007.

[61] P. Muller and G. Parmigiani. Optimal design via curve fitting of monte carlo experiments. *Journal of the American Statistical Association*, 90(432):1322–1330, 1995.

[62] E.F. Murphy, S.G. Gilmour, and M.J.C. Crabbe. Efficient and accurate experimental design for enzyme kinetics: Bayesian studies reveal a systematic approach. *Journal of Biochemical and Biophysical Methods*, 55(2):155–178, 2003.

[63] A. Nabifar, N.T. McManus, E. Vivaldo-Lima, P.M. Reilly, and A. Penlidis. Optimal bayesian design of experiments applied to nitroxide-mediated radical polymerization. *Macromolecular Reaction Engineering*, 4(6-7):387–402, 2010.

[64] H. Najm. Uncertainty quantification and polynomial chaos techniques in computational fluid dynamics. *Annual Review of Fluid Mechanics*, 41(1), 2009.

[65] W.L. Oberkampf, S.M. DeLand, B.M. Rutherford, K.V. Diegert, and K.F. Alvin. Error and uncertainty in modeling and simulation. *Reliability Engineeringa and System Safety*, 75(3):333–357, 2002.

[66] N. Omidbakhsh, T.A. Duever, A. Elkamel, and P.M. Reilly. A bayesian experimental design approach for assessing new product performance: An application to disinfectant formulation. *The Canadian Journal of Chemical Engineering*, 88(1):88–94, 2010.

[67] B.D. Phenix, J.L. Dinaro, M.A. Tatang, J.W. Tester, J.B. Howard, and G.J. McRae. Incorporation of parametric uncertainty into complex kinetic mechanisms: Application to hydrogen oxidation in supercritical water. *Combustion and Flame*, 112(1-2):132–146, 1998.

[68] F. Pukelsheim. *Optimal design of experiments.* Classics in applied mathematics, 50. SIAM/Society for Industrial and Applied Mathematics, Philadelphia, 2006.

[69] C.P. Robert and G. Casella. *Monte Carlo statistical methods.* Springer texts in statistics. Springer, New York, 2004.

[70] C.P. Robert and G. Casella. *Introducing Monte Carlo methods with R.* Springer, New York; London, 2009.

[71] K.J. Ryan. Estimating expected information gains for experimental designs with application to the random fatigue-limit model. *Journal of Computational and Graphical Statistics*, 2003.

[72] W.H.A. Schilders, H.A.v.d. Vorst, and J. Rommes. Model order reduction theory, research aspects and applications, 2008 2008. (Online service).

[73] D. Skanda and D. Lebiedz. An optimal experimental design approach to model discrimination in dynamic biochemical systems. *Bioinformatics*, 26(7):939–945, 2010.

[74] J.C. Spall. *Introduction to stochastic search and optimization : estimation, simulation, and control.* Wiley-Interscience series in discrete mathematics and optimization. Wiley-Interscience, Hoboken, N.J., 2003.

[75] A.H. Stroud. *Numerical quadrature and solution of ordinary differential equations; a textbook for a beginning course in numerical analysis.* Springer-Verlag, New York, 1974.

[76] M.A. Tatang. *Direct incorporation of uncertainty in chemical and environmental engineering systems.* PhD thesis, Massachusetts Institute of Technology, 1995.

[77] M.A. Tatang, W. Pan, R.G. Prinn, and G.J. McRae. An efficient method for parametric uncertainty analysis of numerical geophysical models. *J. Geophys. Res.*, 102, 1997.

[78] C. Tommasi and J. Lpez-Fidalgo. Bayesian optimum designs for discriminating between models with any distribution. *Computational Statistics & Data Analysis*, 54(1):143–150, 2010.

[79] A. Vikhansky and M. Kraft. A monte carlo methods for identification and sensitivity analysis of coagulation processes. *Journal of Computational Physics*, 200(1):50–59, 2004.

[80] A. Vikhansky and M. Kraft. Two methods for sensitivity analysis of coagulation processes in population balances by a monte carlo method. *Chemical Engineering Science*, 61(15):4966–4972, 2006.

[81] M; Villadsen, J.M. *Solution of Differential Equation Models by Polynomial Approximation.* Prentice-Hall, Englewood Cliffs, N.J. :, 1978.

[82] L. Vogel and W. Peukert. Separation of the influences of material and machine in impact comminution - modelling with population balances. *AUFBEREITUNGSTECHNIK*, 2002.

[83] L. Vogel and W. Peukert. Modelling of grinding in an air classifier mill based on a fundamental material function. *Kona*, 2003.

[84] L. Vogel and W. Peukert. From single particle impact behaviour to modelling of impact mills. *Chemical Engineering Science*, 60(18):5164–5176, 2005.

[85] D. Whitley. A genetic algorithm tutorial. *Statistics and Computing*, 4:65–85, 1994.

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

[87] D. Xiu. Fast numerical methods for stochastic computations: A review. *Commun. Comput. Phys.*, 5(2-4):242–272, 2009.

[88] D. Xiu. *Numerical methods for stochastic computations : a spectral method approach.* Princeton University Press, Princeton, N.J., 2010.

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

[90] D. Xiu and G.E. Karniadakis. The wiener–askey polynomial chaos for stochastic differential equations. *SIAM J. Sci. Comput.*, 24(2):619–644, 2002.