A Bayesian approach to feed reconstruction

by

Naveen Kartik Conjeevaram Krishnakumar

B.Tech Chemical Engineering, Indian Institute of Technology, Madras (2011)

Submitted to the School of Engineering
in partial fulfillment of the requirements for the degree of

Master of Science in Computation for Design and Optimization

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2013

Author .....
School of Engineering
May 23, 2013

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

Accepted by .....
Nicolas Hadjiconstantinou
Co-Director, Computation for Design and Optimization

A Bayesian approach to feed reconstruction

by

Naveen Kartik Conjeevaram Krishnakumar

Submitted to the School of Engineering
on May 23, 2013, in partial fulfillment of the
requirements for the degree of
Master of Science in Computation for Design and Optimization

Abstract

In this thesis, we developed a Bayesian approach to estimate the detailed composition of an unknown feedstock in a chemical plant by combining information from a few bulk measurements of the feedstock in the plant along with some detailed composition information of a similar feedstock that was measured in a laboratory. The complexity of the Bayesian model combined with the simplex-type constraints on the weight fractions makes it difficult to sample from the resulting high-dimensional posterior distribution. We reviewed and implemented different algorithms to generate samples from this posterior that satisfy the given constraints. We tested our approach on a data set from a plant.

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

Acknowledgments

First off, I would like to acknowledge my parents Gowri and Krishnakumar, to whom I owe all my past, present and future success. Their unconditional love and support enabled me to pursue my dreams and desires, and become the person I am today. I would also like to thank my advisor Professor Youssef Marzouk, who shaped my journey through MIT. He is intelligent, inspirational, and most of all, the best friend a student could ask for. I could fill this entire page, and still not convey the deep sense of gratitude that I feel today. However, a simple thank you will have to suffice for now. I would like to thank the members of the ACDL community and the UQ lab, for being amazing friends, co-workers and labmates. In particular, I would like to thank Alessio Spantini, who endured many a sleepless night solving problem sets and assignments with me, and Tiangang Cui, who would always patiently listen to my research ramblings.

I would also like to thank my all my friends, both in and outside Boston, who helped me helped me maintain my sanity through the crazy times at MIT. Finally, I would like to thank Barbara Lechner, our program administrator, who passed away recently. She was a truly warm and kind person, who would put the needs of students before herself. We will all miss her wonderful smile.

I would also like to acknowledge BP for supporting this research, and thank Randy Field for his help through the project.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Feed reconstruction in chemical processes

In chemical plants and oil refineries, the typical process stream contains a mixture of a large number of molecular species. In oil refineries for example, it is not uncommon to observe streams with several thousand hydrocarbon species [19]. When these streams are used as a feed to reactors or unit operations in the plant, the detailed composition of the stream becomes important. Accurate knowledge of the stream composition in terms of the molecular constituents allows us to build good operating models (such as kinetic models), which in turn can be used control the reactor conditions, product yield and quality. Recent advances in analytical chemistry techniques (such as conventional gas chromatography (1D-GC) and comprehensive two-dimensional gas chromatography (GC x GC) [35]) allow us to to obtain detailed molecular compositions of various streams. However, these techniques are expensive, time consuming and rarely done in-plant. This means that it is not possible to analytically obtain the detailed composition of various streams in the plant on a regular basis. However, the dynamic nature of the modern refinery or chemical plant results in a changing stream composition on a daily, if not hourly, basis. To overcome this difficulty, it is common practice to resort to feed reconstruction techniques.

Feed reconstruction or composition modeling techniques are a class of algorithms that are used to estimate the detailed composition of a mixture starting from a limited

number of bulk property measurements (such as average molar mass, distillation data, specific density, atomic concentrations, etc.) [34]. In the absence of detailed experimental measurements, feed reconstruction algorithms provide a rapid way to obtain detailed composition data. These techniques have become increasingly popular with the rise of cheap computational power and the high cost of analytical techniques.

To handle the sheer number of molecular components that are present in a typical feed, it is common practice to represent groups of molecules with a single representative molecule called a pseudo-compound [19]. This procedure is also employed in analyzing the kinetics of complex mixtures of molecules, and is sometimes referred to as a lumping approach [22]. The pseudo-compound framework allows us to represent the stream with a significantly smaller set of species. However, it is still diffucult to experimentally ascertain the exact concentrations of these pseudo-compounds. We shall first elaborate on the pseudo- compound/lumping framework that we utilize here before we describe the general feed reconstruction process.

1.2 Pseudo-compound framework

The pseudo-compounds that are chosen to represent a particular feedstock must be sufficiently representative of the underlying actual molecular species to minimize any resulting lumping errors. In an oil refining context, the large number of hydrocarbons that that constitute any typical feed stream allow for a variety of pseudo-compound representations.

One popular choice of pseudo-compounds [19, 29, 35, 46] is based on identifying chemical families, which may be based on certain parameters, such as structural or reactive attributes. Once the chemical families are identified, the pseudo-compounds are then generated by choosing different carbon number homologues corresponding to each chemical family. For instance, if we identify the Thiophene family (molecules that contain at least one thiophene ring) as one possible constituent of our feedstock, then we may choose C1-Thiophene, C3-Thiophene and C7-Thiophene as pseudo-compounds to represent all members of the Thiophene family in the feedstock. The

chemical families that are identified often depend on the feedstock in question. Analytical techniques can be used to understand the molecular composition of the feedstock and reveal the types of chemical families that are required to model it. In Gas Oil feeds, for example, experimental analyses have been used to define 28 different chemical families that can be processed in a refinery [19], while a larger number of such families have been used to model vacuum gas oil cuts [46].

The carbon numbers corresponding to homologues of the chemical families are chosen based on the boiling range of the feedstock. Gas oil, which has a low boiling range, is typically composed of low carbon number molecules, while high boiling feeds such as vacuum gas oil require a higher carbon number range to model accurately.

This approach to feed modeling, which assumes that the given mixture of compounds can be described by a fixed library of pseudo-compounds , is said to be a deterministic representation of the feed. Stochastic representations, on the other hand, do not rely on a fixed library of pseudo-compounds. Instead, they rely on a distribution of molecular attributes and sample from this distribution to generate a realization of pseudo-compounds [29, 46]. However, stochastic methods are usually difficult to implement because they are computationally intensive and rely on distribution information that is usually hard to obtain [50].

Ultimately, the choice of pseudo-compounds depends on the purpose and the desired accuracy of the modeling exercise. For low-fidelity models, it may be sufficient to pick a few chemical families and carbon number homologues. State of the art modeling [35, 34, 50, 27] techniques rely on generating a large number of chemical families by varying structural attributes and then using finely discretized carbon number ranges. The modeler has to choose the pseudo-compounds appropriately, so as to reduce error in the estimation of any variables of interest.

1.3 Current approaches to feed reconstruction

Liguras and Allen [23] proposed one of the first feed reconstruction algorithms, which was based on minimizing a weighted objective function based on bulk measurements

collected using NMR spectroscopy. Their method was innovative, but it required a large number of property measurements to be effective. Since then, several different feed reconstruction techniques have been proposed.

Typical modern modeling techniques involve minimizing a specific function of deviation of the observed bulk measurements from the calculated bulk measurements by varying the concentrations of the pseudo-compounds that are assumed to constitute the feed. Androulakis *et al.* [4] used a constrained weighted least squares approach to modeling the composition of diesel fuel. Van Geem *et al.* [45] proposed a feed reconstruction scheme that characterizes a given feedstock by maximizing a criterion similar to Shannon's entropy. Quanne and Jaffe propose a similar method, where the pseudo-compounds are replaced by vectors called Structured Oriented Lumping (SOL) vectors. These methods assume that the feedstock being modeled can be completely described by the fixed library of pseudo-compounds.

Alternative approaches to feed reconstruction have focused on coupling the minimization step with stochastic representation of the feed [29]. The pseudo-compounds are first generated using a Monte Carlo routine, and then their concentrations are calculated so that the computed bulk properties match the experimental measurements. Trauth *et al.* [44] used this method to model the composition of petroleum resid, while Verstraete *et al.* [46] adopted a similar idea to model vacuum gas oils. To reduce the computational burden associated with stochastic models, Campbell *et al.* [9] used a Monte Carlo generation method coupled with a quadrature technique to work with a reduced basis of pseudo-compounds. Pyl *et al.* [34] proposed a stochastic composition modeling technique that uses constrained homologous series and empirically verified structural distributions to greatly reduce the number of unknowns in the optimization step.

1.3.1 Drawbacks of current feed reconstruction schemes

While feed reconstruction schemes have been around for quite some time, they all suffer from similar drawbacks. The deterministic approaches to feed reconstruction sometimes require bulk properties that are hard to obtain, while the stochastic ap-

proaches rely on attribute distribution information that is mostly fixed on an ad-hoc or empirical basis.

In typical composition modeling techniques, it is difficult to add any new measurement information in a systematic manner. For example, if a modeler wishes to add a new type of bulk measurement to increase the accuracy of the predicted composition, it is not immediately obvious how one may modify the algorithm to incorporate this new piece of information. In particular, if we have any past detailed measurements that could improve current concentration estimates, we cannot take advantage of this information without significantly modifying the algorithm.

Furthermore, traditional composition modeling techniques do not incorporate a systematic idea of uncertainty in the reconstruction process. Understanding uncertainty in the composition of any feed stream in a chemical plant/refinery is very important. The uncertainty in the concentration of any pseudo-compounds of interest allows us to understand the impact of variation of bulk properties on the feed composition. Furthermore, the output of the feed reconstruction process might be used to model downstream unit operations like chemical reactors. In that case, having a handle on the input uncertainty to a unit operation gives us an idea of the resulting output uncertainty of the process. This becomes particularly important in applications such as oil refining, where the final product is expected to conform to certain quality standards. Having a low output uncertainty ensures that the quality of the product remains consistent across time.

Finally, from an experimental design point of view, an accurate quantification of uncertainty in the feed reconstruction process would allow us to analyze if we can reduce the uncertainty in any particular quantities of interest by incorporating additional experimental measurements. For example, environmental regulations require a strict control on the level of sulphur in diesel produced in oil refineries. Therefore, it is important to estimate the composition of sulphur-containing psuedo-compounds with a high degree of confidence. However, it may only be possible to obtain a small number of bulk measurements. If we can quantify the uncertainty in the feed reconstruction, it is possible to implement an experimental design procedure to choose the

bulk measurements that yield the lowest level of uncertainty (or inversely, the highest degree of confidence) in the concentrations of interest. The drawbacks that we have outlined in this section motivate the construction of a new feed reconstruction approach, by adopting a Bayesian approach to the problem.

1.4 Bayesian inference

Parametric inference is the branch of statistical inference that is concerned with identifying parameters of a model given noisy data that is assumed to be generated from the same model. Traditional parameter inference frameworks (sometimes referred to as frequentist inference) assume that the parameters in a model are constant and any deviation in measurement from the model prediction is attributed to noise [48]. Typical frequentist inference techniques focus on finding a best estimate for each parameter (also called a point estimate), along with an associated confidence interval.

The Bayesian approach to statistics is motivated by the simple idea that a probability value assigned to an event represents a degree of belief, rather than a limiting frequency of the event. We can extend this notion of "probability as a degree-of-belief" to inference using Bayesian statistics. Contrary to frequentist inference, Bayesian inference assumes that the underlying model parameters are random variables. Bayesian parametric inference computes a probability distribution for each unknown parameter from the data. Subsequent point estimates and confidence intervals are then calculated from the probability distributions.

Since the Bayesian inference procedure assigns probability distributions to each parameter, we need to choose a way to update the probability distributions with noisy data from the model. The most popular updating rule is the Bayes rule [41], which we shall describe below.

Let us denote the set of parameters as θ. First, we choose a probability distribution $p(\theta)$, called a prior, that reflects our beliefs about the parameters before observing any data (denoted by $\mathcal{D}$). Then, we choose a function $\mathcal{L}$ called a likelihood, which gives the probability of observing $\mathcal{D}$, given θ. Then, by Bayes rule,

$$p(\theta|\mathcal{D}) \propto \mathcal{L}(\mathcal{D}|\theta)p(\theta) \tag{1.1}$$

The posterior, $p(\theta|\mathcal{D})$, contains our updated belief on the value of θ given $\mathcal{D}$. Sometimes, the proportionality in (1.1) is replaced by an equality.The proportionality constant, denoted as, $p(\mathcal{D})$ is termed as the evidence. It can be evaluated as

$$p(\mathcal{D}) = \int_{\theta} \mathcal{L}(\mathcal{D}|\theta)p(\theta)d\theta$$

The Bayes rule provides a systematic way to update our beliefs regarding the values of the parameters of a model upon observing data that is generated from the model. The choice of the prior ($p(\theta)$) and the likelihood ($\mathcal{L}(\mathcal{D}|\theta)$) are very important, since they directly determine the accuracy of the estimated parameters.

1.5 Research objectives

The objectives of this research are twofold:

- Address the drawbacks of existing feed reconstruction schemes by recasting the problem into a Bayesian framework
- Review and develop efficient computational techniques to explore and analyze the Bayesian model

1.6 Thesis outline

In chapter 2, we outline a Bayesian solution to the feed reconstruction problem. In chapter 3, we review some computational techniques to analyze the result of the Bayesian model. In chapter 4, we test the computational algorithms on some sample problems, and then use the best algorithm to analyze a Bayesian model developed on a dataset from a real refinery. In chapter 5, we summarize our conclusions from this study and provide some future directions to extend this work.

Chapter 2

A Bayesian inference approach to feed reconstruction

In this chapter, we shall describe a Bayesian approach to the feed reconstruction problem. While we shall apply this feed reconstruction approach specifically to model crude fractions in an oil refinery, the general framework can be extended to other chemical systems.

We are interested in developing a Bayesian model for the following framework: We have the detailed description for a particular type of feedstock in terms of the chosen pseudo-compounds from analytical chemistry techniques in a laboratory setting. These experiments are time-consuming and expensive, so we cannot repeat them on a daily basis in the chemical plant/refinery. Instead, we would like to predict the detailed composition from properties like distillation curves and bulk concentrations that can be easily measured in the plant. In this framework, a feed reconstruction model should combine the detailed information from the laboratory with the bulk information of the feedstock measured in the plant to infer the detailed composition of the feedstock in the refinery (in terms of weight percents/fractions). A schematic representation of this feed reconstruction procedure is presented in figure (2-1).

In the following sections, we shall first explain type of laboratory and plant information that we assume is available to the feed reconstruction process. Then, we shall outline the Bayesian inference procedure and elaborate on the details.

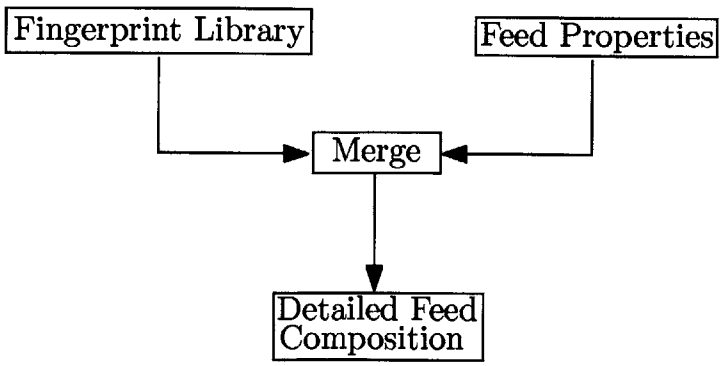

Figure 2-1: Feed reconstruction framework used to develop Bayesian model

2.1 Available data

2.1.1 Laboratory data

Detailed analytical techniques such as two-dimensional gas chromatography allow experimentalists to identify the concentrations (in weight percent) of individual species with great precision. By applying these techniques on a sample feedstock (which is assumed to be representative of any typical feed that you might encounter in the refinery), we can first identify a set of pseudo-compounds that can sufficiently describe the feed along with with their concentrations (in weight fractions) and molecular weights. The pseudo-compounds themselves are assumed belong to certain families, while individual pseudo-compounds within each family are differentiated by carbon number (as described in the previous chapter).

Since each pseudo-compound refers to a collection of molecules, it is not going to have a single boiling temperature. Instead, each pseudo-compound will boil over a temperature range. Along with their concentrations, we shall also assume that we can measure the initial and final boiling point of each pseudo- compound that is identified[1].

[1]Since boiling information is a by-product of most detailed analytical techniques, this assumption is not unjustified

2.1.2 Plant data

The Plant Data refers to any bulk measurements of the feed that are collected in the refinery. While bulk measurements (or assays) could refer to a wide range of physical properties (such as atomic concentrations, refractive indices, specific gravities etc.), we assume that any assay that is collected in the refinery includes some distillation analysis (such as the ASTM D86 distillation [12]). At the simplest level, the resulting data can be understood as a set of boiling ranges and fraction of the crude that boils in those corresponding ranges.

These assumptions on data availability that we have made so far might seem restrictive at first glance. The subsequent analysis however, is general enough to deal with the absence of any type of information (albeit with slight modifications). We have chosen this particular structure in our data available since this refers to the most general type of data that is used as an input to a feed reconstruction procedure.

2.2 Bayesian formulation of feed reconstruction

The first step in constructing a Bayesian model for a given problem is to identify the parameters in the problem. Once we identify the parameters, we need to assign a prior distribution on the range of values that the parameters can take. Then, we need to define a likelihood function on the data that we observe. Finally, we need to use equation (1.1) to compute the posterior distribution on the parameters. In the following subsections, we shall construct a Bayesian model for the feed reconstruction problem using the same outline.

2.2.1 Target variable

Before we begin constructing probability distributions, we must first identify the parameters that we wish to infer in modeling the composition of any feedstock. First, we assume that the modeler has the pseudo-compounds of interest from the laboratory data and the boiling temperature ranges for the feedstock from the plant data. We

have to infer the concentrations of the various pseudo-compounds that are present in the feed in the plant. We shall infer the concentrations (in terms of weight fractions) as the entries of a matrix, where the rows correspond to various pseudo-compounds (that are identified in the laboratory), and the columns refer to the temperature ranges that the feed boils over (that is measured in the plant). The (i, j)-th entry in this matrix denotes the the weight-fraction concentration of pseudo-compound i that boils in the j-th temperature interval. However, from the laboratory, we know that each pseudo-compound does not boil over every temperature range: For example, a pseudo-compound that boils in between 100 and 150° C, would not be expected to boil in a column corresponding to a temperature range of 200 to 400° C. On the other hand, a pseudo-compound that boils over 150 and 350° C would be expected to have nonzero entries in the columns corresponding to 100 to 200° C and 200 to 400° C. This idea enforces a great deal of sparsity in the structure of the matrix, which is sketched in figure (2).

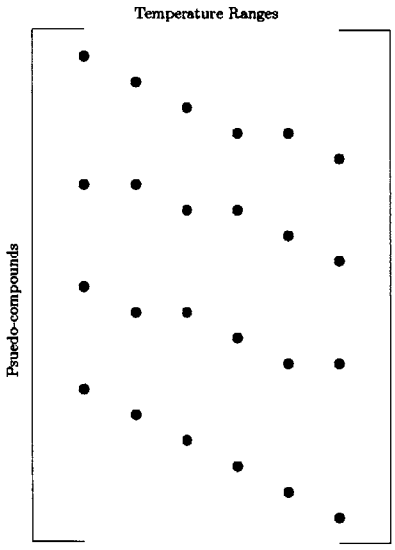

Figure 2-2: Grid of inference target weight fractions

It should be noted that there appears to be some additional structure in the matrix, since neighboring pseudo-compounds boil in consecutive temperature intervals. This is an artifact of the choice of our pseudo-compounds, since they are assumed to be members of the same family with differing carbon numbers. Our Bayesian

inference procedure will be focused only on the nonzero entries of the matrix. So while the procedure is explained in terms of this matrix, the final computations and mathematical operations are performed on a vector whose elements correspond to the nonzero entries of the matrix. We shall call this vector as our target variable of interest Y.[2]

2.2.2 Prior distribution

The prior distribution has to contain all the information about a feedstock before making any bulk measurements. In this case, the prior distribution has to encode the detailed laboratory data of a sample of crude that is similar to the feedstock in question. The choice of prior probability distribution has to ensure that the entries of Y are nonnegative and sum to one hundred (since they represent weight fractions). We model the distribution of entries in Y as a multivariate truncated normal distribution.

Truncated normal distribution

The truncated multivariate normal distribution is a probability distribution of a n-dimensional multivariate random variable that is restricted to to a subset ($\mathbb{S}$) of $\mathbb{R}^n$. Mathematically, this distribution is calculated as

$$p(Y; \mu, \Sigma, \mathbb{S}) \propto \exp\left(-\frac{(Y-\mu)^T\Sigma^{-1}(Y-\mu)}{2}\right) 1_{\mathbb{S}}(Y) \tag{2.1}$$

where μ and Σ are the mean and covariance parameters of the distribution respectively[3], and $1_{\mathbb{S}}(Y)$ is an indicator function defined as

$$1_{\mathbb{S}}(Y) = \begin{cases} 1 & \text{if } Y \in \mathbb{S} \\ 0 & \text{otherwise} \end{cases}$$

[2] Y can imagined as an "unrolling" of the matrix into a vector. For the purposes of understanding the Bayesian model construction, Y can be imagined as either a matrix or a vector.

[3] Note that these do not necessarily correspond to the mean and covariance of the truncated normal random variable. In this case, they are just parameters to the distribution.

Since $p(Y)$ is a probability distribution, the normalizing constant (say, C) in equation (2.1) can be calculated as

$$C = \int_{\mathbb{S}} - \exp\left(\frac{-(Y-\mu)^T\Sigma^{-1}(Y-\mu)}{2}\right) dY$$

In the feed reconstruction case, the subset $\mathbb{S}$ is defined as

$$\mathbb{S} = \{Y \in \mathbb{R}^n | \sum_{i=1}^{n} Y_i = 1,\ Y_i \geq 0,\ i = 1 \ldots n\}$$

The choice of the prior mean and prior covariance parameters for the distribution in equation (2.1) are detailed in the following sections.

Prior mean

The choice of the prior mean parameter is non-unique, and is a function of the type of information that is available. One particular choice (in this case, also the maximum likelihood choice) is the estimate of the entries of Y from the detailed analytical data. These estimates are not readily available, since there is is no accurate one-to-one correspondence between the the laboratory data and the entries of Y.

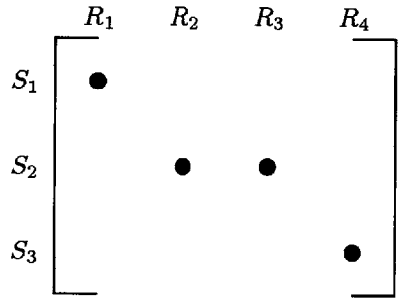

Figure 2-3: A schematic illustrating the "straddling" issue with assigning prior means, with a section for the matrix Y

For instance, in figure (2-3), the pseudo-compound S_1 boils completely in the temperature range R_1. In this case, the prior mean of the nonzero entry corresponding to pseudo-compound S_1 would simply be concentration of S_1 that is measured in the laboratory.On the other hand, the boiling range of a pseudo-compound could potentially straddle the temperature ranges corresponding to the columns of Y. The

pseudo-compound S_2 has a boiling range that straddles temperature ranges R_2 and R_3. This would imply that S_2 has nonzero entries in the column corresponding to R_4. However, there is no easy way to calculate the prior means corresponding to those nonzero entries.

One way to work around the mismatch in temperature ranges in the laboratory and the plant is to use an assumption on how the pseudo-compound boils in its temperature range. One valid assumption, for instance, is to assume that the pseudo-compound boils uniformly between its initial and final boiling point. In that case, the fraction of the pseudo-compound that boils in any arbitrary temperature range would be the ratio of the temperature range to the total boiling range. This assumption, although simple, is not really representative of how boiling occurs in practice. In this case, a more sophisticated assumption on the boiling profile of a pseudo-compound can improve the accuracy of the estimate of the prior mean.

Pyl *et al.* [34] observe that there is a correlation between the carbon number of a hydrocarbon within a homologous series and its boiling point. In this particular feed reconstruction case, this idea would imply that there is a correlation between the boiling range and the carbon number of each pseudo-compound within a particular family (which would be equivalent to a homologous series). Pyl *et al.* further suggest using an exponential or chi-squared distribution to model the relationship. Whitson *et al.* [49] suggest fitting a curve which takes the same functional form as a gamma distribution to petroleum fractions. While there are many choices to model the boiling profile of a feedstock , the gamma distribution is flexible and well-suited to the Bayesian reconstruction scheme. The details of the gamma distribution model are discussed below.

Mathematically, the gamma distribution is a function that models the probability distribution of a gamma random variable. The expression for the probability distribution is given by

$$p(T = t; \alpha, \beta, t_0) = \begin{cases} \frac{\beta^\alpha}{\Gamma(\alpha)}(t - t_0)^{\alpha-1} e^{-\beta(t-t_0)} \text{ , if } t \geq t_0 \\ 0 \text{ otherwise} \end{cases} \tag{2.2}$$

where α and β are 2 positive parameters of the gamma distribution, called shape and scale respectively. The parameter t_0 is a shift term, which represents the lowest possible value of T that has a nonzero density value.

This gamma distribution for the reconstruction case would be a function of the boiling point (T), and the area under the curve between any two boiling points would correspond to the weight fraction of the substance that boils in that temperature range. Furthermore, since equation (2.2) is a probability distribution,

$$\int_0^\infty p(t;\alpha,\beta)dt = 1$$

Using Pyl *et al.* 's suggestion, this gamma distribution is used to model the boiling profile within each family. If the shape (α^k), scale (α^k) and shift term (t_0^k) are available for each family, it is possible to compute the weight fraction of any pseudo-compound that boils in a particular temperature range (as a fraction of the total family) using a simple procedure.

Suppose that pseudo-compound i (which belongs to a family k) boils in a temperature range $[L_i, U_i]$. The fraction of pseudo-compound i that boils in any arbitrary temperature range $[T_1, T_2]$ can be calculated as

$$w_i = \begin{cases} 0 \text{ , if } T_2 \leq L_i \\ \int_{L_i}^{T_2} p(t;\alpha^k,\beta^k,t_0^k)dt \text{ , if } T_1 < L_i \text{ and } T_2 \leq U_i \\ \int_{T_1}^{T_2} p(t;\alpha^k,\beta^k,t_0^k)dt \text{ , if } T_1 \geq L_i \text{ and } T_2 \leq U_i \\ \int_{T_1}^{U_i} p(t;\alpha^k,\beta^k,t_0^k)dt \text{ , if } T_1 \geq L_i \text{ and } T_2 > U_i \\ 0 \text{ , if } T_1 \geq U_i \end{cases} \tag{2.3}$$

The resulting w_i is a fraction of the concentration of the family k that boils as pseudo-compound i in the temperature range $[T_1, T_2]$. To convert this fraction into a concentration of the pseudo-compound, an additional normalizing parameter for each family (say, c^k) is required. Then, the final concentration of pseudo-compound i that boils in $[T_1, T_2]$ is given by $c^k w_i$.

Given the parameters $\psi^k = \{\alpha^k, \beta^k, t_0^k, c^k\}$, this procedure provides a systematic way to populate the prior mean entries of Y. Now, all that remains is to to compute the parameters ψ^k from the laboratory data.

Ideally, family k should not boil at any temperature below the lowest boiling point of all members (say, L_0^k). In other words,

$$\int_0^{L_0^k} p(t; \alpha^k, \beta^k, t_0^k) dt = 0$$

This is consistent with a choice of $t_0^k = L_0^k$.

Suppose that there are n_k pseudo-compounds that belong to a particular family k. The concentrations (c_i, $i = 1, \ldots, n_k$) of these pseudo-compounds are determined in the laboratory, along with their individual boiling ranges ($[L_i, U_i]$, $i = 1, \ldots, n_k$). First, the concentrations are normalized by their sum to result in weight fractions for each pseudo-compound $w_i = c_i / \sum_i c_i$.

Now, the values of α^k, β^k and c^k have to be estimated from these quantities. First, c_k is chosen to be $\sum_{i=1}^{n_k} c_i$. In other words, the total concentration of each family serves as an estimate of the normalization constant for each family. Then, the parameters α^k and β^k have to be chosen such that

$$w_i = \int_{L_i}^{U_i} p(t; \alpha^k, \beta^k, t_0^k) dt \quad i = 1, \ldots, n_k \tag{2.4}$$

Since p is a probability distribution, we can replace the integral in equation (2.4) with the cumulative distribution function F, which is defined as

$$F(T = t; \alpha^k, \beta^k, t_0^k) = \int_{-\infty}^{t} p(t; \alpha^k, \beta^k, t_0^k) dt \tag{2.5}$$

When p is a gamma distribution, the expression for F can be written as

$$F(T = t; \alpha^k, \beta^k, t_0^k) = \frac{\gamma\left(\alpha^k, \beta^k (t - t_0^k)\right)}{\Gamma(\alpha)}$$

where Γ is the gamma function, and γ is the lower incomplete gamma function [2].

Combining equations (2.4) and (2.5), we get,

$$w_i = F(U_i; \alpha^k, \beta^k, t_0^k) - F(L_i; \alpha^k, \beta^k, t_0^k) \quad i = 1, \ldots, n_k \tag{2.6}$$

Given the values of w_i and $[L_i, U_i]$, $i = 1, \ldots, n_k$, a statistical inference procedure can estimate the values of α_k and β_k. A frequentist maximum likelihood approach can be used to solve this problem to yield the "best" estimates for α^k and β^k for each family. This prior parameter estimation process is outlined in figure (2-4).

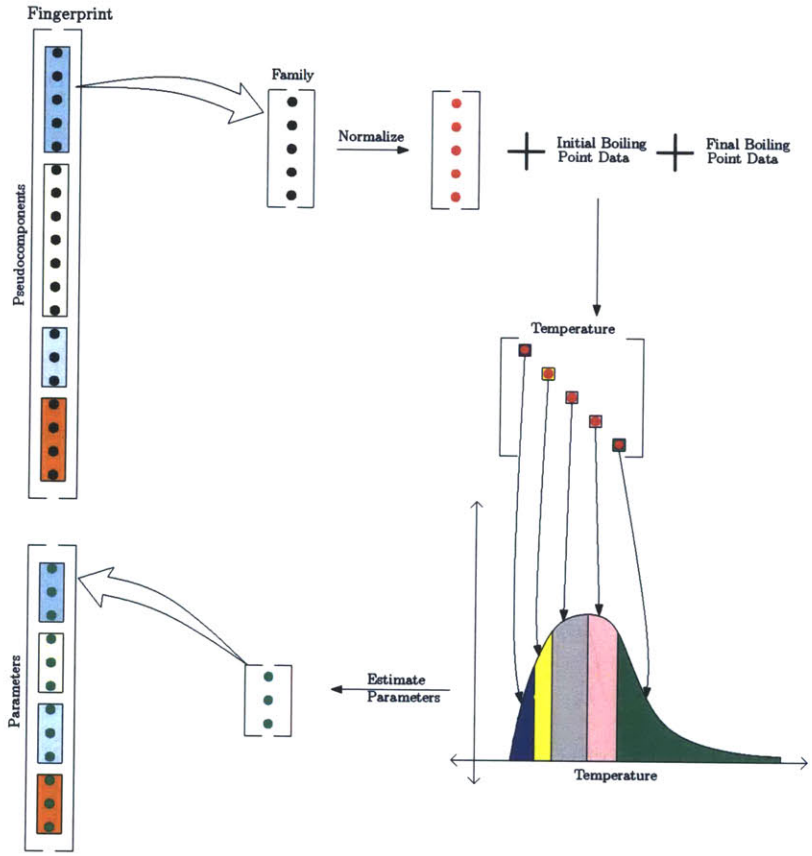

Figure 2-4: A schematic illustrating the prior parameter estimation procedure

By this outlined procedure, it is possible to estimate a set of parameters $\psi^k = \{\alpha^k, \beta^k, t_0^k, c^k\}$ (or in this case, hyperparameters to the prior mean) for each family from the detailed information from the laboratory. Once the parameters are estimated, they can be used to compute the prior mean by the formula detailed in equation (2.3) and the following section.

Prior covariance

Ideally, the prior covariance is a matrix whose (i, j)-th entry captures the how the concentrations of pseudo-compound i and j vary together. However, as a result of the large underconstrained nature of the feed reconstruction problem, it is often better to use some simplifying assumption to reduce the number of unknown covariance parameters. In this case, the prior covariance matrix Σ is assumed to be of the form $\sigma^2 \mathbb{I}_m$, where $\sigma > 0$ is a variance parameter and $\mathbb{I}_m$ is the $m \times m$ identity matrix. Now, the only parameter that has to be tuned is σ.

The variance parameter indicates the degree-of-belief in the similarity of the feed that is analyzed in the laboratory and the feed that is being analyzed in the refinery. If the 2 types of feed are believed to be similar (sourced from the same reservoir or neighboring reservoirs, for example), then a lower value of the variance parameter can be used. If the two feeds are very different, then a higher value of variance would be appropriate.

2.2.3 Likelihood

If $\mathcal{D}_p$ denotes the vector of bulk measurements that are collected in the plant, then the likelihood function $\mathcal{L}(\mathcal{D}_p|Y)$ is a function that quantifies how likely the observed plant data are, given realization of Y.

To construct this likelihood function, a measurement model is required. A measurement model is a function that calculates the bulk properties for a given realization of Y. If the vector of calculated bulk properties is denoted by $\mathcal{D}_{calc}$, the measurement model is a function f, such that $\mathcal{D}_{calc} = f(Y)$. For the sake of convenience, the Bayesian feed reconstruction model assumes that this function f is linear in Y. While this idea ensures that the expression for the posterior distribution can be derived analytically, this is not a necessary condition.

While the linearity assumption might seem unjustified, the choice of the structure of Y makes it convenient to reconstruct several types of bulk measurements typically encountered in a refinery in a linear fashion. In the pseudo-compound and tempera-

ture matrix (which was described in the previous subsection), the columns correspond to the distillation temperature ranges. So, the weight fraction of the feed that is distilled in each temperature range can be recovered as a simple column sum, which is a linear operation on Y. Bulk concentrations of atomic species, such as sulphur, can be calculated as

$$S_{calc} = \sum_i \left(\frac{MW_S}{MW_{Y_i}} \right) Y_i \tag{2.7}$$

where MW_S is the molecular weight of sulphur, MW_{Y_i} is the molecular weight of the pseudo-compound corresponding to Y_i. This is also a linear function of Y.

Since the calculated bulk properties can never match the actual observed bulk properties ($\mathcal{D}_p$), an error model on the observation is also required. For simplicity's sake, a Gaussian measurement error is associated with each measurement. If the vector of observed bulk properties is denoted as $\mathcal{D}_{obs}$, then

$$(\mathcal{D}_{obs})_i = (\mathcal{D}_{calc})_i + \epsilon_i, \ \epsilon_i \sim \mathcal{N}(0, \sigma_i^2), \quad i = 1, \ldots, n_{bulk} \tag{2.8}$$

where n_{bulk} is the number of bulk measurements. Here, ϵ_i is a Gaussian measurement noise in the i-th bulk measurement with zero mean and variance σ^2 (which is usually available as instrument error). It is assumed that all the measurement errors are independent of each other.

With these assumptions, the likelihood function can be written as

$$\mathcal{L}(\mathcal{D}_p|Y) = \exp\left(-\frac{(\mathcal{D}_p - f(Y))^T \Sigma_m^{-1} (\mathcal{D}_p - f(Y))}{2} \right) \tag{2.9}$$

Σ_m is a diagonal matrix where the (i,i)-th entry corresponds to the measurement error in the i-th bulk measurement.

2.2.4 Posterior distribution

With the prior distribution and the likelihood, the posterior distribution can now be computed using equation (1.1) as

$$p(Y|\mathcal{D}_p) \propto \mathcal{L}(\mathcal{D}_p|Y)p(Y) \tag{2.10}$$

If the prior is a truncated normal distribution, and the likelihood in equation (2.9) is used with a linear function $f(Y) = GY$ (say), then a simple expression for the posterior can be derived. From equation (2.10),

$$\begin{aligned} p(Y|\mathcal{D}_p) &\propto \mathcal{L}(\mathcal{D}_p|Y)p(Y) \\ &\propto \exp\left(-\frac{(\mathcal{D}_p - GY)^T\Sigma_m^{-1}(\mathcal{D}_p - GY)}{2}\right)\exp\left(-\frac{(Y-\mu)^T\Sigma^{-1}(Y-\mu)}{2}\right)1_{\mathbb{S}}(Y) \end{aligned}$$

Using some algebra, this expression can be rearranged [40], to give

$$p(Y|\mathcal{D}_p) \propto \exp\left(-\frac{(Y-\mu_{post})^T\Sigma_{post}^{-1}(Y-\mu_{post})}{2}\right)1_{\mathbb{S}}(Y) \tag{2.11}$$

where

$$\Sigma_{post} = (G^T\Sigma_m^{-1}G + \Sigma)^{-1} \; ; \; \mu_{post} = \Sigma_{post}G^T\Sigma_m^{-1}\mathcal{D}_p \tag{2.12}$$

From equation (2.11), it is clear that the posterior is a truncated normal distribution as well, with the parameters stated in equation (2.12).

Equation (2.11) is a distribution over the nonzero entries of the matrix of pseudo-compounds and temperature ranges. The final distribution over the concentrations of the pseudo-compounds can be obtained with a simple row-wise sum. If Z represents a vector of concentrations of the pseudo-compounds, then Z can be computed using a linear of transformation of Y, say, HY. Then, the posterior distribution[4] of Z is

$$p(Z|\mathcal{D}_p) \propto \exp\left(-\frac{(Z-H\mu_{post})^T(H^T\Sigma_{post}H)^{-1}(Y-H\mu_{post})}{2}\right)1_{\mathbb{S}}(Z) \tag{2.13}$$

[4]Note that this expression is valid only when $\dim(Z) \leq \dim(Y)$, which is always true. Otherwise, the resulting covariance matrix will be improper

2.3 Advantages of the Bayesian approach

The Bayesian feed reconstruction procedure constructs a posterior distribution over the concentrations of each pseudo-compound by combining the laboratory information and plant data in a systematic manner. Unlike previous composition modeling techniques, the explicit identification of the prior distribution and the likelihood makes it easy to add new sources of information while retaining intuitive knobs on the reliability of the new data, compared to existing information.

If a new kind of laboratory measurement is made, for instance, it can be incorporated into the reconstruction process by suitably modifying the prior distribution. On the other hand, if a new bulk measurement is made, it can be appended to the existing measurement vector $\mathcal{D}_p$, as long as it has an appropriate measurement model (preferably, linear) and an associated measurement error (σ_i^2).

The notion of uncertainty is directly built into the Bayesian inference procedure. The concentrations Z are assumed to be realizations of a random variable with a multivariate probability distribution. Techniques to perform sensitivity analysis and experimental design on the reconstruction or any subsequent downstream process are readily available.

2.4 Exploring the posterior distribution

Inference procedures on random variables usually focus on expectations or integrals of the probability distribution function. For example, one quantity of interest to the modeler might be the average concentration, expressed as $\mathbb{E}(Z)$. To analyze the variation of the concentration of a pseudo-compound, the modeler might choose to analyze the percentiles of the probability distribution function (which is an integral of the density function).

The posterior probability distribution $p(Z|\mathcal{D}_p)$ is a high-dimensional distribution (since reconstruction procedures employ a high number of pseudo-compounds to improve modeling accuracy). While the expression for this distribution is simple, there

are no closed-form expressions to estimate the moments or expectations of this distribution $p(Z|\mathcal{D}_p)$.

To understand and utilize the resulting posterior distribution, there is a clear need to evaluate arbitrary expectations of Z. In the next section, we shall motivate the idea on how to evaluate these high-dimensional integrals.

Chapter 3

Computational aspects of feed reconstruction

3.1 Introduction

In the previous section, a Bayesian formulation for the feed reconstruction problem was proposed and an expression for the posterior density function was derived. To analyze the posterior distribution, it often becomes necessary to evaluate integrals of the form

$$\mathbb{E}_Z[f(z_1,\ldots,z_n)] = \int\int_{\mathbb{S}\subset\mathbb{R}^n}\cdots\int f(z_1,\ldots,z_n)p(z_1,\ldots,z_n)d\mathbb{S}(z_1,\ldots,z_n) \tag{3.1}$$

Where Z is a random variable with a multivariate probability density function $p(z_1,\ldots,z_n)$ that is truncated on a set $\mathbb{S}$. In the feed reconstruction case, the probability density function would be the Gaussian density function with some mean and covariance parameters, and $\mathbb{S}$ would be the set

$$\mathbb{S} = \{z \in \mathbb{R}^n | \sum_{i=1}^{n} z_i = 1,\ z_i \geq 0,\ i = 1\ldots n\} \tag{3.2}$$

This set corresponds to a scaled version of the standard $(n-1)$-simplex, which is

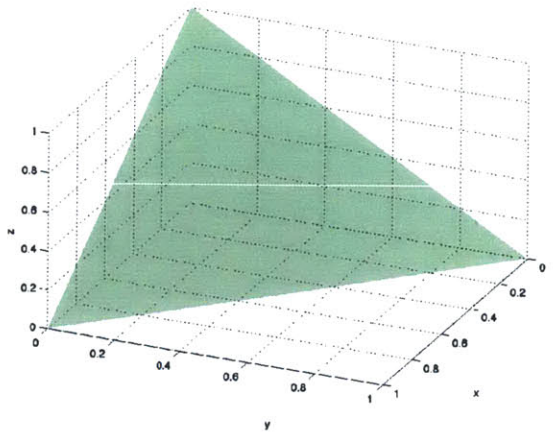

Figure 3-1: The standard 2-simplex corresponding to the set $\mathbb{S} = \{z \in \mathbb{R}^3 | \sum_{i=1}^{3} z_i = 1,\ z_i \geq 0,\ i = \{1,2,3\}\}$

sketched in figure (3-1).

In general, there are no analytical techniques to evaluate these integrals for cases where the density $p(z)$ is the multivariate truncated normal distribution. These distributions play an important role in a wide range of modeling applications even outside feed reconstruction [3, 28], so devising a good algorithm to evaluate these integrals is important. Numerical integration (or, quadrature) techniques are algorithms that are used to compute approximations definite integrals. If the integrals are univariate, there are several quadrature techniques that are readily available[32]. If the integrals are multivariate, numerical integration (or, cubature) using repeated application of 1D quadrature techniques does not scale well with the number of dimensions. For high-dimensional numerical integration problems, Monte Carlo methods or sparse grid techniques are usually the methods of choice.

3.1.1 Monte Carlo methods

Monte Carlo (MC) methods are computationally intensive stochastic sampling techniques that can be used to numerically evaluate integrals. They rely on repeatedly sampling from random variables to carry out numerical computations. The idea was first proposed and developed by mathematicians and physicists at the Los Alamos labs during World War II. With the exponential increase in computing power, Monte

Carlo methods [1] are now very popular and widely used in different areas of computational science.

One of the biggest advantages of MC methods lies in the scaling of the error in the numerical estimate I_{est}. If equation (3.1) is evaluated by repeatedly applying 1D quadrature techniques, the error bound $|I_{est} - I|$ is $O(N^{-r/d})$, where d is the number of dimensions, and r is typically 2 or 4. As a result, typical quadrature methods become too computationally expensive outside low-dimensional ($d < 7$) problems.

However, the error bound in MC methods is $O(N^{-1/2})$, independent of the number of dimensions. This makes MC methods particularly attractive for high dimensional problems[1].

3.2 Numerical integration using Monte Carlo sampling

The Monte Carlo integration algorithm numerically computes equation (3.1) by generating N random points z^i, $i = 1, \ldots, N$, such that $z^i \sim p(z)$), where $p(z)$ is assumed to be supported on $\mathbb{S}$. Then, the value of the integral can be approximately computed as the arithmetic mean of the function evaluations at z^i [24]. Mathematically,

$$\mathbb{E}_Z[f(z_1, \ldots, z_n)] \approx I_{MC} = \frac{1}{N} \sum_{i=1}^{N} f(z_1^i, z_2^i, \ldots, z_n^i), \quad i = 1, 2, \ldots, N \tag{3.3}$$

Note that this method can also be used to numerically integrate any function, if the probability density is chosen to be the uniform density function (assuming that $\mathbb{S}$ is compact) [2].

In the feed reconstruction problem, the probability distribution of interest $p(\mathbf{Z})$ is

[1] It is worth mentioning that MC methods nowadays refer to a very broad class of algorithms. We are particularly interested in MC methods for numerical integration, so any mention of MC methods are in the context of numerical integration

[2] While it might be tempting to evaluate $\mathbb{E}_\mathbf{Z}[f(z_1, \ldots, z_n)]$ by numerically integrating $g(Z) = f(Z).p(Z)$ using a uniform distribution, it should be noted that doing so would drastically reduce the accuracy of the Monte Carlo method[24]

a multivariate Gaussian distribution that is truncated on a simplex[3].

Monte Carlo integration relies on the ability to generate (almost) independent samples from our truncated normal distribution $p(z)$, to ensure the accuracy of the Monte Carlo estimate[43]. While several sampling techniques exist in literature[24], the next section focuses on a particular class of methods called Markov Chain Monte Carlo methods.

3.3 Markov chain Monte Carlo

Markov chain Monte Carlo (MCMC) methods are a class of algorithms that sample a probability distribution, $p(z)$, by generating a Markov chain that has the same stationary distribution [18]. With the rise of computer processing power, there has been an explosion in the research and development of MCMC algorithms [8].

3.3.1 Literature Review

Geweke [17] first addressed the problem of sampling from a normal distribution subject to linear constraints using an MCMC algorithm. However, the method was limited and allowed only for a fixed number of inequality constraints. Rodriguez-Yam *et al.* [38] extended Geweke's method to remove the restriction on the number of constraints, and introduced additional scaling in the problem to improve the sampling. Dobigeon *et al.* [13] explicitly address the problem of sampling from a multivariate Gaussian that is truncated on a simplex, by combining a Gibbs algorithm along with an accept-reject framework to sample from the 1-dimensional truncated Gaussian distribution [37].

Apart from the Gibbs family of algorithms, Smith *et al.* [6] proposed a particular type of random walk samplers called Hit-and-Run samplers that can be used to sample from distributions restricted to convex domains. Recently, Pakman *et al.* [30]

[3]For the sake of simplicity, this section will only discuss sampling from truncated normal distributions. Sampling from arbitrary posterior distributions truncated on simplex-type constraints is an even harder problem.

proposed a Hamiltonian Monte Carlo algorithm to sample from truncated Gaussian distributions.

3.3.2 Sampling problem

To analyze the results of the Bayesian model, we require samples from of the posterior distributions, which are multivariate normal distributions truncated on the standard simplex. However, for the sake of completeness, we shall analyze samplers that can generate samples from normal distributions that are truncated on convex polytopes[4]. Before we explore the different types of algorithms that can be used to sample from normal distributions truncated on convex polytopes, we can simplify the given sampling problem.

Consider sampling from the multivariate Gaussian distribution $p(z) = N(\mu, \Sigma)$, such that the samples z obey

$$\begin{aligned} A_{m\times n}z &= b_{m\times 1} \\ C_{k\times n}z &\leq d_{k\times 1} \end{aligned} \tag{3.4}$$

where m and k are the number of equality and inequality constraints respectively. These feasibility constraints correspond to a convex polytope in $\mathbb{R}^n$.

Note that the case of the truncated The Gaussian distribution can be conditioned such that the resulting function is a distribution only in the subspace defined by the equality constraints. This can be achieved by considering a new random variable $W = AZ$, and deriving a joint distribution for the random variable pair $V = (Z, W)$. Once the joint is obtained, setting $W = b$, will result in a conditional distribution $p(Z|W = b)$.

Mathematically, the random variable W is Gaussian (since affine transformations of Gaussian random variables are Gaussian). This means that the joint distribution of Z and W is also Gaussian, with mean

[4]Simplices are special cases of convex polytopes

$$\widetilde{\mu} = [\mu^T (A\mu)^T]^T \tag{3.5}$$

and covariance

$$\widetilde{\Sigma} = \begin{bmatrix} \Sigma & \Sigma^T A^T \\ A\Sigma & A\Sigma A^T \end{bmatrix} \tag{3.6}$$

Then, the conditional distribution $p(Z|W = b) = N(\mu_c, \Sigma_c)$ [39], where

$$\mu_c = \mu + \Sigma\Sigma^T A^T (A\Sigma A^T)^{-1}(\underline{b} - A\mu) \tag{3.7}$$

and

$$\Sigma_c = \Sigma - \Sigma^T A^T (A\Sigma A^T)^{-1} A\Sigma \tag{3.8}$$

Now, any sample $Z \sim \mathcal{N}(\mu_c, \Sigma_c)$[5], will obey the equality constraints in equation (3.4).

Sampling from the $n - m$ dimensional space spanned by the eigenvectors corresponding to nonzero eigenvalues ensures that the resulting samples always obey the equality constraints. If we represent the eigenvalue decomposition of Σ_c as

$$\Sigma_c = Q\Lambda Q^T \tag{3.9}$$

If the zero eigenvalues and corresponding eigenvectors are deleted, then the truncated eigenvalue decomposition may be denoted as

$$\Sigma_c = \widetilde{Q}\widetilde{\Lambda}\widetilde{Q}^T \tag{3.10}$$

This means that the problem can now be reparametrized in terms of an independent multivariate normal random variable $W \sim \mathcal{N}(0, I)$, where I denotes an identity

[5] Σ_c is a rank-deficient covariance matrix, with rank $= n - \widetilde{m}$, where $\widetilde{m}$ is the number of independent equality constraints ($\leq m$) in equation (3.4)

matrix, using the linear relation

$$Z = MW + \mu_c \tag{3.11}$$

where

$$M = \widetilde{Q}\widetilde{\Lambda}^{1/2} \tag{3.12}$$

Note that W is an $(n-m)$-dimensional random vector. The inequality constraints can be rewritten by substituting the expression in (3.11) in equation (3.4) as

$$Mw + C\mu_c \leq d \tag{3.13}$$

where w denotes a sample of W. Now, the original problem reduces to one of sampling from a truncated $(n-m)$-dimensional independent multivariate Gaussian random variable, truncated to a convex polytope defined by equation (3.13).

In the following sections, 3 types of samplers Markov chain Monte Carlo samplers will be discussed.

3.3.3 Gibbs sampler

The Gibbs sampler [18] was introduced in MCMC literature by Gelfand and Smith [16] as a special case of the Metropolis-Hastings algorithm. The Gibbs algorithm for sampling from an arbitrary distribution $p(w)$ truncated to a set $\mathbb{S} \subset \mathbb{R}^n$ is presented below:

Gibbs sampler

Step 1: Choose a starting point $w^0 \in \mathbb{S}$, and set $i = 0$

Step 2: Update each component of w^i by sampling from the full conditional distributions along each coordinate, (i.e.)

$$w_j^{i+1} \sim p(w_j^i | w_1^{i+1}, \ldots, w_{j-1}^{i+1}, w_{j+1}^i, \ldots, w_n^i), \text{ for } j = 1 \ldots n$$

Step 3: Set $i = i + 1$, and go to Step 2

In general, it is hard to obtain conditional distributions truncated to the feasible set $\mathbb{S}$. However, when the set $\mathbb{S}$ corresponds to a simplex (or more generally, a polytope), closed form expressions for the conditional distributions are easily available [37].

A Gibbs type sampler can be implemented for this problem, since the conditional distribution for each component is the standard normal distribution $\mathcal{N}(0,1)$ truncated to a range. The lower and upper limits of this range can be computed using equation (3.13).

Suppose that at iteration k, we wish to generate a sample of the i-th component w^k, say w_i^k. The upper and lower limits of the conditional distribution range is given by the minimum and the maximum of the vector $w_{\neg i}^k$, which is calculated as

$$w_{\neg i}^k = d - C\mu_c - M_{\neg i}\widehat{w}_{\neg i}^k \tag{3.14}$$

where $M_{\neg i}$ denotes the matrix M with the i-th column removed, and $\widehat{w}_{\neg i}^k$ is the vector $[w_1^k, \ldots, w_{i-1}^k, w_{i+1}^{k-1}, \ldots, w_{n-m}^{k-1}]$. The modified Gibbs algorithm can be summarized as below:

Modified Gibbs sampler

Step 1: Choose a starting point w^0 that satisfies equation (3.13), and set $i = 0$

Step 2: Update each component of w^i by sampling from a truncated normal distribution along each coordinate, (i.e.)

$$w_j^{i+1} \sim \mathcal{N}(0,1), w_j^{i+1} \in [a_j^{i+1}, b_j^{i+1}]; \quad j = 1 \ldots n$$

where a_j^{i+1} and b_j^{i+1} are the minimum and maximum of $w_{\neg j}^{i+1}$, given by equation (3.14)

Step 3: Set $i = i + 1$, and go to Step 2

To complete this procedure, we need to sample from a series of 1-dimensional truncated normal random variables. There are several algorithms [11, 37] available in literature to solve this problem, and any one can be suitably applied to sample from the conditional distribution at each step.

3.3.4 Hit-and-run sampler

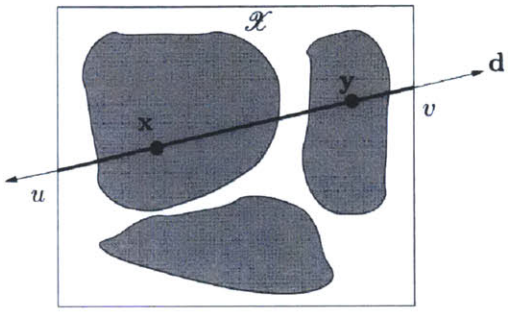

Figure 3-2: Visual Representation of the Hit-and-Run algorithm. Image Courtesy: [20]

The Hit-and-run sampler is an MCMC algorithm that was first proposed by Smith [42] as an alternative to techniques such as rejection sampling and transformation methods for generating uniformly distributed points in a bounded region. The Hit-and-run algorithm generates a Markov chain of samples by by first generating a direction d, and then sampling from the restriction of the bounded region on to the line passing through the current state, along the direction d. This idea is visually represented in figure (3-2).

Belisle *et al.* [6] extended the original Hit-and-run sampler to generate samples from arbitrary multivariate distributions restricted to bounded regions. Chen *et al.* [10] modified and generalized the algorithm to the form that is used in this study. The general Hit-and-run algorithm for generating samples from a random variable $\mathbf{W}$ with probability distribution $p(w)$ supported on $\mathbb{S}(\subset \mathbb{R}^n)$ is presented below:

Hit-and-run sampler

Step 1: Choose a starting point $w^0 \in \mathbb{S}$, and set i = 0

Step 2: Generate a direction d^i on the boundary of the unit sphere in $\mathbb{R}^n$ from a distribution ν

Step 3: Find the set $\widehat{S}^i(d^i, w^i) = \{\lambda \in \mathbb{R} | w^i + \lambda d^i \in S\}$

Step 4: Generate a signed distance $\lambda^i \sim g(\lambda | d^i, w^i)$, where $\lambda \in \widehat{S}^i$

Step 5: Generate $u^i \sim$ Unif $[0, 1]$

Step 6: Set $y = w^i + \lambda^i d^i$, and set $w^{i+1} = \begin{cases} y, & \text{if } u \leq \min\{1, a^i(w^i, y)\} \\ w^i, & \text{otherwise} \end{cases}$

Step 7: Set $i = i + 1$, and go to Step 2

where a^i denotes an acceptance probability. Note that this sampler works even if the function $p(w)$ is only known up to a multiple of a density function[6]. To complete the algorithm, we have to specify the direction distribution ν from Step 2, the density g in Step 4, and the acceptance probability a^i in Step 6. Belisle *et al.* [6] prove that if ν is chosen to be the uniform distribution on the n-dimensional unit hypersphere [7], and $g(\lambda|d, w)$ is the restriction of $p(w)$ on the set S^i, then a^i always equal to 1.

In general, restrictions of arbitrary probability distribution on line sets (such as S^i) are not easy to compute.However, closed form expressions exist for the multivariate uniform and normal distributions.

In particular, we are interested in sampling from the restriction of a truncated normal distribution on to a line along d. The expression for the restriction of a truncated normal density on a line passing through a point x along a direction d is derived in appendix B.

[6]This means that $\int_{\mathbb{S}} p(w_1, \ldots, w_n) dw_1 \ldots dw_n \neq 1$

[7]We can generate uniform directions on the surface of the n-dimensional hypersphere by first generating a sample from an n-dimensional multivariate i.i.d Gaussian, and setting the direction d^i to the normalized sample. Mathematically, $\hat{z} \sim \mathcal{N}(0_{n\times 1}, \mathbb{I}_{n\times n})$; $d^i = \hat{z}/||\hat{z}|| \Rightarrow d^i$ is uniformly distributed on the unit sphere (0 and $\mathbb{I}$ denote a vector of zeros and an identity matrix respectively)

As a final note on hit-and-run sampling algorithms, it is worth observing that if the direction distribution ν is restricted to be an equally weighted discrete probability distribution along n directions, where the n directions are the coordinate directions, and the density of λ^i, is the restriction of the density p on the set S^i, then this algorithm becomes the random scan Gibbs sampling algorithm [21].

As in the Gibbs sampling case, consider the problem of sampling from the standard multivariate normal distribution in $(n-m)$ dimensions truncated to the polytope defined by equation (3.13). If we denote the feasible set as $\mathbb{S}$, the hit-and-run algorithm to sample from this density is summarized below:

Modified hit-and-run sampler

Step 1: Choose a starting point $w^0 \in \mathbb{S}$, and set i = 0

Step 2: Generate a direction d^i by first generating $z^i \sim \mathcal{N}(0, \mathbb{I})$, then setting $d^i = z^i / \|z^i\|$

Step 3: Find the set $\widehat{S}^i(d^i, x^i) = [a^i, b^i]$, where a^i and b^i are the largest and smallest values of λ, such that $w^i + \lambda d^i$ satisfies equation (3.13)

Step 4: Generate a signed distance λ^i from the truncated normal distribution $\mathcal{N}(\hat{\mu}^i, (\hat{\sigma}^i)^2)$, $\lambda^i \in [a^i, b^i]$, where $\hat{\mu}^i$ and $\hat{\sigma}^i$ are given by [8]

$$\hat{\mu} = \frac{-(w^i)^T d^i}{(d^i)^T d^i} = -(w^i)^T d^i$$
$$\hat{\sigma}^2 = \frac{1}{(d^i)^T d^i} = 1$$

Step 5: Generate $u^i \sim$ Unif $[0, 1]$

Step 6: Set $y = w^i + \lambda^i d^i$, and set $w^{i+1} = \begin{cases} y, & \text{if } u \leq \min\{1, a^i(y|w^i)\} \\ w^i, & \text{otherwise} \end{cases}$

Step 7: Set $i = i + 1$, and go to Step 2

[8] For a proof, the reader is referred to appendix B

3.3.5 Directional independence sampler

The Gibbs sampling and Hit-and-run algorithms fall under the class of random walk samplers, where the next sample depends on the value of the current sample. While these type of samplers are convenient, they often result in samples that are increasingly correlated as the dimensionality of the underlying distribution rises [30].

Independence samplers are a class of Metropolis-Hastings algorithms, where the sample that is proposed does not depend on the value of the previous sample. These types of samplers are advantageous, since a well designed independence sampler should scale better than the Gibbs or Hit-and-run algorithm with an increase in the dimension of the distribution. The convex nature of the simplex constraint combined with the unimodality of the normal distribution make the problem particularly amenable to the design of an independence sampler. In this section, we shall construct an independence sampler, which is a modification of the Directional metropolis algorithm proposed by Eidsvik *et al.* [14].

As before, consider the problem of sampling from a random variable W distributed with an $(n-m)$-dimensional multivariate standard normal density truncated to the polytope defined by

$$dMw + C\mu_c \leq d$$

where w is a sample of W. To generate samples from this distribution, consider an independence sampler of the form

$$w = w_{map} + \alpha u \ , \ \alpha \geq 0 \tag{3.15}$$

where w_{map} refers to the maximum a posteriori probability (MAP) estimate of the truncated normal distribution corresponding to W [9]. The p thus generated is a point on the line passing through w_{map}, along a direction u at a euclidean distance of α from w_{map} (Note that (Note that $\|u\| = 1$). By sampling from appropriate distributions, it is possible to generate every point in the polytope defined by equation (3.13) with

[9]This MAP estimate can be computed using a simple constrained optimization routine

a unique combination of u and α. A general independence sampler [24] is outlined below:

Metropolized independence sampler

Step 1: Choose a starting value w^0 that satisfies equation (3.13), and set $i = 0$

Step 2: Draw $\widehat{w}^{\,i+1} \sim q(\widehat{w}^{\,i+1})$

Step 3: Draw $c^i \sim$ Unif $[0, 1]$

Step 4: Set

$$w^{i+1} = \begin{cases} \widehat{w}^{\,i+1} \text{ , if } c^i \leq min\{1, r(w^i, \widehat{w}^{\,i+1})\} \\ w^i \text{ , otherwise} \end{cases}$$

Step 4: Set $i = i + 1$, and go to Step 2

Here, $q(w)$ is called a proposal density, $r(\cdot, \cdot)$ denotes the acceptance ratio at iteration t, and Unif $[0, 1]$ represents the uniform distribution over $[0, 1]$. Let us denote the underlying standard truncated normal distribution of the random variable W as $\pi(w)$.

The Metropolis algorithm is completely specified once the proposal density and the acceptance ratio are fixed. The acceptance ratio is a quantity that ensures that the samples that are generated from the Metropolis algorithm represent samples from the $\pi(w)$, and is fixed by the detailed balance condition. The proposal density, on the other hand, is a distribution whose choice determines the quality of the samples that are generated. In general, the closer the proposal density $q(w)$ to the actual density $\pi(w)$, the better the quality of samples generated. First, we shall decide on the proposal mechanism for the independence sampler. Once the proposal distribution is fixed, the acceptance ratio can be derived by enforcing the detailed balance condition.

Any sample w is generated by first choosing a particular direction u, and then a distance α, such that the w calculated from equation (3.15) remains in the polytope defined by equation (3.13). As a modeling choice, we model the direction using a

projected normal distribution, and the distance α using a Gaussian random variable. The 2 proposal distributions are discussed in the following subsections.

Directional proposal distribution

The direction u is generated from an $(n-m)$-variate projected normal distribution [25] which is parametrized by a mean vector (μ_d) and a covariance matrix (Σ_d). The sample u from the projected normal distribution is generated by first sampling $x \sim \mathcal{N}(\mu_d, \Sigma_d)$, then setting $u = x/\|x\|$. In this case, the sample u lies on the $(n-m)$ dimensional unit sphere. For more information about the projected normal distribution and directional statistics, the reader is referred to appendix C.

Pukkila et al [33] derived an analytical expression for the density function of the projected normal distribution with respect to the surface element $d\omega_{n-m}$ on the $(n-m)$-dimensional unit sphere, as

$$p(u|\mu_d, \Sigma_d) = |2\pi\Sigma_d|^{1/2} Q_3^{(m-n)/2} I_{n-m}(Q_2 Q_3^{-1/2}) exp\left[-2^{-1}(Q_1 - Q_2^2/Q_3)\right] \quad (3.16)$$

where $Q_1 = \mu_d^T \Sigma_d^{-1} \mu_d$, $Q_2 = \mu_d^T \Sigma_d^{-1} u$ and $Q_3 = u^T \Sigma_d^{-1} u$.

$I_{n-m}(\cdot)$ is an integral that can be evaluated using a recursive relation, which is listed in appendix C. This distribution reduces to the uniform distribution on the unit sphere if μ_d is chosen to be zero and Σ_d is chosen to be the identity matrix.

For the sake of convenience, the covariance matrix Σ_d is chosen to be an identity matrix. The vector μ_d acts as a concentration parameter (refer appendix C) that controls the clustering of the samples from the projected normal distribution. This provides a useful tuning parameter for the direction proposal, since it is reasonable to expect the directions of samples to cluster based on the location of the MAP point within the feasible region. For example, if the MAP point were to lie at the barycenter at the polytope, the directions to that originate from the MAP can be expected to be more or less uniform. On the other hand, if the MAP point were to lie close to an edge or a vertex, most of the samples would have directions towards the interior of the polytope.

μ_d can be fixed either using a heuristic (such has a scaling of the line joining the MAP point and the barycenter of the polytope) or using some form of past information. In this case, we shall generate some Gibbs samples (say, p^i_{gibbs}, $i = 1, \ldots, N$) of the original distribution $\pi(w)$ truncated on the polytope, and calculate samples of the directional data (u^i_{gibbs}, $i = 1, \ldots, N$) using the relation

$$u^i_{gibbs} = \frac{p^i_{gibbs} - p_{map}}{\|p^i_{gibbs} - p_{map}\|}, \; i = 1, \ldots, N \tag{3.17}$$

The Gibbs samples can be assumed to realizations of a projected normal distribution parametrized μ_d, and an identity covariance matrix. Now, the problem becomes one of estimation of μ_d.

To compute μ_d, first calculate the mean resultant length (ρ) and mean resultant direction (t) of u^i_{gibbs}, $i = 1, \ldots, N$. Then, if we set $\mu_d = \gamma t$, Presnell *et al.* [31] proved that γ and ρ are related as

$$\rho = \frac{\Gamma\left(\frac{d+1}{2}\right)}{\sqrt{2}\Gamma\left(\frac{d+2}{2}\right)} \gamma \overline{M}\left(\frac{1}{2}, \frac{d}{2} + 1, -\frac{\gamma^2}{2}\right)$$

where $\overline{M}(\cdot, \cdot, \cdot)$ represents a confluent hypergeometric function, called Kummer's function [2]. Once μ_d is calculated, the proposal distribution for the directions is completely specified.

Distance distribution

Once the direction u is sampled, a point along u is generating by simulating $\alpha \geq 0$ from a proposal distribution. If p_{map} is assumed to be zero, then every new sample p is generated as $p = \alpha u$.

This is equivalent to first picking a direction, then picking a point along that direction. Since $\|u\| = 1$, and $\alpha \geq 0$, it implies that $\alpha = \|p\|$. If p was simulated from a standard normal distribution, then α will be distributed like a chi random variable [7] with $n - m$ degrees of freedom and mean

$$\mu_\alpha = \sqrt{2}\,\frac{\Gamma((n-m+1)/2)}{\Gamma((n-m)/2)}$$

where $\Gamma(\cdot)$ denotes the Gamma function. In other words, this means that most of the samples from a high-dimensional Gaussian distribution will lie on a shell of radius $\mu_\alpha \approx \sqrt{n-m}$. This phenomenon is sometimes referred to in literature as concentration of the Gaussian measure [15][10].

This means that a good proposal for α is a Gaussian that is centered at μ_α, with unit variance. Note that the variance of the chi distribution ($\sigma_\alpha^2 = n - m - \mu_\alpha^2$) can be used as a good guess for the variance of the Gaussian proposal. However, for the sake of convenience, we shall use the unit variance proposal. While the lower bound for α is zero, the upper bound b is the largest positive real number that satisfies

$$b(Mu) \leq d - Mp_{map} - C\mu_c$$

This implies that the proposal distribution for alpha is a normal distribution $\mathcal{N}(\mu_\alpha, 1)$, truncated to the closed interval $[0, b]$. The probability density function at any point $\alpha \in [0, b]$ for the distance proposal is

$$q(\alpha|u) = \frac{\phi(\alpha - \mu_\alpha)}{\Phi(b - \mu_\alpha) - \Phi(\mu_\alpha)} \tag{3.18}$$

where $\phi(\cdot)$ and $\Phi(\cdot)$ represent the probability density and cumulative density functions of the standard normal density respectively.

Now that the proposal distributions have been fully specified, the acceptance ratio can be derived by writing the detailed balance condition. To avoid cumbersome notation, let x denote the current sample (w^i), and y denote the sample at the next iteration (w^{i+1}). For the Metropolis algorithm to work, the stationary distribution of the Markov chain that is generated should be the density that is being sampled[11].

[10]For a good introduction to the phenomenon of measure concentration, the reader is referred to [5]

[11]Another important property, called reversibility, can be easily verified to hold for the proposed algorithm [36]

The stationarity condition can be expressed as

$$\pi(y) = \int \pi(x)A(x,y)dx \tag{3.19}$$

Here, $A(x,y)$ denotes the actual transition function from x to y, which is not necessarily the proposal distribution [24]. In fact, for the Metropolis algorithm, the actual transition function can be written as

$$A(x,y) = q(y)\min\{1, r(x,y)\}$$

where $q(\cdot)$ is the independent proposal density and $r(\cdot,\cdot)$ is the acceptance ratio. Since the samples themselves are described in terms of u's and α's, equation (3.19) can be re-written as

$$\pi(\alpha_y, u_y) = \int \pi(\alpha_x, u_x)A([\alpha_x, u_x], [\alpha_y, u_y])\alpha_x^{n-m}d\alpha_x d\omega_x \tag{3.20}$$

where $d\omega_x$ is an infinitesimal surface element on the $n-m$ dimensional unit sphere, and

$$\begin{aligned} x &= p_{map} + \alpha_x u_x \\ \text{and} \quad y &= p_{map} + \alpha_y u_y \end{aligned}$$

where the directions d_x and d_y is simulated from a projected normal distribution parametrized by μ_d. The distance α_x is a sample from a normal distribution with mean μ_α^x and unit variance, truncated on the interval $[0, b_x]$, while α_y is a sample from a normal distribution with mean μ_α^y and unit variance, truncated on the interval $[0, b_y]$.

The detailed balance [24] condition automatically ensures that equation (3.20) is always satisfied. The detailed balance in this case can be written as

$$\pi(\alpha_y, u_y)A([\alpha_y, u_y], [\alpha_x, u_x])\alpha_y^{n-m} = \pi(\alpha_x, u_x)A([\alpha_x, u_x], [\alpha_y, u_y])\alpha_x^{n-m} \tag{3.21}$$

The detailed balance condition in turn can be enforced by choosing $r([\alpha_x, u_x], [\alpha_y, u_y])$ [24] as

$$r([\alpha_x, u_x], [\alpha_y, u_y]) = \frac{\pi(\alpha_y, u_y) q(\alpha_x, u_x) \alpha_y^{n-m}}{\pi(\alpha_x, u_x) q(\alpha_y, u_y) \alpha_x^{n-m}} \tag{3.22}$$

where $r(\cdot, \cdot)$ denotes the acceptance ratio in the Metropolis algorithm. The joint distribution $q(\cdot)$ is rewritten using a conditional distribution, as

$$r([\alpha_x, u_x], [\alpha_y, u_y]) = \frac{\pi(\alpha_y, u_y) q(\alpha_x|u_x) q(u_x) \alpha_y^{n-m}}{\pi(\alpha_x, u_x) q(\alpha_y|u_y) q(u_y) \alpha_x^{n-m}} \tag{3.23}$$

Here, $q(u_x)$ is the proposal (projected normal) density for the direction, and $q(\alpha_x|u_x)$ is the distance proposal (truncated Gaussian) density from the previous subsections. The expressions for these densities can be evaluated and substituted into equation (3.23), to get

$$r(x, y) = r_1.r_2.r_3.r_4 \tag{3.24}$$

where

$$r_1 = \frac{\exp(-y^T y/2)}{\exp(-x^T x/2)}$$

$$r_2 = \left(\frac{R_3^{-(n-m)/2} I_{(n-m)}(R_2 R_3^{-1/2}) exp\left[R_2^2/(2R_3)\right]}{Q_3^{-(n-m)/2} I_{(n-m)}(Q_2 Q_3^{-1/2}) exp\left[Q_2^2/(2Q_3)\right]} \right)$$

$$r_3 = \left(\left[\frac{\Phi(b_y - \hat{\mu}_y) - \Phi(-\hat{\mu}_y)}{\Phi(b_x - \hat{\mu}_x) - \Phi(-\hat{\mu}_x)} \right] \left[\frac{\phi(\alpha_x - \hat{\mu}_x)}{\phi(\alpha_y - \hat{\mu}_y)} \right] \right)$$

$$r_4 = \left[\frac{\alpha_y}{\alpha_x} \right]^{n-m-1}$$

and

$$Q_1 = \mu_d^T \mu_d, \; Q_2 = \mu_d^T u_x \text{ and } Q_3 = 1$$

$$R_1 = \mu_d^T \mu_d, \; R_2 = \mu_d^T u_y \text{ and } R_3 = 1$$

With this proposal distribution and acceptance ratio the modified Metropolized independence sampler is presented below:

Directional independence sampler

Step 1: Calculate the MAP point of the truncated normal distribution, w_{map}, using an optimization routine

Step 2: Choose a starting value w^0 that satisfies equation (3.13), by simulating a value α^0 and d^0 from the proposal densities and set $i = 0$

Step 3: Draw $\widehat{d}^{\,i+1} \sim q(\widehat{d}^{\,i+1})$, and $\widehat{\alpha}^{\,i+1} \sim q(\widehat{\alpha}^{\,i+1}|\widehat{d}^{\,i+1})$,

$\widehat{w}^{\,i+1} = w_{map} + \widehat{\alpha}^{\,i+1}\widehat{d}^{\,i+1}$

Step 4: Draw $c^i \sim$ Unif $[0, 1]$

Step 5: Set

$$w^{i+1} = \begin{cases} \widehat{w}^{\,i+1} \text{ , if } c^i \leq min\{1, r(w^i, \widehat{w}^{\,i+1})\} \\ w^i \text{ , otherwise} \end{cases}$$

where $r(\cdot, \cdot)$ is the acceptance ratio defined in equation (3.24).

Step 6: Set $i = i + 1$, go to Step 3

3.4 Summary

In this section, we have outlined some of the computational issues that can arise when analyzing the posterior distribution of the Bayesian feed reconstruction model. To overcome these challenges, we decided to adopt a Monte Carlo approach. Since the Monte Carlo approach requires an efficient generation of samples from the posterior distribution, we reviewed 3 sampling algorithms. In the next chapter, we shall implement the samplers and compare their relative performance in a set of test cases.

Chapter 4

Results

In the following sections, we shall compare the performance of the different samplers that were discussed in the previous chapter. First, we shall implement a few low dimensional test cases, to test the accuracy of the various samplers. Once we study the performance of these algorithms in the lower dimensional test cases, we shall apply them to sample a posterior (high dimensional) distribution from a feed reconstruction example on a real life data set.

4.1 Low dimensional examples

We construct three different three dimensional Gaussian distributions that are truncated on the standard 2-simplex (say, $\mathbb{S}$), where $\mathbb{S} = \{x \in \mathbb{R}^3 | x_1 + x_2 + x_3 = 1, x_i \geq 0, i = 1, 2, 3\}$. In the three cases, we shall vary the mean (μ) and covariance matrix (Σ) parameters associated with each Gaussian distribution.

$$\text{Case 1: } \mu = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \Sigma = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\text{Case 2: } \mu = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \Sigma = \begin{bmatrix} 0.1 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 0.1 \end{bmatrix}$$

$$\text{Case 3: } \mu = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \Sigma = \begin{bmatrix} 25 & -25 & -50 \\ -25 & 25 & -50 \\ -50 & -50 & 100 \end{bmatrix}$$

In all these three cases, the normal distribution is truncated to the set $\mathbb{S}$. To visually verify the working , we run the algorithm on each of the test cases, and produce a scatter plot of the samples generated on the simplex, along with the trace of the samples in each coordinate, and the autocorrelation of the samples generated along each dimension. The scatter plot and the sample trace allow us to visually inspect if the algorithm generates chains that mixing well [8]. Autocorrelation along each sample dimension numerically tells us how well the chain is mixing. Low autocorrelation implies that the samples that are being generated are nearly independent, so any Monte Carlo estimate would require a relatively low number of samples to reach a desired accuracy. On the other hand, large autocorrelation in the samples implies that one would require a significantly larger number of samples to reach the same accuracy [43].

To numerically verify the samplers, we compute the mean of the truncated normal distribution using a 2-D quadrature routine (such as `integral2` in MATLAB) on the simplex, and then compare it with the mean of 10^4 samples generated by the algorithm. By visually inspecting the samples and comparing the first moment of the truncated distribution along with the sample mean, we can decide if the sampler functions efficiently and accurately.

4.1.1 Gibbs sampler

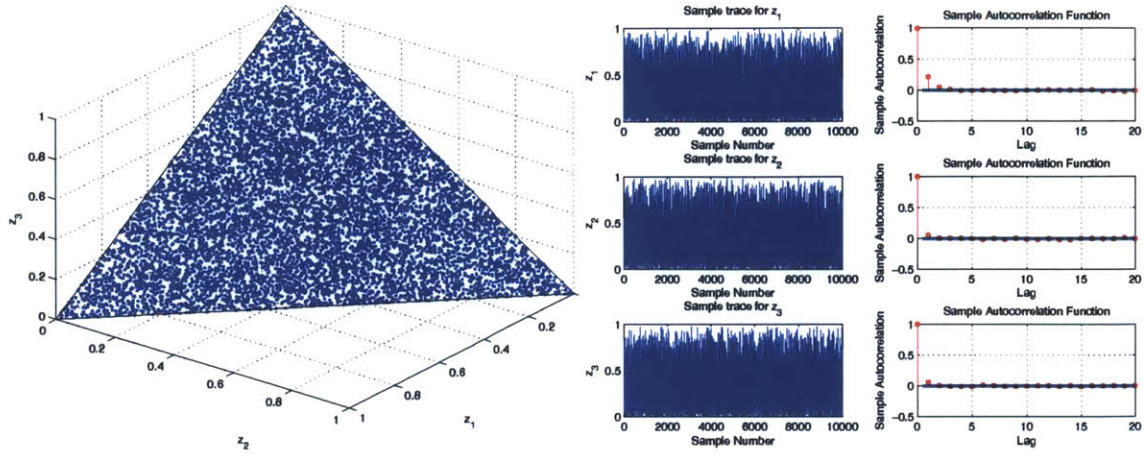

Figure 4-1: Case 1: Scatter plot of samples generated by the Gibbs sampling algorithm, along with the sample trace and autocorrelation plots

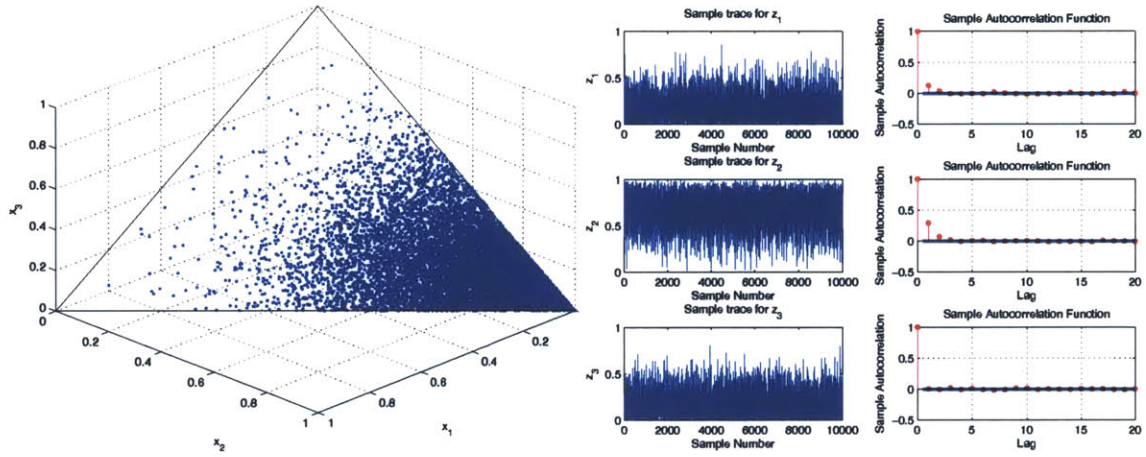

Figure 4-2: Case 2: Scatter plot of samples generated by the Gibbs sampling algorithm, along with the sample trace and autocorrelation plots

The Gibbs sampler performs quite well in low-dimensions, producing samples that are almost uncorrelated. However, it is worth noting that 1 iteration of the Gibbs sampler actually samples 3 independent truncated Gaussian distributions. In higher dimensions, one Gibbs sample requires n conditional samples, which may become prohibitive in high dimensions.

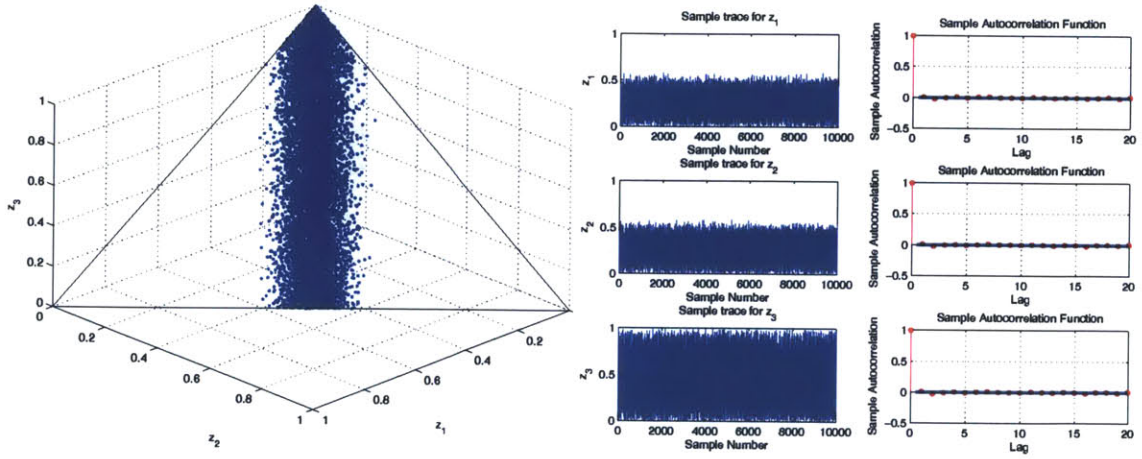

Figure 4-3: Case 3: Scatter plot of samples generated by the Gibbs sampling algorithm, along with the sample trace and autocorrelation plots

Case No.	Coordinate	Computed Mean	Sample Mean
1	z_1	0.3333	0.3275
	z_2	0.3333	0.3378
	z_3	0.3333	0.3347
2	z_1	0.1544	0.1533
	z_2	0.6912	0.6929
	z_3	0.1544	0.1538
3	x_1	0.2620	0.2627
	x_2	0.2620	0.2630
	x_3	0.4759	0.4743

Table 4.1: Comparing the mean of the truncated normal distribution calculated using 2-D quadrature and the sample mean for the Gibbs sampler

4.1.2 Hit-and-run sampler

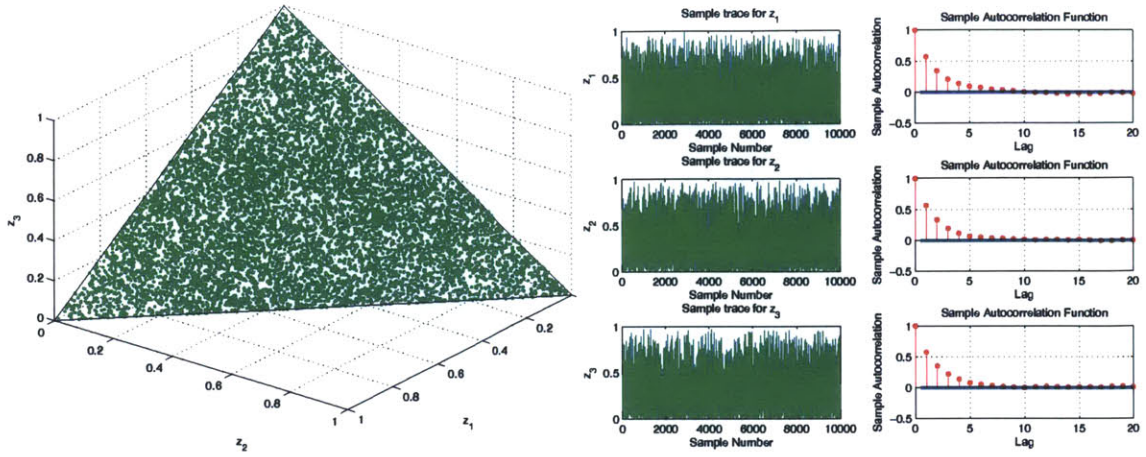

Figure 4-4: Case 1: Scatter plot of samples generated by the hit-and-run sampling algorithm, along with the sample trace and autocorrelation plots

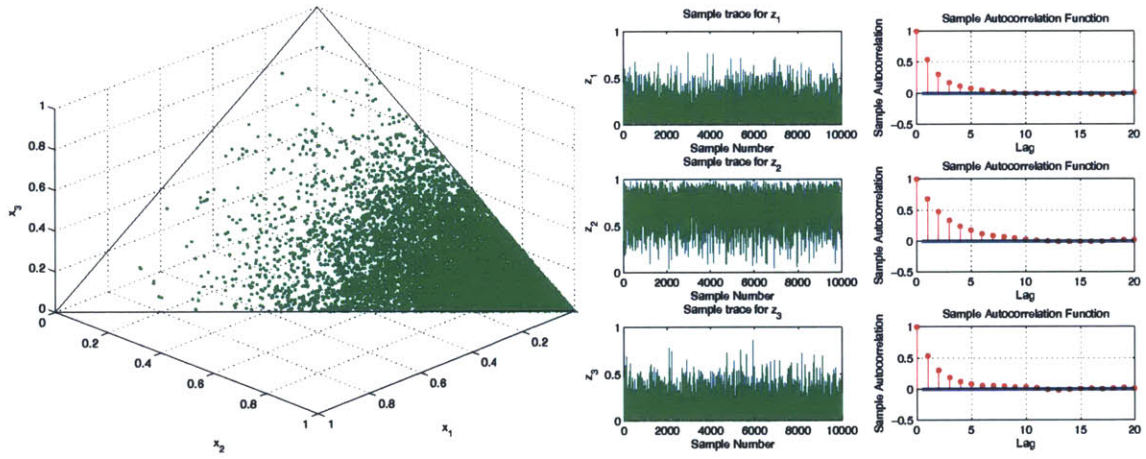

Figure 4-5: Case 2: Scatter plot of samples generated by the hit-and-run sampling algorithm, along with the sample trace and autocorrelation plots

The hit-and-run sampler also performs well in low dimensions. The samples from the hit-and-run sampler appear to be more correlated than the corresponding Gibbs samples for the same distribution. While this might indicate that the hit-and-run algorithm is not as effective as the Gibbs algorithm, it should be remembered that the hit-and-run samples are cheaper to generate (sample from one conditional distribution per sample generated) than the Gibbs sample (samples from two conditional distributions per sample generated).

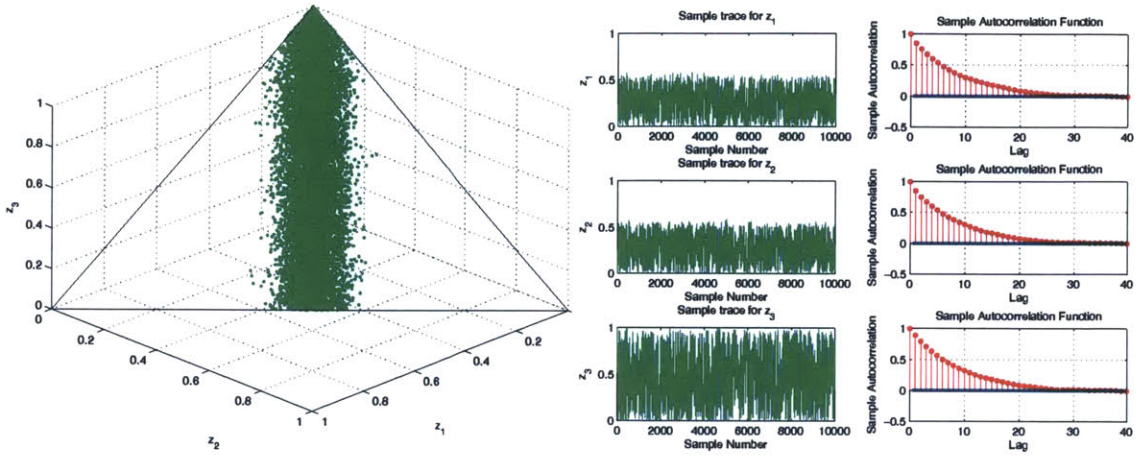

Figure 4-6: Case 3: Scatter plot of samples generated by the hit-and-run sampling algorithm, along with the sample trace and autocorrelation plots

Case No.	Coordinate	Computed Mean	Sample Mean
1	z_1	0.3333	0.3317
	z_2	0.3333	0.3250
	z_3	0.3333	0.3433
2	z_1	0.1544	0.1528
	z_2	0.6912	0.6967
	z_3	0.1544	0.1505
3	x_1	0.2620	0.2687
	x_2	0.2620	0.2682
	x_3	0.4759	0.4631

Table 4.2: Comparing the mean of the truncated normal distribution calculated using 2-D quadrature and the sample mean for the hit-and-run sampler

4.1.3 Directional independence sampler

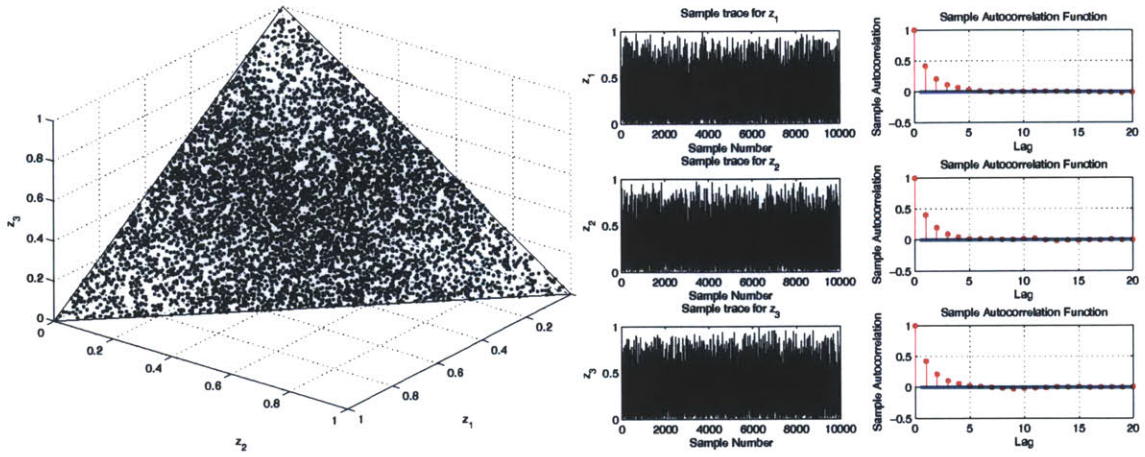

Figure 4-7: Case 1: Scatter plot of samples generated by the directional independence sampler, along with the sample trace and autocorrelation plots

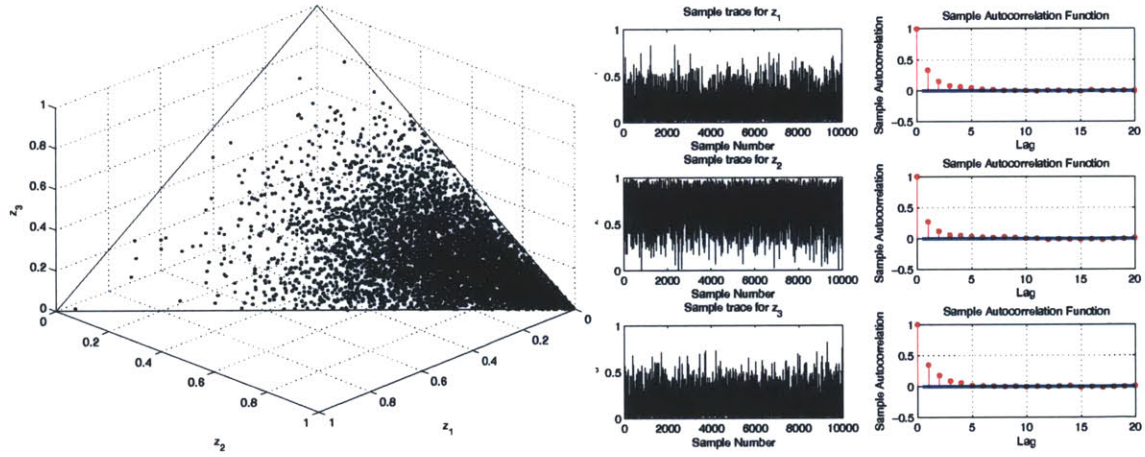

Figure 4-8: Case 2: Scatter plot of samples generated by the directional independence sampler, along with the sample trace and autocorrelation plots

The directional independence sampler performs reasonably well in all three test cases. In the first two cases, the samples appear to be less correlated than the corresponding hit-and-run samples for the same target distribution. In the third example, the directional sampler performs significantly worse (in terms of autocorrelation). It is worth noting that the acceptance ratio [1] in the first two cases is around 60 percent, while the acceptance ratio in the third case is around 10 percent. This indicates that

[1] Acceptance ratio = (number of samples accepted in accept-reject step)/(total number of samples)

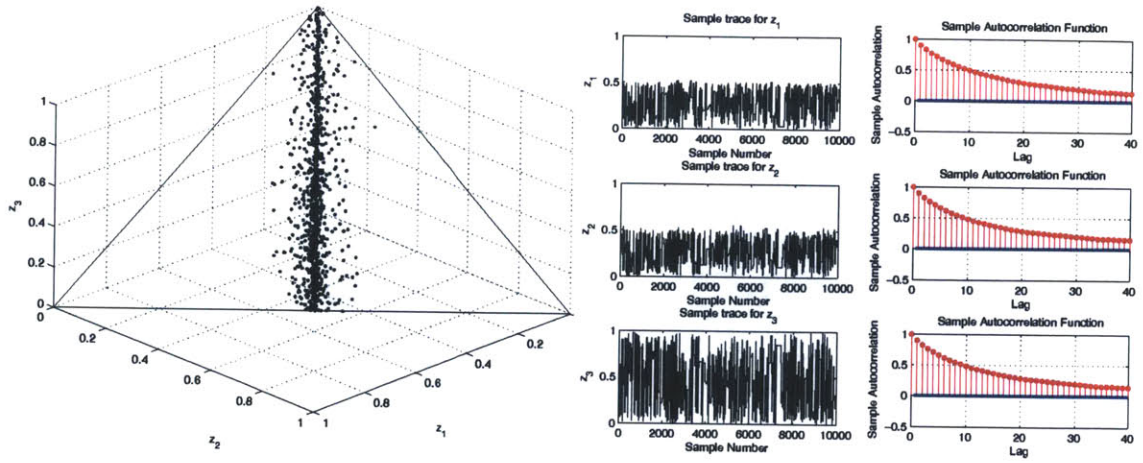

Figure 4-9: Case 3: Scatter plot of samples generated by the directional independence sampler, along with the sample trace and autocorrelation plots

Case No.	Coordinate	Computed Mean	Sample Mean
1	z_1	0.3333	0.3386
	z_2	0.3333	0.3276
	z_3	0.3333	0.3339
2	z_1	0.1544	0.1540
	z_2	0.6912	0.6921
	z_3	0.1544	0.1538
3	x_1	0.2620	0.2475
	x_2	0.2620	0.2548
	x_3	0.4759	0.4977

Table 4.3: Comparing the mean of the truncated normal distribution calculated using 2-D quadrature and the sample mean for the directional independence sampler

the directional sampling algorithm might perform poorly in the presence of strong correlations in the posterior. The low acceptance ratio manifests itself in the sample trace and the increased autocorrelation.

4.2 Feed reconstruction example

To test our Bayesian model, we applied this algorithm to a feed reconstruction problem on a real dataset from a plant. Detailed measurements of some sample feedstock in terms of pseudo-compounds were collected in a laboratory setting. The objective of the feed reconstruction is to use these detailed measurements along with some bulk concentrations and rapid distillation analysis of some unknown feedstock in the refinery to infer the detailed composition of the unknown feedstock in terms of the pseudo-compounds.

The Bayesian model is applied to this problem and the posterior truncated Gaussian mean and covariance parameters are derived.To explore this high-dimensional posterior distribution, we attempt to use the three sampling algorithms that were proposed in the previous section. The presence of high posterior correlations coupled with the large dimensionality of the problem makes this a difficult distribution to analyze.

Among the three algorithms, only the Gibbs sampler manages to scale with the number of dimensions and produce reasonable results. The hit-and-run sampler and the directional independence sampler produce poorly mixing chains even with an extremely large number of samples. This tells us that while the hit-and-run and directional independence sampler produce reasonable results in low-dimensions, they do not scale well with the number of dimensions, and in the presence of high correlations (as in the feed reconstruction case), become ineffective. Thus, the Gibbs sampler is used to sample the posterior distribution of the Bayesian model.

Figure (TODO) shows the sample trace along 2 particular coordinates for the Gibbs samples generated from the truncated normal posterior distribution, and its corresponding autocorrelation. It can be observed that along some dimensions, the autocorrelation in the samples is very large. This is one of the main drawbacks of using the Gibbs sampler, and requires generating a large number ($\sim$ millions) of samples to overcome the heavy correlation.

Figure (4-11) shows the marginal densities of pseudocomponents that belong to

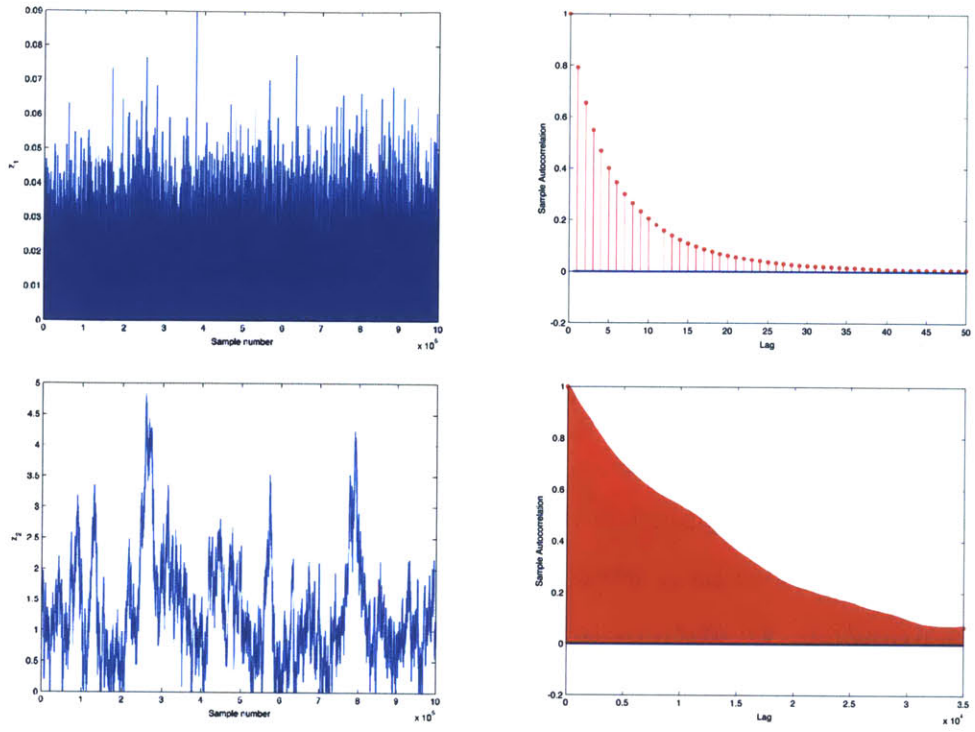

Figure 4-10: Sample trace and autocorrelations along selected coordinates in the Bayesian feed reconstruction example

2 different reacting families. Instead of a single "best" concentration, the Bayesian model produces a distribution over a range of possible concentrations. The width of these marginal densities provides an idea of the uncertainty associated with the concentration of the particular pseudo-compound.

The Bayesian model allows us to understand the impact of the measured bulk properties on the uncertainty of the concentrations of the pseudo-compounds. To illustrate this, consider the following example. In figure (4-12), the histogram on the left shows the marginal posterior densities of the pseudo-compounds in a particular nitrogen containing family. When the measured bulk nitrogen concentration is increased, the uncertainty in the concentrations of the nitrogen containing psuedo-compounds also increases, which seems reasonable.

Finally, the Bayesian framework provides a fine control on the on the degree-of-belief in the laboratory and bulk measurements. For example, in figure (4-13), the marginal densities on the left correspond to a scenario in which the variance associated

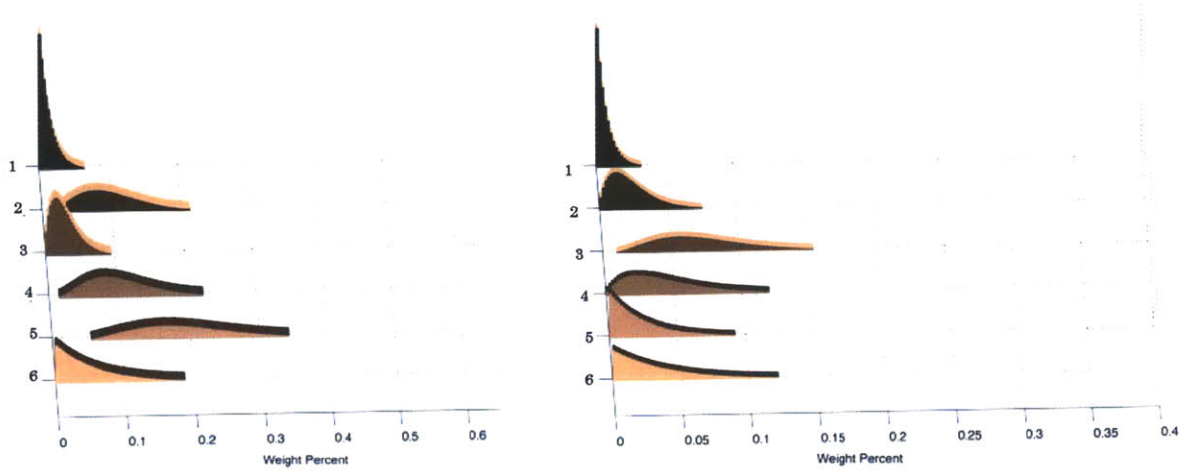

Figure 4-11: Marginal densities of samples generated from Bayesian posterior distribution for 2 different families of pseudo-compounds

with the bulk measurements is very low (corresponding to a high degree-of-belief in the bulk measurements). The marginal densities on the right corresponds to the opposite case, where the variance associated with the prior is very low (which implies a high belief in the laboratory information).

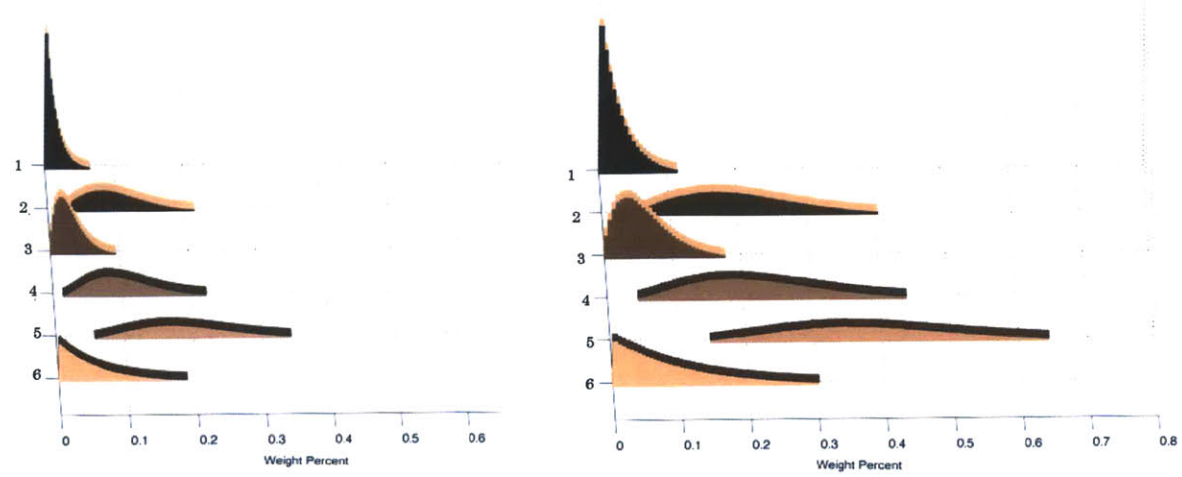

Figure 4-12: Marginal densities of concentrations of pseudo-compounds in a particular family for varying bulk concentrations

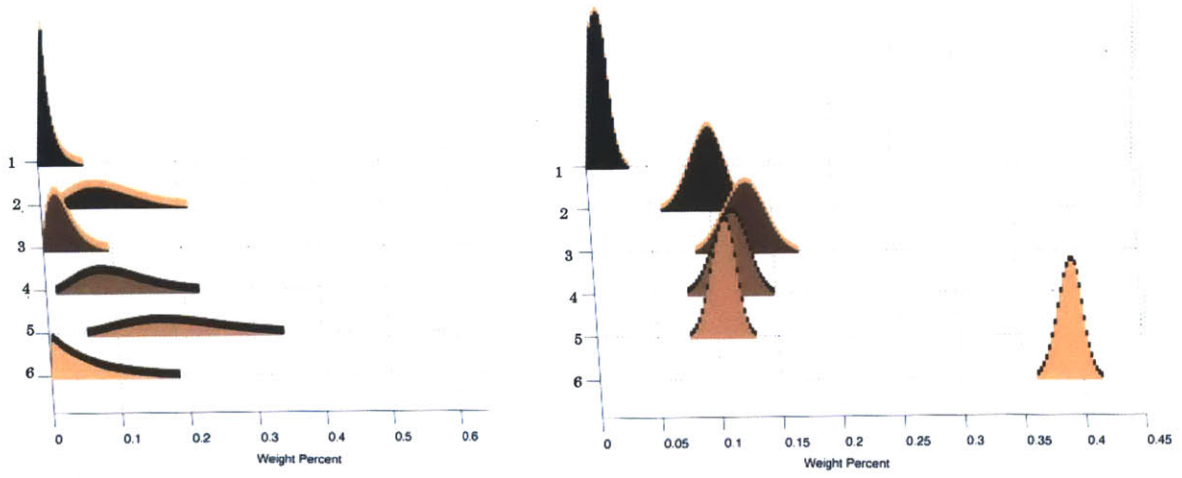

Figure 4-13: Comparing marginal densities of concentrations of pseudo-compounds in a particular family for varying degrees-of-belief in the bulk and laboratory measurements respectively

4.3 Summary

In this chapter, we tested the performance of the various samplers that were discussed in the previous chapter. While all the three samplers produce reasonable results in low dimensions, only the Gibbs sampler manages to work in high dimensions. Then, we implemented the Bayesian feed reconstruction algorithm for a real data set from a plant, and analyzed the resulting posterior distribution using the Gibbs sampler.

Chapter 5

Conclusions

In this thesis, we developed a Bayesian inference algorithm for the feed reconstruction problem. The Bayesian approach addresses some of the shortcomings of current feed reconstruction schemes in literature, by explicitly incorporating uncertainty into the reconstruction process.

On the computational front, we reviewed and implemented three different algorithms to sample from the truncated normal posterior distribution that results from the Bayesian reconstruction algorithm. In lower dimensions, all three algorithms work satisfactorily.In high dimensions, the hit-and-run and the directional independence sampler become ineffective for most applications. The Gibbs sampler produces samples from the posterior distribution, but these are expensive to compute (since each sample requires sampling n conditional distributions), and are highly correlated.

The Bayesian reconstruction algorithm coupled with the Gibbs sampler was applied to study a feed reconstruction example in a real plant scenario. The resulting model captured the uncertainty in both the bulk measurements and the prior information, while providing an intuitive way to tune the degree-of-belief in each measurement.

We believe that the flexibility of the Bayesian modeling technique, combined with the elegant handling of uncertainty, makes the Bayesian approach a novel and useful alternative to current feed reconstruction algorithms.

5.1 Future work

The following represents some directions for future study that we believe can be used to extend the work in this thesis

- While we have presented a complete Bayesian algorithm to reconstruct any feedstock in the refinery, we are still in the process of validating our approach. The ideal validation framework would require detailed laboratory analysis of a feedstock whose bulk measurements are available in the plant. Then, the detailed analysis from the lab can be compared with the results from the Bayesian model with the bulk measurements, and the quality of the reconstruction can be ascertained. However, the complicated and expensive nature of analytical chemistry prohibits the gathering of such validation data. In the absence of analytical data from the laboratory, a simulation tool such as Aspen HYSYS can be used to simulate the laboratory measurements (like distillation analysis and bulk concentrations) of some known mixture of pseudo-compounds. Then, the detailed analysis can be used in the feed reconstruction procedure, and the inferred concentration distributions can be compared with the known detailed composition.

- In the construction of the likelihood, we assumed that the bulk properties that were measured in the plant could be obtained as a linear function of the weight fractions of the psuedo-compounds. While this is true for a large class of bulk properties that are measured in the refinery, the procedure can still be extended to properties that are related to the weight fractions in a nonlinear fashion. The resulting posterior distribution will not be a truncated normal distribution, and any subsequent analysis will require the use of sophisticated sparse quadrature or sampling techniques.

- In the current feed reconstruction scheme, we assumed that the parameters of the gamma distribution that are estimated from laboratory data are constant. This assumption is not necessary, and can be relaxed by assuming

that the parameters are random. This idea is referred to as a fully Bayesian approach to modeling. The fully Bayesian feed reconstruction model is discussed in appendix A.

- The Gibbs sampling algorithm functions as a reasonable technique to generate samples from high dimensional posterior distributions. However, the algorithm scales very poorly with the number of dimensions, and becomes very slow for high dimensional problems, which are common in feed reconstruction settings. The Gibbs algorithm is very general, and does not leverage many properties of the simplex, such as linearity and convexity. We developed the directional independence sampler to take advantage of some of these properties, but it does not appear to scale well with the number of dimensions thanks to the phenomenon of measure concentration. We believe that the structure of the simplex combined with the Gaussian nature of the distribution allows for a good sampler, that can generate almost independent samples as the dimensionality of the problem rises. A fast sampling algorithm is important to the practical application of the Bayesian approach, both in the context of feed reconstruction and general chemical systems.

Appendix A

Fully Bayesian procedure

In the Bayesian feed reconstruction mode, the parameters of the gamma distribution were estimated using an MLE approach. An alternative approach would be to assume that the parameters of the gamma distribution themselves are random, with their probability distribution parametrized by hyperparameters. Then, instead of inferring just the weight percents, the parameters of the gamma distribution can also be inferred jointly. This type of model is sometimes called a fully Bayesian approach to modeling.

Let Ψ_i denote the vector of parameters of the gamma distribution corresponding to the i-th family. Then, using the same notation as chapter 2, the Bayes equation in this case can be rewritten as

$$p(Y, \psi|\mathcal{D}_p) \propto \mathcal{L}(\mathcal{D}_p|Y, \psi)p(Y|\psi)p(\psi) \tag{A.1}$$

To finally obtain obtain a posterior distribution on Y, the joint posterior distribution from equation (A.1) has to be marginalized over ψ:

$$p(Y|\mathcal{D}_p) \propto \int p(Y, \psi|\mathcal{D}_p)d\psi \tag{A.2}$$

The fully Bayesian version of feed reconstruction is the most general Bayesian framework to analyze the feed reconstruction problem, and our preceding discussion in the body of this thesis can be seen as a special case of this approach.

Appendix B

Restriction of a Gaussian to a line

Given a point $x \in \mathbb{S} \subset \mathbb{R}^n$, and a direction $d \in \mathbb{R}^n$ (note that $\|d\| = 1$), let us suppose we are interested in restricting the Gaussian $\mathcal{N}(\mu, \Sigma)$, truncated on the set $\mathbb{S}$ along the line $y = x + \lambda d$, where $\lambda \in \{\lambda \in \mathbb{R} | y \in \mathbb{S}\}$ [1].

$$
\begin{aligned}
p(x+\lambda d) &= |\Sigma|^{-0.5}(2\pi)^{-n/2} \exp(-(x+\lambda d-\mu)^T\Sigma^{-1}(x+\lambda d-\mu)/2) \\
&\propto \exp(-(x-\mu+\lambda d)^T\Sigma^{-1}(x-\mu+\lambda d)/2) \\
&\propto \exp((-(x-\mu)^T\Sigma^{-1}(x-\mu) - \lambda^2 d^T\Sigma^{-1}d + 2\lambda(x-\mu)^T\Sigma^{-1}d)/2) \\
&\propto \exp((-\lambda^2 d^T\Sigma^{-1}d + 2\lambda(x-\mu)^T\Sigma^{-1}d)/2) \quad \{\because (x-\mu) \text{ constant along } d\} \\
&\propto exp(-(\lambda-\hat{\mu})^2/(2\hat{\sigma}^2)) \qquad \{\text{Completing the square in } \lambda\}
\end{aligned}
$$

where

$$
\begin{aligned}
\hat{\mu} &= \frac{-(x-\mu)^T\Sigma^{-1}d}{d^T\Sigma^{-1}d} \\
\hat{\sigma}^2 &= \frac{1}{d^T\Sigma^{-1}d}
\end{aligned}
$$

This implies that $\lambda \sim \mathcal{N}(\hat{\mu}, \hat{\sigma}^2)$. Furthermore, if λ is truncated, such that $\lambda \in [a, b]$, we can evaluate $p(\lambda)$ by using a standard transformation. First, we write

$$\lambda = \hat{\mu} + \hat{\sigma} z$$

[1] This implies that λ might belong to an interval

where z is the standard normal random variable. Then, z is truncated to the set $[c, e] = [(a - \hat{\mu})/\hat{\sigma}, (b - \hat{\mu})/\hat{\sigma}]$. This makes simulation and density evaluation *much* cheaper. Since we are simulating from the truncated standard normal distribution, there are several fast algorithms and packages that can be employed. The density of λ truncated to $[a, b]$ can be computed as

$$p(\lambda|d) = \frac{(1/\hat{\sigma})\phi(z)}{\Phi(e) - \Phi(c)}$$

where ϕ and Φ denote the probability and cumulative distribution functions of the standard normal distribution. There are fast algorithms available to compute the probability and cumulative densities [26].

Appendix C

Projected Normal Distribution

Directional statistics is a branch of statistics that is used to understand and model random directional data. While traditional random variables are often modeled over $\mathbb{R}^n$, directional data are defined only over the unit sphere. This basic difference promotes a slightly different approach to understanding statistics over spheres. Popular measures of central tendency are still applicable to spherical data, but they are often handled with slight modifications.

One particular measure of interest is the mean. For example, let $x_1, x_2, \ldots, x_n$ represent n points on the unit sphere in p dimensions. Then, their sample mean is given by

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{C.1}$$

Then, we define the mean resultant length, $\rho = \|\bar{x}\| \in [0, 1)$ and the *mean resultant direction* $t = \bar{x}/\rho$.

Like the mean, we can also define different measures of central tendency to directional data, but it is out of the scope of this work. For a comprehensive treatment of directional statistics, the reader is referred to [25].

The projected normal distribution is a convenient distribution that can be used to model directional data. In $n - m$-dimensions, we may denote the unit sphere as $\mathbb{S}^{n-m-1}$, which is a subset of $\mathbb{R}^{n-m}$. If we have $x \sim \mathcal{N}(\mu, \Sigma) \in \mathbb{R}^{n-m}$, and define $u = x/\|x\|_2$, then u is said to be distributed as a projected $n - m$-variate

normal random variable, denoted as $u \sim PN_p(\mu, \Sigma)$ [25] (It should be noted that the parameters μ and Σ cannot be uniquely identified, since taking $a\mu$ and $a^2\Sigma$ does not alter the distribution of directions [31], but we can address this problem by requiring $\det(\Sigma) = 1$). Pukkila *et al.* [33] provide a closed form expression for the pdf of u with respect to the surface element on the p-dimensional unit sphere $d\omega_p$ as

$$p(U|\mu, \Sigma) = |2\pi\Sigma|^{1/2} Q_3^{-p/2} I_p(Q_2 Q_3^{-1/2}) exp\left[-2^{-1}(Q_1 - Q_2^2/Q_3)\right] \qquad \text{(C.2)}$$

where $Q_1 = \mu^T\Sigma^{-1}\mu$, $Q_2 = \mu^T\Sigma^{-1}y$, $Q_3 = y^T\Sigma^{-1}y$

and

$$I_p(\alpha) = \int_0^\infty r^{p-1} exp\left(-\frac{(r-\alpha)^2}{2}\right) dr$$

While this integral may be hard to evaluate, Pukkila *et al.* suggest using the recursive relation

$$I_p(\alpha) = (p-1)I_{p-2}(\alpha) + \alpha I_{p-1}(\alpha) \qquad p > 2$$

with the initial values

$$I_2(\alpha) = e^{-\alpha^2/2} + \alpha I_1(\alpha), \quad I_1(\alpha) = \sqrt{2\pi}\Phi(\alpha)$$

Let us study the effect of varying the parameters of the projected normal distribution. The scalar parameter $\|\mu\|$ serves as a measure of concentration on the unit sphere. To highlight this property of $\|\mu\|$, 4 different examples are considered in figure (C-1), with increasing values of $\|\mu\|$, while maintaining the same direction of μ.

Increasing the value of $\|\mu\|$ causes the samples generated from the projected normal distribution to cluster. If we denote the mean resultant length of the samples as ρ, then it can be empirically observed that ρ increases with an increase in μ. This suggests a relationship between $\|\mu\|$ and ρ, which was discovered by Presnell *et al.* [31], as

$$\rho = \frac{\Gamma\left(\frac{d+1}{2}\right)}{\sqrt{2}\Gamma\left(\frac{d+2}{2}\right)} \gamma M\left(\frac{1}{2}, \frac{d}{2} + 1, -\frac{\gamma^2}{2}\right) \qquad \text{(C.3)}$$

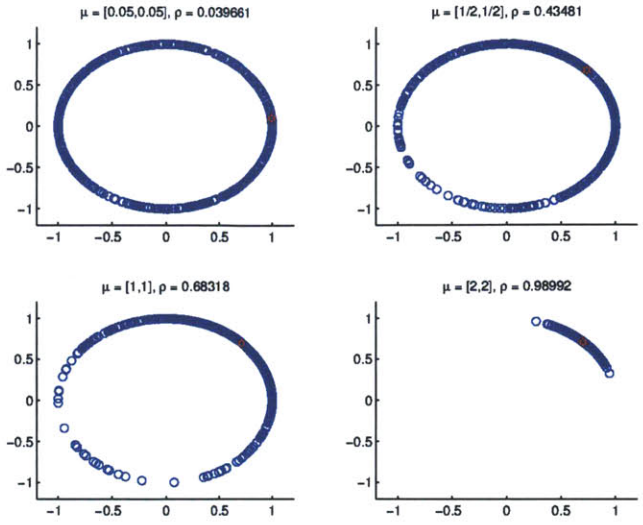

Figure C-1: Plots observed samples for different values of $\|\mu\|$, with the population mean direction highlighted in red

for the case when $\Sigma = I$, where Γ is the gamma function and and M is a confluent hypergeometric function. This concentration phenomenon is useful to model unimodal directional data.

Changing the covariance parameter Σ can produce a bimodal directional distribution, and simultaneously varying μ and Σ can produce many interesting results that can be used to capture complex behavior in directional data. However, this study is quite deep and lies outside the scope of this text. For a detailed introduction to the flexibility of the projected normal distribution to model directional data, the reader is referred to Wang *et al.* [47].

Bibliography

[1] CS294-13: Special Topics in Advanced Computer Graphics. http://inst.eecs.berkeley.edu/ cs294-13/fa09/lectures/scribe-lecture4.pdf. Accessed: 12/12/2012.

[2] Milton Abramowitz and Irene A Stegun. Handbook of mathematical function with formulas, graphs, and mathematical tables. *National Bureau of Standsards, Applied Mathematics Series*, 55, 1970.

[3] James H Albert and Siddhartha Chib. Bayesian analysis of binary and polychotomous response data. *Journal of the American statistical Association*, 88(422):669–679, 1993.

[4] Ioannis P Androulakis, Mark D Weisel, Chang S Hsu, Kuangnan Qian, Larry A Green, John T Farrell, and Kiyomi Nakakita. An integrated approach for creating model diesel fuels. *Energy & fuels*, 19(1):111–119, 2005.

[5] Alenxander Barvinok. Lecture notes on measure concentration. http://www.math.lsa.umich.edu/ barvinok/total710.pdf, 2005.

[6] C.J.P. Bélisle, H.E. Romeijn, and Robert L. Smith. Hit-and-run algorithms for generating multivariate distributions. *Mathematics of Operations Research*, 18(2):255–266, 1993.

[7] D.P. Bertsekas and J.N. Tsitsiklis. *Introduction to probability*, volume 1. Athena Scientific Nashua, NH, 2002.

[8] Steve Brooks, Andrew Gelman, Galin Jones, and Xiao-Li Meng. *Handbook of Markov Chain Monte Carlo*. Chapman and Hall/CRC, 2011.

[9] Darin M Campbell and Michael T Klein. Construction of a molecular representation of a complex feedstock by monte carlo and quadrature methods. *Applied Catalysis A: General*, 160(1):41–54, 1997.

[10] M.H. Chen and B.W. Schmeiser. General hit-and-run monte carlo sampling for evaluating multidimensional integrals. *Operations Research Letters*, 19(4):161–169, 1996.

[11] Nicolas Chopin. Fast simulation of truncated gaussian distributions. *Statistics and Computing*, 21(2):275–288, 2011.

[12] ASTM Subcommittee D02.08. *D86 Standard Test Method for Distillation of Petroleum Products at Atmospheric Pressure.* ASTM International, 28th edition, 2012.

[13] N Dobigeon and JY Tourneret. Efficient sampling according to a multivariate gaussian distribution truncated on a simplex. Technical report, ENSEEIHT, 2007.

[14] Jo Eidsvik and Håkon Tjelmeland. On directional metropolis–hastings algorithms. *Statistics and Computing*, 16(1):93–106, 2006.

[15] B Fleury. Concentration in a thin euclidean shell for log-concave measures. *Journal of Functional Analysis*, 259(4):832–841, 2010.

[16] A.E. Gelfand and A.F.M. Smith. Sampling-based approaches to calculating marginal densities. *Journal of the American statistical association*, 85(410):398–409, 1990.

[17] J. Geweke. Efficient simulation from the multivariate normal and student-t distributions subject to linear constraints and the evaluation of constraint probabilities. In *Computing science and statistics: proceedings of the 23rd symposium on the interface*, pages 571–578. Fairfax, Virginia: Interface Foundation of North America, Inc, 1991.

[18] W.R. Gilks, S. Richardson, and D. Spiegelhalter. *Markov Chain Monte Carlo in practice: interdisciplinary statistics*, volume 2. Chapman & Hall/CRC, 1995.

[19] D Hudebine, J Verstraete, and T Chapus. Statistical reconstruction of gas oil cuts. *Oil & Gas Science and Technology–Revue dIFP Energies nouvelles*, 66(3):461–477, 2009.

[20] D.P. Kroese, T. Taimre, and Z.I. Botev. *Handbook of Monte Carlo Methods*, volume 706. Wiley, 2011.

[21] Richard A Levine, Zhaoxia Yu, William G Hanley, and John J Nitao. Implementing random scan gibbs samplers. *Computational Statistics*, 20(1):177–196, 2005.

[22] Genyuan Li and Herschel Rabitz. A general analysis of exact lumping in chemical kinetics. *Chemical engineering science*, 44(6):1413–1430, 1989.

[23] Dimitris K Liguras and David T Allen. Structural models for catalytic cracking. 1. model compound reactions. *Industrial & Engineering Chemistry Research*, 28(6):665–673, 1989.

[24] J.S. Liu. *Monte Carlo strategies in scientific computing.* Springer, 2008.

[25] Kanti V Mardia and Peter E Jupp. *Directional statistics*, volume 494. Wiley, 2009.

[26] George Marsaglia. Evaluating the normal distribution. *Journal of Statistical Software*, 11(4):1–7, 2004.

[27] Mi Mi Saine Aye and Nan Zhang. A novel methodology in transforming bulk properties of refining streams into molecular information. *Chemical engineering science*, 60(23):6702–6717, 2005.

[28] Brian Neelon and David B Dunson. Bayesian isotonic regression and trend analysis. *Biometrics*, 60(2):398–406, 2004.

[29] Matthew Neurock, Abhash Nigam, Daniel Trauth, and Michael T Klein. Molecular representation of complex hydrocarbon feedstocks through efficient characterization and stochastic algorithms. *Chemical engineering science*, 49(24):4153–4177, 1994.

[30] Ari Pakman and Liam Paninski. Exact hamiltonian monte carlo for truncated multivariate gaussians. *arXiv preprint arXiv:1208.4118*, 2012.

[31] Brett Presnell and Pavlina Rumcheva. The mean resultant length of the spherically projected normal distribution. *Statistics & Probability Letters*, 78(5):557–563, 2008.

[32] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. Numerical recipes in C: the art of scientific computing. 2. *Cambridge: CUP*, 1992.

[33] Tarmo M Pukkila and C Radhakrishna Rao. Pattern recognition based on scale invariant discriminant functions. *Information sciences*, 45(3):379–389, 1988.

[34] Steven P Pyl, Zhen Hou, Kevin M Van Geem, Marie-Françoise Reyniers, Guy B Marin, and Michael T Klein. Modeling the composition of crude oil fractions using constrained homologous series. *Industrial & Engineering Chemistry Research*, 50(18):10850–10858, 2011.

[35] Steven P Pyl, Kevin M Van Geem, Marie-Françoise Reyniers, and Guy B Marin. Molecular reconstruction of complex hydrocarbon mixtures: An application of principal component analysis. *AIChE Journal*, 56(12):3174–3188, 2010.

[36] Christian P Robert and George Casella. *Monte Carlo statistical methods*. Springer, 2004.

[37] C.P. Robert. Simulation of truncated normal variables. *Statistics and computing*, 5(2):121–125, 1995.

[38] G. Rodriguez-Yam, R.A. Davis, and L.L. Scharf. Efficient gibbs sampling of truncated multivariate normal with application to constrained linear regression. 2004.

[39] S Roweis. Gaussian identities. 1999.

[40] Sam Roweis and Zoubin Ghahramani. A unifying review of linear gaussian models. *Neural computation*, 11(2):305–345, 1999.

[41] Devinder S Sivia. *Data analysis: a Bayesian tutorial.* Oxford University Press, 1996.

[42] Robert L. Smith. Efficient monte carlo procedures for generating points uniformly distributed over bounded regions. *Operations Research*, 32(6):1296–1308, 1984.

[43] Alan D Sokal. Monte carlo methods in statistical mechanics: foundations and new algorithms, 1989.

[44] Daniel M Trauth, Scott M Stark, Thomas F Petti, Matthew Neurock, and Michael T Klein. Representation of the molecular structure of petroleum resid through characterization and monte carlo modeling. *Energy & fuels*, 8(3):576–580, 1994.

[45] Kevin M Van Geem, Damien Hudebine, Marie Francoise Reyniers, Francois Wahl, Jan J Verstraete, and Guy B Marin. Molecular reconstruction of naphtha steam cracking feedstocks based on commercial indices. *Computers & chemical engineering*, 31(9):1020–1034, 2007.

[46] Jan J Verstraete, Nadège Revellin, Hugues Dulot, and Damien Hudebine. Molecular reconstruction of vacuum gasoils. *Prepr. Pap.-Am. Chem. Soc., Div. Fuel Chem*, 49(1):20, 2004.

[47] Fangpo Wang and Alan E Gelfand. Directional data analysis under the general projected normal distribution. *Statistical Methodology*, 2012.

[48] Larry Wasserman. *All of statistics: a concise course in statistical inference.* Springer Verlag, 2004.

[49] Curtis H Whitson, Thomas F Anderson, and Ingolf Søreide. Application of the gamma distribution model to molecular weight and boiling point data for petroleum fractions. *Chemical engineering communications*, 96(1):259–278, 1990.

[50] Yongwen Wu and Nan Zhang. Molecular characterization of gasoline and diesel streams. *Industrial & Engineering Chemistry Research*, 49(24):12773–12782, 2010.