Bayesian Adaptive Independence Sampling with Latent Variables

Christian Michael Davey

A thesis submitted for the degree of Doctor of Philosophy at
Monash University in the year 2020
School of Mathematics

Doctoral Advisor
Associate Professor Jonathan Macgregor Keith

Associate Doctoral Advisor
Associate Professor Timothy Michael Garoni

Copyright Notice

Abstract

This dissertation introduces a Bayesian approach for Markov chain Monte Carlo (MCMC) sampling from distributions with many local maxima.

The new approach is called *Bayesian Adaptive Independence Sampling with Latent variables* (BAIS+L) and is an extension of the Bayesian Adaptive Independence Sampler (BAIS) that replaces its single normal distribution proposal with a mixture of normal distributions. As such, BAIS+L simultaneously generates multiple chains of samples using a Metropolis-Hastings-like approach. Just as in BAIS, the current states of these sampling chains are used to update the proposal distribution parameters, by sampling them directly from their posterior distributions, given the current chain states.

By replacing the single normal distribution proposal of BAIS with a mixture of normal distributions, BAIS+L aims to provide greater flexibility in capturing the shape of a target distribution in Euclidean space that has many local maxima. Through this greater flexibility, BAIS+L allows the proposal distribution to more closely approximate the target distribution, thereby improving sampling efficiency.

To enable direct sampling from the proposal distribution, a latent variable is introduced to indicate the component of the mixture from which a sample is drawn. The use of this latent variable complicates the relationship of the underlying Bayesian model used. To bypass this complication, an approximation is introduced into the Metropolis-Hastings acceptance ratio used in BAIS+L. This approximation leads to the stationary distribution of the sampler being an approximation of the target.

Approaches that are commonly used in the literature to guarantee ergodicity of MCMC methods are not applicable to BAIS+L, so conditions are provided to ensure uniform ergodicity of the new sampler, along with proofs of their sufficiency.

BAIS+L is shown to have comparable performance to the Equi-Energy Sampler (EES) when sampling from targets on a Euclidean state space, despite its approximate nature. However, simulations from a two-dimensional spin glass application demonstrate existing pitfalls with BAIS+L, for which possible remedies are suggested.

Finally, an exact version of BAIS+L, called *Exact Bayesian Adaptive Independence Sampling with Latent variables* (EBAIS+L) is introduced and compared to BAIS+L. This exact version is used to justify the use of the approximate BAIS+L, which is demonstrated to be more efficient than its exact counterpart.

Declaration

This thesis is an original work of my research and contains no material which has been accepted for the award of any other degree or diploma at any university or equivalent institution and that, to the best of my knowledge and belief, this thesis contains no material previously published or written by another person, except where due reference is made in the text of the thesis.

Print Name: Christian Michael Davey

Date: January 25, 2020

Acknowledgements

As with any significant undertaking, it would not have been possible for me to complete this dissertation without the support of others.

This research was supported by an Australian Government Research Training Program (RTP) Scholarship. I also wish to express my appreciation for the financial support provided by the Faculty of Science at Monash University in the form of a Dean's Postgraduate Scholarship in the first year of my candidature, prior to being awarded the RTP Scholarship.

I am also grateful for the administrative support provided by the School of Mathematics at Monash University. In particular, I wish to thank Linda Mayer and John Chan for their tireless efforts in helping me to organise presentations and conference attendance during my candidature.

With respect to more technical aspects of candidature, I am grateful to Art Owen of Stanford University and to two anonymous reviewers of a manuscript that I submitted to MCQMC2016, who suggested that I compare the approximate method (BAIS+L) discussed in Chapter 3 of this dissertation, to the equi-energy sampler of Kou et al. (2006). I also thank Christian Robert of Paris Dauphine University and the University of Warwick for suggesting the modification of BAIS+L to make it exact, as studied in Chapter 6 of this dissertation.

Of course, the project studied in this dissertation would not have been realised if it were not for my doctoral advisors. I am grateful to my associate advisor, Tim Garoni, for suggesting that I consider an application of BAIS+L to spin glasses, a field of reasearch that I had never heard of prior to him mentioning it to me.

I especially wish to thank my primary advisor, Jonathan Keith, for suggesting the project that I studied in this dissertation. In particular, I am grateful for his mentorship and guidance, especially with respect to the more theoretical aspects of my project. I also thank him for always being there for me and for encouraging me to achieve my potential as a researcher.

Finally, I am grateful to my family for the love and support that they provide to me, both throughout my candidature and in my life.

Contents

Introduction

In many areas of scientific research and application there is a need to sample from probability distributions in order to evaluate some quantity of interest. Some probability distributions, including the uniform distribution on a subset of $\mathbb{R}$, may be sampled directly, using pseudorandom number generation algorithms, such as the Mersenne Twister (Matsumoto and Nishimura, 1998). Other distributions for which direct sampling methods are not known, may nevertheless be obtained by transforming pseudorandom samples from a distribution for which direct methods do exist. An example of a distribution that can be sampled in this manner is the normal distribution, from which samples may be obtained by applying the Box-Muller transform (Box and Muller, 1958) to uniformly distributed pseudosamples on $[0, 1)$.

There are many other distributions for which no known direct sampling schemes or transformations exist. For these distributions, a stochastic approach is often used in the form of Monte Carlo (MC) or Markov chain Monte Carlo (MCMC) sampling. MC and MCMC approaches vary greatly in their performance, depending on how they are implemented and which distributions they are used to sample. Each has its own requirements and features, which impact its efficiency and the quality of the samples that it produces.

Some of these features include the proposal distributions in both MC and MCMC, the importance functions in some MC samplers, the partitioning scheme of the state vector in Gibbs sampling (an MCMC technique discussed in Section 1.3.2) and the acceptance ratio in Metropolis-Hasting sampling (another MCMC technique, discussed in Section 1.3.1). Early MC and MCMC methods kept these features fixed but, more recently, there has been interest in samplers that adapt them as the sampler progresses. Of particular interest is the area of *adaptive* MCMC methods, which is the primary focus of the current dissertation.

Many adaptive and non-adaptive MC and MCMC methods have already been successfully applied to problems in the natural and social sciences, including the seminal application to thermonuclear research at Los Alamos during World War II (Metropolis, 1987; Eckhardt, 1987). MC and MCMC methods form an ongoing area of research with a wealth of exciting and novel methods and applications just waiting to be discovered.

This dissertation further extends the MCMC practitioner's toolkit by introducing a new adaptive MCMC method: the *Bayesian Adaptive Independence Sampler with Latent variables* (BAIS+L). This new method is an approximate extension of the earlier *Bayesian Adaptive Independence Sampler* (BAIS) of Keith et al. (2008), which progressively homes in on the optimum normal distribution proposal in a Metropolis-Hastings sampler. However, due to its pro-

posal distribution having a single local maximum, BAIS is not appropriate for sampling from target distributions with many local maxima, since its proposal distribution is unable to capture the finer details of this structure. Being an independence sampler, good approximation of the proposal distribution to the target is important for efficient sampling (Gelman et al., 2004, 305–306), as discussed in Section 1.3.1.

BAIS+L increases the applicability of BAIS by replacing this normal proposal distribution with a mixture of normal distributions, thereby allowing it to more closely approximate distributions with multiple local maxima. The sampler uses a latent variable to select a mixture component before performing direct sampling from it. In this dissertation, this new method is developed and its performance studied through a comparison to the equi-energy sampler of Kou et al. (2006). A demonstration is then provided of one possible avenue for using BAIS+L to study a significant computational problem in condensed matter physics; that of the Edwards-Anderson model of the spin glass. Finally, an exact version of BAIS+L is introduced and compared to the approximate version of the sampler.

This dissertation is structured as follows. The first part reviews the relevant literature and is split into two chapters: one that reviews MC and MCMC theory; and another that reviews the target distributions and applications considered in Part II.

The second part of this dissertation is divided into four chapters. The first of these chapters presents a detailed description of the development of BAIS+L and provides sufficient conditions that will guarantee its ergodicity. Proofs are provided for the sufficiency of these conditions as well as a study of how they may be enforced in practice. It then concludes with a discussion of some conjectured properties of the sampler.

The second of these chapters compares BAIS+L to the equi-energy sampler of Kou et al. (2006); a powerful Monte Carlo method designed to sample from distributions with multiple local maxima. It considers both the relative efficiency and accuracy of BAIS+L by testing its performance on three targets in Euclidean space, to which Kou et al. (2006) applied the equi-energy sampler.

The third of these chapters initiates the use of BAIS+L as a tool to study spin glasses. It demonstrates some of the pitfalls that exist with its implementation to such a problem, while suggesting potential solutions to them, as avenues for future work.

The last of these chapters outlines an exact extension of BAIS, the Exact Bayesian Adaptive Independence Sampler with Latent Variables (EBAIS+L), which also uses a mixture of normal distributions to generate proposed states but with a different dependence on the current ones. This chapter then compares BAIS+L to EBAIS+L, highlighting the advantages of using the approximate approach over an exact one.

This dissertation concludes by tying together the presented results before providing suggestions for future work.

Part I

Literature Review

This part of the dissertation provides a review of the current state of knowledge in the fields of adaptive MCMC methods and their applications.

The first chapter is an introduction to MC and MCMC methods, detailing their development from their initial use to the present time. A number of important algorithms from both an historical perspective and a practical one, at the cutting-edge, are presented. The chapter also reviews theorems relevant to Markov chains and MCMC practice, which are referenced in Section 3.3.

The second chapter of this part discusses applications in which MC and MCMC methods play an important role. It introduces some examples, emphasising how and why stochastic numerical methods are used to study them. In particular, spin glass theory is detailed, along with common current approaches used to study them.

Chapter 1

Monte Carlo and Markov Chain Monte Carlo Theory

The advent of electronic computers during the Twentieth Century has made fast sampling from probability distributions practical. The strict rules that govern computers and the fast evaluation of functions that they permit allow algorithmic solutions to problems to be implemented easily and quickly.

As stated in the introduction, pseudorandom samples from specific distributions can be generated using deterministic algorithms, such as the Mersenne Twister (Matsumoto and Nishimura, 1998) for sampling from the uniform distribution on a subset of $\mathbb{R}$. When such algorithms do not exist, MC and MCMC methods provide a straightforward avenue for generating samples. These methods have played an important role in solving many problems that require the approximation of high-dimensional or otherwise analytically intractable integrals (Gilks et al., 1996, pp. 1). MC and MCMC methods essentially filter or transform directly-generated samples, such as those from a uniform or normal distribution, with the end result being a collection of samples from (virtually) any chosen distribution.

This chapter reviews MC and MCMC methods, starting with an introduction to MC. It discusses its initial development and the details of some examples that are used in practice.

It then moves on to MCMC, which is a type of iterative MC approach that produces a Markov chain of arbitrary length, instead of a fixed number of independent samples. This section begins with discussions of two important examples of MCMC approaches: Metropolis-Hastings sampling and Gibbs sampling. Their implementations are outlined and simple proofs are reviewed, which guarantee that their samples have the correct limiting distributions.

The next section then considers a technical aspect of Metropolis-Hastings samplers: the importance of having a well-chosen proposal distribution. In particular, it looks at why it is important for the proposal distribution in an independence sampler (cf. Section 1.3.1) to be as close to the target distribution as possible, and the ensuing pitfalls when this is not the case. Some methods that approximate arbitrary distributions using standard distributions that can be sampled directly are then reviewed. Specifically, this review consists of those methods that fit a normal mixture approximation to the distribution of interest.

This study then leads to the concept of adaptive MCMC. Adaptive MCMC methods attempt to optimise the choice of transition kernel in an MCMC sampler automatically, by using information gathered during the simulation, such as measurements of past performance (Robert and Casella, 2004, pp. 284). This section primarily focusses on adaptive MCMC methods that seek to select an optimum proposal distribution.

Some adaptive MCMC methods update their transition kernels using samples drawn from multiple previous iterations, which breaks the Markov property of the resulting stochastic process. This introduces complications to their implementation that are not present in non-adaptive approaches. Therefore, a section is devoted to reviewing the theory of adaptation schemes, highlighting when they can be safely applied as well as how researchers have shown that they produce the desired output.

This chapter also describes some examples of adaptive MCMC samplers, grouped by their methods of adaptation. In particular, it reviews three samplers that are designed to ease the difficulty of sampling from distributions with multiple local maxima. One of these is parallel tempering (PT), which is used extensively in the study of disordered systems, including spin glasses (Contucci and Giardinà, 2013, pp. 165). Another is the Equi-Energy Sampler (EES) of Kou et al. (2006), which its authors showed to be very powerful. The other is the Adaptive Independent Metropolis-Hastings (AIMH) sampler of Giordani and Kohn (2010), which adaptively refines a normal mixture proposal using existing samples.

The review then takes an in-depth look at the sampler that is the main motivation of the current dissertation: the Bayesian Adaptive Independence Sampler (BAIS) of Keith et al. (2008). Its design and implementation are discussed, and a theorem that guarantees that this sampler has the correct limiting distribution is reviewed. The shortcomings of the sampler are then considered with respect to targets with multiple local maxima.

The final section of this chapter is dedicated to the theoretical properties of Markov chains and sequences. It reviews important results that are referenced in the proofs of Section 3.3 and a result of Besag et al. (1995) that justifies the exact algorithm presented in Chapter 6. This section concludes with descriptions of the convergence diagnostics that are used in the numerical simulations presented later in this dissertation.

1.1 Monte Carlo Methods

This section reviews Monte Carlo (MC) methods. It first discusses their foundations, beginning with their motivation and subsequent development. It then describes the general idea behind MC methods before reviewing some notable examples.

1.1.1 Foundations

The following discussion reviews the main ideas behind MC methods, starting with their motivation. It is then followed by a brief description of the general MC approach.

Motivation

With the advent of computer technology during the first half of the Twentieth Century, the ability to perform many calculations quickly became possible. This enabled the computation of quantities that previously required many hands to compute (Metropolis, 1987) and permitted the growth of new fields of numerical research and application. In particular, their speed and the strict rules governing their operation made it possible to automate the generation of pseudosamples from probability distributions. Large numbers of such pseudosamples are required in diverse fields for the estimation of pertinent quantities that cannot be evaluated analytically (cf. Chapter 2 for examples).

As stated in the introduction to this dissertation, the uniform distribution on a subset of $\mathbb{R}$ may be (pseudo-)sampled using a deterministic algorithm, such as the Mersenne Twister of Matsumoto and Nishimura (1998). The particular example that they introduced is an iterative procedure that produces a deterministic sequence of integers using bitwise operations on a collection of n-bit integers, such that the resulting sequence has the properties of a random sequence of independent and identically uniformly distributed (i.i.d.) samples from the set of integers $\{0, \ldots, 2^{n-1}\}$. It is trivial to rescale such a sequence to one of n-bit floating-point numbers, representing real numbers on any finite sub-interval of the real line.

It is also possible to transform (appropriately-scaled) uniform variates into normal ones using the Box-Muller transform, introduced by Box and Muller (1958). This method takes two independent uniform random variates,

$$U_1, U_2 \sim \mathcal{U}(0, 1),$$

and transforms them via Equation (1.1),

$$X_1 = \sqrt{-2 \log U_1} \cos(2\pi U_2), \tag{1.1}$$

to produce a standard normal random variate $X_1 \in \mathbb{R}$. Box and Muller (1958) also noted that, using the same two uniform variates, it is possible to produce a second normal random variate X_2, which is independent of X_1, via Equation (1.2),

$$X_2 = \sqrt{-2 \log U_2} \sin(2\pi U_1). \tag{1.2}$$

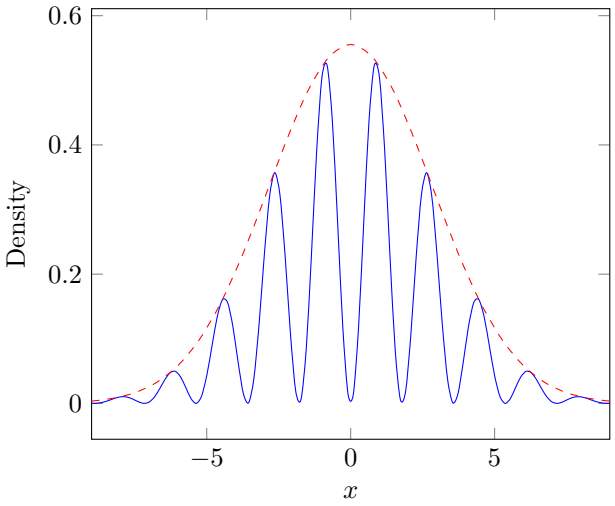
In both Equations (1.1) and (1.2) "log" represents the natural logarithm.

Unfortunately, it is not always practical to use a direct sampling approach for a given probability distribution, so another approach is required to sample from them. One such class of approaches is that of MC methods.

General Idea

Like direct sampling methods, MC sampling generates a collection of samples that in some way resemble samples drawn from the target distribution. However, these samples are not generated directly, using number-theoretic properties to produce a pseudo-independent sequence that mimics random samples from the target distribution. Instead, MC simulation modifies direct samples, using probabilistic criteria. It does this by first sampling points at random in the support of the function of interest, but from an alternative distribution. This alternative distribution is known by a number of different names, depending on

Figure 1.1: An example of a univariate target distribution $\pi(x)$ (solid line), whose mean is to be estimated by first drawing samples randomly from the normal distribution represented by the dashed line. The height of $\pi(x)$ is used only for illustrative purposes.



the context. These include "importance distribution" in importance sampling (cf. Section 1.1.2), or "proposal" or "jumping distribution" (Gelman et al., 2004, pp. 298) in MCMC (cf. Section 1.3). Throughout most of this dissertation, the name proposal distribution will be preferred for its more general connotation.

The proposal distribution is one that can be sampled easily (Gelman et al., 2004, pp. 292) or even directly. MC then filters or reweights the generated samples based on their densities with respect to the target function (Gelman et al., 2004, pp. 285).

As an example, consider Figure 1.1, in which the solid line represents the shape of a target density $\pi(x)$ that cannot be sampled directly. Also assume that the normalisation constant of $\pi(x)$ is unknown, so its height in the figure is used only for illustrative purposes.

Say that the estimand of interest is the mean $\mathbb{E}_\pi[f(x)]$ of some functional f, with respect to π. This may not be able to be computed directly but, as already seen, it is possible to sample points directly from a normal distribution $\mathcal{N}(\cdot|\boldsymbol{\mu}, \boldsymbol{\Sigma})$, whose support includes that of the target. This normal distribution is represented by the dashed line in Figure 1.1.

Once a collection of, say N, samples $\{x_1, \ldots, x_N\}$ has been drawn from the normal distribution, an MC technique, such as importance sampling (cf. Section 1.1.2) may be used to appropriately weight them. The sum of the product of these weights and their corresponding values of $f(x_n)$ is then used to give an estimate of $\mathbb{E}_\pi[f(x)]$.

Many other approaches exist, which use more sophisticated proposal distributions, refinement of the samples or sample rejection schemes. These methods vary in both their efficiency and complexity, depending on the target distribution.

1.1.2 Notable Examples

This section reviews some notable MC methods from the literature. These methods have been selected for discussion due to their historical significance or their relation to more sophisticated methods reviewed later in the current chapter.

Rejection Sampling

The Rejection Sampling (RS) method was introduced by John von Neumann (Forsythe, 1972) as a means to filter samples that have already been generated but which do not represent samples from the target distribution. As its name suggests not all samples necessarily become part of the final collection because they are filtered using a rejection criterion. This criterion places a probability of acceptance on each sample. A uniform random number is then generated and if it is less than this probability the sample is retained. Otherwise it is discarded.

RS is attractive because of its simplicity and ease of implementation. However, if the proposed samples are generated in low density parts of the state space they will have a high chance of being discarded, thereby making the sampler inefficient.

A rejection sampling approach, described by Gelman et al. (2004, pp. 284–285), is summarised in Algorithm A.1 of Appendix A.

Importance Sampling

The idea of importance sampling is to place weights on samples in order to determine which of them contribute the most to the final result of a Monte Carlo run (Hammersley and Morton, 1954). An example of such a scheme is the Bayesian Importance Sampler (BIS) of Geweke (1989), which aims to estimate the mean of a functional $f : \mathcal{X} \to \mathbb{R}$ with respect to an unnormalised measure with density $\pi(x)$, as in Equation (1.3),

$$\mathbb{E}_\pi[f(x)] = \frac{\int_\mathcal{X} f(x)\pi(x)dx}{\int_\mathcal{X} \pi(x)dx}. \tag{1.3}$$

To do so, BIS uses a collection of N i.i.d. samples $\{x_n\}_{n=1}^N$, generated from an *importance function* g, whose normalising constant is known. It then approximates the expectation using a weighted sum of these samples, as in Equation (1.4),

$$\mathbb{E}_\pi[f(x)] \approx \overline{f_N(x)} = \frac{\sum_{n=1}^N f(x_n)\pi(x_n)/g(x_n)}{\sum_{n=1}^N \pi(x_n)/g(x_n)}, \tag{1.4}$$

which they showed converges to the true mean as the number of samples N increases, provided

1. π is proportional to a proper probability density on $\mathcal{X}$,
2. g has support on all possible values of x,
3. $f(x)$ has a finite expectation with respect to the measure that π is assumed to admit.

To ensure the last point, Geweke (1989) assumed that f was integrable with respect to the measure mentioned in the last condition.

The remaining sections of this review of MC methods will take a look at three methods that are based on importance sampling.

Sequential Importance Sampling

An iterative importance sampler that is based on BIS is the Sequential Importance Sampler (SIS) of Doucet et al. (2000), which represents a "unified framework" that they referred to as a "sequential MC filter". Doucet et al. (2000) were motivated by the Bayesian filtering problem, which considers an unobserved process $\left\{\mathbf{x}^{(t)}\right\}_{t=0}^{\infty}$ in p-dimensional Euclidean space with transition density $p[\mathbf{x}^{(t)}|\mathbf{x}^{(t-1)}]$. Given an observed time-series $\left\{\mathbf{y}^{(t)}\right\}_{t=0}^{\infty}$ that is related to the process by the density given by Equation (1.5),

$$\pi\left[\left\{\mathbf{y}^{(t)}\right\}_{t=0}^{\infty}\middle|\left\{\mathbf{x}^{(t)}\right\}_{t=0}^{\infty}\right], \tag{1.5}$$

the goal at each time step T, is to evaluate estimates of a π-integrable function $f^{(T)}$, with respect to the posterior distribution of the unobserved process, which is given by Equation (1.6),

$$\pi\left[\left\{\mathbf{x}^{(t)}\right\}_{t=0}^{T}\middle|\left\{\mathbf{y}^{(t)}\right\}_{t=0}^{T}\right]. \tag{1.6}$$

Doucet et al. (2000) followed the BIS approach of Equation (1.4), using N weighted samples from the unobserved process of Equation (1.6) at times $t = 0$ to $t = T$, as given by Equation (1.7),

$$\begin{aligned}\mathbb{E}\left[f^{(T)}(\mathbf{x})\right] &= \int f^{(T)}\left[\left\{\mathbf{x}^{(t)}\right\}_{t=0}^{T}\right]\pi\left[\left\{\mathbf{x}^{(t)}\right\}_{t=0}^{T}\middle|\left\{\mathbf{y}^{(t)}\right\}_{t=0}^{T}\right]d\left\{\mathbf{x}^{(t)}\right\}_{t=0}^{T}\\ &\approx \overline{f_N^{(T)}(\mathbf{x})}\\ &= \sum_{n=1}^{N} f^{(T)}\left[\left(\left\{\mathbf{x}_n^{(t)}\right\}_{t=0}^{T}\right)\right]\tilde{w}_n^{(T)}.\end{aligned} \tag{1.7}$$

Here $\tilde{w}_n^{(T)}$ is the normalised weight of the nth sampling time series $\left\{\mathbf{x}_n^{(t)}\right\}_{t=0}^{T}$, with $\mathbf{x}^{(t)}$ sampled from the importance function at time step t. Doucet et al. (2000) wrote this weight differently from Geweke (1989), highlighting the dependence on the observed sequence up to time T. The unnormalised weight is given in Equation (1.8),

$$w_n^{(T)} = \frac{\pi\left[\left\{\mathbf{y}^{(t)}\right\}_{t=0}^{T}\middle|\left\{\mathbf{x}_n^{(t)}\right\}_{t=0}^{T}\right]\pi\left[\left\{\mathbf{x}_n^{(t)}\right\}_{t=0}^{T}\right]}{g\left[\left\{\mathbf{x}_n^{(t)}\right\}_{t=0}^{T}\middle|\left\{\mathbf{y}^{(t)}\right\}_{t=0}^{T}\right]}, \tag{1.8}$$

where g is the importance function, and $\pi(\mathbf{x})$ is the prior distribution on the unobserved sequence.

To extend BIS to this sequential setting, Doucet et al. (2000) assumed that samples could be drawn from π and that the likelihood (the first factor in

the numerator of Equation (1.8)) of the observed sequence, given the inferred hidden sequence, could be evaluated pointwise. To satisfy these assumptions they chose the importance function at time step T to be dependent on the observed sequence. By noting the time-dependence between the hidden variables they factored the importance function as in Equation (1.9).

$$g\left[\mathbf{x}^{(0)},\ldots,\mathbf{x}^{(T)}\middle|\mathbf{y}^{(t)},\ldots,\mathbf{y}^{(T)}\right] = g\left[\mathbf{x}^{(0)},\mathbf{y}^{(0)}\right]\prod_{t=1}^{T} g\left[\mathbf{x}^{(t)}\middle|\mathbf{x}^{(0)},\ldots,\mathbf{x}^{(t-1)},\mathbf{y}^{(0)},\ldots,\mathbf{y}^{(t)}\right]. \tag{1.9}$$

SIS infers each hidden variable sequentially in time (hence the name), starting with each $\mathbf{x}_n^{(0)}$, by proposing a new state for each inferred unobserved sequence from its corresponding factor in Equation (1.9). Once all inferred unobserved sequences at time t have been updated, the algorithm computes their unnormalised weights recursively from the weights at time step $t-1$, using Equation (1.10),

$$w_n^{(t)} = w_n^{(t-1)}\frac{\pi\left[\mathbf{y}^{(t)}\middle|\mathbf{x}_n^{(t)}\right]\pi\left[\mathbf{x}_n^{(t)}\middle|\mathbf{x}_{n-1}^{(t)}\right]}{g\left[\mathbf{x}_n^{(t)}\middle|\mathbf{x}_n^{(0)},\ldots,\mathbf{x}_n^{(t-1)},\mathbf{y}^{(0)},\ldots,\mathbf{y}^{(t)}\right]}. \tag{1.10}$$

To limit degeneracy of the algorithm to placing all importance weight on just one population member n, Doucet et al. (2000) suggested that each factor of the importance function in Equation (1.9) be the one that results in minimum variance of the importance weights. They showed that this requirement is satisfied by using the posterior distribution as the importance function, as in Equation (1.11),

$$g\left[\mathbf{x}_n^{(t)}\middle|\mathbf{x}_n^{(t-1)},\mathbf{y}^{(t)}\right] = \pi\left[\mathbf{x}_n^{(t)}\middle|\mathbf{x}_n^{(t-1)},\mathbf{y}^{(t)}\right]. \tag{1.11}$$

They argued that one way to measure the level of degeneracy encountered in an importance sampler run is to use the importance weights to compute the effective number of samples N_{eff} (Kong et al., 1994). To do this Doucet et al. (2000) suggested the estimator in Equation (1.12),

$$\hat{N}_{\text{eff}}^{(t)} = \frac{N}{\sum_{n=1}^{N}\left[w_n^{(t)}\right]^2}. \tag{1.12}$$

When $\hat{N}_{\text{eff}}^{(t)}$ is below a pre-determined level, Doucet et al. (2000) recommended following the advice of Rubin (1988) by resampling the population members via (1.13),

$$i'_n \sim \text{Categorical}\left[\tilde{w}_1^{(t)},\ldots,\tilde{w}_N^{(t)}\right], \forall n \in \{1,\ldots,N\}. \tag{1.13}$$

Combining the sequential importance sampling approach with this resampling step, Doucet et al. (2000) produced the general procedure outlined in Algorithm A.2 of Appendix A.

Doucet et al. (2000) showed that their algorithm generalises earlier, more specific examples of importance sampling. These include the methods of Handschin and Mayne (1969) and Handschin (1970) for multi-stage non-linear filtering, the Bayesian Importance Sampler of Geweke (1989) and the Bootstrap filter of Gordon et al. (1993).

Similar filtering methods were introduced by Del Moral et al. (1993) and Del Moral et al. (1995) for processing radar and sonar signals.

Population Monte Carlo

Population Monte Carlo (PMC) is another extension of importance sampling, introduced by Cappé et al. (2004). Like importance sampling, it produces a set of samples in one step from a known generating distribution and then weights them based on their relative densities with respect to the target. Unlike importance sampling, it is an iterative process, which can be repeated indefinitely to achieve a desired level of precision.

Cappé et al. (2004) designed PMC on the premise of Mengersen and Robert (2003), that instead of sampling from the desired target $\pi(x)$ on its support $\mathcal{X}$, one can sample from the product of N copies of it, thereby creating a "population" of samples. At every iteration t of the sampler, this population is updated by first randomly selecting indicators $\{z_1, \ldots, z_N\}$, representing from which of the K generating distributions $\left\{g_1^{(t)}, \ldots, g_K^{(t)}\right\}$ each population member $n \in \{1, \ldots, N\}$ is to be sampled. It then draws each new state n from its chosen distribution and computes its importance weight $\tilde{w}_n^{(t)}$, whose unnormalised form is given by Equation (1.14),

$$w_n^{(t)} \propto \frac{\pi\left[x_n^{(t)}\right]}{g_{z_n}^{(t)}\left[x_n^{(t)}\right]}. \tag{1.14}$$

As in the sequential importance sampler, Cappé et al. (2004) followed the advice of Rubin (1987) and prescribed resampling of the population to mitigate degeneracy of the algorithm.

Cappé et al. (2004) stressed that an important property of their method is the freedom of the generating distributions $\left\{g_1^{(t)}, \ldots, g_K^{(t)}\right\}$ to vary with time, possibly depending on (any number of) samples from previous iterations.

Algorithm A.3 of Appendix A details the general structure of a PMC sampler.

Sequential Monte Carlo

The Sequential Monte Carlo (SMC) method of Liu and Chen (1998) is a "general framework" that combines RS, SIS and MCMC (cf. Section 1.3) to perform sequential sampling of a dynamic probability distribution $\left\{\pi^{(t)}\right\}_{t=0}^{\infty}$. The dynamic setting considered by Liu and Chen (1998) was similar to the Bayesian filtering problem that was reviewed earlier, in the discussion of SIS (cf. Section 1.1.2). Like SIS, SMC aims to approximate the evolving target distribution by drawing weighted samples at each time step $t \in \mathbb{Z}^+$. That is, at each time t, the approximation in Equation (1.15) is used to estimate the mean of a $\pi^{(t)}$-integrable function $f^{(t)}$.

$$\mathbb{E}_{\pi^{(t)}}\left[f^{(t)}(x)\right] \approx \sum_{n=1}^{N} w_n^{(t)} f\left[x_n^{(t)}\right], \tag{1.15}$$

where $x_n^{(t)}$ is the nth sample at time t and $w_n^{(t)}$ is its corresponding importance weight.

Liu and Chen (1998) combined multiple techniques into their approach for a variety of reasons. Firstly, SIS is used to draw new samples for the N "streams" at time step t, given those at time step $t-1$ and their corresponding weights. This is achieved by sampling from a distribution $g^{(t)}$, which Liu and Chen (1998) recommended be set to the posterior estimation function, as demonstrated by Equation (1.16),

$$g^{(t)}\left[x^{(t)}\left|\mathbf{x}^{(0)},\ldots,\mathbf{x}^{(t-1)}\right.\right]=\pi^{(t)}\left[x^{(t)}\left|\mathbf{x}^{(0)},\ldots,\mathbf{x}^{(t-1)}\right.\right], \tag{1.16}$$

and setting incremental weights as in Equation (1.17)

$$w_n^{(t)}=u_n^{(t)}w_n^{(t)}, \tag{1.17}$$

where $u_n^{(t)}$ is given as the ratio of the density of the samples up to time $t-1$ under the target at time t to their density under the target at time $t-1$, as in Equation (1.18)

$$u_n^{(t)}=\frac{\pi^{(t)}\left[x_n^{(0)},\ldots,x_n^{(t)}\right]}{\pi^{(t-1)}\left[x_n^{(0)},\ldots,x_n^{(t)}\right]}. \tag{1.18}$$

Resampling is used to focus future sampling on the streams with highest weight to improve efficiency. Liu and Chen (1998) suggested doing so when the estimate of the square of the coefficient of variation (Hammersley and Handscomb, 1964, pp. 14),

$$\left[c_v^{(t)}\right]^2=\frac{\mathrm{Var}\left[\mathbf{w}^{(t)}\right]}{\left\{\mathbb{E}\left[\mathbf{w}^{(t)}\right]\right\}^2},$$

of the weights $\mathbf{w}^{(t)}=\left[w_1^{(t)},\ldots,w_N^{(t)}\right]$ is higher than some threshold c_{thresh} or at pre-determined resampling times.

MCMC methods are used when direct sampling from $\pi^{(t)}$ is not possible. Liu and Chen (1998) suggested and outlined the details of a Hastings Independence sampler (cf. Section 1.3.1) for this purpose.

Finally, Liu and Chen (1998) suggested rejection sampling (cf. Section 1.1.2) as an alternative to MCMC sampling that does not suffer from the need to run multiple iterations. However, they pointed out that a "covering constant" must be computed in this case, which they also provided.

Algorithm A.4 of Appendix A summarises the general framework of Liu and Chen (1998).

1.2 Markov Chain Theory

An alternative approach to MC as a means to generate samples for inference, is to generate the samples as a sequence, in which each successive sample is dependent on those preceding it. When the probability of a sample in such a sequence, conditional on all those preceding it, is the same the probability, conditional only on the immediate predecessor, then the sequence is called a *Markov chain* (Gelman et al., 2004, pp. 286). MC methods that produce their samples as a Markov chain are called *Markov chain Monte Carlo* (MCMC) methods (Gelman et al., 2004, pp. 285–286).

Before discussing such methods, however, it is important to understand the theory behind Markov chains. This section briefly reviews the theoretical Markov chain concepts that are needed to understand concepts discussed throughout this dissertation. Most importantly, it discusses the topic of ergodicity, which is used to justify the validity of the main sampler introduced in this dissertation, as discussed in Section 3.3. It also reviews some convergence diagnostics used in practice, which will be useful in Part II.

1.2.1 Using Markov Chains for Inference

Since the primary interest of this dissertation is a new MCMC method for sampling from a density π, it is imperative that the Markov chains constructed using it have particular properties that make it suitable for inference. The first of these properties is that estimates of any particular quantity of interest adequately represent their true values. That is, they need to be *consistent*.

The following discussion follows the definitions of consistency provided by Lehmann and Casella (1998, pp. 54), which they described in terms of convergence in probability. This concept is formalised by Definition 1.

Definition 1 (Convergence in probability)**.** *(Lehmann and Casella, 1998, pp. 54) Let $\{X^{(t)}\}_{t=1}^{\infty}$ be a sequence of random variables defined on state space $\mathcal{X}$. If the probability of a difference of at least r between $X^{(t)}$ and some constant $c \in \mathcal{X}$ tends to zero as t increases, for all $r > 0$, then $\{X^{(t)}\}_{t=1}^{\infty}$ is said to* converge in probability *to c. That is, if it satisfies Equation (1.19),*

$$\mathbb{P}(|X^{(t)} - c| \geq r) \to 0 \text{ as } t \to \infty \quad \forall r > 0. \tag{1.19}$$

Consistency of an estimator is then given by Definition 2.

Definition 2 (Consistent estimator)**.** *(Lehmann and Casella, 1998, pp. 54) Let f be a scalar of interest on state space $\mathcal{X}$ and let $\{\hat{f}^{(t)}(x)\}_{t=1}^{\infty}$ be a sequence of estimators $\hat{f}$ of f. Then $\hat{f}$ is said to be* consistent *if $\hat{f}^{(t)}(x)$ converges in probability to f for every $x \in \mathcal{X}$.*

Normally, estimates made from a Markov chain, started at some state $X^{(0)} = x^{(0)}$, are made after some initial period representing an initial portion of the chain that does not yet resemble a sequence of samples from a *stationary* or *limiting distribution* of the chain (if it has one) (Gelman et al., 2004, pp. 295).

Informally, a stationary distribution of a process is one that does not change with time, and a process that has such a distribution is called a *stationary process*. Formally, a stationary process is described by Definition 3.

Definition 3 (Stationary process)**.** *(Meyn and Tweedie, 1993, pp. 235) Let $\{X^{(t)}\}_{t=1}^{\infty}$ be a stochastic process on $\mathcal{X}$. If the marginal distribution of $\{X^{(t)}, \ldots, X^{(t+k)}\}$ does not change with t for any $k \in \mathbb{N}$, then $\{X^{(t)}\}_{t=1}^{\infty}$ is said to be* stationary*.*

The concept of a stationary distribution on a general state space is formalised in terms of an *invariant measure*. Before discussing what an invariant measure is, a few extra definitions are required.

First, let $\mathcal{X}$ be a topological (state) space with Borel σ-algebra $\mathscr{B}(\mathcal{X})$ and let $\pi : \mathscr{B}(\mathcal{X}) \to (0, \infty)$ be a positive σ-finite measure. Then π is σ-finite if it satisfies Definition 4.

Definition 4. *(Meyn and Tweedie, 1993, pp. 521) A σ-finite measure μ is one that is positive $\mu(A) \geq 0$, on any $A \in \mathscr{B}(\mathcal{X})$ and finite on each set $A_k \in \mathscr{B}(\mathcal{X})$ in a countable cover of $\mathcal{X} = \cup_{k=1}^{\infty} A_k$.*

An invariant measure is then formalised by Definition 5.

Definition 5 (Invariant measure)**.** *(Meyn and Tweedie, 1993, pp. 234) Let $\pi : \mathscr{B}(\mathcal{X}) \to (0, \infty)$ be a σ-finite measure on $\mathcal{X}$ and let $\mathbb{P}(x, \cdot)$ be the probability of a transition from state $x \in \mathcal{X}$. π is said to be* invariant *with respect to $\mathbb{P}(x, \cdot)$ if it is unchanged by $\mathbb{P}(x, \cdot)$. That is, if it satisfies Equation (1.20),*

$$\pi(A) = \int_{\mathcal{X}} \pi(dx)\mathbb{P}(x, A), \tag{1.20}$$

for every $A \in \mathscr{B}(\mathcal{X})$.

This dissertation is concerned with probability measures, for which $\pi(\mathcal{X}) = 1$ (Meyn and Tweedie, 1993, pp. 521).

While stationarity of a Markov chain is important for making use of the samples that it generates, it is useless if the stationary distribution is not the unique limiting distribution of the chain. When using MCMC to sample from a target density π, the resulting Markov chain must be designed to produce π as a stationary distribution. This leads to the important property of *ergodicity*, which guarantees that inferences made using a chain increase in precision and accuracy the longer a chain is run. Due to the great importance of this property, Section 1.2.2 is dedicated to its review.

1.2.2 Ergodicity

Ergodicity is a crucial concept, as it guarantees convergence of a Markov chain to a stationary distribution (Meyn and Tweedie, 1993, pp. 313). This essentially means that the chain, if allowed to run for an infinite time, will appear to be a collection of samples from the stationary distribution.

Before delving into a discussion of ergodicity, it is first important to understand the notions of the total variation of a measure and the periodicity of a Markov chain.

First note that the set of finite measures on the Borel σ-algebra $\mathscr{B}(\mathcal{X})$ of a topological space $\mathcal{X}$ trivially forms a vector space under the usual operations of addition $(\mu + \nu)(S) = \mu(S) + \nu(S)$ and scalar multiplication $(\lambda\mu)(S) = \lambda\mu(S)$, where μ, ν are measures on $\mathscr{B}(\mathcal{X})$ and $S \in \mathscr{B}(\mathcal{X})$.

The total variation of a measure is understood in terms of its *total variation norm*, which is given by Definition 6.

Definition 6 (Total variation norm)**.** *(Meyn and Tweedie, 1993, pp. 315) The* total variation norm $\|\mu\|$ *of a measure $\mu : \mathscr{B}(\mathcal{X}) \to \mathbb{R}$ is the largest possible difference between any two values in its range, as given by Equation (1.21),*

$$\|\mu\| := \sup_{A \in \mathscr{B}(\mathcal{X})} \mu(A) - \inf_{A \in \mathscr{B}(\mathcal{X})} \mu(A). \tag{1.21}$$

The periodicity of a Markov chain on a general state space is defined in terms of *small sets*. Theorem 1.3(ii) of Mengersen and Tweedie (1996) defined a small set according to Definition 7, which is a modified version of the definition given by Meyn and Tweedie (1993, pp. 109).

Definition 7 (Small set). *If the probability $\mathbb{P}^n(x, B)$ of an n-step transition from any state $x \in A \in \mathscr{B}(\mathcal{X})$ into any set $B \in \mathscr{B}(\mathcal{X})$ is bounded from below by some non-trivial probability measure $\nu : \mathscr{B}(\mathcal{X}) \to \mathbb{R}$, scaled by positive constant $\delta > 0$, then the set A is said to be* small. *In other words, a small set satisfies Equation (1.22),*

$$\mathbb{P}^n(x, B) \geq \delta\nu(B), \forall x \in A, \forall B \in \mathscr{B}(\mathcal{X}). \tag{1.22}$$

The period of a set $A \in \mathscr{B}(\mathcal{X})^+$ of positive measure $\nu(A) > 0$ is then formalised according to Definition 8.

Definition 8 (Period of a set). *(Meyn and Tweedie, 1993, pp. 119–120) Let $A \in \mathscr{B}(\mathcal{X})^+$ be a set in the Borel σ-algebra of state space $\mathcal{X}$, with positive measure $\nu(A) > 0$, and let E_A be the set of times $n \in \mathbb{Z}^+$ such that A is small with respect to measure $\nu_n = \delta_n\nu$ for some step-dependent constant $\delta_n > 0$,*

$$E_A := \left\{n \in \mathbb{Z}^+ : \mathbb{P}^n(x, B) \geq \delta_n\nu(B), \forall x \in A, \forall B \in \mathscr{B}(\mathcal{X}), \text{ for some } \delta_n > 0\right\}.$$

Then the period *d of A is the greatest common divisor of the elements of E_A.*

An *aperiodic* set is one for which $d = 1$ (Meyn and Tweedie, 1993, pp. 119). A ϕ-irreducible Markov chain on a state space $\mathcal{X}$ is said to be aperiodic if every $A \in \mathscr{B}(\mathcal{X})^+$ is aperiodic (Meyn and Tweedie, 1993, pp. 121).

The concept of *ϕ-irreducibility* is given by Definition 9.

Definition 9 (ϕ-irreducible Markov chain). *(Meyn and Tweedie, 1993, pp. 89) A Markov chain $\left\{X^{(t)}\right\}_{t=1}^{\infty}$ is ϕ-irreducible if every set $A \in \mathscr{B}(\mathcal{X})$ of positive measure $\phi(A) > 0$ is visited in finite time with strictly positive probability, by a Markov chain $\left\{X^{(t)}\right\}_{t=1}^{\infty}$, started at any state $x \in \mathcal{X}$, where ϕ is some common measure on $\mathcal{X}$. That is, if Equation (1.23) is satisfied,*

$$\phi(A) > 0 \implies \mathbb{P}[\tau_A < \infty] > 0, \tag{1.23}$$

where τ_A is the time of first arrival of the Markov chain into A.

A ϕ-irreducible Markov chain $\left\{X^{(t)}\right\}_{t=1}^{\infty}$ with invariant probability measure π is said to be *positive* (Meyn and Tweedie, 1993, pp. 235).

The ergodicity of a Markov chain is described by Definition 10, which summarises the discussion of Meyn and Tweedie (1993, pp. 315–316).

Definition 10 (Ergodic chain). *(Meyn and Tweedie, 1993, pp. 315–316) Let π be the unique invariant probability measure of an aperiodic Markov chain $\left\{X^{(t)}\right\}_{t=1}^{\infty}$ on state space $\mathcal{X}$, and let $\mathbb{P}^n(x, \cdot)$ be the n-step transition probability from state $x \in \mathcal{X}$. If the total variation norm between $\mathbb{P}^n(x, \cdot)$ and π tends to zero for all $x \in \mathcal{X}$, as n increases, then $\left\{X^{(t)}\right\}_{t=1}^{\infty}$ is said to be* ergodic. *That is, if Equation (1.24) is satisfied,*

$$\lim_{n\to\infty} \|\mathbb{P}^n(x, \cdot) - \pi\| = 0, \forall x \in \mathcal{X}. \tag{1.24}$$

The particular version of ergodicity of interest to the sampler introduced in Chapter 3 of this dissertation is that of *uniform ergodicity*. While general ergodicity guarantees convergence to a stationary distribution, this stronger form guarantees an upper bound on the rate of convergence in distribution of a sequence to its stationary distribution, irrespective of the starting position of the chain (Meyn and Tweedie, 1993, pp. 387).

The uniform ergodicity of a Markov chain is formalised by Definition 11.

Definition 11 (Uniformly ergodic chain). *(Meyn and Tweedie, 1993, pp. 388) If the supremum over all $x \in \mathcal{X}$ of the total variation norm between the n-step transition probability with stationary distribution π and n-step transition kernel $\mathbb{P}^n(x, \cdot)$ decreases to zero as $n \to \infty$, as in Equation (1.25), then the Markov chain is said to be* uniformly ergodic.

$$\lim_{n\to\infty} \sup_{x\in\mathcal{X}} \|\mathbb{P}^n(x, \cdot) - \pi(\cdot)\| = 0. \tag{1.25}$$

To guarantee that Definition 11 is met in this dissertation, Theorem 1 will be used. It is summarised from the equivalent forms of Theorem 16.0.2(v) of Meyn and Tweedie (1993, pp. 389) and Theorem 1.3(ii) of Mengersen and Tweedie (1996).

Theorem 1. *A Markov chain $\left\{X^{(t)}\right\}_{t=1}^{\infty}$ on $\mathcal{X}$ is uniformly ergodic if and only if the entire state space is small.*

Recall that the meaning of small was given in Definition 7.

Meyn and Tweedie (1993, pp. 396) showed that uniform ergodicity of an aperiodic Markov chain was equivalent to an important condition known as *Doeblin's Condition*, which is given by Definition 12.

Definition 12 (Doeblin's condition). *(Meyn and Tweedie, 1993) There exists a probability measure ϕ for which a particular n-step probability of a transition from any $x \in \mathcal{X}$ into any measurable set $A \in \mathscr{B}(\mathcal{X})$ is at least δ whenever A has measure of at least ϵ with respect to ϕ, for some $\epsilon < 1$ and $\delta > 0$. This definition is equivalent to Equation (1.26),*

$$\phi(A) > \epsilon \implies \mathbb{P}^n(x, A) \geq \delta, \forall x \in \mathcal{X}. \tag{1.26}$$

Doeblin's Condition has important implications on the rate of convergence of a uniformly ergodic Markov chain, which will be reviewed in Section 1.2.4.

Uniform ergodicity is a strong form of ergodicity that may not be achieved in practice. Nonetheless, a Markov chain may satisfy a weaker form of ergodicity, called *geometric ergodicity*.

Geometric ergodicity requires an understanding of what it means for a set to be *recurrent*. The basic notion of recurrence of a set is given by Definition 13.

Definition 13 (Recurrent set). *(Meyn and Tweedie, 1993, pp. 177) Let η_A be the number of times that a Markov chain $\{X^{(t)}\}_{t=1}^{\infty}$ on state space $\mathcal{X}$ visits set $A \in \mathscr{B}(\mathcal{X})$. Then A is said to be* recurrent *if it is expected that $\{X^{(t)}\}_{t=1}^{\infty}$ visits it infinitely many times. That is, if Equation (1.27) is satisfied,*

$$\mathbb{E}(\eta_A) = \infty. \tag{1.27}$$

For a Markov chain to be geometrically ergodic it is necessary that the sets that it visits satisfy a stronger form of recurrence, known as *Harris recurrent*, which Meyn and Tweedie (1993, pp. 204) defined as follows.

Definition 14 (Harris recurrent set). *(Meyn and Tweedie, 1993, pp. 204) If a Markov chain $\left\{X^{(t)}\right\}_{t=1}^{\infty}$ almost surely visits a measurable set $A \in \mathscr{B}(\mathcal{X})$, in the Borel σ-algebra of $\mathcal{X}$, infinitely often, then A is said to be* Harris recurrent. *That is, if Equation (1.28) is satisfied,*

$$\mathbb{P}_x(\eta_A = \infty) = 1, \tag{1.28}$$

where η_A is the number of visits to A and the subscript x of the probability operator indicates that the initial state is x.

The definition of Harris recurrence extends to a Markov chain by considering *all* sets that can be visited by a Markov chain, according to Definition 15.

Definition 15 (Harris recurrent Markov chain). *(Meyn and Tweedie, 1993, pp. 204). A ϕ-irreducible Markov chain, for which every set $A \in \mathscr{B}(\mathcal{X})$ of positive measure $\phi(A) > 0$ satisfies Definition 14, is also said to be Harris recurrent.*

If every $A \in \mathscr{B}(\mathcal{X})$, satisfies the weaker definition of recurrence given in Definition 13, then a ϕ-irreducible Markov chain on $\mathcal{X}$ is said to be recurrent (Meyn and Tweedie, 1993, pp. 178).

A Harris recurrent Markov chain that is also positive is said to be *positive Harris* (Meyn and Tweedie, 1993, pp. 236).

Meyn and Tweedie (1993, pp. 235) ensured the uniqueness of an invariant measure of a (Harris) recurrent Markov chain when it exists, according to Theorem 2.

Theorem 2 (Uniqueness of the invariant measure). *(Meyn and Tweedie, 1993, pp. 235) A recurrent Markov chain admits a unique invariant measure.*

Definition 16 ties together the preceding definitions, to formally define geometric ergodicity.

Definition 16 (Geometrically ergodic Markov chain). *(Meyn and Tweedie, 1993, pp. 359) If the geometric sum*

$$\sum_{n=1}^{\infty} r^n \left\| \mathbb{P}^n(x, \cdot) - \pi \right\|$$

of the total variation norm between the n-step transition kernel of a positive Harris Markov chain $\left\{X^{(t)}\right\}_{t=1}^{\infty}$ with finite stationary distribution π, with the nth term scaled by the nth power of a constant $r > 1$, is strictly finite, then $\left\{X^{(t)}\right\}_{t=1}^{\infty}$ is said to be geometrically ergodic.

An equivalent definition of geometric ergodicity is that the total variation norm between the stationary density π and the n-step transition kernel from any $x \in \mathcal{X}$ be bounded above by the nth power of a constant $c < 1$ multiplied by some function $h(x)$, as in Equation (1.29) (Chan, 1993),

$$\left\| \mathbb{P}^n(x, \cdot) - f \right\| \leq c^n h(x). \tag{1.29}$$

Chan (1993) noted that in this definition, uniform ergodicity follows if $h(x)$ is bounded.

In practice, for the MCMC practitioner to be satisfied that sampling from the stationary distribution is taking place, a convergence diagnostic may be used (cf. Section 1.2.5). However, in a finite-length chain of N samples, it may appear that a distribution satisfying Definition 3 has been achieved but after a long time $M >> N$ it may appear to have a different stationary distribution. In this situation, the process exhibits the *quasi-ergodic problem*, which is described by Definition 17.

Definition 17 (Quasi-ergodic problem)**.** *(Wood and Parker, 1957) The state space on which a stochastic process is defined can be split into multiple regions of significance, where the probability of transition between them in any (reasonable) finite time is extremely low.*

The result of this problem, according to Wood and Parker (1957), is that such a process may appear to have converged even if it has not. In other words, for a quasi-ergodic process, Definition 3 is satisfied for some measure π for some, but not *all*, $n \in \mathbb{N}$.

1.2.3 Detailed Balance

As stated in the preceding sections, the existence of a stationary distribution of a Markov chain generated by the Metropolis-Hastings Sampler, is guaranteed by detailed balance. The current section describes this concept and why it is sufficient for a stationary distribution to exist.

Detailed balance is summarised by Definition 18, which follows the discussion of Robert and Casella (2004, pp. 230), while not assuming that the invariant measure π has an associated density.

Definition 18 (Detailed balance)**.** *Robert and Casella (2004, pp. 230) Let $\{X^{(t)}\}_{t=1}^{\infty}$ be a Markov chain on a topological (state) space $\mathcal{X}$, with $\mathbb{P}(x, B)$, representing the probability of a transition from state x into a measurable set $B \in \mathscr{B}(\mathcal{X})$ in the Borel σ-algebra of $\mathcal{X}$. Then $\{X^{(t)}\}_{t=1}^{\infty}$ satisfies the* detailed balance condition *if there is a measure π for which Equation (1.30) holds for all measurable sets $A, B \in \mathscr{B}(\mathcal{X})$ in the Borel σ-algebra of $\mathcal{X}$.*

$$\int_A \mathbb{P}(x, B) d\pi(x) = \int_B \mathbb{P}(x, A) d\pi(x). \tag{1.30}$$

As noted by Robert and Casella (2004, pp. 230), taking one of A or B to be the entire state space $\mathcal{X}$ in Equation (1.112) immediately gives the definition of an invariant measure (cf. Definition 5), with π as the invariant measure, as follows,

$$\begin{aligned} \int_{\mathcal{X}} \mathbb{P}(x, A) d\pi(x) &= \int_{\mathcal{A}} \mathbb{P}(x, \mathcal{X}) d\pi(x) \\ &= \int_{\mathcal{A}} d\pi(x) \\ &= \pi(x), \end{aligned}$$

An important consequence of detailed balance is the property of *reversibility*.

Definition 19 (Reversible Markov chain)**.** *(Robert and Casella, 2004, pp. 230) A stationary Markov chain $\{X^{(t)}\}_{t=1}^{\infty}$, for which the conditional distributions of $X^{(t)}$ given $X^{(t-1)}$ and given $X^{(t+1)}$, respectively, are the same, is said to be* reversible.

This implies that the density π is stationary with respect to the transition kernel, as formalised by Theorem 3.

Theorem 3. *(Robert and Casella, 2004, pp. 230) Let $\mathcal{X}$ be a topological (state) space with Borel σ-algebra $\mathscr{B}(\mathcal{X})$ and let $\mathbb{P}(x, B)$ be the probabiliy of a transition of the Markov chain $\{X^{(t)}\}_{t=1}^{\infty}$ on $\mathcal{X}$ from state x into a measurable set $B \in \mathscr{B}(\mathcal{X})$. If this transition satisfies Definition 18 with some probability density π, then the chain is* reversible *with π as an invariant density.*

Recall from Theorem 2 that, on an uncountable state space, π is unique if the Markov chain is recurrent.

Preserving Detailed Balance with Adaptive Proposal Parameters

An important concept that will be key to the exact approach introduced in Chapter 6 is that of preserving detailed balance in a Markov process when the transition mechanism is allowed to vary randomly with time. Besag et al. (1995) showed that, as long as the mechanism to update the state x_n of a sampling chain n has no dependence on x_n, then detailed balance is preserved.

Formally, for product sets, their result is expressed by Theorem 4. In this theorem ν and π are used to represent a product measure and density, respectively, on $\mathcal{X}^N$. To be consistent with the notation of Besag et al. (1995), these symbols are also used to represent the corresponding measures on coordinate subspaces and conditional densities, respectively.

Theorem 4. *(Besag et al., 1995) Consider a topological (state) space $\mathcal{X}$, with Borel σ-algebra $\mathscr{B}(\mathcal{X})$, and denote the Borel σ-algebra of $\mathcal{X}^N$ by $\mathscr{B}(\mathcal{X}^N)$. Let $\mathbf{x} = (x_1, \ldots, x_N) \in \mathcal{X}^N$ be a vector of N parallel sampling chain states, each on state space $\mathcal{X}$, and let $\mathbf{x}_{-n} \in \mathcal{X}^{N-1}$ be $\mathbf{x}$ without the nth element. Denote the probability of a transition of the nth sampling chain from state x_n, taking $\mathbf{x}$ into a measurable set $A \in \mathscr{B}(\mathcal{X}^N)$, by $\mathbb{P}_n^{\boldsymbol{\theta}_n}(\mathbf{x}, A)$, where the parameter vector $\boldsymbol{\theta}_n$ is a random variable with density $p(\boldsymbol{\theta}_n|\mathbf{x}_{-n})$, supported on an appropriate parameter space $\mathcal{T}$. Furthermore, assume that $\mathbb{P}_n^{\boldsymbol{\theta}_n}(\mathbf{x}, \cdot)$ satisfies detailed balance for each $n \in \{1, \ldots, N\}$, each $\mathbf{x}_{-n} \in \mathcal{X}^{N-1}$ and each $\boldsymbol{\theta}_n \in \mathcal{T}$, with common stationary distribution π, with respect to a σ-finite product measure ν, as shown in Equation (1.31),*

$$\int_A \pi(x_n|\mathbf{x}_{-n})\mathbb{P}_n^{\boldsymbol{\theta}_n}(\mathbf{x}, B)d\nu(x_n) = \int_B \pi(x_n|\mathbf{x}_{-n})\mathbb{P}_n^{\boldsymbol{\theta}_n}(\mathbf{x}, A)d\nu(x_n), \tag{1.31}$$

where $d\nu(x) = d\nu(x_1) \times \cdots \times d\nu(x_N)$.

Finally, assume that all sampling chains are updated sequentially. Then the overall transition of all N chains also satisfies detailed balance, with stationary distribution π for all product sets $A, B \in \mathscr{B}(\mathcal{X}^N)$. That is, Equation (1.32) is satisfied,

$$\begin{aligned}\int_A \pi(\mathbf{x}) \int \mathbb{P}_n^{\boldsymbol{\theta}_n}(\mathbf{x}, B)d\mu(\boldsymbol{\theta}_n|\mathbf{x}_{-n})d\nu(\mathbf{x}) \\ = \int_B \pi(\mathbf{x}) \int \mathbb{P}_n^{\boldsymbol{\theta}_n}(\mathbf{x}, A)d\mu(\boldsymbol{\theta}_n|\mathbf{x}_{-n})d\nu(\mathbf{x}),\end{aligned} \tag{1.32}$$

for all product sets $A, B \in \mathscr{B}(\mathcal{X}^N)$.

The proof of Theorem 4 makes use of Tonelli's Theorem, which is provided by Theorem 5.

Theorem 5 (Tonelli). *(Billingsley, 1995, pp. 232–234) Let f be a non-negative function on the product space $\mathcal{X} \times \mathcal{Y}$ with product measure $\mu \times \nu$. Then Equation (1.33) holds,*

$$\int f(x,y)d(\mu \times \nu)(x,y) = \int \int f(x,y)d\nu(y)d\mu(x) = \int \int f(x,y)d\mu(x)d\nu(y). \tag{1.33}$$

Proof of Theorem 4. (Besag et al., 1995) Let $A, B \subseteq \mathcal{X}^N$ be two measurable subsets of the overall product space $\mathcal{X}^N$, where

$$A = A_1 \times \cdots \times A_N$$
$$B = B_1 \times \cdots \times B_N,$$

and A_n, B_n are the projections of A and B, respectively onto the coordinate subspace corresponding to the nth sampling chain. Also let A_{-n} and B_{-n} be the projections of A and B, respectively onto the coordinate subspaces corresponding to all but the nth sampling chain.

Factor $\pi(\mathbf{x})$ into $\pi(\mathbf{x}_{-n})\pi(x_n|\mathbf{x}_{-n})$ and take $\pi(x_n|\mathbf{x}_{-n})$ inside the inner integral. The left-hand side of Equation (1.32) is then expressed according to Equation (1.34),

$$\begin{aligned}
&\int_A \pi(\mathbf{x}) \int \mathbb{P}_n^{\boldsymbol{\theta}_n}(\mathbf{x}, B)d\mu(\boldsymbol{\theta}_n|\mathbf{x}_{-n})d\nu(\mathbf{x}) \\
&\quad = \int_{A_{-n}} \pi(\mathbf{x}_{-n}) \int_{A_n} \int \pi(x_n|\mathbf{x}_{-n}) \\
&\qquad \times \mathbb{P}_n^{\boldsymbol{\theta}_n}(\mathbf{x}, B)d\mu(\boldsymbol{\theta}_n|\mathbf{x}_{-n})d\nu(x_n)d\nu(\mathbf{x}_{-n})
\end{aligned} \tag{1.34}$$

Since π is a probability density and since $\mathbb{P}_n^{\boldsymbol{\theta}_n}$ is a probability measure for each $\boldsymbol{\theta}_n \in \mathcal{T}$, the integrand is non-negative. Therefore, by Tonelli's Theorem (Theorem 5), the order of integration of the inner two integrals may be swapped, to give Equation (1.35),

$$\begin{aligned}
&\int_{A_{-n}} \pi(\mathbf{x}_{-n}) \int_{A_n} \int \pi(x_n|\mathbf{x}_{-n}) \\
&\quad \times \mathbb{P}_n^{\boldsymbol{\theta}_n}(\mathbf{x}, B)d\mu(\boldsymbol{\theta}_n|\mathbf{x}_{-n})d\nu(x_n)d\nu(\mathbf{x}_{-n}) \\
&\quad = \int_{A_{-n}} \pi(\mathbf{x}_{-n}) \int \int_{A_n} \pi(x_n|\mathbf{x}_{-n}) \\
&\qquad \times \mathbb{P}_n(\mathbf{x}, B)d\nu(x_n)d\mu(\boldsymbol{\theta}_n|\mathbf{x}_{-n})d\nu(\mathbf{x}_{-n}).
\end{aligned} \tag{1.35}$$

Invoking the detailed balance of the nth sampling chain, Equation (1.35) is further rewritten according to Equation (1.36),

$$\begin{aligned}
&\int_{A_{-n}} \pi(\mathbf{x}_{-n}) \int \int_{A_n} \pi(x_n|\mathbf{x}_{-n}) \\
&\quad \times \mathbb{P}_n^{\boldsymbol{\theta}_n}(\mathbf{x}, B)d\nu(x_n)d\mu(\boldsymbol{\theta}_n|\mathbf{x}_{-n})d\nu(\mathbf{x}_{-n}) \\
&\quad = \int_{A_{-n}} \pi(\mathbf{x}_{-n}) \int \int_{B_n} \pi(x_n|\mathbf{x}_{-n}) \\
&\qquad \times \mathbb{P}_n^{\boldsymbol{\theta}_n}(\mathbf{x}, A)d\nu(x_n)d\mu(\boldsymbol{\theta}_n|\mathbf{x}_{-n})d\nu(\mathbf{x}_{-n}).
\end{aligned} \tag{1.36}$$

Note that, under a transition of sampling chain n, all other sampling chains remain fixed. Since $\mathbb{P}_n^{\boldsymbol{\theta}_n}(\mathbf{x}, B) = 0$ for all $\mathbf{x}_{-n} \notin B_{-n}$, the integrand is zero on $A_{-n} \setminus B_{-n}$. Therefore, Equation (1.36) may be further rewritten according to Equations (1.37) and (1.38),

$$\begin{aligned}
&\int_{A_{-n}} \pi(\mathbf{x}_{-n}) \int\int_{B_n} \pi(x_n|\mathbf{x}_{-n}) \\
&\qquad \times \mathbb{P}_n^{\boldsymbol{\theta}_n}(\mathbf{x}, A) d\nu(x_n) d\mu(\boldsymbol{\theta}_n|\mathbf{x}_{-n}) d\nu(\mathbf{x}_{-n}) \\
&= \int_{A_{-n}\cap B_{-n}} \pi(\mathbf{x}_{-n}) \int\int_{B_n} \pi(x_n|\mathbf{x}_{-n}) \qquad (1.37) \\
&\qquad \times \mathbb{P}_n^{\boldsymbol{\theta}_n}(\mathbf{x}, A) d\nu(x_n) d\mu(\boldsymbol{\theta}_n|\mathbf{x}_{-n}) d\nu(\mathbf{x}_{-n}). \\
&= \int_{B_{-n}} \pi(\mathbf{x}_{-n}) \int\int_{B_n} \pi(x_n|\mathbf{x}_{-n}) \qquad (1.38) \\
&\qquad \times \mathbb{P}_n^{\boldsymbol{\theta}_n}(\mathbf{x}, A) d\nu(x_n) d\mu(\boldsymbol{\theta}_n|\mathbf{x}_{-n}) d\nu(\mathbf{x}_{-n}).
\end{aligned}$$

Observe that the right-hand side Equation (1.38) is just the right-hand side of Equation (1.35), with the roles of A and B swapped. Therefore it may finally be expressed according to Equation (1.39),

$$\begin{aligned}
&\int_{B_{-n}} \pi(\mathbf{x}_{-n}) \int\int_{B_n} \pi(x_n|\mathbf{x}_{-n}) \\
&\qquad \times \mathbb{P}_n^{\boldsymbol{\theta}_n}(\mathbf{x}, A) d\nu(x_n) d\mu(\boldsymbol{\theta}_n|\mathbf{x}_{-n}) d\nu(\mathbf{x}_{-n}) \\
&= \int_B \pi(\mathbf{x}) \int \mathbb{P}_n^{\boldsymbol{\theta}_n}(\mathbf{x}, A) d\nu(\mathbf{x}). \qquad (1.39)
\end{aligned}$$

Thus, detailed balance is satisfied for all product sets $A, B \in \mathscr{B}(\mathcal{X}^N)$. □

Besag et al. (1995) implied that the result of Theorem 4 extends to more general sets than product sets. This extension is a result of Dynkin's $\pi - \lambda$ Theorem, which first requires definitions of π- and λ-systems. These systems are formalised by Definitions 20 and 21.

Definition 20 (π-System)**.** *(Billingsley, 1995, pp. 41) Let Ω be a set of interest and let $\mathcal{P} \subseteq 2^\Omega$ be a collection of subsets of Ω. If the intersection $A \cap B$ of any two sets $A, B \in \mathcal{P}$ is also in $\mathcal{P}$, then $\mathcal{P}$ is a* π-system*. That is, if the following condition is met,*

$$A, B \in \mathcal{P} \implies A \cap B \in \mathcal{P}.$$

Definition 21 (λ-System)**.** *(Billingsley, 1995, pp. 41) Let Ω be a set of interest and let $\mathcal{L} \subseteq 2^\Omega$ be a collection of subsets of Ω. If $\mathcal{L}$ contains*

1. *the complement of each of its elements,*

$$A \in \mathcal{L} \iff A^c \in \mathcal{L},$$

2. *all countable unions of disjoint elements,*

$$((A_1, A_2, \ldots \in \mathcal{L}) \wedge ((A_i \cap A_j = \emptyset) \forall i \neq j)) \implies \cup_{n=1}^{\infty} A_n \in \mathcal{L},$$

3. *and the generating space Ω,*

$$\Omega \in \mathcal{L},$$

then $\mathcal{L}$ is a λ-system.

Dynkin's $\pi - \lambda$ Theorem is then formalised by Theorem 6.

Theorem 6 (Dynkin's $\pi - \lambda$)**.** *(Billingsley, 1995, pp. 42) Let $\mathcal{L}$ be a λ-system and let $\mathcal{P}$ be a π-system such that $\mathcal{P} \subset \mathcal{L}$. Then the σ-algebra generated by $\mathcal{P}$ is contained within $\mathcal{L}$. That is, $\sigma(\mathcal{P}) \subset \mathcal{L}$.*

Finally, the extension of Theorem 4 will rely on the Monotone Convergence Theorem for Integrals, which is stated in Theorem 7.

Theorem 7 (Monotone Convergence of Integrals)**.** *(Meyn and Tweedie, 1993, pp. 522) Let $(\mathcal{X}, \mathscr{B}(\mathcal{X}), \mu)$ be a measure space and let $\{f_n\}_{n=1}^{\infty}$ be a sequence of μ-measurable functions $f_n : (\mathcal{X}, \mathscr{B}(\mathcal{X})) \to (\mathbb{R}, \mathscr{B}(\mathbb{R}))$ with limit f, as $n \to \infty$, such that it is monotonically increasing,*

$$0 \leq f_1(x) \leq f_2(x) \leq \cdots \leq f(x) \leq 1,$$

for μ-almost every $x \in \mathcal{X}$. Then the limit of the integral of f_n as $n \to \infty$ is equal to the integral of its limit, f,

$$\int_{\mathcal{X}} f(x) d\mu(x) = \lim_{n \to \infty} \int_{\mathcal{X}} f_n(x) d\mu(x).$$

With the Monotone Convergence Theorem and Dynkin's $\pi - \lambda$ Theorem reviewed, the extension of Theorem 4 implied by Besag et al. (1995) follows according to Theorem 8.

Theorem 8. *Let $(\Omega_1, \mathcal{F}_1)$ and $(\Omega_2, \mathcal{F}_2)$ be measurable spaces and let $(\Omega, \mathcal{F}) = (\Omega_1 \times \Omega_2, \mathcal{F}_1 \times \mathcal{F}_2)$ be their product space. Also let $\mathbb{P}(x, A)$ be the probability of a Markov transition from state $x \in \Omega$ into set $A \in \mathcal{F}$. Then a probability measure μ on $\mathcal{F}$ satisfies detailed balance,*

$$\int_A \mathbb{P}(x, B) d\mu(x) = \int_B \mathbb{P}(x, A) d\mu(x),$$

for any μ-measurable sets $A, B \in \mathcal{F}$ if and only if it does so for any μ-measurable sets of the forms $A = A_1 \times A_2$ and $B = B_1 \times B_2$, where $A_1, B_1 \in \mathcal{F}_1$ and $A_2, B_2 \in \mathcal{F}_2$.

While Theorem 8 is a straightforward application of Theorems 6 and 7 to product sets that satisfy detailed balance, to the best of the knowledge of the author of this dissertation, its justification has not been outlined explicitly in the literature. This elusive justification may be due to the simplicity of the argument, which is possibly taken for granted by everyone in the field. Therefore, this argument is outlined in full in the following proof.

Proof. If detailed balance is satisfied for any $A, B \in \mathcal{F}$, then it is, by definition, satisfied for product sets $A, B \in \mathcal{F}$, proving the "only if" part of the statement.

To prove the "if" part of the statement, first let $\mathcal{R}$ be the set of all product sets in $\mathcal{F}$. That is,

$$\mathcal{R} := \{A \in \mathcal{F} : A = A_1 \times A_2; A_1 \in \mathcal{F}_1; A_2 \in \mathcal{F}_2\}$$

Therefore $\sigma(\mathcal{R}) = \mathcal{F}$, by the definition of a product σ-algebra. Observe that $\mathcal{R}$ is a π-system. To see this, note that $\mathcal{R}$ contains at least the empty set, which is trivially a product subset of $\mathcal{F}$. Furthermore, if $A, B \in \mathcal{R}$ they may be expressed as $A = A_1 \times A_2$ and $B = B_1 \times B_2$, respectively, where $A_1, B_1 \in \mathcal{F}_1$ and $A_2, B_2 \in \mathcal{F}_2$. Taking their intersection gives

$$A \cup B = (A_1 \times A_2) \cap (B_1 \times B_2) = (A_1 \cap B_1) \times (A_2 \cap B_2) \in \mathcal{R}.$$

Therefore, $\mathcal{R}$ is a π-system.

Now let A be any set in $\mathcal{R}$ and let $\mathcal{F}_A \subseteq \mathcal{F}$ be the set of all sets $B \in \mathcal{F}$ that satisfy detailed balance with respect to A,

$$\mathcal{F}_A := \left\{ B \in \mathcal{F} : \int_A \mathbb{P}(x, B) d\mu(x) = \int_B \mathbb{P}(x, A) d\mu(x) \right\}.$$

Note that $\mathcal{F}_A$ is a λ-system. To see this, recall that $\mathcal{R} \subseteq \mathcal{F}_A$ for all $A \in \mathcal{R}$. Therefore $\mathcal{F}_A \neq \emptyset$, since $\mathcal{R}$ is non-empty. Let $B \in \mathcal{F}_A$. Then

$$\int_A \mathbb{P}(x, B^c) d\mu(x) = \int_A \mathbb{P}(x, \Omega) d\mu(x) - \int_A \mathbb{P}(x, B) d\mu(x).$$

$\Omega = \Omega_1 \times \Omega_2$ is a product set, so it also satisfies detailed balance, by assumption. Therefore, invoking detailed balance on each integral gives

$$\begin{aligned} \int_A \mathbb{P}(x, \Omega) d\mu(x) - \int_A \mathbb{P}(x, B) d\mu(x) &= \int_\Omega \mathbb{P}(x, A) d\mu(x) - \int_B \mathbb{P}(x, A) d\mu(x) \\ &= \int_{B^c} \mathbb{P}(x, A) d\mu(x), \end{aligned}$$

so $B \in \mathcal{F}_A \iff B^c \in \mathcal{F}_A$.

Now let $B_1, B_2, \ldots \in \mathcal{F}_A$ be a sequence of pairwise disjoint sets. Note that $\left\{ \mathbb{P}\left(x, \cup_{n=1}^N B_n\right) \right\}_{N=1}^\infty$ defines a sequence of μ-measurable functions with limit $\mathbb{P}\left(x, \cup_{n=1}^\infty B_n\right)$ as $N \to \infty$, such that

$$0 \leq \mathbb{P}(x, B_1) \leq \mathbb{P}(x, B_1 \cup B_2) \leq \cdots \leq \mathbb{P}\left(x, \cup_{n=1}^\infty B_n\right) \leq 1.$$

Hence, by the Monotone Convergence Theorem for Integrals (Theorem 7),

$$\int_A \lim_{N \to \infty} \mathbb{P}\left(x, \cup_{n=1}^N B_n\right) d\mu(x) = \lim_{N \to \infty} \int_A \mathbb{P}\left(x, \cup_{n=1}^N B_n\right) d\mu(x).$$

Since the B_n are pairwise disjoint, their probabilities add, giving Equation (1.40),

$$\begin{aligned} \lim_{N \to \infty} \int_A \mathbb{P}\left(x, \cup_{n=1}^N B_n\right) d\mu(x) &= \lim_{N \to \infty} \int_A \sum_{n=1}^N \mathbb{P}(x, B_n) d\mu(x) \\ &= \lim_{N \to \infty} \sum_{n=1}^N \int_A \mathbb{P}(x, B_n) d\mu(x). \end{aligned} \tag{1.40}$$

By definition, each B_n satisfies detailed balance with respect to A, giving

$$\lim_{N \to \infty} \sum_{n=1}^N \int_A \mathbb{P}(x, B_n) d\mu(x) = \lim_{N \to \infty} \sum_{n=1}^N \int_{B_n} \mathbb{P}(x, A) d\mu(x)$$

$$= \int_{\cup_{n=1}^{\infty} B_n} \mathbb{P}(x, A) d\mu(x),$$

where the final equality again holds due to the pairwise disjoint nature of the B_n. Taking the limit as $N \to \infty$ gives

$$\int_A \mathbb{P}\left(x, \cup_{n=1}^{\infty} B_n\right) d\mu(x) = \int_{\cup_{n=1}^{\infty} B_n} \mathbb{P}(x, A) d\mu(x).$$

Hence, all countable unions of disjoint sets in $\mathcal{F}_A$ are also in $\mathcal{F}_A$, meaning $\mathcal{F}_A$ is a λ-system. By Dynkin's $\pi - \lambda$ Theorem (Theorem 6) $\sigma(\mathcal{R}) \subseteq \mathcal{F}_A \subseteq \mathcal{F}$. Therefore, $\mathcal{F}_A = \mathcal{F}$.

Now take any $B \in \mathcal{F}$ and let $\mathcal{F}_B$ be the set of all sets in $\mathcal{F}$ that satisfy detailed balance with respect to B. As just shown, every μ-measurable product set in $\mathcal{F}$ satisfies detailed balance with any μ-measurable set in $\mathcal{F}$. Therefore $\mathcal{R} \subseteq \mathcal{F}_B$.

Note that $\mathcal{F}_B$ is also a λ-system. This result follows by an argument identical to the one used to show that $\mathcal{F}_A$ was a λ-system. Since $\mathcal{R} \subset \mathcal{F}_B$, by Dynkin's $\pi - \lambda$ Theorem (Theorem 6), $\sigma(R) \subset \mathcal{F}_B$. Once again, note that $\sigma(R) = \mathcal{F}$, so $\mathcal{F}_B = \mathcal{F}$.

Therefore, detailed balance is satisfied for every $A, B \in \mathcal{F}$. □

1.2.4 Rates of Convergence to the Stationary Distribution

In practical applications it is not only important to guarantee that the values in the generated Markov chain represent samples from the correct distribution, as noted in Section 1.2.2, but also how efficient a sampler is at generating those values. Since MCMC methods produce samples with autocorrelation, it may take some time before the resulting chain sufficiently represents samples from the stationary distribution; that is, before the distribution of the samples is within some threshold distance from the stationary distribution, with respect to some metric. Exactly how long this takes depends on the problem and the type of sampler used. Determining this convergence rate is not easy (Rosenthal, 1995) but, given appropriate conditions are satisfied, there are results that bound it.

One such bound is obtained when a Markov chain is uniformly ergodic. Meyn and Tweedie (1993) showed, using the equivalence of Doeblin's Condition and the definition of uniform ergodicity, that the n-step transition kernel of a uniformly ergodic Markov chain converges to the stationary distribution of the chain at a rate bounded by a power of the measure of the complement of a (small) subset of the state space, with respect to the measure ν_n that makes the sets small. Their result is formalised by Theorem 9,

Theorem 9 (Geometric convergence of a uniformly ergodic Markov chain)**.** *(Meyn and Tweedie, 1993, pp. 397) Let $\left\{X^{(t)}\right\}_{t=1}^{\infty}$ be a uniformly ergodic Markov chain on $\mathcal{X}$, where each set $A \in \mathscr{B}(\mathcal{X})$ is small with respect to probability measure ν and the m-step transition probability $\mathbb{P}^m(x, A)$ satisfies Equation (1.41),*

$$\mathbb{P}^m(x, A) \geq \delta \nu(A), \forall x \in \mathcal{X}, \tag{1.41}$$

for some $\delta > 0$.

Then the total variation norm between the chain's invariant measure π and its probability of an n-step transition from any $x \in \mathcal{X}$ is bounded above by the size of the complement of the entire state space, according to Equation (1.42),

$$\|\mathbb{P}^n(x, \cdot) - \pi\| \leq [1 - \delta\nu(\mathcal{X})]^{n/m}. \tag{1.42}$$

This property of uniform ergodicity was used by Mengersen and Tweedie (1996) to provide conditions for the uniform ergodicity and, hence, geometric convergence of the transition kernel of the Metropolis and independent Hastings samplers. See Section 1.3 for a discussion of these samplers.

Specifically, for an independent Hastings sampler, that is, one where the proposal density $g(y)$ is independent of the current state x of the chain (cf. Section 1.3.1), they provided Theorem 10.

Theorem 10. *Let g be the proposal density of an independent Hastings algorithm and let π be the target density. If there is a uniform ratio $\beta > 0$ such that Equation (1.43) is satisfied for all $x \in \mathcal{X}$,*

$$\frac{g(x)}{\pi(x)} \geq \beta, \tag{1.43}$$

then the Markov chain generated by the algorithm is uniformly ergodic.

They also showed that, in $\mathbb{R}$, a Markov chain, generated using a Metropolis algorithm with a symmetric proposal density $g(x, y)$ from state x to state y satisfying $g(x, y) = g(y-x) = g(x-y)$, converges geometrically if the tails of the target distribution are exponentially-decreasing and the symmetric proposal density also has a "finite absolute first moment". In $\mathbb{R}$, *exponentially-decreasing in the tails* means that the target π satisfies Equation (1.44) (Mengersen and Tweedie, 1996),

$$\log \pi(x) - \log \pi(y) \geq \begin{cases} \alpha(y - x) & \text{if } y \geq x \geq z \\ \alpha(x - y) & \text{if } y \leq x \leq z, \end{cases} \tag{1.44}$$

where $\alpha > 0$ and $z \in \mathbb{R}$ are constants.

Conditions for geometric convergence of the n-step transition kernel of a Markov chain generated using a Gibbs sampler to its stationary distribution have also been provided by Chan (1993).

1.2.5 Assessing Convergence and Mixing

If the Markov chain generated by an MCMC sampler is ergodic it eventually forgets its initial state and, regardless of where it was started, the stationary distribution will be the same (cf. Section 1.2.2). However, given the inherent dependence of one Markov chain state on the one immediately preceding it, it is reasonable to assume that the samples generated by an MCMC simulation will exhibit autocorrelation. However, note the definition of the lag-τ autocorrelation of the dth univariate marginal of a stationary p-dimensional Markov chain $\{X^{(t)}\}_{t=1}^{\infty}$, which is given in Equation (1.45) (Madras and Slade, 1996, pp. 296–297),

$$\text{Corr}_\tau(X_d) = \frac{\widehat{\text{Cov}}\left[X_d^{(t)}, X_d^{(t+\tau)}\right]}{\widehat{\text{Var}}\left[X_d^{(t)}\right]} \in [-1, 1], \tag{1.45}$$

where $\widehat{\text{Cov}}\left[X_d^{(t)}, X_d^{(t+\tau)}\right]$ is the estimated covariance between the stationary chain at times t and $t+\tau$, and $\widehat{\text{Var}}\left[X_d^{(t)}\right]$ is the estimated variance of the stationary chain.

Provided the correlation between any two successive samples is not perfect (+1), the autocorrelation must diminish with the time lag τ between two subsequences of the Markov chain, as evidenced by Equation (1.45).

The size of this autocorrelation determines how long the generated chain must be run in order for estimates made from it to achieve a desired level of precision. This property of the chain may be quantified by the related concepts of integrated autocorrelation time and the effective number of samples (cf. Section 1.2.5). Furthermore, the early portions of a Markov chain may have behaviour that differs significantly from the long term behaviour of the chain (Gelman et al., 2004, pp. 294).

Therefore, in MCMC simulations, it is customary to discard this initial part of the chain in a practice called *burn-in* (Gelman et al., 2004, pp. 295). The question is how long must the burn-in period be? This is not a straight-forward quantity to know *a priori*, so various diagnostics have been developed to assist MCMC practitioners with this task. These so called *convergence diagnostics* are used to determine if the inferred distribution of the chain of samples generated up to a particular time is significantly different from the stationary one. Hence, they are used to diagnose *non-convergence* of the Markov chain.

The following is a brief look at some common convergence diagnostics.

Geweke's Convergence Diagnostic

Geweke's convergence diagnostic (Geweke, 1991) considers the output of a single sequence of MCMC samples. It compares the mean value of an initial portion of the sequence to a later portion. If the two have the same mean to a chosen level of precision then convergence is not ruled out.

Gelman and Rubin's Convergence Diagnostic

Gelman and Rubin's convergence diagnostic (Gelman and Rubin, 1992) is designed to measure convergence of a simulation involving independent parallel sampling chains. The diagnostic works by comparing the average variance of the sample average within each sampling chain to the variance of the sample average between the chains. Specifically, following the notation of Gelman et al. (2004, pp. 296), the within-chain variance W of the average of a quantity ψ is given by Equation (1.46) and the between-chain variance B is given by Equation (1.47).

$$W = \frac{1}{N(T-1)} \sum_{t=1}^{T} \sum_{n=1}^{N} \left[\psi_n^{(t)} - \bar{\psi}_n\right]^2 . \tag{1.46}$$

$$B = \frac{T}{N-1} \sum_{n=1}^{N} \left[\bar{\psi}_n^{(t)} - \bar{\psi}\right]^2 , \tag{1.47}$$

where $\psi_n^{(t)}$, $\bar{\psi}_n$ and $\bar{\psi}$ are the tth sample of chain n, the MC average of chain n and the MC average over all chains in a simulation with N sampling chains and T iterations (Gelman et al., 2004, pp. 296).

Using these two quantities, an overestimate of the variance of ψ is computed according to Equation (1.48),

$$\widehat{s^+} = \frac{1}{T-1}W + \frac{1}{T}B. \tag{1.48}$$

This variance overestimate is useful to bound the variance of ψ when it is computed from correlated samples.

To assess convergence, the diagnostic of Gelman and Rubin (1992) computes the *potential scale reduction factor* (PSRF), which is given in Equation (1.49),

$$\text{PSRF} = \sqrt{\frac{\hat{V}}{W}\left(\frac{\nu}{\nu-2}\right)}, \tag{1.49}$$

where

$$\hat{V} = \hat{s}^+ + \frac{B}{NT},$$

and ν is the degrees of freedom of the Student's t distribution approximation of the distribution of the samples.

The degrees of freedom is given by Equation (1.50)

$$\nu = 2\frac{\hat{V}^2}{\widehat{\text{Var}}\left(\hat{V}\right)}, \tag{1.50}$$

where, by letting

$$s_n^2 = \frac{1}{T-1}\sum_{t=1}^{T}\left[\psi_n^{(t)} - \bar{\psi}_n\right]^2$$

be the sample variance of the nth chain, Gelman and Rubin (1992) derived the sampling variance $\widehat{\text{Var}}(\hat{V})$ of $\hat{V}$ to be

$$\begin{aligned}\widehat{\text{var}}\left(\hat{V}\right) = {} & \frac{1}{N}\left(\frac{T-1}{T}\right)^2 \widehat{\text{Var}}\left(s_n^2\right) \\ & + \frac{2}{N-1}\left(\frac{N+1}{NT}\right)^2 B^2 + 2\frac{(N+1)(T-1)}{N^2T} \\ & \times \left[\widehat{\text{Cov}}\left(s_n^2, \hat{\psi}_n^2\right) - 2\bar{\psi}\widehat{\text{Cov}}\left(s_n^2, \bar{\psi}_n\right)\right],\end{aligned}$$

where $\widehat{\text{Var}}(\cdot)$ and $\widehat{\text{Cov}}(\cdot,\cdot)$ represent the estimated/sampling variance and covariance, respectively, of the quantities enclosed in the parentheses, over the N sampling chains.

Brooks and Gelman (1998) noted, however, that the ratio $\nu/(\nu-2)$ in Equation (1.50) was incorrect and provided the correct potential scale reduction factor given in Equation (1.51)

$$\text{PSRF} = \sqrt{\frac{\hat{V}}{W}\left(\frac{\nu+3}{\nu+1}\right)}. \tag{1.51}$$

Brooks and Gelman (1998) also provided a number of alternative approaches to assessing convergence that were based on the potential scale reduction factor, including a multivariate version of the diagnostic.

To assess convergence, the PSRF should be plotted over time to see its evolution with the simulation. When it remains sufficiently close to 1 it is safe to assume that the generated Markov chain has reached stationarity.

Note that the preceding convergence diagnostic assumes that the generated chains of samples converge in distribution to the target. To avoid this assumption, Gorham and Mackey (2015) introduced a convergence diagnostic, based on Stein's method (Stein, 1972), which has the ability to explicitly detect convergence againt a target density. However, despite the more robust nature of their diagnostic, it is considerably more involved. Therefore, the diagnostic of Gorham and Mackey (2015) was not utilised in the numerical studies presented in Part II of this dissertation.

Integrated Autocorrelation Time

Another method for assessing convergence of a variable it to study the evolution of its *integrated autocorrelation time* (IAT), which estimates the average distance between uncorrelated samples in a chain. By recording all samples $\mathbf{x} = [x^{(t_0)}, \dots, x^{(T+t_0)}]$ in a simulation of length T starting at time t_0 one uses (1.52) to determine the IAT (Madras and Slade, 1996, pp. 296–297).

$$\text{IAT} = \frac{1}{2} + \sum_{t=1}^{\infty} \text{Corr}_t(\mathbf{x}) \tag{1.52}$$

where $\text{Corr}_t(\mathbf{x})$ is the lag-t autocorrelation of $\mathbf{x}$.

Once this quantity remains constant within a chosen level of tolerance it may be assumed that the variable has converged in distribution.

As in the case of Gelman and Rubin's diagnostic it is possible to use the IAT of a simulation to determine the effective number of independent samples (Madras and Slade, 1996, pp. 297). This is achieved by simply dividing the true number of samples by twice the IAT, as in Equation (1.53),

$$N_{\text{eff}} = \frac{N}{2\text{IAT}}. \tag{1.53}$$

1.3 Markov Chain Monte Carlo

Markov chain Monte Carlo (MCMC) methods represent a subclass of MC methods, in which each sample produced is dependent on the one immediately preceding it (Gelman et al., 2004, pp. 286). This section describes the development of MCMC methods starting with two important examples. These two examples are the Metropolis-Hastings (MH) sampler and the Gibbs sampler.

In addition to descriptions of these two methods, their validities are considered, and reviews of important results provided.

1.3.1 Metropolis-Hastings Sampling

A widely-used class of MCMC methods was introduced by Metropolis et al. (1953). In their paper, the authors proposed a method to generate a Markov chain, whose limiting distribution is the target distribution. The method, which has come to be known as the *Metropolis* algorithm (Betancourt, 2019), first proposes a sample from a distribution that can be sampled directly, before accepting

or rejecting it using a ratio based on the target densities at the current and proposed states of the Markov chain. The distribution used to propose samples in the Metropolis algorithm is symmetric but it was later extended by Hastings (1970) to asymmetric proposal distributions, using a suitable modification of the sampler's acceptance criterion.

Motivation

Metropolis et al. (1953) were interested in solving the state equations of an interacting particle system. They justified their new approach by considering a collection of N particles on a square with periodic boundary conditions. The authors argued against a naïve MC approach consisting of taking the weighted sum of a function evaluated at randomly-selected points on the square in order to compute the expectation of that function. In particular, for close-packed configurations, they noted that there is a high probability of randomly selecting points of low density, leading to an overall configuration of low weight. This, in turn, results in inaccurate estimation of the expectation being sought.

Instead, Metropolis et al. (1953) suggested selecting the points based on their densities under the target of interest and weighting them equally.

Description of the Metropolis Sampler

The method of Metropolis et al. (1953) generates a Markov chain whose stationary distribution is the target. In the case of Metropolis et al. (1953) this target was the Boltzmann distribution over configurations of the N particles within the state space.

Each iteration of the algorithm begins by proposing a sample from a symmetric density g that can be sampled directly, as in Equation (1.54),

$$y \sim g(\cdot|x, \boldsymbol{\theta}). \tag{1.54}$$

In Equation (1.54) the vector of parameters $\boldsymbol{\theta}$ of g has been explicitly stated in order to highlight its importance in the current study.

The symmetry of g means that the density associated with a move from state x to state y is the same as that of a move from state y to state x. This situation is described by Equation (1.55),

$$g(y|x, \boldsymbol{\theta}) = g(x|y, \boldsymbol{\theta}). \tag{1.55}$$

In particular, Mengersen and Tweedie (1996) studied a proposal distribution that can be written $g(|y - x||\boldsymbol{\theta})$ (cf. Section 1.2.4), which determines the size of a jump, relative to the current state x of the Markov chain to a proposed state y, regardless of the current state. As noted by Mengersen and Tweedie (1996), such a proposal distribution is trivially symmetric.

Metropolis et al. (1953) noted that this proposal distribution must have positive support on the entire state space, so as to ensure ergodicity (cf. Section 1.2.2) of the method and that the stationary distribution is indeed the target. In their illustration of the method they moved each of the N particles one at a time, proposing a new position from a uniform distribution centred about the current one, with the allowable moves in each dimension chosen within some maximum distance from the current position.

Once a new state has been proposed, the sampler then uses the ratio of the target density π at the proposed state to the target density at the current state to compute the probability of accepting the proposed move. For their interacting particle model, an individual state was the configuration of the particles (Metropolis et al., 1953). If the ratio is greater than one, indicating that the new state has a higher density under the target than the current one, then acceptance is guaranteed. Otherwise the move is accepted with probability equal to the ratio. This *acceptance ratio* is summarised in Equation (1.56),

$$\alpha(x, y) = \min\left\{1, \frac{\pi(y)}{\pi(x)}\right\}. \tag{1.56}$$

If the proposed move is accepted at the current time step then the Markov chain is updated to the proposed value. Otherwise the chain remains at its current state. However, Metropolis et al. (1953) emphasised that the Markov chain produced by the method must repeat the current state if a proposed move is rejected, in order to preserve the stationary distribution.

Independence Sampling and Random Walk Sampling

The previous section discussed the Metropolis sampler in the context of a *random walk*. By considering moving particles, Metropolis et al. (1953) necessarily conditioned the proposed position of a particle on its current one, as demonstrated by the dependence exhibited in Equation (1.54).

While a random walk had a physical justification in the work of Metropolis et al. (1953), in other problems, a random walk may not be suitable. In situations where the current state has no bearing on a proposed one, a proposal density that is constant with respect to the current state of the Markov chain may be more appropriate. Such lack of a proposed state on the current one describes an *independence sampler*, whose proposal density resembles Equation (1.57),

$$g(\cdot|x, \boldsymbol{\theta}) = g(\cdot|\boldsymbol{\theta}). \tag{1.57}$$

This distinction between independence and random walk samplers is important as it is fundamental to the ideas developed in Part II of the current dissertation.

Hastings' Extension to Asymmetric Proposals

The condition of the Metropolis proposal distribution being symmetric is rather restrictive and care must be taken to ensure that it is satisfied when using the Metropolis sampler. Hastings (1970) overcame this restriction by introducing an extension of the Metropolis sampler that weakens the assumptions on the proposal distribution. That is, Hastings' sampler works with an asymmetric proposal distribution. With this weaker assumption, Equation (1.54) is no longer satisfied and a modified acceptance ratio must be used to account for this difference. Equation (1.58) provides the acceptance ratio introduced by Hastings (1970), which allows the use of an asymmetric proposal distribution,

$$\alpha(x, y) = \min\left\{1, \frac{\pi(y)}{\pi(x)} \cdot \frac{g(x|y, \boldsymbol{\theta})}{g(y|x, \boldsymbol{\theta})}\right\}. \tag{1.58}$$

Note that when the proposal distribution is symmetric, the second fraction is equal to 1, giving the Metropolis acceptance ratio.

Algorithm A.5 describes this general MH sampler, of which the Metropolis sampler is a specific case when the proposal distribution is symmetric.

Stationary Distribution

The Metropolis and Hastings algorithms each induce a Markov process. As with any such process it is important to ascertain its limiting properties. Specifically, the chain needs to be ergodic (cf. Section 1.2.2) and its stationary distribution (cf. Section 1.2.1) needs to be the desired target.

Assume that the target distribution π, with support $\mathcal{X}$, admits a density, which shall also be denoted by π. Also assume that an appropriate proposal density $g(y|x, \boldsymbol{\theta})$, whose support includes $\mathcal{X}$, is used.

Chib and Greenberg (1995) noted that for an accept/reject MCMC sampler, such as MH, the transition kernel from state $x \in \mathcal{X}$ into some π-measurable set $A \in \mathscr{B}(\mathcal{X})$ in the Borel σ-algebra $\mathscr{B}(\mathcal{X})$ of $\mathcal{X}$ considers both accepted moves into A and, if $x \in A$, any rejected moves. Thus, the transition kernel may be written,

$$\mathbb{P}(x, A) = \int_A p_{\boldsymbol{\theta}}(x, y)dy + I_A(x)\left[1 - \int_{\mathcal{X}} p_{\boldsymbol{\theta}}(x, y)dy\right], \tag{1.59}$$

where $I_A(x) = 1$ if $x \in A$ and 0 otherwise, and $p_{\boldsymbol{\theta}}(x, y)$ represents the density associated with an accepted move from state x to state y.

The form of $p_{\boldsymbol{\theta}}$ for an MH sampler is given by Equation (1.60),

$$p_{\boldsymbol{\theta}}(x, y) = g(y|x, \boldsymbol{\theta}) \min\left\{1, \frac{\pi(y)}{\pi(x)} \cdot \frac{g(x|y, \boldsymbol{\theta})}{g(y|x, \boldsymbol{\theta})}\right\}, \tag{1.60}$$

which Chib and Greenberg (1995) noted, satisfies detailed balance (cf. Section 1.8.3 for a discussion of detailed balance) with the target density π, by design, according to Equation (1.61),

$$\begin{aligned} &\pi(x)g(y|x, \boldsymbol{\theta}) \min\left\{1, \frac{\pi(y)}{\pi(x)} \cdot \frac{g(x|y, \boldsymbol{\theta})}{g(y|x, \boldsymbol{\theta})}\right\} \\ &= \pi(y)g(x|y, \boldsymbol{\theta}) \min\left\{1, \frac{\pi(x)}{\pi(y)} \cdot \frac{g(y|x, \boldsymbol{\theta})}{g(x|y, \boldsymbol{\theta})}\right\}. \end{aligned} \tag{1.61}$$

In general, Chib and Greenberg (1995) showed that if $p_{\boldsymbol{\theta}}$ satisfies detailed balance, then π is invariant with respect to the transition kernel of Equation (1.59). This result is given by Theorem 11.

Theorem 11. *Let π denote a target distribution of an accept/reject MCMC sampler, such as MH, on a state space $\mathcal{X}$. Assume that π admits a density, which shall also be denoted by π. Let $p_{\boldsymbol{\theta}}(x, y)$ denote the density associated with an accepted move from state x to state y and assume that it satisfies detailed balance with π, so that*

$$\pi(x)p_{\boldsymbol{\theta}}(x, y) = \pi(y)p_{\boldsymbol{\theta}}(y, x).$$

Then π is a stationary distribution of the generated Markov chain.

Proof. (Chib and Greenberg, 1995) Consider the effect on π of a transition $\mathbb{P}(x, A)$ from any $x \in \mathcal{X}$ into a π-measurable set $A \subseteq \mathcal{X}$

$$\int_{\mathcal{X}} \pi(x)\mathbb{P}(x, A)dx, \tag{1.62}$$

Substituting Equation (1.59) into Equation (1.62) gives

$$\begin{aligned}\int_{\mathcal{X}} \pi(x)\mathbb{P}(x, A)dx = &\int_{\mathcal{X}} \pi(x) \int_A p_{\boldsymbol{\theta}}(x, y)dydx \\ &+ \int_{\mathcal{X}} I_A(x) \left[1 - \int_{\mathcal{X}} p_{\boldsymbol{\theta}}(x, y)dy\right] dx.\end{aligned}$$

Observing that all densities are non-negative, allowing Tonelli's Theorem (cf. Section 1.8.3) to be used to swap the order of integration, the first integral on the right-hand side may be rewritten as

$$\int_{\mathcal{X}} \pi(x) \int_A p_{\boldsymbol{\theta}}(x, y)dydx = \int_A \pi(y) \int_{\mathcal{X}} p_{\boldsymbol{\theta}}(y, x)dxdy.$$

Observe that multiplyng the integrand of the second integral by $I_A(x)$ is equivalent to integrating over A,

$$\begin{aligned}&\int_{\mathcal{X}} I_A(x)\pi(x) \left[1 - \int_{\mathcal{X}} p_{\boldsymbol{\theta}}(x, y)dy\right] dx \\ &= \int_A \pi(y) \left[1 - \int_{\mathcal{X}} p_{\boldsymbol{\theta}}(y, x)dx\right] dy,\end{aligned}$$

where the last equality takes advantage of the fact that swapping the roles of x and y does not change the integral.

Putting together these two equations back together gives

$$\begin{aligned}\int_{\mathcal{X}} \pi(x)\mathbb{P}(x, A)dx &= \int_A \pi(y) \int_{\mathcal{X}} p_{\boldsymbol{\theta}}(y, x)dxdy \\ &\quad + \int_A \pi(y) \left[1 - \int_{\mathcal{X}} p_{\boldsymbol{\theta}}(y, x)dx\right] dy \\ &= \int_A \pi(y)dy.\end{aligned}$$

Hence, the target density π is invariant under MH transitions. □

Corollary 1. *Metropolis transitions between states x and y satisfy detailed balance, with the target π being a stationary distribution of the induced Markov chain.*

Proof. It is clear that the Metropolis algorithm is a special case of the MH algorithm, where g is a symmetric proposal. Hence, the Metropolis sampler admits the target π as its stationary distribution. □

Clearly an issue arises if both $g(x|y, \boldsymbol{\theta})$ and $g(y|x, \boldsymbol{\theta})$ are zero in Theorem 11, as their ratio is then undefined. However, Roberts and Smith (1994) provided conditions that avoid this issue and which ensure ergodicity of the Markov chain, with π as its unique stationary distribution.

They showed that if g (without an accept/reject mechanism) results in an aperiodic and π-irreducible Markov chain (cf. Section 1.2.2) and if $g(x|y, \boldsymbol{\theta}) = 0$ if and only if $g(y|x, \boldsymbol{\theta}) = 0$, then the Markov chain induced by the MH sampler (i.e. with the accept/reject mechanism) is also aperiodic and π-irreducible. As discussed in Section 1.2.2, the conditions of aperiodicity and π-irreducibility are sufficient for such a process to result in an ergodic Markov chain. Roberts and Smith (1994) explained that such conditions are easy to implement in practice.

1.3.2 Gibbs Sampling

The second of the two foundational MCMC samplers reviewed in this section is Gibbs sampler. This sampler was introduced by Glauber (1963) in the context of sampling spins on a regular lattice and independently by Geman and Geman (1984) as a means of sampling from the posterior distribution in a Bayesian image restoration model. The case studied by Glauber (1963) is discussed in Section 2.3.4, as part of the review of spin glasses, while the current section is devoted to the more general framework of Geman and Geman (1984).

As in the review of the MH sampler, this section begins by reviewing the motivation of the Gibbs sampler's development, before describing its general implementation.

Motivation

Geman and Geman (1984) were interested in the problem of image restoration. Given a degraded image represented by a lattice of pixel intensities, they wished to infer the *maximum a posteriori* (MAP) estimate of the original undegraded image. To do so they used a combination of "stochastic relaxation" and an annealing schedule to generate a Markov chain that converged to the MAP estimate of the original image, given a degraded input image. The annealing schedule was used to increase the peakedness of the target distribution over the course of a simulation (cf. Section 1.4.2 for a discussion of its use).

Geman and Geman (1984) noted that the relatively large size of the space of possible images, compared to the small number of significant mass images, prohibited naïve updating of the generated Markov chain by sampling a full image each time. Therefore, they used a simpler approach of updating the state of each lattice individually, based on the current states of all other lattice sites. Their justification was that the conditional distributions are usually easier to sample than the full posterior. In fact, for the image restoration model that they employed, the conditional distribution of the state of each lattice site could be sampled directly. As such, there was no need for the use of acceptance ratios as is the case for MH sampling. This conditional sampling is the heart of the Gibbs sampler.

While their method was originally developed for the discrete state-space problem of image restoration, its applicability has successfully been extended to problems in which the state space is not discrete but in which conditional distributions of the target are still known. Such examples include the latent variable approach of Tanner and Wong (1987), which essentially describes a Gibbs sampler on a two-dimensional Euclidean state space, and the generalisation of that approach to Euclidean state spaces of higher finite dimension, provided by Gelfand and Smith (1990).

Description

Let $\mathcal{X}$ be the p-dimensional state space on which the target distribution π is defined, and let $\mathbf{S} = \cup_{n=1}^{N} S_n$ be a partition of $\{1, \ldots, p\}$ into N subsets.

In the general case (Gelman et al., 2004, p. 284), for a target distribution, represented by the scalar quantity $\pi(\mathbf{x})$, with $\mathbf{x} \in \mathcal{X}$, whose conditional distributions,

$$\pi\left(\{x_i\}_{i \in S_n} \middle| \{x_i\}_{i \in \mathbf{S} \setminus S_n}\right)$$

can be sampled directly, the Gibbs sampler updates each $\{x_i\}_{i \in S_n}$ one at a time from its conditional distribution, with the current state of all the other components held fixed. That is,

$$x_i \sim \pi\left(\cdot \middle| \{x_i\}_{i \in \mathbf{S} \setminus S_n}\right).$$

Geman and Geman (1984) noted that the sampler could be made to run in parallel but did not demonstrate this in their paper, due to the necessary hardware being unavailable to them at the time. If some parts of $\mathbf{x}$ are independent of others then this independence can be exploited to update separate parts in parallel.

Algorithm A.6 of Appendix A outlines the general form of the Gibbs sampler.

The Gibbs sampler has since seen further developments, particularly in the aforementioned cases of Tanner and Wong (1987) and Gelfand and Smith (1990), which extended it to problems outside of image restoration and processing.

The Gibbs sampler has also been combined with MH sampling in cases where direct sampling from the conditional distributions is not possible (Müller, 1991, 1993). In these cases one or more conditional distributions are sampled using a MH sampler, to give what is called a *Metropolis-within-Gibbs* scheme (Robert and Casella, 2004, pp. 393).

Ergodicity

Since the transition kernel in a Gibbs sampler is not constant, Geman and Geman (1984) were unable to use the same approach as Metropolis et al. (1953) to prove ergodicity of the resultant Markov chain. For their image restoration application, they did, nonetheless, show that their method produces an ergodic Markov chain with the target as its stationary distribution.

In Theorem C of Geman and Geman (1984) the authors stated, without proof, the ergodicity of the Gibbs sampler for image restoration. In Theorem A of Geman and Geman (1984) they noted that if each site of the lattice in an image restoration model is visited infinitely often using the Gibbs sampler, then the limiting probability of any particular configuration is equal to the density of that configuration under the target. The proof of Theorem A is considerably more involved than the proof of detailed balance of the MH sampler and the reader is directed towards the appendix of the original paper for its details.

Theorems A and C of Geman and Geman (1984) are specific to applications substantially similar to image restoration. However, Chan (1993) and Roberts and Smith (1994) provided conditions that guarantee ergodicity in more general applications, as well as conditions that guarantee geometric ergodicity of the

induced Markov chain. In particular, the conditions given by Roberts and Smith (1994) are easy to implement in practice.

Roberts and Smith (1994) showed that for a discrete target π it is sufficient for a Gibbs transition kernel to result in a π-irreducible Markov chain in order for it to converge pointwise to π. They also provided another set of conditions to ensure ergodicity of the Markov chain generated by Gibbs sampling.

Specifically, Roberts and Smith (1994) showed that for a probability space whose underlying measure is Lesbegue, if the target is lower semicontinuous at 0, has connected support and the integral $\int \pi(\mathbf{x}) d\mathbf{x}_i$ over each block i of simultaneously-updated coordinates of $\mathbf{x}$ forms a locally bounded family, then pointwise convergence of the transition kernel to the target and, hence, ergodicity of the Markov chain, is guaranteed. For definitions of "connected", "locally bounded" and "lower semicontinuous" see, for example, Tao (2006, pp. 430), Conway (1978, pp. 153) and Dixmier (1984, pp. 77–78), respectively. For a discussion of aperiodicity and π-irreducibility see Section 1.2.2.

As seen in Section 1.2, the concept of geometric ergodicity has important implications on the rate of convergence of the transition kernel to the stationary distribution.

1.4 Targeting Distributions with Multiple Local Maxima

In many practical uses of MC and MCMC methods, the state space may have multiple local maxima. One such situation has already been mentioned in the motivation of Geman and Geman (1984), who noted that an image restoration model may have many local maxima. Other important examples include the energy landscapes of spin glass models (cf. Section 2.3) or the rough "funnel" of a protein's conformational landscape (Onuchic et al., 1995).

1.4.1 The Problem

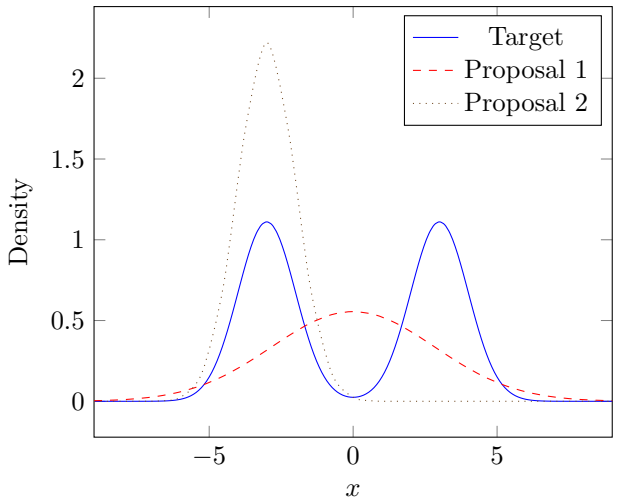
Target distributions with many local maxima present a problem because the high density regions are separated by low density ones. When sampling from such a distribution, an MCMC sampler must traverse these low density regions in order to sample all regions of significant mass.

As the samples generated in an MCMC procedure are realisations of a Markov chain, there is an inherent dependence between successive samples. Chains with high autocorrelations will have a low effective sample size (Gelman et al., 2004, pp. 298). Consequently, a large number of samples may be required before the stationary distribution is adequately represented.

Another concern is the issue of burn-in. Due to the aforementioned autocorrelation, the starting position of the chain will affect early samples. To reduce the impact of the starting position on quantities inferred from the chain, it is beneficial to discard an early portion of them (Gelman et al., 2004, pp. 295). High autocorrelations lead to a stronger effect, so a larger number of samples must be discarded for the remainder to be sufficiently uncorrelated with the starting position.

Recall the fractional component of the MH acceptance ratio given in Equation (1.58), for a transition from state x to state y under the target distribution

Figure 1.2: A naïve implementation of an MH independence sampler with a normal proposal distribution, applied to a target distribution on $\mathbb{R}$ with two local maxima. Note that one proposal distribution (dashed line) places significant mass on the low density region between local maxima, while the other (dotted line) places most mass around one local maximum and very little around the other.



π and having proposal distribution g, with parameter vector $\boldsymbol{\theta}$,

$$\frac{\pi(y)}{\pi(x)} \cdot \frac{g(x|y,\boldsymbol{\theta})}{g(y|x,\boldsymbol{\theta})} = \frac{\pi(y)}{\pi(x)} \Big/ \frac{g(y|x,\boldsymbol{\theta})}{g(x|y,\boldsymbol{\theta})} \tag{1.63}$$

where $\boldsymbol{\theta}$ represents the parameters of g.

The focus of Part II of this dissertation will be on an independence sampler, in which Equation (1.63) simplifies to Equation (1.64),

$$\frac{\pi(y)}{\pi(x)} \cdot \frac{g(x|\boldsymbol{\theta})}{g(y|\boldsymbol{\theta})} = \frac{\pi(y)}{\pi(x)} \Big/ \frac{g(y|\boldsymbol{\theta})}{g(x|\boldsymbol{\theta})} \tag{1.64}$$

In this case, it is clear that the closer that g is to π in distribution, the closer the outer fraction in Equation (1.64), and hence, the acceptance ratio, is to 1. However, if g differs greatly from π then the probability of proposing states with low density with respect to π may be high, resulting in them being likely to be rejected. Obviously, such rejections are wasteful and lead to slow movement of the Markov chain throughout the state space due to long periods of no change in its state.

Now consider the simple example of a normal mixture target on $\mathbb{R}$ with component centres $x = -3$ and $x = 3$, as demonstrated by the solid line in Figure 1.2. The dashed line in Figure 1.2 shows a naïve proposal distribution for an MH independence sampler. This proposal is a single normal distribution centred at the global mean of the target and with variance σ^2. It covers both true local maxima equally. However, it places greatest mass on the region between the local maxima, where the target density is low. As such, the majority of

proposed moves will be rejected due to their low density under the target. These rejections will lead to the sampler moving slowly through the state space.

An alternative approach for this independence sampler is to place the mean of the proposal distribution at one of the true local maxima of the target. For example, it may be placed at the leftmost local maximum (dotted line). While the sampler has a high chance of proposing and accepting moves around this local maximum, there is a low probability of proposing states around the second local maximum, due to the low mass assigned to it by the proposal.

Clearly neither of the discussed proposal distributions can effectively capture the shape of the target distribution on their own, which leads to inefficient exploration of the state space and a Markov chain with large *autocorrelation*.

Recall from Section 1.2.5 that the lag-τ autocorrelation of the dth univariate marginal of a stationary p-dimensional Markov chain $\left\{X^{(t)}\right\}_{t=1}^{\infty}$, is given by Equation (1.65) (Madras and Slade, 1996, pp. 296–297),

$$\text{Corr}_{\tau}(X_d) = \frac{\widehat{\text{Cov}}\left[X_d^{(t)}, X_d^{(t+\tau)}\right]}{\widehat{\text{Var}}\left[X_d^{(t)}\right]} \in [-1, 1], \tag{1.65}$$

where $\widehat{\text{Cov}}\left[X_d^{(t)}, X_d^{(t+\tau)}\right]$ is the estimated covariance between the stationary chain at times t and $t+\tau$, and $\widehat{\text{Var}}\left[X_d^{(t)}\right]$ is the estimated variance of the stationary chain.

The closer the autocorrelation is to zero, the more efficient the process.

If the sampler is an independence sampler, in which the location of the proposal distribution does not depend on the current state of the sampling chain, then the most likely states to be proposed are those around the peak of the proposal distribution. In the case of the proposal density represented by the red dashed line, this peak is centered at a position where the target density is very low. If the current state of the Markov chain is in one of the peaks of the target but the proposed state is in the peak of the proposal then $\pi(y)/g(y|\boldsymbol{\theta})$ will be low and $g(x|\boldsymbol{\theta})/\pi(x) << 1$, resulting in their product being even lower. Since such values of y are the most likely to be proposed, such a low acceptance scenario will be very likely.

Now consider a random walk, in which the proposal distribution is centred on the current state—which is again taken to be in one of the two larger peaks of the target. There still exists the problem of some states that have low acceptance probability being likely proposal candidates but there is now also another issue: the probability of proposing a state in the larger peak on the opposite side of the target is very low. This means that, while high-probability moves are proposed, they will tend to be in the vicinity of the proposal density's maximum. Even though the resulting acceptance rate will be high, it may be a long time before the other peak of the target is visited by the sampling chain.

In all three cases just discussed, the resulting chain of states will mix (move about the state space) slowly and exhibit strong autocorrelation. This high autocorrelation is the result of successive states being the same, due to either low acceptance probabilities or small accepted moves relative to the current state of the chain.

1.4.2 An Important Metaheuristic

One approach to reducing the aforementioned difficulty in passing through low-density regions of the state space into high-density ones is to flatten the distribution, thereby reducing the barrier between local maxima and, hence, the difficulty of switching between them.

Simulated annealing (SA) is a metaheuristic independently introduced by Kirkpatrick et al. (1983) and Černý (1985) . It is similar to MCMC in that it generates a chain of values that is driven by an underlying function, although it has a different goal. Instead of sampling from a probability density, it aims to find the global optimum of a rough function. That is, one with many local optima.

To avoid becoming trapped in local minima, SA employs a series of *temperatures* $T_{\max} = \infty > T_1 > \cdots > T_N = T_{\min}$. The choice of the intermediate temperatures between $T_{\max}$ and $T_{\min}$, as well as the rate at which the optimisation process progresses from one temperature to the next, depends on a *cooling schedule*, which determines the spacing between temperatures. This cooling schedule is dependent on the target function $\pi : \mathcal{X} \to \mathbb{R}$ being optimised.

SA concerns target functions that can be expressed as Boltzmann distributions, with energy function h, as demonstrated in Equation (2.5),

$$\pi_T(x) = \frac{1}{Z_T} \exp\left[-\frac{h(x)}{T}\right], \tag{1.66}$$

where Z_T is the *partition function*, which serves to normalise π_T, and is defined by Equation (1.67)

$$Z_T = \int_{\mathcal{X}} \pi_T(x) dx. \tag{1.67}$$

The roughness of the target at a given step of SA is dependent on the temperature, with higher temperatures leading to distributions with smaller variation in the heights of their local maxima. As the temperature is lowered and the distribution becomes more peaked, the chance of becoming trapped in a local minimum increases. However, it was shown by Geman and Geman (1984) that if the cooling is sufficiently slow then the method is guaranteed to reach the global optimum. Unfortunately, to determine the optimum cooling rate is not a straightforward problem and even if found it may be too slow to be of use to the MCMC practitioner. Geman and Geman (1984) encountered this problem in their image restoration application where they found that the theoretical cooling rate that would guarantee the correct MAP estimate was too slow to be used in practice.

SA is summarised in Algorithm A.7 of Appendix A.

This method has inspired temperature-based MC and MCMC methods, such as population annealing, parallel tempering and equi-energy sampling, all of which will be discussed in Section 1.4.3.

1.4.3 Methods that Use Multiple Temperatures

The following methods all have the same motivation as simulated annealing. That is, they all take advantage of a range of temperatures to enable more effective exploration of the state space. However, the objective of these methods is not optimisation of a function but rather the generation of samples from a

probability distribution. The three methods reviewed in this section are the MCMC approaches of equi-energy sampling and parallel tempering and the MC method of population annealing.

Parallel Tempering

A method that has seen much use in estimating quantities of interest, including in spin glass research (Contucci and Giardinà, 2013, pp. 165), is that of parallel tempering (PT), which was developed independently by Geyer (1991) and Hukushima and Nemoto (1996). PT involves the use of multiple simultaneous instances of the same canonical target distribution $\pi_T(x) = \exp[-h(x)/T]$ simulated at different temperatures.

The use of multiple instances or "replicas" was pioneered by Swendsen and Wang (1986) as a means to accelerate simulation of spin glasses, an important application in statistical mechanics (cf. Section 2.3). They achieved this acceleration by coupling the replicas using an appropriate scheme, and justified their use by noting that in the statistical mechanics literature, functions of interest need to be evaluated at multiple temperatures. Just as in SA, the higher-temperature replicas enable faster exploration of the state space, while the lower-temperature replicas highlight finer details of the target.

PT was formalised as *Metropolis-coupled Markov chain Monte Carlo* by Geyer (1991), for maximum-likelihood estimation and independently as *exchange Monte Carlo* by Hukushima and Nemoto (1996), for simulating disordered systems.

Instead of coupling replicas explicitly, PT first simulates each one independently and then uses a "replica exchange" mechanism to potentially swap randomly-selected pairs of replicas. Geyer (1991) attempted the swap after each iteration, while Hukushima and Nemoto (1996) did so less frequently.

The replica exchange mechanism involves a Metropolis move with the general odds ratio given by Geyer (1991) in Equation (1.68) to determine the probability of swapping thermally-neighbouring replicas n and m,

$$p_{\text{swap}}(n, m) = \min\left\{1, \frac{\pi_n(x_m)}{\pi_n(x_n)} \cdot \frac{\pi_m(x_n)}{\pi_m(x_m)}\right\}, \tag{1.68}$$

where π_n and x_n represent the target density and the state, respectively, of replica n and π_m and x_m represent those of replica m. Hukushima and Nemoto (1996) suggested swapping neighbouring states since the probability of a swap diminishes with the temperature difference between them. They also suggested guidelines for determining the temperatures $T_0, \ldots, T_N$ of the $N + 1$ replicas.

PT produces $N + 1$ sampling chains, each with the stationary distribution equal to the target distribution, albeit at a different temperature. The form of the target distribution of the nth replica is the same as the modified target function of simulated annealing, given in Equation (1.66). Therefore, the model being simulated needs to be able to be expressed in terms of an energy function h.

PT is summarised in Algorithm A.8 of Appendix A.

The preceding method makes use of a temperature to flatten the *target* distribution. An interesting application of this approach is that of *prior parallel tempering* (PPT), which was introduced by van Havre et al. (2015) to aid in fitting

a finite mixture model to observed data. By assuming a mixture model, with kernel f, on a set $\mathbf{y} = \{y_n\}_{n=1}^N$ of observations according to Equation (1.69),

$$p(\mathbf{y}|\mathbf{w}, \boldsymbol{\theta}) = \prod_{n=1}^{N} \sum_{k=1}^{K} w_k f(y_n|\boldsymbol{\theta}_k), \tag{1.69}$$

their goal was to sample component weights $\mathbf{w} = \{w_k\}_{k=1}^K$ and component parameters $\boldsymbol{\theta} = \{\boldsymbol{\theta}_k\}_{k=1}^K$ from their posterior distribution $p(\mathbf{w}, \boldsymbol{\theta}|\mathbf{y})$, given the observations. To increase applicability of their approach, they employed PT on the prior distributions of the weights $\mathbf{w}$. They assumed N different prior distributions on the weights $\{p_n(\mathbf{w})\}_{n=1}^N$, with the same prior distributions on the other components $\boldsymbol{\theta}$, in order to produce N different posterior distributions for sampling. That is, instead of simulating a canonical posterior distribution at different temperatures, they assumed different temperatures on a canonical prior distribution of the weights.

Since the means of flattening is different from standard PT, a modified acceptance ratio must be used for replica exchange. van Havre et al. (2015) noted that the acceptance ratio only depends on the prior distributions $\{p_n\}_{n=1}^N$ on the weights $\mathbf{w} = \{w_k\}_{k=1}^K$. They gave the probability of exchanging the inferred weights w_n and w_{n+1} of replicas n and $n+1$, respectively, by Equation (1.70),

$$\alpha(n, n+1) = \min\left\{1, \frac{p_n\left(w_{n+1}\right) p_{n+1}\left(w_n\right)}{p_n\left(w_n\right) p_{n+1}\left(w_{n+1}\right)}.\right\} \tag{1.70}$$

Lastly, a similar approach to PT, which uses *random* temperatures on a *single* chain, instead of multiple temperatures on multiple chains, is that of *simulated tempering* (ST) (Marinari and Parisi, 1992), which was introduced as an extension of SA.

Population Annealing

Population annealing (PA) is a population MC method of Hukushima and Iba (2003), which applies a modified SA approach to a population of samples. The goal is to calculate averages of a canonical distribution that is parameterised by a quantity β.

To motivate their sampler, Hukushima and Iba (2003) chose this parameter to be the inverse temperature $\beta = 1/T$, so that the canonical distribution was the familiar Boltzmann distribution given in Equation (1.66) of Section 1.4.2. Parameterising by the inverse temperature, Hukushima and Iba (2003) used the form of the Boltzmann distribution given in Equation (1.71),

$$\pi_\beta(x) = \frac{1}{Z_\beta} \exp\left[-\beta h(x)\right], \tag{1.71}$$

where Z_β is the partition function at inverse temperature β.

PA performs a single sweep down a pre-chosen decreasing temperature range $T_1 > \cdots > T_K$. Equivalently, it performs a single sweep up the range of inverse temperatures $\beta_1 < \cdots < \beta_K$.

As a population MC approach, at each temperature index t, PA uses a collection of simultaneous samples $\left\{x_n^{(t)}\right\}_{n=1}^N$ with corresponding weights $\left\{w_n^{(t)}\right\}_{n=1}^N$.

These weights are all initially set equal to 1 and are recursively updated from the weights at the previous index using Equation (1.72),

$$w_n^{(t)} = w_n^{(t-1)} \exp\left\{-(\beta_t - \beta_{t-1})h\left[x_n^{(t)}\right]\right\}. \tag{1.72}$$

Hukushima and Iba (2003) called Equation (1.72) the "Neal-Jarzynksi" factor because it is based on an equality demonstrated by Jarzynski (Jarzynski, 1997b,a), which was then adapted by Neal (Neal, 2001) to MC sampling with temperature annealing. Hukushima and Iba (2003) noted that the approach of Neal (2001) was unsuccessful due to a highly-fluctuating weight factor, so they proposed that the weights be resampled only once every M temperature indices, where the probability of selecting population member n at temperature index t is given by Equation (1.73),

$$p_n^{(t)} = \frac{w_n^{(t)}}{\sum_{m=1}^{N} w_m^{(t)}}. \tag{1.73}$$

This approach is reminiscent of the resampling approach reviewed in Section 1.1.2. After a resampling step Hukushima and Iba (2003) prescribed that each $w_n^{(t)}$ be reset to 1.

The preceding approach is summarised in Algorithm A.9 of Appendix A.

At each inverse temperature β_t, Hukushima and Iba (2003) gave the formula for computing a canonical average $\langle f \rangle_{\beta_t} \equiv \bar{f}_{\beta_t}$ of a function $f(x)$ that is presented in Equation (1.74),

$$\bar{f}_{\beta_t} = \frac{\sum_{n=1}^{N} f\left[x_n^{(t)}\right] w_n^{(t)}}{\sum_{n=1}^{N} w_n^{(t)}}, \tag{1.74}$$

where t represents the index in the temperature ladder corresponding to inverse temperature β_t.

They also noted that the ratio Z_{β_t}/Z_{β_0} of the partition function at inverse temperature β_t to that at inverse temperature β_0 can be estimated directly from the empirical weights,

$$\frac{Z_{\beta_t}}{Z_{\beta_0}} \approx \frac{1}{N} \sum_{n=1}^{N} w_n^{(t)}.$$

Hukushima and Iba (2003) noted that without the weights or resampling their algorithm reduces to N instances of SA. Omitting only the resampling, their method reduces to the *fast-growth method* of Hendrix and Jarzynski (2001) or the method of Neal (2001). They also demonstrated that in the case of a three-dimensional Ising spin glass (cf. Section 2.3 for a discussion of spin glasses), their PA algorithm performed comparably to PT and that both algorithms performed significantly better than SA.

The population annealing algorithm of Hukushima and Iba (2003) bears a striking resemblance to the particle filter of Chopin (2002). However, a distinguishing feature of population annealing is the use of the temperature ladder, which allows the weights to be computed recursively from one temperature to the next.

Equi-Energy Sampling

The Equi-Energy Sampler (EES) was introduced by Kou et al. (2006) and is another method that makes use of a temperature ladder to improve sampling efficiency and state-space exploration. Like both PT and PA, it generates multiple sampling chains at different temperatures, to mitigate the effect of energy barriers. However, unlike PT, it was not developed directly from temperature considerations but energy ones. As such Kou et al. (2006) referred to temperature-motivated approaches as "temperature-domain" methods and their own energy-based approach as an "energy-domain" method.

They noted the duality between temperature and energy in a statistical mechanical system in the form of a Laplace transform pair of the partition function $Z(T)$ of a system at temperature T,

$$Z(T) = \int \exp\left[-\frac{h(x)}{T}\right] dx$$

and the density of states $\Omega(u)$ at energy u,

$$\Omega(u) \propto \int I_u\left[h(x)\right] dx,$$

where $h(x)$ is the Hamiltonian or energy of the system and $I_u(x)$ is the indicator function, which equals 1 when $x = u$.

Kou et al. (2006) associated with the sampling chains an increasing sequence of minimum energy truncation levels,

$$H_0 < H_1 < \cdots < H_N < H_{N+1} = \infty,$$

and an increasing sequence of temperatures

$$1 = T_0 < T_1 < \cdots < T_N < T_{N+1} = \infty,$$

with the minimum energy H_0 of the lowest-temperature chain being at most the lowest possible energy of the system,

$$H_0 \leq \inf_x h(x).$$

EES simulates each chain n from its corresponding Boltzmann distribution, given in Equation (1.75),

$$\pi_n(x) \propto \exp\left[-\frac{h_n(x)}{T_n}\right], \tag{1.75}$$

where the value of the Hamiltonian $h_n(x)$ of the nth chain is restricted to be at least H_n,

$$h_n(x) = \max\{h(x), H_n\}.$$

By considering the energy of each sample, as well as the energy ladder, EES partitions the state space into energy bands $D_0, \ldots, D_N$, where the nth band contains only those samples x with energy $h(x)$ between H_n and H_{n+1}. The estimated bands $\hat{D}_0, \ldots, \hat{D}_N$, generated during simulation, are further partitioned based on the chain from which the samples are generated. Specifically, a

sample with energy between H_k and H_{k+1} simulated from chain n is assigned to estimated energy band $\hat{D}_k^{(n)}$. Kou et al. (2006) referred to this energy band as the nth "order energy ring".

Unlike other parallel update algorithms, EES does not start all sampling chains at the same time. Instead, as a simulation progresses the chains are started in sequence from the highest temperature to the lowest temperature and, once a chain has started, it continues to run for the rest of the simulation.

The specific approach outlined by Kou et al. (2006) involves starting chain $n = N$ (the one at the highest temperature) and sampling from its corresponding Boltzmann distribution using a standard sampling scheme. After an initial burn-in period of duration B, it begins construction of the Nth order energy rings. After a further pre-chosen number of iterations R the sampler then starts sampling the $(N-1)$th chain.

For chains $n = N-1$ down to $n = 0$ the sampler complements the standard MC chain updates with an "equi-energy" jump. Instead of updating every iteration with standard MC the sampler may perform an equi-energy jump with probability p_{ee}. When such a move is attempted, the sampler uniformly samples one of the states y in the estimated energy ring $\hat{D}_n^{(n+1)}$, at the energy one truncation level higher. The proposed state is then accepted to replace the current state x with the probability given in Equation (1.76),

$$\alpha = \min\left\{1, \frac{\pi_n(y)\pi_{n+1}(x)}{\pi_n(x)\pi_{n+1}(y)}\right\} \tag{1.76}$$

where π_n is the target density of chain n.

Just as in the case of the highest-temperate chain, there is a burn-in period before the construction of energy rings commences, and a minimum number of post burn-in samples before the next highest-energy chain is started.

EES is summarised in Algorithm A.10 of Appendix A.

Kou et al. (2006) illustrated their sampler by sampling from a mixture target of Liang and Wong (2001) (cf. Section 2.1.1), before using it to study three applications: a regression model (cf. Section 2.2); a motif-finding example in genetic sequences; and a simplified protein-folding example.

Of these three applications, of most interest to the current study will be the regression model, as it is the only one of the three that specifically targets a non-denumerable state space–the focus of the methods introduced in Part II of this dissertation.

Included in their results, Kou et al. (2006) provided a comparison to PT, finding that EES performed better than PT in terms of mixing times, autocorrelations, the number of local maxima visited and the means and mean squared estimates of the target.

Chapter 4 compares the results of sampling from the mixture target of Liang and Wong (2001) and the mixture exponential regression model to those produced by the sampler that is introduced in Chapter 3 of this dissertation.

1.5 Approximating Distributions by Finite Mixtures

In the context of independence sampling, given the need for a proposal distribution to closely resemble the target while also being possible to sample directly, it is prudent to consider how to construct such a proposal. One type of distribution that is appealing for approximating other distributions on p-dimensional Euclidean space $\mathbb{R}^p$, due to its flexibility, is a mixture of normal distributions (West, 1993).

Mixture models provide an obvious means of creating a proposal distribution that can easily be tuned, due to their inherent flexibility. With a sufficient number of mixture components they can capture the detail of a target with multiple local maxima by centering each component of the mixture near a local maximum of the distribution to be approximated. Obviously, the combined support of all mixture components should be equal to the support of the density/distribution that it approximates.

Due to their flexibility, mixture models can be made as complex as required. By increasing the number of components in the mixture, an arbitrary level of agreement with the target of interest can be achieved (Frühwirth-Schnatter and Pyne, 2010).

This section reviews methods and guidelines for constructing such approximations. Particular focus is paid to the case where the kernel of the mixture is a normal distribution, as this type of mixture will be important to the methods developed later in this dissertation.

1.5.1 Estimating Mixture Parameters by Clustering

One class of methods that can be used to construct mixtures of location-scale distributions, such at the normal distribution, is that of clustering. This subsection reviews three well-established approaches: expectation maximisation; k-means clustering; and k-harmonic means clustering, which is an extension of k-means clustering.

Expectation Maximisation

A well-established concept that can be used to cluster data is that of *expectation maximisation* (EM). This concept has been used in many settings, including more than just clustering, as discussed by Dempster et al. (1977). Their seminal paper collected many already-existing examples of EM and suggested a general framework for its implementation. Since the focus of the current section is on clustering methods, it does not review the general EM approach studied by Dempster et al. (1977) but rather this restricted example.

Dempster et al. (1977) let $\mathbf{y}$ be a vector of N observations, with each observation $n \in \{1, \ldots, N\}$ assumed to have a corresponding latent allocation to one of K components in an underlying mixture model. They also defined an $N \times K$ matrix containing the probabilities of each y_n belonging to each of the K components, given a current estimate of the parameters of the mixture. That is, each $\mathbf{z}_n$ is a vector of length K, with the kth entry indicating the probability that the nth observation originates from mixture component k.

Dempster et al. (1977) then noted that the goal is to maximise the complete-data log-likelihood function given in Equation (1.77),

$$\log \pi(\mathbf{y}, \mathbf{z}|\boldsymbol{\theta}) = \sum_{n=1}^{N} \mathbf{z}_n^{\mathrm{T}} \mathbf{U}(\mathbf{y}_n|\boldsymbol{\theta}) + \sum_{n=1}^{N} \mathbf{z}_n^{\mathrm{T}} \mathbf{V}(\boldsymbol{\theta}), \tag{1.77}$$

where $\mathbf{U}(\mathbf{y}|\boldsymbol{\theta})$ is the vector of the logarithms of the density at $\mathbf{y}$ under each of the K mixture components and $\mathbf{V}(\boldsymbol{\theta})$ is the vector of the logarithms of the conditional weight of each mixture component.

For general EM, Dempster et al. (1977) split the process into two alternating steps to be performed iteratively. The first step, which they referred to as the "E-step", keeps $\mathbf{y}$ and $\boldsymbol{\theta}$ fixed and computes the expected values of the allocations $\mathbf{z}$, given the current estimates of the parameters $\boldsymbol{\theta}^{(t)}$ at iteration t. Then, with $\mathbf{y}$ and $\mathbf{z}$ held fixed, the second step, which they called the "M-step", maximises Equation (1.77) for $\boldsymbol{\theta}$ and sets the solution as $\boldsymbol{\theta}^{(t+1)}$,

The preceding steps are repeated until some stopping criterion for the process is met. An example of such a criterion is a threshold change $\Delta_{\text{thresh}} = \log \pi[\mathbf{y}, \mathbf{z}|\boldsymbol{\theta}^{(t)}] - \log \pi[\mathbf{y}, \mathbf{z}|\boldsymbol{\theta}^{(t)}]$ between the log-likelihood at iterations t and $t-1$. Dempster et al. (1977) showed that, in general, the (log-)likelihood function used in an EM algorithm does not decrease with iteration. Therefore, such a stopping criterion is guaranteed to eventually be satisfied.

EM for finite mixtures is summarised in Algorithm A.11 of Appendix A.

Dempster et al. (1977) noted that the preceding approach will, in general, find *one* parameter set that *locally* maximises $\log \pi(\mathbf{y}, \mathbf{z}|\boldsymbol{\theta})$. However, by starting multiple instances of the algorithm in different parts of the parameter space $\mathcal{T}$, it is possible to find more than one such solution (Gelman et al., 2004, p. 319).

Dempster et al. (1977) also suggested replacing the maximisation step with a step that merely finds a value of $\boldsymbol{\theta}^{(t+1)}$ for which the log-likelihood of the complete-data is higher than it is for $\boldsymbol{\theta}^{(t)}$, thereby alleviating the computational burden at each step.

The current dissertation is concerned with mixtures of normal distributions, for which Dempster et al. (1977) reviewed a number of important cases. In particular, they noted that, in the case where the latent allocations are assumed to be i.i.d., Hasselblad (1966) provided equations for performing iterative EM. His paper considered a truncation approach, a steepest descent method and Newton's method. Dempster et al. (1977) also noted that the work of Hasselblad (1966), which was developed in the context of univariate mixtures, was extended by Wolfe (1970) to multivariate ones.

Considering the significant role of Markov chains in the current work, it is also worth noting that Dempster et al. (1977) reviewed a case where the latent allocations are assumed to evolve as a Markov process. In that case, which was developed by Baum and Eagon (1967), Baum et al. (1970) and Baum (1972), Dempster et al. (1977) noted that the complete-data log-likelihood in Equation (1.77) must be replaced by the one in Equation (1.78), in order to account for the Markovian structure,

$$\log \pi(\mathbf{y}, \mathbf{z}|\boldsymbol{\theta}) = \sum_{n=1}^{N} \mathbf{z}_n^{\mathrm{T}} \mathbf{U}(\mathbf{y}_n|\boldsymbol{\theta}) + \sum_{n=1}^{N} \mathbf{z}_n^{\mathrm{T}} \mathbf{V}(\boldsymbol{\theta}) \mathbf{z}_{n-1}, \tag{1.78}$$

where $\mathbf{V}^*(\boldsymbol{\theta})$ is a Markov transition matrix for the $\mathbf{z}_n$ and $\mathbf{z}_0$ is a vector of probabilities of the Markov chain starting in each component of the mixture.

k-Means Clustering

Another approach to clustering involves partitioning a state space, containing N points, into K disjoint regions. Steinhaus (1956) presented a note on such an idea, with the clusters having centres $\mathbf{q}_k, k \in \{1, \ldots, K\}$. The goal of this problem was to find an optimal choice of regions B_k and points $\mathbf{q}_k$ of the state space to minimise Equation (1.79),

$$S\left[\{(B_k, \mathbf{q}_k)\}_{k=1}^{K}\right] = \sum_{k=1}^{K} I\left(B_k, \mathbf{q}_k\right), \tag{1.79}$$

where the function $I(C, \mathbf{d})$ represents the moment of inertia of a body C about a point $\mathbf{d}$. Steinhaus (1956) showed that there exists such a solution, for which neighbouring centres are equidistant from the mutual boundary of their corresponding regions and that the geometric centres of the regions are also their respective centres of mass. The resulting partition is a Voronoi (Voronoi, 1908) or Dirichlet (Dirichlet, 1850) tessellation of convex subsets.

The following year Lloyd (1957) presented an algorithm to Bell Laboratories, describing an iterative method to produce such a solution in the context of pulse code modulation (PCM), which he eventually published (Lloyd, 1982). At the Eastern North American Region (ENAR) Spring Meeting of the International Biometric Society, Forgy (1965) independently presented a similar approach for the case where the centres are chosen from a fixed set of input data, rather than $\mathbb{R}$.

The basic k-means clustering method of Lloyd (1982) and Forgy (1965) involves iterating through two steps: updating region boundaries; and updating region centres. Starting with an initial arbitrary collection of regions $B_k, k \in \{1, \ldots, K\}$ the algorithm first computes their centres of mass, where the centre of mass of the kth region is given by Equation (1.80),

$$\mathbf{q}_k = \frac{\int_{B_k} \mathbf{y} dF(\mathbf{y})}{\int_{B_k} dF(\mathbf{y})}. \tag{1.80}$$

Here $F(\cdot)$ represents the probability measure associated with the density function of the collection of regions. In the case of a finite number N of equally-weighted samples $\mathbf{y}_n, n \in \{1, \ldots, N\}$ in p-dimensional Euclidean space, treated by Forgy (1965) $\mathbf{q}_k$ is simply the arithmetic mean,

$$\mathbf{q}_k = \frac{\sum_{\mathbf{y} \in B_k} \mathbf{y}}{N}.$$

Once the $\mathbf{q}_k$ have been computed, which Lloyd (1982) noted will generally not be the geometric means of the regions, the region boundaries are then updated. For the scalar approach on $\mathbb{R}$, treated by Lloyd (1982), the method to update the boundaries is the same as that proposed by Steinhaus (1956), namely, setting region boundaries between neighbouring regions to be the points halfway between their respective means. The result of such boundaries is that a point in the state space $\mathbf{y}$ will be assigned to the region to whose centre they are closest. That is, a data point $\mathbf{y}$ is placed in the region k for which Equation (1.81) is satisfied,

$$m(\mathbf{y}, \mathbf{q}_k) \leq m(\mathbf{y}, \mathbf{q}_j), \forall j \in \{1, \ldots, K\}, \tag{1.81}$$

for some metric $m(\cdot,\cdot)$. When Equation (1.81) is satisfied by more than one j then a tie-break procedure must be performed to select the appropriate cluster membership.

This process is then repeated until the region centres do not change more than some given threshold ϵ_{thresh}. That is, until

$$m\left[\mathbf{q}_k^{(t)}, \mathbf{q}_k^{(t-1)}\right] \leq \epsilon_{\text{thresh}}, \forall k \in \{1,\ldots,K\}, \tag{1.82}$$

where t is the iteration number.

Lloyd (1982) showed that at each iteration the sequence $[\mathbf{B}^{(t)}, \mathbf{q}^{(t)}]_{t=1}^{\infty}$ of regions $\mathbf{B}^{(t)} = \{B_1^{(t)},\ldots,B_K^{(t)}\}$ and their centres $\mathbf{q}^{(t)} = \{\mathbf{q}_1^{(0)},\ldots,\mathbf{q}_K^{(t)}\}$ converges to some $(\mathbf{B}', \mathbf{q}')$ that locally minimises Equation (1.79) for the PCM problem, provided that such a minimum exists. This means that the Equation (1.82) will eventually be satisfied.

The k-means method is summarised in Algorithm A.12 of Appendix A.

MacQueen (1967) discussed applications of the approach presented by Forgy (1965) to general distributions, including the use of mixtures of normal distributions to approximate the distributions of the input data. He referred to these algorithms as "k-means", due to the fact that they all involve finding the means of K clusters.

k-Harmonic Means Clustering

An extension of k-means clustering is that of k-harmonic means clustering, introduced by Zhang et al. (1999) and generalised by Zhang (2000) for partitioning a collection of N samples $\{\mathbf{y}_n\}_{n=1}^N$ into K clusters with centres $\{\mathbf{q}_k\}_{k=1}^K$ in a p-dimensional state space $\mathcal{X}$. Unlike the k-means approach, k-harmonic means clustering does not assign each sample to one of the clusters. Instead, each point is simultaneously considered to be a member of all the clusters. As such, k-harmonic means clustering does not produce a Voronoi tessellation, with disjoint regions, but a set of cluster centres, which, together, describe *all* data points.

Like k-means and EM, k-harmonic means uses a function to assess the goodness of the clustering. Zhang et al. (1999) and Zhang (2000) gave this *performance function* is given in Equation (1.83),

$$\text{Perf}\left(\{\mathbf{y}_n\}_{n=1}^N, \{\mathbf{q}_k\}_{k=1}^K\right) = \sum_{n=1}^{N} \frac{K}{\sum_{k=1}^{K}\left[m\left(\mathbf{y}_n, \mathbf{q}_k\right)\right]^{-a}}, \tag{1.83}$$

where $m(\cdot,\cdot)$ is a suitable metric on $\mathcal{X}$ and a is an appropriately-chosen power of it.

To update cluster centres $\{\mathbf{q}_k\}_{k=1}^K$, Zhang (2000) gave the recursive formula presented in Equation (1.84),

$$\mathbf{q}_k^{(t)} = \frac{\sum_{n=1}^{N}\left\{m\left[\mathbf{y}_n, \mathbf{q}_k^{(t-1)}\right]\right\}^{-a-2}\left(\sum_{l=1}^{K}\left\{m\left[\mathbf{y}_n, \mathbf{q}_l^{(t-1)}\right]\right\}^{-a}\right)^{-2}\mathbf{y}_n}{\sum_{n=1}^{N}\left\{m\left[\mathbf{y}_n, \mathbf{q}_k^{(t-1)}\right]\right\}^{-a-2}\left(\sum_{l=1}^{K}\left\{m\left[\mathbf{y}_n, \mathbf{q}_l^{(t-1)}\right]\right\}^{-a}\right)^{-2}}. \tag{1.84}$$

To make Equation (1.84) easier to manage, Zhang et al. (1999) and Zhang (2000) suggested updating the centres using the sequence of formulae given in Equations (1.85), (1.86), (1.87), (1.88) and (1.89), which produce the same result.

$$d_n^{(\min)} = \min_{k \in \{1,\ldots,K\}} \left\{ m\left[\mathbf{y}_n, \mathbf{q}_k^{(t-1)}\right] \right\}, \tag{1.85}$$

$$b_k^{(n)} = \frac{\left[d_n^{(\min)}\right]^{a-2} \left\{ \dfrac{d_n^{(\min)}}{m\left[\mathbf{y}_n, \mathbf{q}_k^{(t-1)}\right]} \right\}^{a+2}}{\left(\sum_{k=1}^{K} \left\{ \dfrac{d_n^{(\min)}}{m\left[\mathbf{y}_n, \mathbf{q}_k^{(t-1)}\right]} \right\}^{a} \right)^2} \tag{1.86}$$

$$b_k = \sum_{n=1}^{N} b_k^{(n)}, \tag{1.87}$$

$$c_k^{(n)} = \frac{b_k^{(n)}}{b_k}, \tag{1.88}$$

$$\mathbf{q}_k^{(t)} = \sum_{n=1}^{N} c_k^{(n)} \mathbf{y}_n. \tag{1.89}$$

As in the case of k-means clustering, the algorithm stops once the change in the performance function from iteration $t-1$ to t drops below some threshold level Δ_{thresh}. That is, once

$$\begin{aligned} \Delta\text{Perf} &= \left| \text{Perf}\left(\{\mathbf{y}\}_{n=1}^{N}, \left\{ \mathbf{q}_k^{(t-1)} \right\}_{k=1}^{K} \right) - \text{Perf}\left(\{\mathbf{y}\}_{n=1}^{N}, \left\{ \mathbf{q}_k^{(t)} \right\}_{k=1}^{K} \right) \right| \\ &\leq \Delta_{\text{thresh}}. \end{aligned}$$

Zhang et al. (1999) and Zhang (2000) prescribed that k-harmonic means algorithm be repeated until the performance function stabilises, implying that a threshold change in performance from one iteration to the next must eventually be met.

The general k-harmonic means algorithm of Zhang et al. (1999) and Zhang (2000) is given in Algorithm A.13 of Appendix A.

Zhang et al. (1999) and Zhang (2000) compared the k-harmonic means method to the k-means and EM approaches, finding that, for the experimental runs that they tested, k-harmonic means outperformed both of the other approaches. This claim was further supported by Hamerly and Elkan (2002), who introduced two more versions of k-harmonic means clustering.

The k-harmonic means approach was used by Giordani and Kohn (2010) to construct an adaptive MH sampler that will be reviewed in Section 1.6.4.

1.5.2 Sampling-Based Approaches

Given the interest of the current study in MCMC sampling, it is natural to consider how such techniques may be used to construct mixture distributions. As it turns out, there exist methods for achieving just this goal. The current section considers two of them.

Adaptive Mixture Refinement

A method introduced by West (1993) that uses MC to infer the parameters of a mixture approximation to a target $\pi(\mathbf{x})$ is that of *adaptive mixture refinement* (AMR). This approach uses an adaptive importance sampling scheme to successively "home in" on a refined kernel density estimate of $\pi(\mathbf{x})$.

As in previous sections, the focus here is on approximations that use a multivariate normal kernel. In this context, AMR starts with an initial overdispersed normal mixture approximation $g^{(0)}\left[\mathbf{x}\middle|\left\{w_k^{(0)},\boldsymbol{\mu}_k^{(0)}\right\}_{k=1}^{K^{(0)}},\boldsymbol{\Sigma}^{(0)}\right]$ to $f(\mathbf{x})$, as given by Equation (1.90),

$$\pi(\mathbf{x}) \approx g^{(0)}\left[\mathbf{x}\middle|\left\{w_k^{(0)},\boldsymbol{\mu}_k^{(0)}\right\}_{k=1}^{K^{(0)}},\boldsymbol{\Sigma}^{(0)}\right] = \sum_{k=1}^{K^{(0)}} w_k^{(0)}\mathcal{N}\left(\mathbf{x}\middle|\boldsymbol{\mu}_k^{(0)},\boldsymbol{\Sigma}^{(0)}\right), \quad (1.90)$$

where $w_k^{(t)}$ and $\boldsymbol{\mu}_k^{(t)}$ are the weight and mean, respectively, of component k and $\boldsymbol{\Sigma}^{(t)}$ is a variance-covariance matrix, which is common to all components. To compute the weight of the kth component of refinement t, West (1993) used the importance weight of Geweke (1989) at the component's mean, as demonstrated by Equation (1.91),

$$w_k^{(t)} = \frac{\pi\left[\boldsymbol{\mu}_k^{(t)}\right]}{g^{(t-1)}\left[\mathbf{x}_k^{(t)}\right]\left\{\sum_{k=1}^{K^{(t)}}\frac{\pi\left[\boldsymbol{\mu}_k^{(t)}\right]}{g^{(t-1)}\left[\boldsymbol{\mu}_k^{(t)}\right]}\right\}}. \quad (1.91)$$

West (1993) then prescribed following the Bayesian importance sampler of Geweke (1989) (cf. Section 1.1.2) with importance function $g^{(0)}\left[\mathbf{x}\middle|\left\{w_k^{(0)},\boldsymbol{\mu}_k^{(0)}\right\}_{k=1}^{K^{(0)}},\boldsymbol{\Sigma}^{(0)}\right]$ to produce $K^{(1)}$ samples $\left\{\mathbf{x}_k^{(1)}\right\}_{k=1}^{K_1}$, whose weighted average approximates the expected value of $\pi(\mathbf{x})$. Kernel density estimation is then performed on these samples to produce a refined approximation $g^{(1)}\left[\mathbf{x}\middle|\left\{w_k^{(1)},\boldsymbol{\mu}_k^{(1)}\right\}_{k=1}^{K^{(1)}},\boldsymbol{\Sigma}^{(1)}\right]$. If this refined approximation is insufficiently accurate, then the process is repeated, using this new approximation as an updated importance function. In general, at refinement t, this kernel density estimate, is given by Equation (1.92),

$$g^{(t)}\left[\mathbf{x}\middle|\left\{w_k^{(t)},\boldsymbol{\mu}_k^{(t)}\right\}_{k=1}^{K^{(t)}},\boldsymbol{\Sigma}^{(t)}\right] = \sum_{k=1}^{K_t} w_k^{(t)}\mathcal{N}\left(\mathbf{x}\middle|\boldsymbol{\mu}_k^{(t)},\boldsymbol{\Sigma}_k^{(t)}\right), \quad (1.92)$$

where $\left\{\boldsymbol{\mu}_k^{(t)}\right\}_{k=1}^{K^{(t)}}$ and $\boldsymbol{\Sigma}^{(t)}$ are appropriately computed from $\{\mathbf{x}_k^{(t)}\}_{k=1}^{K_t}$ West (1993) suggested that the common matrix $\boldsymbol{\Sigma}^{(t)}$ be an appropriately-scaled estimate $\hat{\boldsymbol{\Sigma}}$ of the true variance-covariance matrix $\boldsymbol{\Sigma}$ of $\pi(\mathbf{x})$, defined according to Equation (1.93),

$$\boldsymbol{\Sigma}^{(t)} = h^{(t)}\hat{\boldsymbol{\Sigma}}, \quad (1.93)$$

where $h^{(t)}$ is a "smoothing parameter".

West (1993) noted that a typically-used scale for a p-dimensional $\mathbf{x}$ at refinement t is the form of $h^{(t)}$ given in Equation (1.94),

$$h^{(t)} = \left[\frac{4}{K^{(t)}(1+2p)}\right]^{1/(1+4p)}. \tag{1.94}$$

Multiplying $h^{(t)}$ by the MC estimate of the variance-covariance matrix of $\pi(\mathbf{x})$ gives Equation (1.95),

$$\boldsymbol{\Sigma}^{(t)} = \sum_{k=1}^{K^{(t)}} w_k^{(t)} \left[\mathbf{x}_k^{(t)} - \bar{\mathbf{x}}^{(t)}\right]\left[\mathbf{x}_k^{(t)} - \bar{\mathbf{x}}^{(t)}\right]^{\mathrm{T}}. \tag{1.95}$$

Here $\bar{\mathbf{x}}^{(t)}$ is the MC mean of refinement t, which is given by Equation (1.96),

$$\bar{\mathbf{x}}^{(t)} = \sum_{k=1}^{K^{(t)}} w_k^{(t)} \mathbf{x}_k^{(t)}. \tag{1.96}$$

West (1993) noted that the resulting refinement $g^{(t)}\left[\mathbf{x}\middle|\left\{w_k^{(t)}, \boldsymbol{\mu}_k^{(t)}\right\}_{k=1}^{K^{(t)}}, \boldsymbol{\Sigma}^{(t)}\right]$ is always more disperse than $\pi(\mathbf{x})$. For an initial approximation (refinement 0) this property may be acceptable but the ultimate goal is to obtain an approximation that is close to the target being approximated. For this reason West (1993) used shrinkage to correct the overdispersion of the refinement in his examples, so that their variances were equal to those of their targets. To achieve this goal, he used the approach of West (1990), by setting the mean in the expression for refinement t, given by Equation (1.92), according to Equation (1.97),

$$\boldsymbol{\mu}_k^{(t)} = \mathbf{x}_k^{(t)}\sqrt{1 - \left[h^{(t)}\right]^2} + \bar{\mathbf{x}}^{(t)}\left\{1 - \sqrt{1 - \left[h^{(t)}\right]^2}\right\}. \tag{1.97}$$

This choice results in component means that are closer to the target distribution's mean than they would otherwise be, thereby preserving the overall variance of the target.

To determine when to stop refinement of the approximation it is necessary to have some measure of change in the approximation. For this purpose West (1993) suggested plotting a time-series of the variance of weights (the relative numerical efficiency of Geweke (1989)) or using the entropy relative to uniformity, which is given in Equation (1.98),

$$H\left[\left\{w_k^{(t)}\right\}_{k=1}^{K^{(t)}}\right] = -\sum_{k=1}^{K^{(t)}} w_k^{(t)} \frac{\log\left[w_k^{(t)}\right]}{\log\left[K^{(t)}\right]}. \tag{1.98}$$

Once Equation (1.98) remains below some threshold, then refinement stops.

AMR is summarised in Algorithm A.14 of Appendix A.

Bayesian Estimation with Data Augmentation

An example of using MCMC for approximating a target by fitting mixtures of skew-normal or skew-t distributions was given by Frühwirth-Schnatter and Pyne (2010), who used a Gibbs-like approach to infer the parameters of the mixture.

The most general kernel that they considered was that of a p-variate skew-t, with density in the form given by Azzalini and Capitanio (2003). This density is presented in Equation (1.99),

$$\mathcal{ST}(\mathbf{y}|\boldsymbol{\xi},\boldsymbol{\Omega},\boldsymbol{\alpha},\nu) = 2t_p(\mathbf{y}|\boldsymbol{\xi},\boldsymbol{\Omega},\nu)T_{p+\nu}\left\{\boldsymbol{\alpha}^{\mathrm{T}}\left[\mathrm{Diag}(\boldsymbol{\Omega})\right]^{-\frac{1}{2}}(\mathbf{y}-\boldsymbol{\xi})\right. \\ \left.\times\sqrt{\frac{\nu+p}{\nu+(\mathbf{y}-\boldsymbol{\xi})^{\mathrm{T}}\boldsymbol{\Omega}^{-1}(\mathbf{x}-\boldsymbol{\xi})}}\right\}, \tag{1.99}$$

where $t_p(\cdot|\boldsymbol{\xi},\boldsymbol{\Omega})$ is the PDF of a p-variate Student's t-distribution with location $\boldsymbol{\xi}$, scale matrix $\boldsymbol{\Omega}$ and degrees of freedom ν, $T_{p+\nu}(\cdot)$ is the CDF of the univariate standard Student's t-distribution with $p+\nu$ degrees of freedom, $\boldsymbol{\alpha}$ is a skewness vector of dimension p and $\mathrm{Diag}(\boldsymbol{\Omega})$ is a diagonal matrix with diagonal entries equal to those of the variance-covariance matrix $\boldsymbol{\Omega}$.

Azzalini and Capitanio (2003) decomposed such a p-variate skew-t-distributed random variable $\mathbf{Y}$ using Equation (1.100),

$$\mathbf{Y} = \boldsymbol{\xi} + \frac{\mathbf{Z}}{\sqrt{V}}, \tag{1.100}$$

where $V \sim \chi^2_\nu/\nu$ and $\mathbf{Z}$ is a zero-located p-variate truncated-normal random variable on $[0,\infty)^p$.

Frühwirth-Schnatter and Pyne (2010) also considered skew-normal mixtures, whose kernel was given by Azzalini and Dalla Valle (1996) according to Equation (1.101),

$$\mathcal{SN}(\mathbf{y}|\boldsymbol{\xi},\boldsymbol{\Omega},\boldsymbol{\alpha}) = 2\mathcal{N}(\mathbf{y}|\boldsymbol{\xi},\boldsymbol{\Omega})\Phi\left\{\boldsymbol{\alpha}^{\mathrm{T}}\left[\sqrt{\mathrm{Diag}(\boldsymbol{\Omega})}\right]^{-1}(\mathbf{y}-\boldsymbol{\xi})\right\}, \tag{1.101}$$

where $\Phi(\cdot)$ is the CDF of a univariate standard normal random variable.

As a χ^2_ν random variate, V can be obtained by first noting the relation of a χ^2_ν distribution to a gamma distribution, given in Equation (1.102) (Gelman et al., 2004, pp. 580),

$$\chi^2_\nu(x) = \mathrm{Gamma}\left(x\middle|\frac{\nu}{2},\frac{1}{2}\right). \tag{1.102}$$

Then an appropriate specialised gamma generator, such as one of those discussed in Devroye (1986, Chapter 9, Section 3), may be used to sample from the gamma distribution before transforming the result using Equation (1.102) to obtain V.

To enable posterior inference of the parameters in Equations (1.99) and (1.101), Frühwirth-Schnatter and Pyne (2010) expressed them as random effects models, according to Equation (1.103),

$$\mathbf{Y} = \boldsymbol{\xi} + \boldsymbol{\psi}\mathbf{Z} + \epsilon, \tag{1.103}$$

where $\epsilon \sim \mathcal{N}(\cdot\,|\mathbf{0}_p,\boldsymbol{\Sigma})$ for a skew-normal model and $\epsilon \sim \mathcal{N}(\cdot\,|\mathbf{0}_p,\boldsymbol{\Sigma}/b)$ for a skew-t one. Here $\mathbf{0}_p$ is the p-dimensional zero vector and b is a scaling factor that depends on the degrees of freedom of the skew-t distribution.

This approach leads to a reparameterisation of the skew-normal and skew-t distributions in terms of $(\boldsymbol{\xi},\boldsymbol{\psi},\boldsymbol{\Sigma})$ and $(\boldsymbol{\xi},\boldsymbol{\psi},\boldsymbol{\Sigma},b)$, respectively. The location

parameter $\boldsymbol{\xi}$ is the same as in the expressions of Azzalini and Dalla Valle (1996) and Azzalini and Capitanio (2003).

Frühwirth-Schnatter and Pyne (2010) noted that the scaling factor b is gamma-distributed, according to Equation (1.104),

$$b \sim \text{Gamma}\left(\frac{\nu}{2}, \frac{\nu}{2}\right). \tag{1.104}$$

Using the preceding transformations, Frühwirth-Schnatter and Pyne (2010) inferred the parameters $(\boldsymbol{\xi}, \boldsymbol{\psi}, \boldsymbol{\Sigma}, \nu)$ in their study.

With these reparameterisations, Frühwirth-Schnatter and Pyne (2010) considered the mixtures of normal and skew-t distributions given in Equations (1.105) and (1.106), respectively,

$$p_{\mathcal{SNM}}(\mathbf{y}|\boldsymbol{\xi}, \boldsymbol{\Sigma}, \boldsymbol{\alpha}) = \sum_{k=1}^{K} d_k \mathcal{SN}(\mathbf{y}|\boldsymbol{\xi}_k, \boldsymbol{\Sigma}_k, \boldsymbol{\alpha}_k). \tag{1.105}$$

$$p_{\mathcal{STM}}(\mathbf{y}|\boldsymbol{\xi}, \boldsymbol{\Sigma}, \boldsymbol{\alpha}, \boldsymbol{\nu}) = \sum_{k=1}^{K} d_k \mathcal{ST}(\mathbf{y}|\boldsymbol{\xi}_k, \boldsymbol{\Sigma}_k, \boldsymbol{\alpha}_k, \nu_k), \tag{1.106}$$

where

$$\begin{aligned}
\boldsymbol{\xi} &= \{\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_K\}, \\
\boldsymbol{\psi} &= \{\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_K\}, \\
\boldsymbol{\Sigma} &= \{\boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_K\}, \\
\boldsymbol{\alpha} &= \{\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_K\}, \\
\boldsymbol{\nu} &= \{\nu_1, \ldots, \nu_K\},
\end{aligned}$$

are vectors of the transformed parameters for each component k and d_k are the corresponding component weights.

They used the data augmentation approach of Diebolt and Robert (1994) in a blocked Gibbs sampler, to iteratively update the parameters in Equations (1.105) and (1.106), given a fixed set of observations $\mathbf{y} = (\mathbf{y}_1, \ldots, \mathbf{y}_N)$ from their underlying distribution $\pi(\mathbf{y})$. This involved introducing latent indicator variables $\mathbf{S} = (S_1, \ldots, S_N)$ for the observations, indicating the (unknown) mixture component with which each is associated. The vectors $\mathbf{z} = (z_1, \ldots, z_N)$ and $\mathbf{b} = (b_1, \ldots, b_N)$ describe the (unobserved) allocations and random effects, respectively, for the observations. Given a suitable prior distribution,

$$p(\boldsymbol{\xi}, \boldsymbol{\psi}, \boldsymbol{\Sigma})$$

on the parameters that are common to both distributions, Frühwirth-Schnatter and Pyne (2010) noted that closed-form posterior distributions,

$$p_{\mathcal{SNM}}\left(\boldsymbol{\xi}, \boldsymbol{\psi}, \boldsymbol{\Sigma}, \boldsymbol{\alpha} \,|\mathbf{d}, \mathbf{S}, \mathbf{z}, \mathbf{y}\right),$$

and

$$p_{\mathcal{STM}}\left(\boldsymbol{\xi}, \boldsymbol{\psi}, \boldsymbol{\Sigma}, \boldsymbol{\alpha} \,|\mathbf{d}, \mathbf{S}, \mathbf{z}, \mathbf{b}, \mathbf{y}\right),$$

given all other variables, exist in closed form.

By sampling from such a posterior, estimates of the skew-t mixture distribution's parameters are obtained. This, of course, requires transforming the transformed parameters back into the desired form. Each $\boldsymbol{\xi}_k$ is already in the form required. Frühwirth-Schnatter and Pyne (2010) related the scale $\boldsymbol{\Omega}$ and the skewness $\boldsymbol{\alpha}$ to these new variables according to Equations (1.107) and (1.108), respectively,

$$\boldsymbol{\Omega}_k = \boldsymbol{\Sigma}_k \boldsymbol{\psi}_k \boldsymbol{\psi}_k^{\mathrm{T}}, \tag{1.107}$$

$$\boldsymbol{\alpha}_k = \frac{1}{\sqrt{1 - \boldsymbol{\psi}_k^{\mathrm{T}} \boldsymbol{\Omega}_k^{-1} \boldsymbol{\psi}_k}} \sqrt{\mathrm{Diag}(\boldsymbol{\Omega}_k)} \boldsymbol{\Omega}_k^{-1} \boldsymbol{\psi}_k. \tag{1.108}$$

Frühwirth-Schnatter and Pyne (2010) summarised their method as two and three block Gibbs samplers for the skew-normal and skew-t mixtures, respectively. At iteration t, the first block for a skew-normal mixture is given by Equations (1.109) and (1.110), while the second block is given by Equation (1.111),

$$\mathbf{d}^{(t)} \sim p_{\mathcal{SNM}} \left[\cdot \middle| \mathbf{S}^{(t-1)}, \mathbf{z}^{(t-1)}, \mathbf{y} \right], \tag{1.109}$$

$$\boldsymbol{\xi}^{(t)}, \boldsymbol{\psi}^{(t)}, \boldsymbol{\Sigma}^{(t)} \sim p_{\mathcal{SNM}} \left[\cdot \middle| \mathbf{S}^{(t-1)}, \mathbf{z}^{(t-1)}, \mathbf{y} \right], \tag{1.110}$$

$$\mathbf{S}^{(t)}, \mathbf{z}^{(t)} \sim p_{\mathcal{SNM}} \left[\cdot \middle| \boldsymbol{\xi}^{(t)}, \boldsymbol{\psi}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \mathbf{d}^{(t)}, \mathbf{y} \right]. \tag{1.111}$$

The first block for a skew-t mixture is given by Equations (1.112) and (1.113), while the second is given by Equation (1.114).

$$\mathbf{d}^{(t)} \sim p_{\mathcal{ST}\mathcal{M}} \left[\cdot \middle| \mathbf{S}^{(t-1)}, \mathbf{z}^{(t-1)}, \mathbf{b}^{(t-1)}, \mathbf{y} \right], \tag{1.112}$$

$$\boldsymbol{\xi}^{(t)}, \boldsymbol{\psi}^{(t)}, \boldsymbol{\Sigma}^{(t)} \sim p_{\mathcal{ST}\mathcal{M}} \left[\cdot \middle| \mathbf{S}^{(t-1)}, \mathbf{z}^{(t-1)}, \mathbf{b}^{(t-1)}, \mathbf{y} \right], \tag{1.113}$$

$$\mathbf{S}^{(t)}, \mathbf{z}^{(t)} \sim p_{\mathcal{ST}\mathcal{M}} \left[\cdot \middle| \boldsymbol{\xi}^{(t)}, \boldsymbol{\psi}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \mathbf{d}^{(t)}, \mathbf{b}^{(t-1)}, \mathbf{y} \right]. \tag{1.114}$$

The third block consists of updating the degrees of freedom and the scaling factors. Frühwirth-Schnatter and Pyne (2010) noted that the posterior density to update the degrees of freedom $\boldsymbol{\nu}$ is not available in closed form and must be updated using an MH sampler inside the Gibbs sampler to generate $\boldsymbol{\nu}^{(t)}$ and $\mathbf{b}^{(t)}$, given all other parameters, observations and latent allocations.

Frühwirth-Schnatter and Pyne (2010) left the update equations arbitrary, which highlights that their forms depend on the chosen prior distributions.

The preceding approaches is summarised in Appendix A in Algorithm A.15 for skew-normal mixtures and Algorithm A.16 for skew-t mixtures.

1.5.3 Selecting the Number of Mixture Components

As the number of components in a mixture distribution increases so does its flexibility. Increasing the number of components allows an approximation in Euclidean space to more closely resemble the desired density/distribution (Frühwirth-Schnatter and Pyne, 2010). However, there may exist a point of diminishing return, where the rate of increase in improvement of the approximation is offset by the extra computational cost involved in using them (West, 1993). This section reviews an approach for decreasing the number of components in a kernel density estimate, as well as a collection of techniques for selecting a suitable number of components.

Collapsing Mixtures

Starting with an overfitted mixture approximation to a distribution, West (1990) demonstrated a method to cut down excess components in the mixture, for reasons of computational efficiency, without unduly harming the approximation. The idea behind the presented method was to merge the closest neighbouring mixture component to the one with the largest weight. West (1993) employed this approach in an iterative procedure to reduce an N-component mixture to an M-component one, reducing the current mixture at each step by one.

Without loss of generality, assume that the lowest-weighted component is indexed by $k = 1$ at step $t-1$. In the context of a normal kernel, the neighbour of a mixture component is the component, out of all other components, whose mean $\boldsymbol{\mu}_k^{(t-1)}$ minimises Equation (1.115),

$$\Delta_k^{(t-1)} = \left\| \boldsymbol{\mu}_1^{(t-1)} - \boldsymbol{\mu}_k^{(t-1)} \right\|. \tag{1.115}$$

Once the corresponding index i of the nearest neighbour has been found, the smallest component is replaced by setting its weight according to Equation (1.116),

$$w_1^{(t)} = w_1^{(t-1)} + w_i^{(t-1)}, \tag{1.116}$$

and its mean according to Equation (1.117),

$$\boldsymbol{\mu}_1^{(t)} = \frac{w_1^{(t-1)} \boldsymbol{\mu}_1^{(t-1)} + w_i^{(t-1)} \boldsymbol{\mu}_i^{(t-1)}}{w_1^{(t)}} \tag{1.117}$$

After collapsing the smallest component and its nearest neighbour, the indices of the parameter vector are updated. For components $k = 2$ to $i-1$ the new parameters are set according to Equation (1.118),

$$\left[w_k^{(t)}, \boldsymbol{\mu}_k^{(t)} \right] = \left[w_k^{(t-1)}, \boldsymbol{\mu}_k^{(t-1)} \right]. \tag{1.118}$$

and for components $k = i+1$ to $k = N-t$, the parameters are set according to Equation (1.119),

$$\left[w_k^{(t)}, \boldsymbol{\mu}_k^{(t)} \right] = \left[w_{k+1}^{(t-1)}, \boldsymbol{\mu}_{k+1}^{(t-1)} \right]. \tag{1.119}$$

This process effectively eliminates component i of the mixture at step t to reduce the approximation from one with $N-t+1$ components to one with $N-t$ components.

The method of West (1990) for collapsing normal mixtures is summarised in Algorithm A.17 of Appendix A.

This process was advocated by West (1993) in the context of adaptive mixture refinement, which was reviewed in Section 1.5.2. To know how many components should be used in the final mixture, West (1993) suggested monitoring the change in the approximation as the number of components is decreased and stopping the procedure before too large a change is observed. In order to reflect the change in the number of components, he also suggested that the smoothing parameter h be recalculated using Equation (1.94) with the new number of components.

Model Choice

Another approach for selecting the number of mixture components in a mixture approximation is that of using an information criterion for model selection. This strategy was advocated by Frühwirth-Schnatter and Pyne (2010), who explored its application to the assessment of their method for fitting skew-normal and skew-t distributions, which was reviewed in Section 1.5.2. They considered five different quantities.

The first quantity that they considered was the *Bayesian Information Criterion* (BIC), which was introduced by Schwarz (1978). In what follows, the component weights of a model are represented by $\mathbf{w} = (w_1, \ldots, w_K) \in \boldsymbol{\Delta}^K$ (where $\boldsymbol{\Delta}^K$ is the K-simplex in $\mathbb{R}^K$) and the component-specific parameters by $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K) \in \mathcal{T}$.

For the a fixed kernel type, the BIC of a K-component mixture is given by the expression in Equation (1.120), which follows the notation of Frühwirth-Schnatter and Pyne (2010),

$$\mathrm{BIC}_K = -2 \log p \left(\mathbf{y} \,\middle|\, \hat{\mathbf{w}}, \hat{\boldsymbol{\theta}}\,, K \right) + \phi_K \log N. \tag{1.120}$$

Here N is the number of observations, $\mathbf{y} = (\mathbf{y}_1, \ldots, \mathbf{y}_N)$ is the vector of those observations, $\hat{\mathbf{w}}$ and $\hat{\boldsymbol{\theta}}$ are the joint maximum-likelihood estimates (MLE) of the weights $\mathbf{w}$ and kernel parameters $\boldsymbol{\theta}$, respectively, in $\log p(\mathbf{y}|\mathbf{w}, \boldsymbol{\theta}, K)$ and

$$\phi_K = (2p+1)K - 1 + \frac{Kp(p+1)}{2} + K I_{\text{skew-}t},$$

where p is the dimension of the state space and $I_{\text{skew-}t} = 1$ if the mixture kernel is skew-t and 0 if it is normal.

The second quantity that they considered was the *Approximate Weight of Evidence* (AWE) for a model to have K clusters, introduced by Banfield and Raftery (1993). The AWE is given by Equation (1.121),

$$\mathrm{AWE}_K = \begin{cases} \sum_{k=1}^{K-1} E_k & K > 1, \\ 0, & K = 1, \end{cases} \tag{1.121}$$

where $c \in \mathbb{R}$ is a constant and E_k is an approximation of twice the negative of the natural logarithm of the Bayes factor, which represents the ratio of the odds of a model with k components to one with $k+1$ components. The Bayes Factor is defined according to Equation (1.122) (Banfield and Raftery, 1993),

$$\mathrm{Bayes}(k, k+1) = \frac{p(\mathbf{y}|K=k)}{p(\mathbf{y}|K=k+1)}. \tag{1.122}$$

As its name suggests, the AWE involves a number of approximations. As just stated, one of these approximations is the relationship between E_k and the Bayes Factor. Banfield and Raftery (1993) expressed this relationship according to Equation (1.123),

$$E_k = \lambda_k - 2\delta_k \left[\frac{3}{2} + \log(pN_{k,k+1}) \right] \approx -2 \log \mathrm{Bayes}(k, k+1), \tag{1.123}$$

where p is the dimension of the state space, λ_k is the asymptotically χ^2-distributed test statistic of the likelihood ratio of the model with k components

to the one with $k+1$ components and δ_k is the reduction in the number of parameters when going from the $k+1$-component model to the k-component one. $N_{k,k+1}$ is the number of observations in the component formed by merging two of the components in the $k+1$-component model to produce the k-component model.

The other approximation is for $K > 1$, for which Banfield and Raftery (1993) used Equation (1.124),

$$E_K \approx c + 2\log p(K|\mathbf{y}). \tag{1.124}$$

For the purposes of assessing their skew-normal and skew-t mixtures with latent allocations $\mathbf{S}$, Frühwirth-Schnatter and Pyne (2010) used the form of AWE given in Equation (1.125),

$$\text{AWE}_K = -2\log p\left(\mathbf{y}, \hat{\mathbf{S}} \middle| \hat{\boldsymbol{\theta}}\right) + 2\phi_K\left(\frac{3}{2} + \log N\right), \tag{1.125}$$

with N being the number of observations, ϕ_K defined as earlier and $(\hat{\mathbf{S}}, \hat{\boldsymbol{\theta}})$ being the combination of $\mathbf{S}$ and $\boldsymbol{\theta}$ that jointly maximises the complete-data log-likelihood or its logarithm, which is given in Equation (1.126),

$$(\hat{\mathbf{S}}, \hat{\boldsymbol{\theta}}) = \underset{(\mathbf{S},\boldsymbol{\theta})\in(\{1,\ldots,K\}^N,\mathcal{T})}{\arg\max} \sum_{n=1}^{N} \log\left[w_{S_n} p(\mathbf{y}_n|\boldsymbol{\theta}_{S_n})\right]. \tag{1.126}$$

The third criterion considered by Frühwirth-Schnatter and Pyne (2010) was the difference between *Integrated Completed/Classification Likelihood* (ICL) of Biernacki (2000) and the BIC. Biernacki (2000) defined the ICL for a generic mixture kernel f approximation of $\mathbf{y}$, according to Equation (1.127)

$$\text{ICL} = \log p\left(\mathbf{y} \middle| f, K, \hat{\mathbf{w}}, \hat{\boldsymbol{\theta}}\right) - \frac{g_f^{(K)}}{2}\log N, \tag{1.127}$$

where $\hat{\mathbf{w}}$ and $\hat{\boldsymbol{\theta}}$ are the joint MAP estimate of the weights $\mathbf{w}$ and parameters $\boldsymbol{\theta}$, respectively, and $g_f^{(K)}$ is the number of free parameters in a model with kernel type f and K mixture components.

Instead of computing this quantity directly, Frühwirth-Schnatter and Pyne (2010) used the approximation of McLachlan and Peel (2000, pp. 216) that is given in Equation (1.128),

$$\text{ICL} - \text{BIC}_K \approx 2\text{BIC}_K + \text{EN}\left[\hat{\mathbf{w}}, \hat{\boldsymbol{\theta}}, K\right], \tag{1.128}$$

where the second term on the right-hand side is the entropy, which Frühwirth-Schnatter and Pyne (2010) gave according to Equation (1.129) for the mixture model,

$$\begin{aligned}\text{EN}\left[\hat{\mathbf{w}}, \hat{\boldsymbol{\theta}}, K\right] = -\sum_{n=1}^{N}\sum_{k=1}^{K} \mathbb{P}\left[S_n = k \middle| \mathbf{y}_n \hat{\mathbf{w}}, \hat{\boldsymbol{\theta}}, K\right] \\ \times \log\left\{\mathbb{P}\left[S_n = k \middle| \mathbf{y}_n, \hat{\mathbf{w}}, \hat{\boldsymbol{\theta}}, K\right]\right\}.\end{aligned} \tag{1.129}$$

Finally, Frühwirth-Schnatter and Pyne (2010) studied two instances of the *Deviance Information Criterion* (DIC) of Spiegelhalter et al. (2002), whose general form, in the notation of the current section, is given by Equation (1.130),

$$\mathrm{DIC}_K = D\left(\tilde{\mathbf{w}}, \tilde{\boldsymbol{\theta}}\right) + 2p_D\left[\mathbf{y}, \boldsymbol{\Delta}^K, \mathcal{T}, \left(\tilde{\mathbf{w}}, \tilde{\boldsymbol{\theta}}\right)(\mathbf{y})\right]. \tag{1.130}$$

Spiegelhalter et al. (2002) referred to $D(\mathbf{w}, \boldsymbol{\theta})$ as the *Bayesian deviance* and defined it according to Equation (1.131),

$$D(\mathbf{w}, \boldsymbol{\theta}) = -2\log[p(\mathbf{y}|\mathbf{w}, \boldsymbol{\theta})] + 2\log[g(\mathbf{y})], \tag{1.131}$$

where $g(\mathbf{y})$ is some "fully-specified" function of the observations, which serves to standardise the posterior distribution of the weights $\mathbf{w} \in \boldsymbol{\Delta}^K$ and parameters $\boldsymbol{\theta} \in \mathcal{T}$, given the observations $\mathbf{y}$. Spiegelhalter et al. (2002) gave the form of p_D stated in Equation (1.132),

$$\begin{aligned} p_D\left[\mathbf{y}, \boldsymbol{\Delta}^K, \mathcal{T}, \left(\tilde{\mathbf{w}}, \tilde{\boldsymbol{\theta}}\right)(\mathbf{y})\right] &= \mathbb{E}_{\boldsymbol{\theta}|\mathbf{y}}\{-2\log[p(\mathbf{y}|\mathbf{w}, \boldsymbol{\theta})]\} \\ &\quad + 2\log\left\{p\left[\mathbf{y}\,\middle|\left(\tilde{\mathbf{w}}, \tilde{\boldsymbol{\theta}}\right)(\mathbf{y})\right]\right\}, \end{aligned} \tag{1.132}$$

where $\left(\tilde{\mathbf{w}}, \tilde{\boldsymbol{\theta}}\right)(\mathbf{y})$ is an appropriately-chosen estimator of the true parameters. Taking the estimator to be the sample mean, $\overline{(\mathbf{w}, \boldsymbol{\theta})}$, Spiegelhalter et al. (2002) showed that p_D may be expressed according to Equation (1.133),

$$p_D\left[\mathbf{y}, \boldsymbol{\Delta}^K, \mathcal{T}, \left(\tilde{\mathbf{w}}, \tilde{\boldsymbol{\theta}}\right)(\mathbf{y})\right] = \overline{D(\mathbf{w}, \boldsymbol{\theta})} + D\left[\overline{(\mathbf{w}, \boldsymbol{\theta})}\right], \tag{1.133}$$

with $\overline{D(\mathbf{w}, \boldsymbol{\theta})}$ representing the mean of $D(\mathbf{w}, \boldsymbol{\theta})$ over $(\mathbf{w}, \boldsymbol{\theta}) \in \left(\boldsymbol{\Delta}^K, \mathcal{T}\right)$.

Both instances of the DIC used by Frühwirth-Schnatter and Pyne (2010) were explicitly derived and evaluated by Celeux et al. (2006) for missing data problems, using the approach illustrated by Spiegelhalter et al. (2002). The first of these, which they denoted $\mathrm{DIC}_{2,K}$ sets the estimators $\tilde{\mathbf{w}}$ and $\tilde{\boldsymbol{\theta}}$ of the weights and parameters, respectively, to be those corresponding to their posterior mode, as given in Equation (1.134),

$$\left(\tilde{\mathbf{w}}, \tilde{\boldsymbol{\theta}}\right)(\mathbf{y}) = \underset{(\mathbf{w}, \boldsymbol{\theta}) \in (\boldsymbol{\Delta}^K, \mathcal{T})}{\arg\max}\; p(\mathbf{w}, \boldsymbol{\theta}|\mathbf{y}). \tag{1.134}$$

With this choice they derived the form given in Equation (1.135)

$$\mathrm{DIC}_{2,K} = -4\mathbb{E}_{\mathbf{w}, \boldsymbol{\theta}}\left[\log p\left(\mathbf{y}\,\middle|\,\tilde{\mathbf{w}}, \tilde{\boldsymbol{\theta}}\right)\middle|\,\mathbf{y}\right] + 2\log p\left[\mathbf{y}\,\middle|\,\tilde{\mathbf{w}}, \tilde{\boldsymbol{\theta}}, \mathbf{y}\right]. \tag{1.135}$$

Their rationale behind this choice of estimator was that simply using the posterior mean is inappropriate for models with unidentifiability, such as mixture models, which experience label-switching of the mixture components.

The other version of DIC used by Frühwirth-Schnatter and Pyne (2010) was one involving complete data, which Celeux et al. (2006) derived using a complete data estimator of the posterior mean $\mathbb{E}_{\mathbf{w}, \boldsymbol{\theta}, \mathbf{S}}\left(\mathbf{w}, \boldsymbol{\theta}\,|\mathbf{y}, \mathbf{S}\right)$ of the parameters. The result was the form given by in Equation (1.136),

$$\mathrm{DIC}_{4,K} = -4\mathbb{E}_{\mathbf{w}, \boldsymbol{\theta}, \mathbf{S}}\left[\log p(\mathbf{y}, \mathbf{S}|\mathbf{w}, \boldsymbol{\theta})|\mathbf{y}\right]$$

$$+ 2\mathbb{E}_{\mathbf{S}} \left\{ \log p \left[\mathbf{y}, \mathbf{S} \left| \mathbb{E}_{\mathbf{w},\boldsymbol{\theta}} \left(\mathbf{w}, \boldsymbol{\theta} \left| \mathbf{y}, \mathbf{S} \right. \right) \right| \mathbf{y} \right. \right] \right\}. \tag{1.136}$$

As pointed out by Frühwirth-Schnatter and Pyne (2010), Celeux et al. (2006) noted that it is not possible to analytically evaluate the complete data estimator of the posterior mean in the case of skew finite mixture. However, Celeux et al. (2006) derived an approximation that adds twice the expected posterior entropy, as inferred by MC simulation, to DIC_2. They gave this approximation according to Equation (1.137),

$$\mathrm{DIC}_4 \approx \mathrm{DIC}_2 + 2\mathbb{E}_{\mathbf{w},\boldsymbol{\theta}} \left\{ \mathrm{EN} \left[p(\mathbf{S}|\mathbf{y}, \mathbf{w}, \boldsymbol{\theta}) | \mathbf{y} \right] \right\}. \tag{1.137}$$

Each of the preceding criteria are suited to different types of applications. For example, Frühwirth-Schnatter and Pyne (2010) recalled a result of Keribin (2000), which proved the consistency of BIC under correct model specification and a sufficiently large number of observations. However, Biernacki (2000) demonstrated the superior robustness of ICL to model misspecification over that of BIC, while Banfield and Raftery (1993) introduced AWE as a method that can account for commonality or differences in features of the components of the mixture model.

In addition to the information criteria discussed by Frühwirth-Schnatter and Pyne (2010), two other noteworthy information criteria are Akaike's information criterion (AIC) (Akaike, 1987) and the consistent Akaike's information criterion (CAIC) (Bozdogan, 1987).

AIC and CAIC are defined according to Equations (1.138) and (1.139), respectively,

$$\mathrm{AIC}_K = -2 \log p \left(\mathbf{y} \left| \tilde{\mathbf{w}}, \tilde{\boldsymbol{\theta}}, K \right. \right) + 2g^{(K)} \tag{1.138}$$

$$\mathrm{CAIC}_K = -2 \log p \left(\mathbf{y} \left| \tilde{\mathbf{w}}, \tilde{\boldsymbol{\theta}}, K \right. \right) + g^{(K)} \left[(\log N) + 1 \right], \tag{1.139}$$

where $g^{(K)}$ represents the number of free parameters in a model with K components.

Bozdogan (1987) acknowledged that the AIC was an important early approach to model selection that addressed the trade-off between model accuracy and model complexity (*Occam's razor*). However, he also noted that contemporary literature criticised AIC for not being consistent in terms of K as the number of samples N is increased to infinity. That is, K is not guaranteed to converge to a (hypothetical) "true" value with N. Therefore, Bozdogan (1987) proposed CAIC (and a second information criterion, not discussed here) as a consistent extension of AIC.

1.6 Adaptive Markov Chain Monte Carlo

One solution to the problem of inefficient sampling of the target distribution is to use adaptive MCMC. In this approach, the proposal distribution may change with time. Various adaptive schemes have been been proposed but they can be sorted into two main groups: *externally adaptive* and *internally adaptive* (Atchadé et al., 2011).

As Atchadé et al. (2011) noted, adaptive methods may adapt for only the first portion of a simulation because the adaptive scheme may not necessarily ensure

ergodicity of the process. Once the proposal distribution is sufficiently tuned to the given problem, it is fixed, thereby avoiding issues of ensuring ergodicity of the process under a non-adaptve proposal distribution during the rest of the simulation. Otherwise, if the adaptive process is chosen more carefully, the requirement of ergodicity may be satisified, ensuring that the stationary distribution is the target or sufficiently close to it if an approximation is used. When such care is taken, adaptation can continue indefinitely.

If the transition kernel is allowed to change during a simulation, it can home in on a more efficient one automatically.

1.6.1 The Benefits of Adaptation

The algorithms discussed so far are efficient when the sampling distribution g is similar to the target but they are not necessarily optimal. The MCMC practitioner is unlikely to know *a priori* the optimum proposal distribution of a particular form that best approximates the target. As such, it is beneficial to learn it as a simulation progresses.

1.6.2 Types of Adaptation

As just stated, Atchadé et al. (2011) sorted adaptive Markov chain Monte Carlo methods into two categories, which they called "external" and "internal".

Under internal adaptation, the parameters of the transition kernel are adjusted according to a combination of the histories of the simulated draws and of the parameters themselves. This includes samplers in which history earlier than the state immediately preceding the current one is used, thereby breaking the Markov property.

The purpose of internal adaptation is to find a locally optimum transition kernel from a particular family of transition kernels. Atchadé et al. (2011) noted that, although the rate of adaptation may diminish as a simulation progresses, the total amount by which the transition kernel is modified (under some appropriate metric) is allowed to tend towards infinity. This allows a wide variety of internal adaptation schemes. Due to the breaking of the Markov property, approaches such as detailed balance (cf. Section 1.2.3) cannot be used to prove existence of a stationary distribution (cf. Section 1.2.3), so care must be taken to ensure that the stationary distribution exists and is the target of interest. Section 1.6.3 reviews some guidelines on how this may be achieved.

Some examples of internal adaptation mentioned by Atchadé et al. (2011) include the Adaptive Direction sampler of Gilks et al. (1994) (cf. Section 1.6.4), the Adaptive Proposal sampler of Haario et al. (1999) (cf. Section 1.6.4), the Adaptive Metropolis sampler of Haario et al. (2001) (cf. Section 1.6.4) and the Bayesian Adaptive Independence Sampler of Keith et al. (2008) (cf. 1.8).

Under external adaptation, an auxiliary, possibly heuristic, approach is used to guide sampling. Atchadé et al. (2011) highlighted that three key examples of this are SA (cf. Section 1.4.2), PT (cf. Section 1.4.3) and EES (cf. Section 1.4.3).

1.6.3 Ensuring the Correct Stationary Distribution

As mentioned in Section 1.3.2, Geman and Geman (1984) proved the ergodicity of the Gibbs sampler on discrete targets. Tanner and Wong (1987) also proved convergence to a desired stationary distribution of a Markov chain generated by a latent variable approach that essentially represents a two-variable Gibbs sampler on a Euclidean state space, which Chan (1993) further generalised to arbitrary finite dimensions. In Sections 1.3.1 and 1.3.2, respectively, it was noted that Roberts and Smith (1994) provided conditions to ensure ergodicity of both the Metropolis-Hastings sampler and Gibbs sampler, with the target as their stationary distributions.

Unfortunately, the conditions and proofs provided in those references assumed that the samplers obey the Markov property. That is, given a history of samples up to time $t-1$

$$x^{(0)}, x^{(1)}, \ldots, x^{(t-1)},$$

the probability density at $x^{(t)}$, the state at time t, conditional on all of this history is the same as the probability density at $x^{(t)}$, conditional only on the state at time $t-1$ (Bertsekas and Tsitsiklis, 2002, pp. 314).

$$p\left[x^{(t)} \middle| x^{(0)}, x^{(1)}, \ldots, x^{(t-1)}\right] = p\left[x^{(t)} \middle| x^{(t-1)}\right].$$

As mentioned in the introduction to the current section, there exist sampling algorithms that adapt their transition kernels using information from more than just the previous iteration, thereby breaking this property. As such, the conditions of Chan (1993) and Roberts and Smith (1994) do not apply, nor do other properties of stationary Markov chains, such as detailed balance. Therefore, different approaches are required to guarantee that samples produced come from the correct stationary distribution.

Atchadé and Rosenthal (2005) and Andrieu and Moulines (2006) independently provided sufficient conditions for the consistency of estimators (Definition 2 of Section 1.2.1) and for the ergodicity (cf. Section 1.2.2) of an adaptive MCMC process.

In the case of Atchadé and Rosenthal (2005), the authors essentially required that the convergence in n of the n-step transition kernel, at a given iteration t, towards the stationary distribution up to that iteration be "uniform-in-time" and that adaptation of the transition kernel diminish with each successive iteration. Atchadé and Rosenthal (2005) stated that their conditions generalised those used in Haario et al. (1999) and Haario et al. (2001) to prove ergodicity of the Adaptive Proposal (cf. Section 1.6.4) and Adaptive Metropolis (cf. Section 1.6.4) algorithms, respectively.

Andrieu and Moulines (2006) required two conditions for ergodicity of an adaptive sampler whose proposal density depends on the entire history of samples. These were a minorisation condition, equivalent to the existence of a small set (cf. Section 1.2.2), and a drift condition. This drift condition requires the existence of a finite measurable function $V : \mathcal{X} \to [1, \infty)$, on a separable state space $\mathcal{X}$ with a countably-generated σ-field $\mathscr{B}(\mathcal{X})$, as well as two constants $b \in (0, \infty)$ and $\lambda \in [0, 1)$, such that the integral with respect to V, of the transition kernel $P_{\boldsymbol{\theta}}(x, \cdot)$, parameterised by $\boldsymbol{\theta}$, is bounded according to Equation (1.140)

$$\int_{\mathcal{X}} \mathbb{P}_{\boldsymbol{\theta}}(x, dy) V(y) \leq \begin{cases} \lambda V(x) & \text{if } x \notin C, \\ b & \text{if } x \in C, \end{cases} \tag{1.140}$$

for any $\boldsymbol{\theta} \in \mathcal{S} \subset \mathcal{T}$ and some $C \in \mathscr{B}(\mathcal{X})$, where $\mathcal{S}$ is a compact subset of the parameter space $\mathcal{T}$.

Andrieu and Moulines (2006) noted that b and λ depend on V, C and $\mathcal{S}$.

Other techniques have also been used to prove consistency and ergodicity in an adaptive MCMC setting. For example, Holden et al. (2009) used a strong Doeblin condition (cf. Definition 12 of Section 1.2.2) to guarantee convergence of their adaptive extension of an independent MH algorithm.

Alternatively, some algorithms are designed to simply restrict or even halt adaptation altogether after a finite time or once sampler performance has reached some pre-determined threshold. One example of such an algorithm is the Adaptive Independent Metropolis-Hastings sampler of Giordani and Kohn (2010), which initially uses adaptive regime that does not guarantee ergodicity of the generated Markov chain, followed by a more restrictive one, which does. Section 1.6.4 takes an in-depth look at this sampler.

1.6.4 Examples

Three externally adaptive MC algorithms have already been reviewed in this chapter: PT, PA and EES. This section reviews in detail some of the aforementioned *internally*-adaptive sampling schemes.

Adaptive Direction Sampler

An algorithm introduced as an attempt to improve slow convergence of a Gibbs sampler is the *adaptive direction sampler* (ADS) of Gilks et al. (1994); Roberts and Gilks (1994). As its name suggests, this sampler grants more flexibility in the sampling direction of a Gibbs sampler, by allowing it to change at each iteration.

ADS updates, at each iteration t, a set of N p-dimensional points,

$$\mathbf{S}^{(t-1)} = \left\{ \mathbf{x}_1^{(t-1)}, \ldots, \mathbf{x}_N^{(t-1)} \right\},$$

where $N > p$.

It does this by first uniformly selecting one of the points $\mathbf{x}_c^{(t-1)} \in \mathbf{S}^{(t-1)}$, which Gilks et al. (1994) referred to as the "current point". The sampler then updates this point, relative to its current value, to produce a new point $\mathbf{x}^{(t)}$. This aspect of ADS makes the chain of each $\{\mathbf{x}_t\}_{t \in \mathbb{N}}$ a random walk (cf. Section 1.3.1).

The manner in which $\mathbf{x}_c^{(t-1)}$ is updated to $\mathbf{x}_c^{(t)}$ depends on two distributions, a multivariate one $D_{\mathbf{v}}$ and a univariate one D_u. Both of these distributions take the set $\mathbf{S}^{(t-1)} \setminus \left\{ \mathbf{x}_c^{(t-1)} \right\}$ as their argument and, according to Gilks et al. (1994), they are allowed to take any form. These distributions are used to generate variates $\mathbf{v}$ and u, respectively, which are then combined with a real-valued jumping distance r to provide a *jump* from $\mathbf{x}_c^{(t-1)}$ to $\mathbf{x}_c^{(t)}$, according to (1.141).

$$\mathbf{x}_c^{(t)} = \mathbf{x}_c^{(t-1)} + r \left[\mathbf{v} + u \mathbf{x}_c^{(t-1)} \right]. \tag{1.141}$$

Gilks et al. (1994) chose the rescaling r so as to ensure that the stationary distribution of the process was the intended target. ADS achieves this

by sampling it from an "adjusted full conditional" distribution of the target $\pi\left\{\mathbf{x}_c^{(t-1)} + r\left[\mathbf{v} + u\mathbf{x}_c^{(t-1)}\right] \middle| \mathbf{x}_c^{(t-1)}, \mathbf{v}, u\right\}$, weighted by an "adjustment factor" $J(r)$. Roberts and Gilks (1994) showed that this prescription produces the correct stationary distribution.

The general form of the adjustment factor was given by Gilks et al. (1994) to be that of Equation (1.142)

$$J(r) = |1 + ru|^{p-1}, \tag{1.142}$$

resulting in the form of the adjusted full conditional distribution given in Equation (1.143) (Roberts and Gilks, 1994),

$$\pi\left\{r \middle| \mathbf{x}_c^{(t-1)}, \mathbf{v}, u\right\} = \frac{\pi\left\{\mathbf{x}_c^{(t-1)} + r \cdot \left[\mathbf{v} + u\mathbf{x}_c^{(t-1)}\right]^{p-1}\right\}}{\int_{\mathbb{R}} \pi\left\{\mathbf{x}_c^{(t-1)} + s \cdot \left[\mathbf{v} + u\mathbf{x}_c^{(t-1)}\right]^{p-1}\right\} ds}. \tag{1.143}$$

Note that Equation (1.143) is a univariate density.

By sampling r with an appropriate method, $\mathbf{x}_c^{(t-1)}$ is then updated to $\mathbf{x}_c^{(t)}$. ADS is summarised in Algorithm A.18 of Appendix A.

Adaptive Proposal

Haario et al. (1999) introduced a modified random walk Metropolis sampler, intended to produce a more efficient sampling scheme, by adapting the parameters of the proposal distribution. They employed a heuristic approach that updates the proposal distribution at time t using information from the previous H iterations $\mathbf{x}^{(t-H)}, \ldots, \mathbf{x}^{(t-1)} \in \mathbb{R}^p$. As a result, the chain of samples produced by this sampler is not Markovian.

Specifically, they chose a normal proposal distribution, centred at the current sample, with covariance matrix $c_p^2\hat{\mathbf{\Sigma}}^{(t)}$,

$$\mathbf{y} \sim \mathcal{N}\left[\mathbf{x}^{(t)}, c_p^2\hat{\mathbf{\Sigma}}^{(t)}\right],$$

where $c_p = 2.38/\sqrt{p}$ is the scaling factor recommended by Gelman et al. (1996) for a normal proposal distribution and $\hat{\mathbf{\Sigma}}^{(t)}$ is the sample covariance matrix of the previous H draws, given by Equation (1.144),

$$\hat{\mathbf{\Sigma}}^{(t)} = \frac{1}{H-1}\sum_{i=t-H}^{t-1}\left[\mathbf{x}^{(i)} - \bar{\mathbf{x}}^{(t)}\right]\left[\mathbf{x}^{(i)} - \bar{\mathbf{x}}^{(t)}\right]^{\mathrm{T}}, \tag{1.144}$$

with corresponding sample mean,

$$\hat{\mathbf{\Sigma}}^{(t)} = \frac{1}{H}\sum_{i=t-H}^{t-1}\mathbf{x}^{(i)}.$$

The acceptance ratio is given by the Metropolis acceptance ratio of the target density π at the proposed and current samples, respectively, which is restated in Equation (1.145) for convenience,

$$\alpha\left[\mathbf{x}^{(t-1)}, \mathbf{y}\right] = \min\left\{1, \frac{\pi(\mathbf{y})}{\pi\left[\mathbf{x}^{(t-1)}\right]}\right\}. \tag{1.145}$$

The sampler then progresses at each iteration by first following this adaptive scheme, before performing a Metropolis update of the state. This Adaptive Proposal Sampler (APS) is summarised in Algorithm A.19 of Appendix A.

Haario et al. (1999) noted that the stationary distribution of the chain of samples produced by the adaptive proposal sampler is not the same as the intended target but, nonetheless, they found it to be acceptably close in the examples to which they applied it.

Adaptive Metropolis

Haario et al. (2001) also developed a modified version of APS. Instead of using a fixed-length history of the previous draws, their modified sampler uses the entire history. As such, the chain of samples generated by this sampler is, once again, not Markovian.

The proposal distribution still is p-variate normal, centred on the most recently-sampled state $\mathbf{x}^{(t-1)}$, so that at time t it is given by Equation (1.146),

$$g^{(t)}\left[\mathbf{y}\left|\mathbf{x}^{(0)},\ldots,\mathbf{x}^{(t-1)}\right.\right]=\mathcal{N}\left\{\mathbf{y}\left|\mathbf{x}^{(t-1)},\boldsymbol{\Sigma}^{(t)}\right.\right\}. \tag{1.146}$$

$\boldsymbol{\Sigma}^{(t)}$ is a covariance matrix associated with all samples up to but not including iteration t. It is not the sample covariance matrix but it is related to it, using the form given by Equation (1.147),

$$\boldsymbol{\Sigma}^{(t)}=\begin{cases}\boldsymbol{\Sigma}^{(0)} & \text{if } t\leq t_0\\ c_d^2\left[\hat{\boldsymbol{\Sigma}}^{(t-1)}+\epsilon\mathbf{I}_p\right] & \text{if } t>t_0\end{cases}, \tag{1.147}$$

where $c_p=2.38/\sqrt{d}$ is the recommended scaling of Gelman et al. (1996), as used in Section 1.6.4, t_0 is an initial non-adaptive sampling period, $0<\epsilon\ll 1$ is a positive constant, which Haario et al. (2001) required to ensure ergodicity, $\mathbf{I}_p$ is the p-dimensional identity matrix and $\hat{\boldsymbol{\Sigma}}^{(t-1)}$ is the sample covariance matrix of $\left\{\mathbf{x}^{(i)}\right\}_{i=1}^{t-1}$,

$$\hat{\boldsymbol{\Sigma}}^{(t-1)}=\frac{1}{t-1}\sum_{i=0}^{t-1}\left[\mathbf{x}^{(i)}-\bar{\mathbf{x}}^{(t-1)}\right]\left[\mathbf{x}^{(i)}-\bar{\mathbf{x}}^{(t-1)}\right]^{\mathrm{T}},$$

with the usual sample mean up to time $t-1$,

$$\bar{\mathbf{x}}^{(t-1)}=\frac{1}{t}\sum_{i=0}^{t-1}\mathbf{x}^{(i)}.$$

$\boldsymbol{\Sigma}^{(0)}$ is an arbitrary positive-definite covariance matrix, selected according to the target being studied.

For the adaptive portion of the chain ($t>t_0$), Haario et al. (2001) computed $\boldsymbol{\Sigma}^{(t)}$ online, by updating it as each new sample is drawn, using (1.148),

$$\begin{aligned}\boldsymbol{\Sigma}^{(t)}=\frac{t-2}{t-1}\boldsymbol{\Sigma}^{(t-1)}+\frac{c_p}{t-1}\Big\{&(t-1)\bar{\mathbf{x}}^{(t-2)}\left[\bar{\mathbf{x}}^{(t-2)}\right]^{\mathrm{T}}\\&-t\bar{\mathbf{x}}^{(t-1)}\left[\bar{\mathbf{x}}^{(t-1)}\right]^{\mathrm{T}}+\mathbf{x}^{(t-1)}\left[\mathbf{x}^{(t-1)}\right]^{\mathrm{T}}+\epsilon\mathbf{I}_p,\Big\}\end{aligned} \tag{1.148}$$

where

$$\bar{\mathbf{x}}^{(t-1)} = \frac{t-2}{t-1}\bar{\mathbf{x}}^{(t-2)} + \frac{1}{t-1}\mathbf{x}^{(t-1)}.$$

Unlike APS, Haario et al. (2001) demonstrated that their Adaptive Metropolis Samper (AMS) has the target π as its stationary distribution.

This sampler has formed the basis of later approaches, including examples by Atchadé and Rosenthal (2005) and Roberts and Rosenthal (2009).

Adaptive Independent Metropolis-Hastings

Giordani and Kohn (2010) introduced an approximate MH method called *adaptive independent Metropolis-Hastings* (AIMH), which allows fast sampling using mixture proposals. Their method is comprised of two adaptive phases, the latter of which continues until the end of sampling. The first stage does not ensure ergodicity, as it aims to find a selection of parameters that leads to a high acceptance rate. Once a sufficiently high acceptance rate has been achieved, the adaptation schedule is restricted, to ensure the correct form of the stationary distribution.

In the first phase of their sampler, each candidate new state $\mathbf{x}'$ is proposed according to Equation (1.149),

$$q^{(t)}\left[\mathbf{x}\middle|\boldsymbol{\lambda}^{(t)}\right] = \omega_1 g^{(0)}(\mathbf{x}) + (1-\omega_1)\, g^{(t)}\left[\mathbf{x}\middle|\boldsymbol{\lambda}^{(t)}\right], \tag{1.149}$$

where $g^{(0)}(\mathbf{x})$ is an initial proposal distribution given in Equation (1.150),

$$g^{(0)}(\mathbf{x}) = 0.6\phi^{(0)}(\mathbf{x}) + 0.4\tilde{\phi}^{(0)}(\mathbf{x}), \tag{1.150}$$

$\phi^{(0)}$ is an appropriately-chosen mixture of normal distributions and $\tilde{\phi}^{(0)}$ is another mixture that is obtained from $\phi^{(0)}$ by scaling its variance-covariance matrices each by a factor of 25.

$g^{(t)}$ is an adaptive density, given by Equation (1.151),

$$g^{(t)}\left[\mathbf{x}\middle|\boldsymbol{\lambda}^{(t)}\right] = \omega_2' \tilde{g}_\star^{(t)}\left[\mathbf{x}\middle|\tilde{\boldsymbol{\lambda}}_\star^{(t)}\right] + (1-\omega_2')\, g_\star^{(t)}\left(\mathbf{x}\middle|\boldsymbol{\lambda}_\star^{(t)}\right). \tag{1.151}$$

The mixture parameters $\boldsymbol{\lambda}^{(t)}$ are clear from Equation (1.151), provided the parameters $\boldsymbol{\lambda}_\star^{(t)}$ and $\tilde{\boldsymbol{\lambda}}_\star^{(t)}$ are known. Giordani and Kohn (2010) prescribed that $g_\star^{(t)}$ be a mixture of k normal distriutions, whose parameters $\boldsymbol{\lambda}_\star^{(t)}$ are generated using k-harmonic means clustering (cf. Section 1.5.1) and that the parameters $\tilde{\boldsymbol{\lambda}}_\star^{(t)}$ of another mixture $\tilde{g}_\star^{(t)}$ of k normal distributions then be set equal to $\boldsymbol{\lambda}_\star^{(t)}$ before scaling its variance-covariance matrices by a factor $k > 1$.

Giordani and Kohn (2010) required that $0 < \omega_1, \omega_2 < 1$, and $\omega_1 + \omega_2 < 1$, and they defined ω_2' according to Equation (1.152),

$$\omega_2' = \frac{\omega_2}{1-\omega_1}. \tag{1.152}$$

$\mathbf{x}'$ is then accepted via the standard MH acceptance ratio Equation (1.153),

$$\alpha\left[\mathbf{x}^{(t-1)}, \mathbf{x}'\right] = \min\left\{1, \frac{\pi\left(\mathbf{x}'\right) q^{(t)}\left[\mathbf{x}^{(t-1)}\middle|\boldsymbol{\lambda}^{(t)}\right]}{\pi\left[\mathbf{x}^{(t-1)}\right] q^{(t)}\left[\mathbf{x}'\middle|\boldsymbol{\lambda}^{(t)}\right]}\right\}. \tag{1.153}$$

The procedure for updating the chain state is summarised in Algorithm A.21 of Appendix A.

The procedure for updating the parameters is summarised in Algorithm A.22, while AIMH is summarised in Algorithm A.23.

Giordani and Kohn (2010) asserted that AIMH is geometrically ergodic (cf. Section 1.2.2) provided that a modified version of the necessary and sufficient conditions of Mengersen and Tweedie (1996) hold. That is, they required that the ratios of the target density and the successive time-varying components of the proposal densities to the initial proposal at every point $\mathbf{x} \in \mathcal{X}$ be no greater than some positive constant $K > 0$,

$$\frac{\pi(\mathbf{x})}{g_0(\mathbf{x})}, \frac{g^{(t)}\left[\mathbf{x}\left|\boldsymbol{\lambda}^{(t)}\right.\right]}{g^{(0)}(\mathbf{x})} \leq K.$$

They also required that the supremum of the absolute difference between time-varying proposal components, relative to the fixed component of the proposal be bounded from above by some positive constant $a^{(t)}$,

$$\sup_{\mathbf{x}\in\mathcal{X}} \left| \frac{g^{(t)}\left[\mathbf{x}\left|\boldsymbol{\lambda}^{(t)}\right.\right] - g^{(t+1)}\left[\mathbf{x}\left|\boldsymbol{\lambda}^{(t+1)}\right.\right]}{g^{(0)}(\mathbf{x})} \right| \leq a^{(t)}.$$

1.7 Adaptive Proposals with Multiple Local Maxima

Section 1.6.4 reviewed an adaptive MCMC sampler that utilised a mixture of normal distributions as its proposal. This was just one of a number of such algorithms that adaptively update parameters of a mixture proposal.

Other more recent methods, involving mixture proposals, include a modification of the adaptive Metropolis sampler (cf. Section 1.6.4), introduced by Ji and Schmidler (2013). Their modified version replaces the proposal distribution of AMS with a mixture proposal involving normal and point-mass components.

There have also been recent developments that, like the AIMH of Giordani and Kohn (2010), use clustering to adapt the parameters of a mixture proposal. One such method is that of Li and Lin (2015), which is an extension of importance sampling (cf. Section 1.1.2), used to solve Bayesian inverse problems. Unlike MCMC, their sampler is not designed to continue for an arbitrary length of time. Rather, its goal is to construct a normal mixture approximation of the posterior distribution of the parameters in a model under study. From this mixture, the parameters and their variances may be estimated. At each time step, it uses clustering to update the normal mixture approximation to the posterior, before using polynomial chaos expansions of Wiener (1938) to allow fast computation of approximate state samples from the model described by the current set of parameter samples. It then uses the likelihood of the state samples to adaptively temper the algorithm for computing importance weights of the simulated parameters. Once the temperature of the sampler reduces to zero by this tempering scheme the sampler stops and the most recent parameters and hyperparameters are used for inference.

Another recent method that combines tempering and clustering to solve Bayesian inverse problems is that of Feng and Li (2018). Like the AIMH of

Giordani and Kohn (2010), this algorithm progresses in multiple phases: two adaptation phases and one non-adaptive one. During the first adaptive phase it uses a tempering scheme to enable fast exploration of the state space, in order to find local maxima, using a clustering approach. In the second phase it continues the same adaptive method intermittently without tempering. Finally, it halts adaptation entirely after a certain number of steps and continues the sampler with a fixed proposal, thereby avoiding the issues of proving ergodicity, which arise in ongoing adaptation (cf. Section 1.6.3).

Another method with a mixture proposal that halts adaptation after some time is the *Fast universal self-tuned sampler within Gibbs* (FUSS) of Martino et al. (2014). It starts with a refined mesh of "support points" that are used to construct a piecewise uniform distribution on a region of interest to the MCMC practitioner. Outside these points it uses an appropriate distribution to approximate the tails. FUSS then cuts down the number of regions until it arrives at a sufficiently simple and accurate proposal density that it fixes for the rest of the sampler. For this purpose Martino et al. (2014) provided three pruning algorithms. The first retains the support points with the most significant density with respect to the target, the second retains those with density above some threshold value, and the third retains those whose densities are sufficiently large compared to those of the other support points, given some threshold.

A method that takes advantage of modern parallel computing is the *Jumping Adaptive Multimodal Sampler* (JAMS) of Pompe et al. (2018). Their method is a hybrid approach that involves a mechanism to detect local maxima running alongside a sampling mechanism. During a run, the mechanism to find local maxima seeks out new local maxima, while the sampling method performs either a local move or a jump between regions surrounding different local maxima, followed by a local move within the new region. In a local move it uses a circular distribution to sample in a given region using an MH sampler, while a jump move is made using an adaptive jump probability, given the current parameters associated with the detected local maxima.

JAMS continues adaptation indefinitely, using all post-burn-in samples to adjust the parameters associated with the detected local maxima as it learns more about the target.

Pompe et al. (2018) showed that JAMS is a particular instance of a more general class of adaptive methods that use auxilliary variables, for which they provided proofs of sufficiency of conditions that guarantee their ergodicity.

Also taking into consideration parallel sampling is the *Parallel Metropolis-Hastings Coupler* (PMHC) of Llorente et al. (2019). Their method combines multiple parallel chains, with each performing standard MH updates independently of the other chains. At pre-determined intervals, PMHC then switches to sampling from a mixture proposal that couples the information of the parallel chains. These mixture components are centred on the current states of each of the parallel chains, using a fixed dispersion in each. PMHC then uses a Metropolis-within-Gibbs approach, by randomly selecting one population member to be updated and using standard MH to update it.

Another MH-like sampler, which does not use a mixture proposal to capture local structure of the target, but a locally-adaptive one, is the *Kameleon algorithm* of Sejdinovic et al. (2014). Their approach uses a normal proposal, centred on the current state of the Markov chain. The variance-covariance ma-

trix of this proposal distribution is constructed by first randomly selecting a subset of the history of the generated Markov chain. It then computes gradients from the current state to the randomly-selected ones, with respect to some pre-chosen probability kernel, whose gradients are easily computed. These gradients are then combined to orient the normal proposal distribution in such a way that local high-acceptance probability moves are favoured.

The result is a sampler that achieves efficient local sampling in a moving region, without the need to specify the number of components in a mixture proposal. Such an approach avoids the need to specify a number of components in some mixture. However, the motivation of the Kameleon algorithm was for sampling from highly non-linear targets (Sejdinovic et al., 2014), not necessarily from targets with many well-separated local maxima.

The idea of estimating gradients in the state space using sampler history, rather than explicit numerical computation, has also been applied by Strathmann et al. (2015) to Hamiltonian Monte Carlo (HMC) (Neal, 2011) (introduced as "Hybrid Monte Carlo" by Duane et al. (1987)), to produce an HMC sampler that does not require explicitly computing derivatives.

Note that in all of the preceding approaches, the adaptive aspect of the sampler either adjusts the proposal distribution by adjusting its current form or uses a clustering algorithm on the samples to create a new one. That is, their proposal parameters are *constructed*. An alternative approach to constructing the parameters is to *sample* them. A recent framework that employs such an approach is that of *Posterior-Based Proposals* (PBPs) of Pooley et al. (2019).

Their method aims to sample parameters and latent variables of a model from their posterior distribution, given some data. PBPs break down the problem of sampling from the posterior into one of successively sampling the parameters and latent variables, based on their dependence structure within the model in question. The result is sequential sampling from a collection of univariate distributions with standard form, which approximate the true conditional densities of the parameters and latent variables.

By performing an accept or reject step once after sampling all parameters and latent variables, PBPs take into account the correlation structure of the underlying model, thereby improving mixing (Pooley et al., 2019).

Section 1.8 will review another procedure that makes use of sampling to update its proposal distribution but which does not make reference to the target distribution in doing so.

1.8 Bayesian Adaptive Independence Sampling

An adaptive technique that is crucial to the current dissertation is the Bayesian Adaptive Independence Sampler (BAIS) of Keith et al. (2008). This MCMC method will form the starting point for the algorithm presented in Part II.

1.8.1 Idea

Like AMS and APS, BAIS uses a normal distribution to propose new chain states. Also like the adaptive proposal sampler it adjusts the parameters of this normal proposal at each iteration. However, the means by which this adaptation takes place is very different.

Keith et al. (2008) cast the problem of inferring these parameters as one in Bayesian inference, hence the name. They considered not only the samples from the target but also the proposal parameters as random variables to be inferred. By running N simultaneous chains, each representing samples from the target, they used the current states of these chains to infer the posterior distribution of the proposal parameters. BAIS then samples new proposal parameters directly from this distribution, which it then uses in the proposal distribution to perform MH-like sampling of new chain states.

1.8.2 Description

There are two key differences between the MH sampler discussed in Section 1.3.1 with a normal proposal with parameters $\boldsymbol{\theta} = (\boldsymbol{\mu}, \boldsymbol{\Sigma})$ and BAIS. The first is that there are N simultaneous samples at each iteration

$$\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)$$

instead of just one. The second is that $\boldsymbol{\theta}$ is updated at each iteration, given the current population of samples.

The sampler alternates between updating the proposal parameters and the sampling chain states so that the Markov chain generated by it consists of chains of N p-dimensional sampling chain states, a p-dimensional proposal mean $\boldsymbol{\mu}$ and a $p \times p$ proposal variance-covariance matrix $\boldsymbol{\Sigma}$,

$$\left\{ \left[\boldsymbol{\mu}^{(0)}, \boldsymbol{\Sigma}^{(0)}, \mathbf{x}^{(0)} \right], \ldots, \left[\boldsymbol{\mu}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \mathbf{x}^{(t)} \right], \ldots \right\}.$$

It is this chain that must be ergodic and, naturally, the marginal stationary distribution over each $\left\{ \mathbf{x}_n^{(t)} \right\}_{t=0}^{\infty}$ must be the target distribution π.

Keith et al. (2008) developed BAIS so that these two requirements are guaranteed. In particular, stationarity of π is easily verified using detailed balance (cf. Section 1.8.3).

Proposal Distribution

As stated previously, BAIS uses a single normal distribution to propose samples, according to Equation (1.154),

$$\mathbf{y} | \boldsymbol{\mu}, \boldsymbol{\Sigma} \sim \mathcal{N}(\cdot | \boldsymbol{\mu}, \boldsymbol{\Sigma}). \tag{1.154}$$

A proposed state $\mathbf{y}$ is generated from this distribution independently for each of the N sampling chains.

Prior Model on Proposal Parameters

In order to update the proposal parameters $\boldsymbol{\mu}, \boldsymbol{\Sigma}$ using a Bayesian approach a prior model is required. Keith et al. (2008) used a conjugate prior on the parameters. Following Gelman et al. (2004, p. 88), they chose it to be the uninformative improper Jeffreys prior, given in Equation (1.155),

$$p(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \propto |\boldsymbol{\Sigma}|^{-(p+1)/2}. \tag{1.155}$$

Section 3.1.2 will discuss why such a prior is not suitable when the proposal distribution is a mixture, rather than just a single normal density.

Other prior models are also possible, such as the joint conjugate prior of both $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, given by a normal-inverse-Wishart distribution (Gelman et al., 2004, pp. 87–88), as in Equations (1.156) and (1.157),

$$\boldsymbol{\Sigma} \sim \text{Inv-W}_{\nu^{(0)}} \left\{ \cdot \left| \left[\boldsymbol{\Lambda}^{(0)} \right]^{-1} \right. \right\}, \tag{1.156}$$

$$\boldsymbol{\mu} | \boldsymbol{\Sigma} \sim \mathcal{N} \left[\cdot \left| \boldsymbol{\mu}^{(0)}, \frac{\boldsymbol{\Sigma}}{\kappa^{(0)}} \right. \right], \tag{1.157}$$

with the prior degrees of freedom $\nu^{(0)}$ and the prior scale matrix $\boldsymbol{\Lambda}^{(0)}$ on $\boldsymbol{\Sigma}$, the prior mean $\boldsymbol{\mu}^{(0)}$ and the prior number of observations $\kappa^{(0)}$ of the scale of $\boldsymbol{\Sigma}$ chosen appropriate to the model under study.

The prior given in Equations (1.156) and (1.157) was used by Keith and Davey (2013) in their random walk extension of BAIS (cf. Section 1.8.5).

Posterior Distribution of the Proposal Parameters

Taking likelihood function to be the proposal distribution given in Equation (1.154), the posterior distribution of proposal parameters corresponding to the uninformative prior used by Keith et al. (2008) is a product of a normal distribution on $\boldsymbol{\mu}$ and an inverse-Wishart distribution on $\boldsymbol{\Sigma}$. Specifically, the inverse-Wishart distribution on the proposal variance-covariance matrix is dependent on the number of sampling chains and their current states, (Keith et al., 2008) (Gelman et al., 2004, pp. 88), as shown in Equation (1.158),

$$\boldsymbol{\Sigma} | \mathbf{x} \sim \text{Inv-W}_{N-1}(\cdot | \mathbf{S}), \tag{1.158}$$

where $\mathbf{S}$ is the sample sum of squared errors,

$$\mathbf{S} = \sum_{n=1}^{N} (\mathbf{x}_n - \bar{\mathbf{x}}) (\mathbf{x}_n - \bar{\mathbf{x}})^{\mathrm{T}}$$

and $\bar{\mathbf{x}}$ is the sample mean,

$$\bar{\mathbf{x}} = \frac{1}{N} \sum_{n=1}^{N} \mathbf{x}_n.$$

The posterior distribution on the proposal mean $\boldsymbol{\mu}$ is a normal distribution with mean $\bar{\mathbf{x}}$ and variance-covariance matrix $\boldsymbol{\Sigma}/N$, as shown in Equation (1.159),

$$\boldsymbol{\mu} | \boldsymbol{\Sigma}, \mathbf{x} \sim \mathcal{N} \left(\cdot \left| \bar{\mathbf{x}}, \frac{\boldsymbol{\Sigma}}{N} \right. \right). \tag{1.159}$$

Only states from the iteration immediately preceding the current one are used to update the proposal parameters, hence the Markov property is preserved.

Acceptance Ratio

Since the proposal distribution changes at each iteration, the proposed sample $\mathbf{y}$ and the current state $\mathbf{x}_n^{(t-1)}$ of sampling chain n are drawn from different proposal distributions. Hence, the standard MH acceptance ratio is no longer valid.

Updating each sampling chain n in series, Keith et al. (2008) defined the current population mean $\bar{\mathbf{x}}$, the proposed population mean $\bar{\mathbf{x}}_{\mathbf{y}}$, the current population sum of squared errors $\mathbf{S}$ and the proposed population sum of squared errors $\mathbf{S}_{\mathbf{y}}$ according to Equations (1.160), (1.161), (1.162) and (1.163).

$$\bar{\mathbf{x}} = \frac{1}{N}\left[\sum_{i=1}^{n-1}\mathbf{x}_i^{(t)} + \sum_{i=n}^{N}\mathbf{x}_i^{(t-1)}\right], \tag{1.160}$$

$$\bar{\mathbf{x}}_{\mathbf{y}} = \frac{1}{N}\left[\sum_{i=1}^{n-1}\mathbf{x}_i^{(t)} + \mathbf{y} + \sum_{i=n+1}^{N}\mathbf{x}_i^{(t-1)}\right], \tag{1.161}$$

$$\begin{aligned}\mathbf{S} = &\sum_{i=1}^{n-1}\left[\mathbf{x}_i^{(t)} - \bar{\mathbf{x}}\right]\left[\mathbf{x}_i^{(t)} - \bar{\mathbf{x}}\right]^{\mathrm{T}} \\ &+ \sum_{i=n}^{N}\left[\mathbf{x}_i^{(t)} - \bar{\mathbf{x}}\right]\left[\mathbf{x}_i^{(t-1)} - \bar{\mathbf{x}}\right]^{\mathrm{T}},\end{aligned} \tag{1.162}$$

$$\begin{aligned}\mathbf{S}_{\mathbf{y}} = &\sum_{i=1}^{n-1}\left[\mathbf{x}_i^{(t)} - \bar{\mathbf{x}}_{\mathbf{y}}\right]\left[\mathbf{x}_i^{(t)} - \bar{\mathbf{x}}_{\mathbf{y}}\right]^{\mathrm{T}} \\ &+ \left(\mathbf{y} - \bar{\mathbf{x}}_{\mathbf{y}}\right)\left(\mathbf{y} - \bar{\mathbf{x}}_{\mathbf{y}}\right)^{\mathrm{T}} \\ &+ \sum_{i=n+1}^{N}\left[\mathbf{x}_i^{(t)} - \bar{\mathbf{x}}_{\mathbf{y}}\right]\left[\mathbf{x}_i^{(t)} - \bar{\mathbf{x}}_{\mathbf{y}}\right]^{\mathrm{T}}.\end{aligned} \tag{1.163}$$

Taking into account the parameter update distributions, Keith et al. (2008) arrived at the acceptance ratio given in (1.164),

$$\begin{aligned}\alpha\left[\mathbf{x}_n^{(t-1)}, \mathbf{y}\right] = \min\Bigg\{1, &\frac{\pi(\mathbf{y})}{\pi[\mathbf{x}^{(t-1)}]} \cdot \frac{\text{Inv-W}_{N-1}(\boldsymbol{\Sigma}|\mathbf{S}_{\mathbf{y}})}{\text{Inv-W}_{N-1}(\boldsymbol{\Sigma}|\mathbf{S})} \\ &\times \frac{\mathcal{N}(\boldsymbol{\mu}|\bar{\mathbf{x}}_{\mathbf{y}}, \boldsymbol{\Sigma})}{\mathcal{N}(\boldsymbol{\mu}|\bar{\mathbf{x}}, \boldsymbol{\Sigma})} \frac{\mathcal{N}[\mathbf{x}^{(t-1)}|\boldsymbol{\mu}, \boldsymbol{\Sigma}]}{\mathcal{N}(\mathbf{y}|\boldsymbol{\mu}, \boldsymbol{\Sigma})}\Bigg\}.\end{aligned} \tag{1.164}$$

Due to the importance of BAIS to the current dissertation, it is summarised here, in Algorithm 1.1. Note that the redundant minimum on Line 12 of Algorithm 1.1 has been included to highlight that α is an acceptance probability.

1.8.3 Satisfying Detailed Balance

With the modified Metropolis-Hastings acceptance ratio in Equation (1.164), Keith et al. (2008) implied that BAIS still produces an ergodic Markov chain that has the target distribution as its stationary distribution. The stationarity of the target is summarised by Theorem 12, whose result Keith et al. (2008) noted holds due to detailed balance.

Algorithm 1.1 BAIS (Keith et al., 2008).

Require: $\left[\mathbf{x}_1^{(0)}, \ldots, \mathbf{x}_N^{(0)}\right] \in \mathbb{R}^{pN}$ initial chain states.

Ensure: N chains of samples $\left\{\left[\mathbf{x}_n^{(t)}\right]_{n=1}^{N}\right\}_{t=1}^{\infty}$ from the target distribution π.

1: **for** $t \in \mathbb{Z}^+$ **do**
2: Set $\bar{\mathbf{x}} = \sum_{n=1}^{N} \mathbf{x}_n^{(t-1)} \Big/ N$.
3: Set $\mathbf{S} = \sum_{n=1}^{N} \left[\mathbf{x}_n^{(t-1)} - \bar{\mathbf{x}}\right] \left[\mathbf{x}_n^{(t-1)} - \bar{\mathbf{x}}\right]^{\mathrm{T}}$.
4: Generate $\boldsymbol{\Sigma} \sim \text{Inv-W}_{N-1}(\cdot|\mathbf{S})$.
5: Generate $\boldsymbol{\mu} \sim \mathcal{N}(\cdot|\bar{\mathbf{x}}, \boldsymbol{\Sigma}/N)$.
6: **for** Sampling chain $n = 1$ to $n = N$ **do**
7: Generate $\mathbf{y} \sim \mathcal{N}(\cdot|\boldsymbol{\mu}, \boldsymbol{\Sigma})$.
8: Set $\bar{\mathbf{x}} = \left[\sum_{i=1}^{n-1} \mathbf{x}_i^{(t)} + \sum_{i=n}^{N} \mathbf{x}_i^{(t-1)}\right] \Big/ N$.
9: Set $\mathbf{S} = \sum_{i=1}^{n-1} \left[\mathbf{x}_i^{(t)} - \bar{\mathbf{x}}\right] \left[\mathbf{x}_i^{(t)} - \bar{\mathbf{x}}\right]^{\mathrm{T}} + \sum_{i=n}^{N} \left[\mathbf{x}_i^{(t)} - \bar{\mathbf{x}}\right] \left[\mathbf{x}_i^{(t-1)} - \bar{\mathbf{x}}\right]^{\mathrm{T}}$.
10: Set $\bar{\mathbf{x}}_{\mathbf{y}} = \left[\sum_{i=1}^{n-1} \mathbf{x}_i^{(t)} + \mathbf{y} + \sum_{i=n+1}^{N} \mathbf{x}_i^{(t-1)}\right] \Big/ N$.
11: Set $\mathbf{S}_{\mathbf{y}} = \sum_{i=1}^{n-1} \left[\mathbf{x}_i^{(t)} - \bar{\mathbf{x}}_{\mathbf{y}}\right] \left[\mathbf{x}_i^{(t)} - \bar{\mathbf{x}}_{\mathbf{y}}\right]^{\mathrm{T}} + \left[\mathbf{y} - \bar{\mathbf{x}}_{\mathbf{y}}\right] \left[\mathbf{y} - \bar{\mathbf{x}}_{\mathbf{y}}\right]^{\mathrm{T}} + \sum_{i=n+1}^{N} \left[\mathbf{x}_i^{(t)} - \bar{\mathbf{x}}_{\mathbf{y}}\right] \left[\mathbf{x}_i^{(t)} - \bar{\mathbf{x}}_{\mathbf{y}}\right]^{\mathrm{T}}$.
12: Set $\alpha = \min\left\{1, \dfrac{\pi(\mathbf{y})\text{Inv-W}_{N-1}(\boldsymbol{\Sigma}|\mathbf{S}_{\mathbf{y}})\mathcal{N}(\boldsymbol{\mu}|\bar{\mathbf{x}}_{\mathbf{y}}, \boldsymbol{\Sigma})\mathcal{N}[\mathbf{x}^{(t-1)}|\boldsymbol{\mu}, \boldsymbol{\Sigma}]}{\pi[\mathbf{x}^{(t-1)}]\text{Inv-W}_{N-1}(\boldsymbol{\Sigma}|\mathbf{S})\mathcal{N}(\boldsymbol{\mu}|\bar{\mathbf{x}}, \boldsymbol{\Sigma})\mathcal{N}[\mathbf{y}|\boldsymbol{\mu}, \boldsymbol{\Sigma}]}\right\}$.
13: Generate $u \sim \mathcal{U}(0, 1)$.
14: **if** $u \leq \alpha$ **then**
15: Set $\mathbf{x}^{(t)} = \mathbf{y}$.
16: **else**
17: Set $\mathbf{x}^{(t)} = \mathbf{x}^{(t-1)}$.
18: **end if**
19: **end for**
20: **end for**

Theorem 12. *The augmented target distribution over the chain states* $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)$ *and the proposal distribution parameters* $\boldsymbol{\theta} = (\boldsymbol{\mu}, \boldsymbol{\Sigma})$*, given by Equation* (1.165)*,*

$$h(\mathbf{x}_1, \ldots, \mathbf{x}_N, \boldsymbol{\theta}) = p(\boldsymbol{\theta}|\mathbf{x}) \prod_{n=1}^{N} \pi(\mathbf{x}_n) \tag{1.165}$$

is stationary with respect to BAIS transitions.

Proof. Keith et al. (2008) considered the BAIS update process in two stages: a parameter update step; and a sequence of sampling chain update steps. Updating the parameters from $\boldsymbol{\theta}$ to $\boldsymbol{\theta}'$ does not change the marginal stationary distribution of $\mathbf{x}$. Now consider the transition from $(\mathbf{x}, \boldsymbol{\theta}')$ to $(\mathbf{x}', \boldsymbol{\theta}')$.

To see that this transition does not affect the stationary distribution of $\mathbf{x}$ it is sufficient to check that it satisfies detailed balance in a similar way as Keith and Davey (2013) did for their sampler.

That is, when updating sampling chain n, let the current $\mathbf{x}$ and updated $\mathbf{x}'$ populations of samples be given by Equations (1.166) and (1.167), respectively,

$$\mathbf{x} = \left[\mathbf{x}_1^{(t)}, \mathbf{x}_2^{(t)}, \ldots, \mathbf{x}_{n-1}^{(t)}, \mathbf{x}_n^{(t-1)}, \mathbf{x}_{n-1}^{(t-1)}, \ldots, \mathbf{x}_{N-1}^{(t-1)}, \mathbf{x}_N^{(t-1)}\right], \tag{1.166}$$

$$\mathbf{x}' = \left[\mathbf{x}_1^{(t)}, \mathbf{x}_2^{(t)}, \ldots, \mathbf{x}_{n-1}^{(t)}, \mathbf{x}_n^{(t)}, \mathbf{x}_{n-1}^{(t-1)}, \ldots, \mathbf{x}_{N-1}^{(t-1)}, \mathbf{x}_N^{(t-1)}\right]. \tag{1.167}$$

Then Equation (1.168) needs to hold,

$$h\left(\mathbf{x}, \boldsymbol{\theta}\right) p\left(\mathbf{x}'|\mathbf{x}, \boldsymbol{\theta}\right) = h\left(\mathbf{x}', \boldsymbol{\theta}\right) p\left(\mathbf{x}|\mathbf{x}', \boldsymbol{\theta}\right). \tag{1.168}$$

Observe that showing the truth of Equation (1.168) is equivalent to showing that

$$h\left(\mathbf{x}, \boldsymbol{\theta}\right) p\left[\mathbf{x}_n^{(t)} \middle| \mathbf{x}, \boldsymbol{\theta}\right] = h\left(\mathbf{x}', \boldsymbol{\theta}\right) p\left[\mathbf{x}_n^{(t-1)} \middle| \mathbf{x}', \boldsymbol{\theta}\right] \tag{1.169}$$

holds, since a transition between $\mathbf{x}$ and $\mathbf{x}'$ only only changes the state of $\mathbf{x}_n$, according to

$$p(\mathbf{x}'|\mathbf{x}, \boldsymbol{\theta}) = p\left[\mathbf{x}_n^{(t)} \middle| \mathbf{x}, \boldsymbol{\theta}\right]$$
$$p(\mathbf{x}|\mathbf{x}', \boldsymbol{\theta}) = p\left[\mathbf{x}_n^{(t-1)} \middle| \mathbf{x}', \boldsymbol{\theta}\right]$$

Substituting the appropriate quantities into Equation (1.169) gives

$$\begin{aligned}
\text{L.H.S.} &= h\left(\mathbf{x}, \boldsymbol{\theta}\right) p\left[\mathbf{x}_n^{(t)} \middle| \mathbf{x}, \boldsymbol{\theta}\right] \\
&= p(\boldsymbol{\theta}|\mathbf{x}) \pi\left[\mathbf{x}_n^{(t-1)}\right] \prod_{i=1}^{n-1} \pi\left[\mathbf{x}_n^{(t)}\right] \prod_{i=n+1}^{N} \pi\left[\mathbf{x}_n^{(t-1)}\right] \\
&\quad \times p\left[\mathbf{x}_n^{(t)} \middle| \mathbf{x}, \boldsymbol{\theta}\right] \\
&= p(\boldsymbol{\theta}|\mathbf{x}) \pi\left[\mathbf{x}_n^{(t-1)}\right] \prod_{i=1}^{n-1} \pi\left[\mathbf{x}_n^{(t)}\right] \prod_{i=n+1}^{N} \pi\left[\mathbf{x}_n^{(t-1)}\right] \\
&\quad \times \mathcal{N}\left[\mathbf{x}_n^{(t)} \middle| \boldsymbol{\theta}\right] \min\left\{1, \frac{\pi\left[\mathbf{x}_n^{(t)}\right]}{\pi\left[\mathbf{x}_n^{(t-1)}\right]} \frac{p(\boldsymbol{\theta}|\mathbf{x}')}{p(\boldsymbol{\theta}|\mathbf{x})} \frac{\mathcal{N}\left[\mathbf{x}_n^{(t-1)} \middle| \boldsymbol{\theta}\right]}{\mathcal{N}\left[\mathbf{x}_n^{(t)} \middle| \boldsymbol{\theta}\right]}\right\}
\end{aligned}$$

$$
\begin{aligned}
&= p(\boldsymbol{\theta}|\mathbf{x}')\pi\left[\mathbf{x}_n^{(t)}\right]\prod_{i=1}^{n-1}\pi\left[\mathbf{x}_n^{(t)}\right]\prod_{i=n+1}^{N}\pi\left[\mathbf{x}_n^{(t-1)}\right] \\
&\quad\times\mathcal{N}\left[\mathbf{x}_n^{(t-1)}\middle|\boldsymbol{\theta}\right]\min\left\{\frac{\pi\left[\mathbf{x}_n^{(t-1)}\right]}{\pi\left[\mathbf{x}_n^{(t)}\right]}\frac{p(\boldsymbol{\theta}|\mathbf{x})}{p(\boldsymbol{\theta}|\mathbf{x}')}\frac{\mathcal{N}\left[\mathbf{x}_n^{(t)}\middle|\boldsymbol{\theta}\right]}{\mathcal{N}\left[\mathbf{x}_n^{(t-1)}\middle|\boldsymbol{\theta}\right]},1\right\} \\
&= p(\boldsymbol{\theta}|\mathbf{x}')\pi\left[\mathbf{x}_n^{(t)}\right]\prod_{i=1}^{n-1}\pi\left[\mathbf{x}_n^{(t)}\right]\prod_{i=n+1}^{N}\pi\left[\mathbf{x}_n^{(t-1)}\right] \\
&\quad\times p\left[\mathbf{x}_n^{(t-1)}\middle|\mathbf{x}',\boldsymbol{\theta}\right] \\
&= h(\mathbf{x}',\boldsymbol{\theta})p\left[\mathbf{x}_n^{(t-1)}\middle|\mathbf{x}',\boldsymbol{\theta}\right] \\
&= \text{R.H.S.}
\end{aligned}
$$

Therefore, $h(\mathbf{x},\boldsymbol{\theta})$ is unchanged by sampling chain updates.

Since BAIS is an accept/reject MCMC sampler, whose accepted moves from state $(\mathbf{x},\boldsymbol{\theta})$ to state $(\mathbf{x}',\boldsymbol{\theta})$ satisfy detailed balance with the augmented target $h(\mathbf{x},\boldsymbol{\theta})$, by Theorem 11, $h(\mathbf{x},\boldsymbol{\theta})$ is stationary with respect to sampling chain updates. □

1.8.4 Shortcomings and Differences from Other Parallel Adaptive Methods

While Keith et al. (2008) showed that BAIS is an effective sampler, demonstrating that it is useful in many situations, it is not a natural option for sampling from target distributions with many local maxima, such as the rough energy landscapes seen in spin glass simulations or protein structure prediction (cf. Section 1.4). The single normal proposal distribution of BAIS only encapsulates the overall mean and variance of the target distribution but ignores local peaks and troughs. That is it does not capture the local shape of the target distribution and, hence, cannot approximate a rough target density well, meaning that the issue illustrated in Figure 1.2 will still be exhibited by a BAIS simulation from a target with multiple local maxima.

The use of parallel sampling chains is not unique to BAIS. For example, in a non-adaptive setting, Craiu and Meng (2005) used antithetically-coupled parallel sampling chains in an MCMC sampler to reduce the variance of estimates inferred from the samples. In an adaptive setting, introduced shortly after BAIS, Craiu et al. (2009) combined two approaches, which they called "Interchain Adaptation" (INCA) and "Regional Adaptation" (RAPT). INCA uses N parallel chains, started at different points in the state space, to allow the exploration of N regions of the state space simultaneously, while RAPT uses a mixture proposal that enables each region of the state space to be explored in a tailored fashion.

The method of Craiu et al. (2009) is an alternative approach to addressing the main problem addressed in Part II of this dissertation: sampling from a target with many local maxima. However, unlike the approach introduced in this dissertation or BAIS, the method of Craiu et al. (2009) does not explicitly exploit the Bayesian relationship between the prior distribution of the proposal

parameters and the posterior distribution of the proposal parameters, given the generated chain states.

1.8.5 Bayesian Adaptive Metropolis-Hastings Sampling

BAIS is a special instance of a more general approach, called *Bayesian Adaptive Metropolis-Hastings Sampling* (BAMS), due to Keith and Davey (2013). Instead of using a normal distribution proposal with mean $\boldsymbol{\mu}$ and variance-covariance matrix $\boldsymbol{\Sigma}$ in an independence sampler, BAMS uses any proposal distribution $p(\mathbf{x}|\boldsymbol{\theta})$, where the vector of proposal parameters $\boldsymbol{\theta}$ is dependent on the choice of proposal distribution.

As in BAIS, these parameters are updated at the start of each iteration from their posterior distribution given the current states of the N sampling chains. The hyperparameters $\boldsymbol{\Lambda}^{(t)}$ of the posterior distribution at iteration t are given by a deterministic function that computes the summary statistics of the posterior distribution, given $\mathbf{x}$, as in Equation (1.170),

$$\boldsymbol{\Lambda}^{(t)} = \boldsymbol{\Lambda}\left[\mathbf{x}_1^{(t-1)}, \ldots, \mathbf{x}_N^{(t-1)}\right]. \tag{1.170}$$

The proposal parameters are then updated according to Equation (1.171),

$$\boldsymbol{\theta}^{(t)} \sim p\left[\cdot \left| \mathbf{x}^{(t-1)} \right.\right] = p\left[\cdot \left| \boldsymbol{\Lambda}^{(t)} \right.\right]. \tag{1.171}$$

With $\boldsymbol{\theta}^{(t)}$ generated, Keith and Davey (2013) then prescribed that a new state $\mathbf{y}_n$ be proposed for each sampling chain n from the proposal distribution, Equation (1.172).

$$\mathbf{y}_n \sim p\left[\cdot \left| \mathbf{x}^{(t-1)}, \boldsymbol{\theta}^{(t)} \right.\right]. \tag{1.172}$$

BAMS then updates each chain state $\mathbf{x}_n^{(t)}$ in sequence to either $\mathbf{x}_n^{(t-1)}$ or $\mathbf{y}_n$ through a Metropolis-Hastings accept/reject step. To do so, like BAIS, it requires that the significant statistics $\boldsymbol{\Lambda}_n^{(t)}$ of the posterior parameter update distribution be recalculated for each chain, according to Equation (1.173).

$$\boldsymbol{\Lambda}_n^{(t)} = \boldsymbol{\Lambda}\left[\mathbf{x}_1^{(t)}, \ldots, \mathbf{x}_{n-1}^{(t)}, \mathbf{x}_n^{(t-1)}, \ldots, \mathbf{x}_N^{(t-1)}\right]. \tag{1.173}$$

It similarly requires that alternative significant statistics $\boldsymbol{\Lambda}_{n*}^{(t)}$ also be computed in the same manner as $\boldsymbol{\Lambda}_n^{(t)}$ but with $\mathbf{x}_n^{(t-1)}$ replaced with $\mathbf{y}_n$, as in Equation (1.174),

$$\boldsymbol{\Lambda}_{n*}^{(t)} = \boldsymbol{\Lambda}\left[\mathbf{x}_1^{(t)}, \ldots, \mathbf{x}_{n-1}^{(t)}, \mathbf{y}_n, \mathbf{x}_{n+1}^{(t-1)}, \ldots, \mathbf{x}_N^{(t-1)}\right]. \tag{1.174}$$

The acceptance ratio for sampling chain n is then given by Equation (1.175),

$$\begin{aligned} &\alpha_n\left[\mathbf{x}_1^{(t)}, \ldots, \mathbf{x}_{n-1}^{(t)}, \mathbf{y}, \mathbf{x}_{n+1}^{(t-1)}, \ldots, \mathbf{x}_N^{(t-1)}, \boldsymbol{\theta}^{(t)}\right] \\ &= \min\left\{1, \frac{\pi(\mathbf{y}) p\left[\boldsymbol{\theta}^{(t)} \left| \boldsymbol{\Lambda}_n^{(t)} \right.\right] p\left[\mathbf{x}_n^{(t-1)} \left| \mathbf{y}, \boldsymbol{\theta}^{(t)} \right.\right]}{\pi\left[\mathbf{x}_n^{(t-1)}\right] p\left[\boldsymbol{\theta}^{(t)} \left| \boldsymbol{\Lambda}_{n*}^{(t)} \right.\right] p\left[\mathbf{y} \left| \mathbf{x}_n^{(t-1)}, \boldsymbol{\theta}^{(t)} \right.\right]}\right\}. \end{aligned} \tag{1.175}$$

BAMS is summarised in Algorithm A.24 of Appendix A.

1.9 Connections to the Current Study

As mentioned in Section 1.8, the sampling framework developed in Part II of this dissertation will be an extension of BAIS that uses a proposal distribution that can have multiple local maxima, instead of BAIS's, which can only have one. This means that, unlike the other adaptive MCMC algorithms, discussed in Sections 1.6 and 1.7, it explicitly samples new parameters of its proposal distribution from their posterior distribution, given the current states of a population of sampling chains. That is, the parameters are inferred *theoretically* rather than *empirically*, representing an alternative paradigm for adaptive MCMC.

This is not to say that the much-used empirical approach is not without merit. On the contrary, all of the preceding algorithms were shown by their creators to perform well. However, the approach of Part II seeks to further develop the foundations put in place by Keith et al. (2008) in their introduction of BAIS, in order to open up another avenue to adaptation that has, thus far, been largely ignored.

In terms of the problem of constructing a proposal distribution that can have multiple local maxima, Part II will not make recourse to any external adaptive parameters. This sets it apart from the temperature- and energy-based methods of Section 1.4.3. As such, it will not suffer from the practical difficulty of having to determine a complex cooling schedule before a simulation is run.

Section 1.5 outlined the merits of approximation by finite mixtures and discussed best practices in terms of selection of the number of components. Its details must be kept in mind when working with such distributions and, in particular, the selection methods for the number of mixture components discussed in Section 1.5.3 will play a role in Chapter 6.

Finally, the theory reviewed in Section 1.2 will be used to provide a rigorous justification of the methods developed in this dissertation.

1.9.1 Scientific Contribution of the Current Study

The methods and studies presented in Part II of the current dissertation supplement the adaptive MCMC methods reviewed in Sections 1.6 and 1.7 in the form of two new MCMC sampling methods. Like the methods reviewed in Section 1.7, these methods tackle the problem of sampling from target distributions with many local maxima, which was discussed in Section 1.4, by incorporating mixture approximations (cf. Section 1.5) as proposal distributions inside an adaptive MCMC framework. The novelty in the methods of this dissertation lies in the extension of the framework put in place by Keith et al. (2008) in their introduction of BAIS, proposing samples from a mixture proposal. This dissertation also provides theoretical and practical justification of the study of these new techniques.

Chapter 2

Applications of Monte Carlo and Markov Chain Monte Carlo

This chapter reviews applications of MC and MCMC methods.

The first section of this chapter considers test functions used in the literature to assess the performance of earlier MCMC and optimisation methods. These test functions will appear later in this dissertation, where they will be sampled, in order to test the performance of the samplers that will be introduce in Chapters 3 and 6.

The second section reviews the regression model studied by Kou et al. (2006), which will be used to compare the method of Chapter 3 to the EES and to demonstrate guidelines developed for its implementation.

Finally, the third section reviews spin glass simulation, which is suggested as a potential application for the new method of Chapter 5. It begins with a description of what a spin glass is, followed by how they are modelled in the literature. The section finishes with examples of computational techniques used to study spin glasses.

2.1 Test Functions

This section reviews several existing test functions from the literature that are particularly suited to the needs of the current study. Specifically, these functions all have multiple local maxima, which makes them suitable for testing methods that are designed to sample from such distributions.

The first portion of the current section reviews a mixture target distribution to which Kou et al. (2006) applied EES. The second portion then describes a set of optimisation test functions collected by Storn and Price (1997), which will be used in Chapter 6 as targets for the comparison of the two methods that will developed in this current work.

2.1.1 A Mixture Target

Kou et al. (2006) illustrated EES (cf. Section 1.4.3) by sampling from a mixture target. The distribution that they studied was an example from Liang and Wong (2001) with the expression

$$f(\mathbf{x}) = \sum_{i=1}^{20} \frac{2.5}{\pi} \exp\left\{-50\left(\mathbf{x}-\boldsymbol{\mu}_i\right)^{\mathrm{T}}\left(\mathbf{x}-\boldsymbol{\mu}_i\right)\right\},$$

where

$$\begin{aligned}
\boldsymbol{\mu}_1 &= \begin{pmatrix}2.18\\5.76\end{pmatrix}, & \boldsymbol{\mu}_2 &= \begin{pmatrix}8.67\\9.59\end{pmatrix}, & \boldsymbol{\mu}_3 &= \begin{pmatrix}4.24\\8.48\end{pmatrix}, & \boldsymbol{\mu}_4 &= \begin{pmatrix}8.41\\1.68\end{pmatrix},\\
\boldsymbol{\mu}_5 &= \begin{pmatrix}3.93\\8.82\end{pmatrix}, & \boldsymbol{\mu}_6 &= \begin{pmatrix}3.25\\3.47\end{pmatrix}, & \boldsymbol{\mu}_7 &= \begin{pmatrix}1.70\\0.50\end{pmatrix}, & \boldsymbol{\mu}_8 &= \begin{pmatrix}4.59\\5.60\end{pmatrix},\\
\boldsymbol{\mu}_9 &= \begin{pmatrix}6.91\\5.81\end{pmatrix}, & \boldsymbol{\mu}_{10} &= \begin{pmatrix}6.87\\5.40\end{pmatrix}, & \boldsymbol{\mu}_{11} &= \begin{pmatrix}5.41\\2.65\end{pmatrix}, & \boldsymbol{\mu}_{12} &= \begin{pmatrix}2.70\\7.88\end{pmatrix},\\
\boldsymbol{\mu}_{13} &= \begin{pmatrix}4.98\\3.70\end{pmatrix}, & \boldsymbol{\mu}_{14} &= \begin{pmatrix}1.14\\2.39\end{pmatrix}, & \boldsymbol{\mu}_{15} &= \begin{pmatrix}8.33\\9.50\end{pmatrix}, & \boldsymbol{\mu}_{16} &= \begin{pmatrix}4.93\\1.50\end{pmatrix},\\
\boldsymbol{\mu}_{17} &= \begin{pmatrix}1.83\\0.09\end{pmatrix}, & \boldsymbol{\mu}_{18} &= \begin{pmatrix}2.26\\0.31\end{pmatrix}, & \boldsymbol{\mu}_{19} &= \begin{pmatrix}5.54\\6.86\end{pmatrix}, & \boldsymbol{\mu}_{20} &= \begin{pmatrix}1.69\\8.11\end{pmatrix}.
\end{aligned}$$

In their simulations, Kou et al. (2006) used a minimum energy of 0.2, five temperatures and an equi-energy jump probability of 0.1. They initialised their simulation by drawing the initial states from a uniform distribution on the unit square $[0,1]^2$ and set the initial MH step size for chain n to $0.25\sqrt{T_n}$, where T_n is the temperature associated with the nth energy truncation level in the EES (cf. Section 1.4.3). After burn-in they ran the sampler for 50000 iterations. This was repeated 20 times. In total each simulation resulted in 250000 recorded samples.

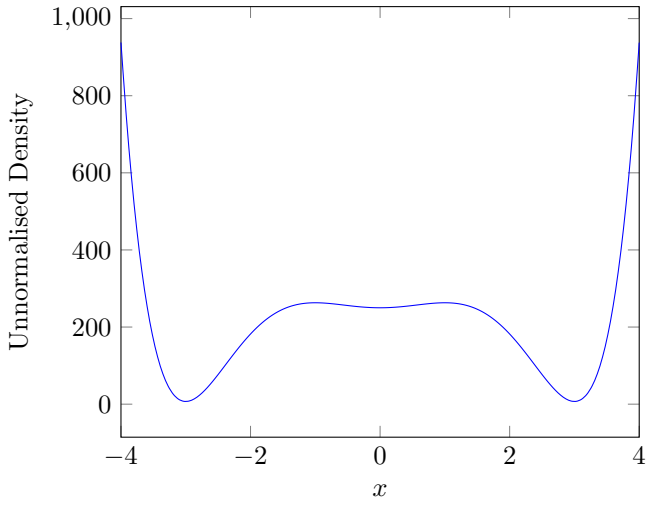
This target was chosen for its multiple local maxima. As such, it has the potential to exhibit the quasi-ergodic problem (cf. Section 1.2.2) if not sampled carefully. Having also been used to test the EES by Kou et al. (2006), results were already available for a direct comparison against the method developed in Chapter 3 of this dissertation.

2.1.2 An Optimisation Test Bed

Functions used for testing optimisation algorithms are particularly interesting because they are usually designed to exhibit many local extreme points but only one global one. This makes them challenging to optimise because an inefficient optimisation strategy may become stuck in any one of these numerous local extremes; not just the global one. These functions may be converted into densities by treating them as an energy function inside a Boltzmann distribution or by restricting their supports. For the same reasons they are difficult to optimise, they are also difficult to sample.

This section reviews one such class of test functions, collected by Storn and Price (1997) for demonstrating their differential evolution algorithm for optimisation. They considered in total 30 different types of test functions, which they optimised for a range of different values of their respective parameters. Not all of these functions were appropriate for the needs of the current study, so this section only discusses those that were converted to probability densities for sampling in Part II.

Figure 2.1: Goldstein function on the domain $[-4, 4]$.



Four functions were selected from test beds 1, 2 and 3 of Storn and Price (1997), ranging in dimension from one to three. The subscripts on the functions correspond to those given by Storn and Price (1997).

The first of these test targets was adapted by Storn and Price (1997) from the De Jong functions (De Jong, 1975) into their first test bed. It was the univariate Goldstein function, which Storn and Price (1997) initialised on the interval $[-10, 10]$. To convert this function into a density, it was restricted to the smaller domain of $[-4, 4]$ to give the target in Equation (2.1).

$$p_{16}(x) \propto \begin{cases} x^6 - 15x^4 + 27x^2 + 250, & \text{if } x \in [-4, 4] \\ 0, & \text{otherwise.} \end{cases} \tag{2.1}$$

On this restricted domain the resulting density has a maximum at each extreme (cf. Figure 2.1). Between these two extremes the function decreases rapidly, with a small double-hump in the middle, as demonstrated in Figure 2.1.

The second modified De Jong function considered corresponds to function 5 of Storn and Price (1997). This function is Shekel's foxholes, which is a two-dimensional function with 25 local minima. These minima make the function suitable for testing a sampler designed to study distributions with multiple local maxima. To convert this function into a density, its support was restricted to $(x_1, x_2) \in [-40, 40]$.

The formula for Shekel's foxholes is stated in Equation (2.2).

$$p_5(x_1, x_2) \propto \begin{cases} \dfrac{1}{0.002 + \sum_{i=0}^{24} \dfrac{1}{i + [x_1 - a_{(i \mod 5)+1}]^6 + (x_2 - a_{\lfloor i/5 \rfloor + 1})^6}}, & \text{if } x_1, x_2 \in [-40, 40] \\ 0, & \text{otherwise.} \end{cases} \tag{2.2}$$

In Equation (2.2) $\mathbf{a} = (a_i)_{i=1}^5 = (-32, -16, 0, 16, 32)$, " mod " is the modulo operator and $\lfloor \cdot \rfloor$ is the floor function.

Figure 2.2: Filled contour plot of Shekel's foxholes function on the domain $[-40, 40]^2$.

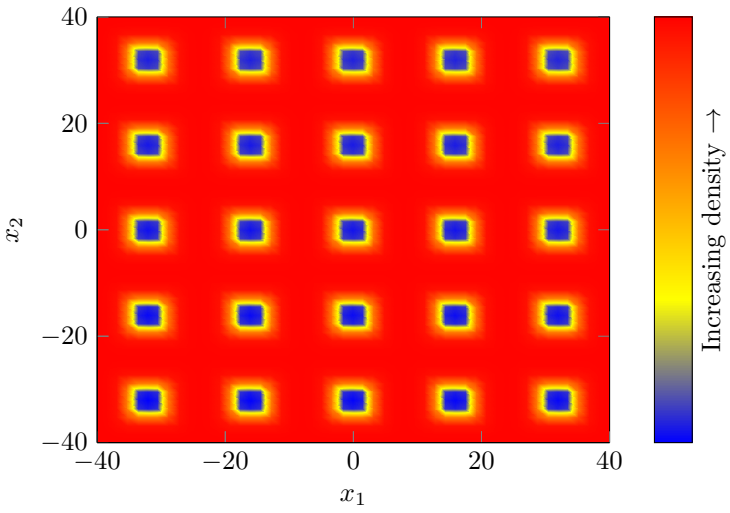


Figure 2.2 sketches Shekel's foxholes as a filled contour plot. The darker regions in Figure 2.2 represent regions of lower density.

Finally considered, were the modified versions of Rastrigin's and Ackley's (Ackley, 1987, pp. 13–14) functions in $p = 1$, $p = 2$ and $p = 3$ dimensions. Their formulae are given by Equations (2.3) and (2.4), respectively,

$$p_{13}(\mathbf{x}) \propto 10p + \sum_{d=1}^{p} \left[x_d^2 - 10\cos(2\pi x_d)\right], \tag{2.3}$$

$$p_{15}(\mathbf{x}) \propto -20\exp\left(-\frac{1}{50}\sqrt{\frac{1}{p}\sum_{d=1}^{p} x_d^2}\right) - \exp\left[\frac{1}{d}\sum_{d=1}^{p}\cos\left(2\pi x_d\right)\right] + 20 + \exp(1). \tag{2.4}$$

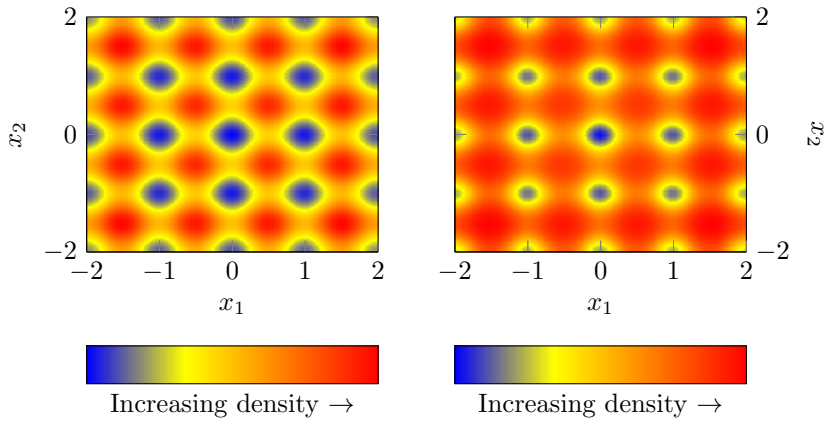
These two functions are very similar, with multiple local maxima of varying height in the same locations.

Figure 2.3 demonstrates the two-dimensional versions of Rastrigin's and Ackeley's functions.

While "flipping" the preceding targets (so that maxima become minima and minima become maxima) may have posed a greater sampling challenge, this would have entailed further changes to their expressions. Since the difficulty of sampling these targets was not of primary concern in this dissertation, this modification was not implemented.

Like the target of Section 1.4, these test functions were selected for their multiple or non-trivial local maxima and because they provide further examples for testing the developed methods of this dissertation.

Figure 2.3: Rastrigin's function in two dimensions on the domain $[-2, 2]$ (left) and Ackley's function in two dimensions on the domain $[-2, 2]$ (right).



2.2 Mixture Exponential Regression

The mixture exponential regression problem studied by Kou et al. (2006) was also considered, as a further comparison against the equi-energy sampler and to illustrate the guidance given in Part II of this dissertation.

Kou et al. (2006) generated $M = 200$ data pairs $(y_i, x_i)_{i=1}^{M} \in \mathbb{R}^2$ with the following distributions,

$$x_i \sim \mathcal{U}(0, 2)$$
$$y_i \sim \begin{cases} \text{Exp}\left[\exp(\beta_{11} + x_i\beta_{12})\right], & \text{if } z_i = 0 \\ \text{Exp}\left[\exp(\beta_{21} + x_i\beta_{22})\right], & \text{if } z_i = 1 \end{cases},$$

where $z_i = 1$ with probability α and 0 otherwise, $\mathcal{U}(a, b)$ is the uniform distribution on the interval $[a, b]$ and $\text{Exp}(\lambda)$ represents an exponential distribution with parameter λ.

In this model the parameters of interest are $\alpha, \beta_{11}, \beta_{12}, \beta_{21}$ and β_{22}.

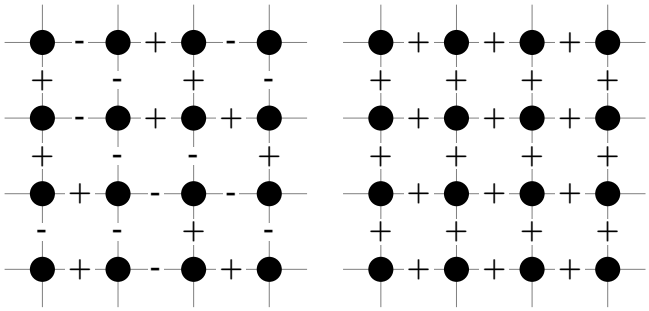
In order to produce the M data pairs, Kou et al. (2006) set the true values of these parameters to be $\alpha = 0.3, \beta_{11} = 1, \beta_{12} = 2, \beta_{21} = 4$ and $\beta_{22} = 5$.

By setting a beta prior on α with shape 1 and scale 1 and a normal prior with mean 0 and variance $\sigma^2 = 100$ on each of the βs, Kou et al. (2006) derived the log-posterior density of the model parameters to be

$$\sum_{i=1}^{M} \log\left\{\frac{\alpha}{\exp(\beta_{11} + x_i\beta_{12})} \exp\left[-\frac{y_i}{\exp(\beta_{11} + x_i\beta_{12})}\right] + \frac{\alpha}{\exp(\beta_{21} + x_i\beta_{22})} \exp\left[-\frac{y_i}{\beta_{21} + x_i\beta_{22}}\right]\right\} - \frac{1}{2\sigma^2}(\beta_{11}^2 + \beta_{12}^2 + \beta_{21}^2 + \beta_{22}^2) + \log(C),$$

where C is the normalising constant of the posterior density.

Figure 2.4: An example of a disordered magnet (left) and a ferromagnet (right). These examples do not show the orientations of the spins (black circles), only whether two spins on an edge prefer to be the same (+) or opposite (−).



They used the equi-energy sampler to simulate drawing from this distribution, using an initial minimum temperature of −1740.8 and eight sampling chains.

2.3 Spin Glass Simulation

Spin glasses represent an important problem of computational interest in theoretical physics. The most common mathematical models used to describe them have simple descriptions but the resulting probability distributions over their configuration space have many local minima, providing a challenging task for MC methods (Fischer and Hertz, 1991, pp. 130–131). Therefore, spin glasses present an ideal application for algorithms that are designed to sample from distributions with many local maxima, such as the one introduced in Part II of the current dissertation.

2.3.1 Background

Spin glasses are disordered magnets, which, on a microscopic scale, have a "frozen" random arrangement of interactions between the spins of their constituent particles (Fischer and Hertz, 1991, pp. 2). When an interaction is satisfied, the resulting energy of the system is lower and hence, more preferable, than when it is not. This random arrangement of spin preferences, demonstrated in the left-hand lattice of Figure 2.4, contrasts with the aligned preference of ferromagnets (right-hand lattice of Figure. 2.4) and the anti-aligned preferences of antiferromagnetic interactions.

Due to the random arrangement of ferromagnetic and antiferromagnetic interactions, local competition between neighbouring spin directions is present. This competition leads to *frustration* (Toulouse, 1977), where not all interactions can be simultaneously satisfied. Such frustration is exemplified in simple terms by a triangle with two ferromagnetic interactions and one antiferromagnetic one, as depicted in Figure 2.5. Not all of the spin preferences can be satisfied at once, with at least one of them being unsatisfied. To see this, note that if $a = b = c$, then the two + bonds will be satisfied but then $a = c$, making the − bond unsatisfied. Similarly, if $a = b \neq c$ then the bonds between a and

Figure 2.5: An example of spin frustration.

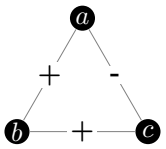
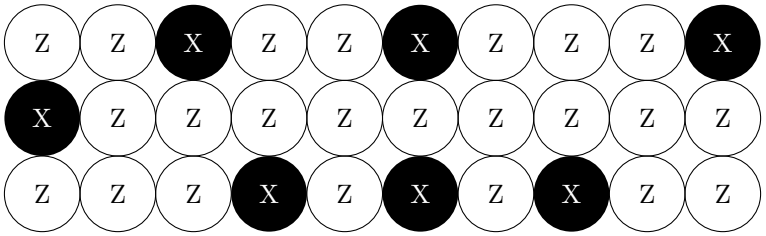


Figure 2.6: A simplified illustration of a dilute alloy of a paramagnetic solute X inside a diamagnetic matrix of element Z.



b and between a and c will be satisfied but the one between b and c will not, with the same problem arising if $b = c \neq a$. The complexity induced by frustration leads to questions regarding its effect on the macroscopic properties of such systems, both from theoretical and experimental perspectives.

Researchers have investigated spin glass behaviour experimentally by testing a variety of dilute metallic alloys whose microscopic structures involve competing interactions. Examples of such alloys include paramagnetic manganese atoms in a copper matrix (e.g. Anderson (1970)), paramagnetic vanadium atoms in a gold matrix (e.g. Beck (1971)) or ferromagnetic iron atoms in a gold (e.g. Cannella and Mydosh (1972)) or aluminium (e.g. Beck (1971)) matrix.

These examples all involve a ferromagnetic or paramagnetic metal solute X inside a matrix of a purely diamagnetic one Z, as illustrated in Figure 2.6. While the spins of the electrons of the solute (the para- or ferromagnetic element in lower quantity) will tend to orient themselves with an externally-applied magnetic field, the spins of the electrons of the matrix (the diamagnetic metal of higher quantity) will tend to orient against that same field.

A key property of spin glasses that makes them an interesting subject of study, is the cusp in their magnetic susceptibility, demonstrated by Cannella and Mydosh (1972) for iron in gold.

From a theoretical perspective the focus lies in inferring the properties of spin glass models by averaging them over all possible configurations and random couplings of a model in question. This requires sampling from the probability distribution of configurations, which is given by a Boltzmann distribution,

$$p(\mathbf{s}) \propto \exp\left[-\frac{h(\mathbf{s})}{T}\right], \tag{2.5}$$

where $\mathbf{s}$ is the configuration of spins, the Hamiltonian $h(\mathbf{s})$ is the energy associated with configuration $\mathbf{s}$ and T is the temperature of the simulation. For a physical temperature, T should be replaced by $k_B T$, where k_B is Boltzmann's constant. Since this replacement would only constitute a change of units of T, the current work will continue to use T, for ease of exposition.

The normalising constant $Z(T)$ of this distribution is known as its *partition function* in the statistical mechanics literature and is described at temperature T by Equation (2.6)

$$Z(T) = \sum_{\mathbf{s} \in \mathbb{S}} \exp\left[-\frac{h(\mathbf{s})}{T}\right], \tag{2.6}$$

where $\mathbb{S}$ is the space of all possible spin configurations.

The partition function, from which important thermodynamic properties of the system of study may be derived (Baxter, 1982, pp. 8), is generally not readily computed in an exact form (Baxter, 1982, pp. 9). Instead, quantities that are derived from it must be approximated numerically, using methods such as MC or MCMC (cf. Section 2.3.4). Researchers are especially interested in their computation as the temperature is reduced to zero, where, due to the form of the configurational distribution, any local maxima in the distribution are amplified, making sampling progressively more difficult.

The rest of this section reviews mathematical models of magnets, starting with those of ferromagnets. It then progresses onto the most studied models of spin glasses and their relationships with their real-world counterparts, before reviewing methods that have been proposed to sample from their configurational distributions.

2.3.2 Ferromagnetic Models

This section takes a look at three models of interacting spins on a regular lattice. It begins with a review of the Ising model, which has a simple description, before discussing two more general models, which have more realistic but more complex descriptions.

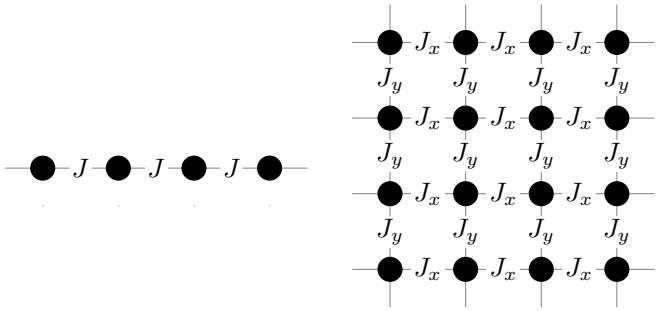
The Ising Model

The Ising model, introduced by Lenz (1920), is a simplified representation of the spin interactions on a ferromagnetic lattice, in which each spin is oriented in one of two opposing directions, denoted by $+1$ and -1. The specific form of its partition function was derived by Lenz's student Ising (1925) for the one-dimensional case in zero magnetic field (described below). The partition function that he found was continuously differentiable with respect to temperature $T \in \mathbb{R}^+$ (cf. Equation 2.8), indicating that the Ising model does not exhibit a phase transition in one dimension.

What follows is a review of the description of the one- and two-dimensional Ising models in zero magnetic field, given by Onsager (1944), who derived the partition function in two dimensions, subsequently deriving an expression for the temperature T_c at which the predicted phase transition (Peierls, 1936) takes place.

Onsager (1944) started with a review of the one-dimensional model studied by Ising (1925), where spins are modelled linearly, as in the lattice on the left-hand side of Figure 2.7. The circles represent the spins, which each equal ± 1, and the lines between them, the interaction energies of the pairs, which equal $\pm J$. The sign before the coupling constant J depends on the states of the interacting spins and is positive if they are the same and negative otherwise.

Figure 2.7: An Ising spin glass in one dimension (left) and in two dimensions (right). Note that in two dimensions the coupling strength between neighbouring atoms may differ in each of the dimensions.



The Hamiltonian corresponding to the N-spin instance of this model is given by Equation (2.7),

$$h(\mathbf{s}) = -J \sum_{n=1}^{N} s_n s_{n+1}, \tag{2.7}$$

where $\mathbf{s} = (s_1, \ldots, s_N)$ is the vector of spin states. Thus, the partition function for an N-spin one-dimensional system is given by Equation (2.8),

$$Z(T) = \sum_{\mathbf{s} \in \{\pm 1\}^N} \exp\left(-\frac{J}{T} \sum_{n=1}^{N-1} s_n s_{n+1} \right). \tag{2.8}$$

The infinite two-dimensional case is illustrated on the right-hand side of Figure 2.7, with separate coupling constants J_x and J_y in the horizontal and vertical directions, respectively.

Onsager (1944) provided a detailed derivation of the corresponding partition function and showed that a phase transition occurs at a critical temperature T_c satisfying Equation (2.9)

$$\sinh\left(\frac{2J_x}{T_c}\right) \sinh\left(\frac{2J_y}{T_c}\right) = 1, \tag{2.9}$$

where the system transitions from an ordered state above T_c to a disordered one below it.

This result extends one of Kramers and Wannier (1941), who found the transition temperature for the case where $J_x = J_y = J$ to be given by Equation (2.10),

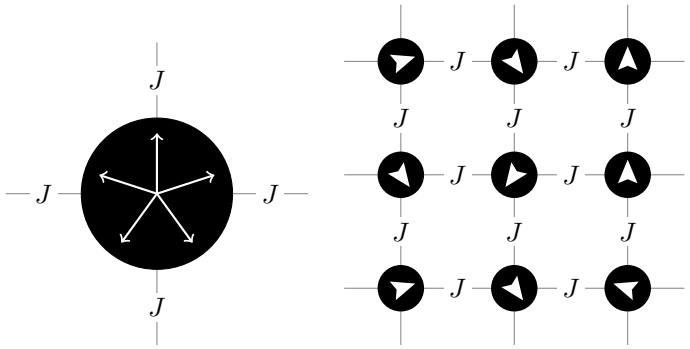
$$T_c = \frac{0.8814}{J}. \tag{2.10}$$

The Potts Model

A generalisation of the two-dimensional Ising model is the Potts model, due to Potts (1952), which generalises a more specific model that was introduced by Ashkin and Teller (1943).

Unlike the Ising model, which only considers two types of spins, namely + and −, on a regular lattice, the Potts model considers r different spin types,

Figure 2.8: A single lattice site in a Potts model with five possible spin directions, indicated by vectors. Adjacent spin interact with coupling strength $J = J_0$ when they are aligned and $J = J_1$ otherwise.



occurring in equal proportions, on any graph. Potts (1952) represented each spin type as a vector pointing in one of r directions, equal angles apart, as illustrated in Figure 2.8. He then defined two different interaction energies, J_0 for spins pointing in the same directions and J_1 for those pointing in different ones.

The $r = 2$ case, with $J_0 = -J_1$, is simply the two-dimensional Ising model, for which Kramers and Wannier (1941) discovered the critical point given at the end of the previous subsection. The case where $r = 4$ corresponds to the model studied by Ashkin and Teller (1943).

Potts (1952) also mentioned a more general model, where he stated that the interaction between nearest neighbours is "proportional to the scalar product of the vectors representing them". That is, the interaction is represented by a rescaling $J\mathbf{s}_i \cdot \mathbf{s}_j$ of the dot product between spins i and j, represented, by vectors $\mathbf{s}_i$ and $\mathbf{s}_j$, respectively. This relates to the more specific case described above, where J_0 and J_1 are expressed by Equations (2.11) and (2.12), respectively,

$$J_0 = J \tag{2.11}$$
$$J_1 = J\mathbf{s}_i \cdot \mathbf{s}_j. \tag{2.12}$$

The limiting case of this more general model as $r \to \infty$ describes a model, in which spins may point in any direction and where every interaction in the lattice has a component in each lattice direction.

The Heisenberg and n-Vector Models

The Heisenberg model is a three-dimensional description of a lattice of interacting spins, first introduced by Heisenberg (1928), in which each spin may point in any direction. As such, a spin is depicted as a unit 3-vector.

For general dimension n Stanley (1968) introduced the *n-vector* model, which has the Hamiltonian given in Equation (2.13),

$$H(\mathbf{s}) = -2J \sum_{\langle i,j \rangle} \mathbf{s}_i \cdot \mathbf{s}_j, \tag{2.13}$$

where $\cdot$ represents the dot product and $\mathbf{s}_i, \mathbf{s}_j \in \mathbb{R}^n$ are the unit-length spin vectors at lattice sites i and j, respectively. He noted that $n = 1$ describes the Ising model and $n = 3$, the Heisenberg model.

2.3.3 Spin Glass Models

To describe characteristic properties of spin glasses, a number of mathematical models have been proposed in the literature. The most studied of these, due to their ease of description and their agreement with experimental results, are related to the ferromagnetic models of interacting spins reviewed in the previous section. Unlike those models, which their corresponding authors studied using a constant interaction J, in spin glass models this interaction is allowed to vary across neighbouring pairs. The two most commonly studied such descriptions of the spin glass are the Edwards-Anderson (EA) model and the Sherrington-Kirkpatrick (SK) model, which involve Ising spin interactions.

This section reviews both of these models.

The Edwards-Anderson Model

Edwards and Anderson (1975) proposed a model of spin glasses, to explain a cusp in the magnetic susceptibility, discovered by Cannella and Mydosh (1972). Their model considers a p-dimensional lattice of N spin sites, in which the interaction between sites i and j, indicated by J_{ij}, is a random variable with known distribution. The resulting Hamiltonian $h(\mathbf{s})$ of the configuration $\mathbf{s}$, is then given by Equation (2.14) (Fischer and Hertz, 1991, pp. 19),

$$h(\mathbf{s}) = -\frac{1}{2} \sum_{\langle i,j \rangle} J_{ij} s_i s_j, \tag{2.14}$$

where $\sum_{\langle i,j \rangle}$ represents a sum over all interacting lattice sites and $\mathbf{s} = (s_1, \ldots, s_N)$ is a spin configuration.

Edwards and Anderson (1975) assumed the net interaction between all spins to be zero on any scale,

$$\sum_{i,j} J_{ij} = 0.$$

They considered the relationship between two independent realisations of the spin configurations, a and b, for the same set of interactions, defining an order parameter (Stein and Newman, 2013, Sec. 4.8),

$$q_{\mathrm{EA}} = \left\langle s_i^{(a)} s_i^{(b)} \right\rangle = \lim_{N \to \infty} \frac{1}{N} \sum_{i=1}^{N} \langle s_i \rangle^2 ,$$

where $\left\langle s_i^{(a)} s_i^{(b)} \right\rangle$ represents a sample average of $s_i^{(a)} s_i^{(b)}$ over all lattice sites i and all replica pairs a, b, while $\langle s_i \rangle$ represents a sample average of the state of the ith spin in a lattice with N spin sites.

Edwards and Anderson (1975) proceeded from the argument that for a spin glass, in the absence of an external magnetic field, $q_{\mathrm{EA}} = 0$ above some critical temperature T_c, indicating no net macroscopic magnetisation, but below it $q_{\mathrm{EA}} \neq 0$, indicating a net latent magnetisation. They noted that the group

of A. T. Fiory observed such a change in experimental investigations (Murnick et al., 1976), which they attributed to "freezing" of the spin directions.

Using their definition of q_{EA}, its experimentally-observed nature above and below the transition temperature and the expression of T_c, Edwards and Anderson (1975) derived a number of thermodynamic properties, including the existence of a cusp in magnetic susceptibility as observed by Cannella and Mydosh (1972), and the specific heat of the transition at T_c.

The Sherrington-Kirkpatrick Model

Sherrington and Kirkpatrick (1975) considered a spin glass model consisting of infinite-range interactions between N Ising spins $\mathbf{s} = (s_1, \ldots, s_N) \in \{\pm 1\}^N$ (cf. Section 2.3.2), with Hamiltonian given by Equation (2.15),

$$h(\mathbf{s}) = -\frac{1}{2} \sum_{i \neq j} J_{ij} s_i s_j. \tag{2.15}$$

In their model Sherrington and Kirkpatrick (1975) specified a Gaussian distribution over the interactions by Equation (2.16),

$$p(J_{ij}) = \frac{1}{\sqrt{2\pi J^2}} \exp\left[-\frac{(J_{ij} - J_0)^2}{2J^2} \right]. \tag{2.16}$$

As in the more general model of Edwards and Anderson (1975), Sherrington and Kirkpatrick (1975) considered multiple configurations or "replicas" with the same set of couplings. They specified two order parameters, m and q, given by Equations (2.17) and (2.18), respectively,

$$m := \mathbb{E}_J \left[\langle s_i \rangle \right], \tag{2.17}$$

$$q := \mathbb{E}_J \left[\langle s_i \rangle^2 \right], \tag{2.18}$$

where $\langle \cdot \rangle$ represents a configurational average and $\mathbb{E}_J$ represents an average over the sets of couplings.

Studying their model down to the limit of $n = 0$ replicas and averaging over all possible sets of couplings, they non-rigorously demonstrated that m and q are independent of the lattice site i. They further explained the significance of these two variables, noting that magnetic order results when $q \neq 0$. When there is order present, Sherrington and Kirkpatrick (1975) interpreted $m = 0$ as indication of the ferromagnetic phase, and $m \neq 0$ as an indication of the spin glass phase.

Sherrington and Kirkpatrick (1975) then used the preceding formulation to describe thermodynamic properties of their model with respect to "intensive" (scale-independent) variables $\tilde{J}_0 = J_0 N$ and $\tilde{J} = J\sqrt{N}$.

2.3.4 Computer Simulation

Glauber Dynamics and Single-Spin-Flip MC

An early computational approach to simulating spin glasses was that of single-spin-flip dynamics, due to Glauber (1963). This approach is essentially an early version of Gibbs sampling, which was reviewed in Section 1.3.2.

Specifically, Glauber (1963) considered a one-dimensional closed loop of N spins that are either up or down ($s_n = \pm 1, n \in \{1, \ldots, N\}$). His model then updated each spin n individually, while keeping all others fixed, by considering the constant probability p_n of a sign-change in the spin's value (due to its interaction with an external "heat reservoir"), as well as its interactions ("correlations") with the other spins in the system. The "master" differential equation of his model is given by Equation (2.19),

$$\frac{d}{dt}\mathbb{P}_t\left(s_1, \ldots, s_N\right) = -\mathbb{P}_t\left(s_1, \ldots, s_N\right) \sum_{n=1}^{N} p_n\left(s_n\right) + \sum_{n=1}^{N} p_n\left(-s_n\right) \mathbb{P}_t\left(s_1, \ldots, s_{n-1}, -s_n, s_{n+1}, \ldots, s_N\right). \tag{2.19}$$

For a sequence of discrete-time spin updates in an Ising model, as considered in Chapter 5, the flip at lattice site n is effectively a Metropolis move, with proposal that selects the configuration with all spins the same, except for the nth. A standard Metropolis accept-reject step (cf. Section 1.3.1) completes the move, with probability α given by the ratio of densities with respect to the target (Glauber, 1963), according to Equation (2.20),

$$\alpha = \min\left\{1, \frac{\exp[-(J/T)s_n \sum_{i \in \mathcal{V}(n)} s_i}{\exp[(J/T)s_n \sum_{i \in \mathcal{V}(n)} s_i}\right\}, \tag{2.20}$$

where J is the interaction strength, s_n is the spin being updated and $\mathcal{V}(n)$ is the set of indices of the neighours of site n.

Section 5.2.1 considers a multi-spin updating approach using the main MCMC method developed in this dissertation, which will be compared to single-spin-flip in Section 5.3.

The Swendsen-Wang Algorithm

Instead of performing single-spin flips in a Monte Carlo simulation of a system of interacting spins, Swendsen and Wang (1987) introduced an algorithm to cluster the spins in a given configuration into N_c distinct clusters. They achieved this goal on the basis of the existence of satisfied spin preferences between neighbouring spins.

Swendsen and Wang (1987) described their approach for a (ferromagnetic) Potts model but noted that it extended to spin glasses in a straightforward manner. In Chapter 5, where spin glasses will be explored further, the focus will be on Ising spin glasses, so its algorithm is described here in that context.

The Swendsen and Wang (1987) algorithm considers a Hamiltonian over the system. For an Ising spin glass this Hamiltonian is given by Equation (2.21),

$$h(\mathbf{s}) = -\frac{1}{T} \sum_{\langle i,j \rangle} J_{ij}\left(s_i s_j - 1\right), \tag{2.21}$$

where J_{ij} is the coupling energy between spins i and j, based on their respective states s_i and s_j.

Given a set of couplings $\mathbf{J}$ and a configuration $\mathbf{s} = (s_1, \ldots, s_N)$, the algorithm of Swendsen and Wang (1987) proceeds by first randomly assigning bonds

between neighbouring spins that have a satisfied interaction. The probability of assigning a bond between satisfied spins i and j is thus given by Equation (2.22),

$$\mathbb{P}_{\text{bond}}(s_i, s_j, J_{ij}) = 1 - \exp\left(-\frac{1}{T}\max\{J_{ij}s_is_j, 0\}\right), \tag{2.22}$$

where T is the temperature.

The result of performing the preceding assignments is a partition of the lattice into $N_c \leq N$ disjoint clusters that covers the entire lattice.

Swendsen and Wang (1987) justified their algorithm on the premise of Fortuin and Kasteleyn (1972), who demonstrated that a Potts (and, hence, Ising) model may be represented as a percolation model. The unnormalised probability of a particular clustering is, therefore, given by Equation (2.23) (Fortuin and Kasteleyn, 1972),

$$\begin{aligned}\mathbf{P}_{\text{clustering}}(\mathbf{s}, \mathbf{J}, \mathbf{B}) \propto \sum_{i=1}^{N} \prod_{j\in\{k:k\in\mathcal{V}(i),k>i\}} &[\mathbb{P}_{\text{bond}}(s_i, s_j, J_{ij})]^{B_{i,j}} \\ &\times [1 - \mathbb{P}_{\text{bond}}(s_i, s_j, J_{ij})]^{1-B_{i,j}},\end{aligned} \tag{2.23}$$

where $\mathbf{B} = \{\{B_{ij}\}_{j\in\{k:k\in\mathcal{V}(i),k>i\}}\}_{i=1}^{N}$ is a set of indicators, such that $B_{ij} = 1$ if there is a bond between lattice sites i and j and 0 otherwise. Note that $k > i$ in the product discounts double-consideration of any interaction.

In the case of a p-dimensional Ising model, Fortuin and Kasteleyn (1972) gave a simple formula for the partition function, which is restated in Equation (2.24),

$$Z = \sum_{\mathbf{B}\in\{0,1\}^{pN}} q^{\sum\mathbf{B}}(1-q)^{pN-\sum\mathbf{B}}2^{N_c(\mathbf{B})}, \tag{2.24}$$

where $\sum\mathbf{B}$ represents the total number of bonds, $N_c(\mathbf{B})$ is the number of clusters induced by bonds $\mathbf{B}$ and $q = 1 - \exp[-J/T]$ is the common probability of a bond between neighbouring spins of the same orientation. This probability is not common between lattice sites in a spin glass, which complicates the computation of the partition function.

Algorithm A.25 of Appendix A summarises the clustering algorithm of Swendsen and Wang (1987).

In their introduction of replica MC, Swendsen and Wang (1986) suggested not using a Hamiltonian over interacting spins when using a cluster-updating approach, but one over neighbouring clusters, in the form of a cluster Hamiltonian. The same approach applies to the Swendsen-Wang cluster model, in which case, the cluster Hamiltonian is given by Equation (2.25),

$$H_{cl} = -\frac{1}{T}\sum_{\langle c,d\rangle} K_{cd} \tag{2.25}$$

where the effective coupling K_{cd} is equal to the sum of all interactions between clusters c and d.

Formally, for a spin glass, let C_i be the cluster to which lattice site i is assigned and set $I_c(C_i) = 1$ if lattice site i is in cluster c and 0 otherwise. Then K_{cd} is defined according to Equation (2.26),

$$K_{cd} = \sum_{i\in N}\sum_{j\in\{k\in\mathcal{V}(i):k>i\}} J_{ij}s_is_jI_c(C_i)I_d(C_j). \tag{2.26}$$

Since all spin preferences inside a single cluster are satisfied, there are only two possible sets of orientations, which differ by total spin reversal. Call these two sets $\mathbf{a}_c$ and $\mathbf{a}'_c$, where $\mathbf{a}_c = -\mathbf{a}'_c$.

To use the Swendsen-Wang algorithm inside a Metropolis sampler is analogous to the type of single-spin-flip approach discussed at the end of the previous subsection, but instead of updating one spin at a time, it updates one cluster at a time. In this case, a flip is proposed of the spin at each lattice site within the cluster. The probability of accepting this flip is then computed in much the same manner as the probability of a spin flip. That is, cluster c is updated in a Metropolis move, with the proposed configuration always being the same as the current one but with the cth cluster flipped. It is then accepted with a standard Metropolis acceptance ratio.

Section 5.2.2 will introduce the foundations for an alternative approach to Swendsen-Wang cluster-updating, which may be used to propose new states in the main MCMC sampler introduced in this dissertation.

Part II

BAIS with Latent variables (BAIS+L)

This part of the dissertation introduces and discusses a novel approach to sampling from distributions with many local maxima: the *Bayesian Adaptive Independence Sampler with Latent variables* (BAIS+L). This sampler extends the Bayesian Adaptive Independence Sampler of Keith et al. (2008), by utilising a more flexible proposal distribution, in the form of a mixture of normals.

Chapter 3 begins with a detailed development of the new sampler, including a discussion of an approximation that makes the sampler possible. It outlines the quantities necessary for its implementation and which are used to define its algorithm. Chapter 3 also provides sufficient conditions for uniform ergodicity, with proofs of their sufficiency. Considerations of how these sufficient conditions may be enforced are presented, along with a study of their effectivenss. Chapter 3 concludes with some conjectured properties of BAIS+L.

Chapter 4 compares BAIS+L to the equi-energy sampler of Kou et al. (2006). It does so by using BAIS+L to sample the same targets in Euclidean state space that were studied by Kou et al. (2006). This chapter explores the relative efficiency of BAIS+L, as well as its accuracy, compared to those of the equi-energy sampler. In particular, it considers how the number of sampling chains and mixture components in the proposal distribution affect performance. From the results of these runs, guidelines are then developed for how to implement BAIS+L in practice. These guidelines are demonstrated when sampling from the mixture exponential application studied by Kou et al. (2006).

Chapter 5 initiates a study of how BAIS+L may be used to study spin glasses. It discusses two potential approaches, before applying the simpler of the two to a two-dimensional spin glass. The results of this chapter highlight pitfalls with BAIS+L and suggest potential solutions and alternative avenues for future study.

Finally, Chapter 6 introduces an exact version of BAIS+L, called EBAIS+L. This approach is justified by a theorem of Besag et al. (1995), which was reviewed in Section 1.2.3. The new method is compared to BAIS+L and used to highlight the efficiency gained in using the approximate approach of BAIS+L in place of the exact one offered by EBAIS+L.

Chapter 3

BAIS+L Development

This chapter provides the details of the *Bayesian Adaptive Indepenence Sampler with Latent variables* (BAIS+L). It starts with a description of its motivation, with reference to the issues discussed in earlier chapters, especially with respect to targets with many local maxima. It then outlines specific details of BAIS+L's design, including its proposal distribution and acceptance ratio. Finally, it combines these design details and summarises them as an algorithm.

After highlighting the novelty of BAIS+L in Section 3.2, Section 3.3 is devoted to considerations of the ergodicity of BAIS+L, in the form of sufficient conditions for uniform ergodicity. Proofs of their sufficiency are then provided, before a discussion of how they may be implemented in practice. This discussion is supplemented by some numerical simulations, which are used to provide practical guidance.

3.1 Motivation and Description

BAIS does not completely solve the problem of approximating a target distribution with many local maxima, which was illustrated in Figure 1.2, as its proposal distribution contains only one one (local) maximum. Therefore, the goal of the current study is to combine the flexibility of an adaptive approach with a mixture proposal distribution, which can approximate a continuous density to an arbitrary precision, given enough mixture components (Frühwirth-Schnatter and Pyne, 2010).

3.1.1 Extension of BAIS

This section introduces a novel approach to adaptive MCMC, which extends BAIS (cf. Section 1.8) by replacing its single normal proposal distribution with a mixture of K normal distributions. That is, a proposed state $\mathbf{y}$ is sampled from Equation (3.1),

$$\mathbf{y} \sim \sum_{k=1}^{K} d_k \mathcal{N}\left(\cdot \,|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\right). \tag{3.1}$$

This new proposal distribution has the following parameters for each component $k \in \{1, \ldots, K\}$: a mixture proportion d_k; a mean $\boldsymbol{\mu}_k$; and a covariance matrix $\boldsymbol{\Sigma}_k$.

For each sampling chain, the state-space is augmented with a latent variable z_n, indicating a component of the proposal; hence the "+L" in the name of the sampler. This is the same idea that was used by Frühwirth-Schnatter and Pyne (2010) to indicate component allocations of samples from skew-normal and skew-t distributions.

When updating sampling chain $n \in \{1, \ldots, N\}$, a proposed component membership w is first drawn from a categorical distribution, based on the component weights, before proposing a chain state $\mathbf{y}$ from the corresponding normal distribution component,

$$\begin{aligned} w &\sim \text{Categorical}\left(\cdot\,|d_1, \ldots, d_K\right), \\ \mathbf{y}|w &\sim \mathcal{N}\left(\cdot\,|\boldsymbol{\mu}_w, \boldsymbol{\Sigma}_w\right). \end{aligned}$$

This process is repeated in parallel for each of the N sampling chains.

Once a new membership and state have been proposed for a sampling chain n it is accepted or rejected via a Metropolis-Hastings-like acceptance step. The details of this step will be outlined later, in Section 3.1.4. Before doing so, however, a closer look must be taken at the proposal parameter update procedure.

3.1.2 The Prior Model of the Proposal Parameters

Following BAIS, BAIS+L casts the selection of new proposal parameters as a problem in Bayesian inference. Given a current collection of sampling chain states $(\mathbf{x}_1, \ldots, \mathbf{x}_N)$ and component memberships $(z_1, \ldots, z_N)$ the goal is to determine a new set of component weights, means and covariance matrices $(d_1, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1, \ldots, d_K, \boldsymbol{\mu}_K, \boldsymbol{\Sigma}_K)$.

Assuming conjugate priors on the proposal parameters, the prior distribution of the component weights $\mathbf{d} = (d_1, \ldots, d_K)$, is given by Equation (3.2),

$$\mathbf{d} \sim \text{Dirichlet}\left[\cdot\,\middle|\,\alpha_1^{(0)}, \ldots, \alpha_K^{(0)}\right], \tag{3.2}$$

where $\alpha_k^{(0)}$ is the prior (unnormalised) weight of component k.

The non-informative Jeffrey's prior used by Keith et al. (2008) in their development of BAIS was deemed inappropriate for a mixture proposal, as it can only capture the overall variance of the target distribution. Furthermore, non-informative nature of the prior distribution on the mean of a component limits its flexibility, since it does not allow any distinction between the prior positions of each component. Therefore, following Gelman et al. (2004, pp. 87–88), the prior distributions of the variance-covariance matrix $\boldsymbol{\Sigma}_k$ and the mean $\boldsymbol{\mu}_k$ of component k of the proposal distribution are given by Equations (3.3) and (3.4), respectively,

$$\boldsymbol{\Sigma}_k \sim \text{Inv-W}_{\nu_k^{(0)}}\left[\cdot\,\middle|\,\boldsymbol{\Lambda}_k^{(0)}\right] \tag{3.3}$$

$$\boldsymbol{\mu}_k \sim \mathcal{N}\left[\cdot\,\middle|\,\boldsymbol{\mu}_k^{(0)}, \frac{\boldsymbol{\Sigma}_k^{(0)}}{\kappa_k^{(0)}}\right], \tag{3.4}$$

In Equations (3.4) and (3.3), $\kappa_k^{(0)}$ is the prior number of observations of the scale of $\boldsymbol{\Sigma}_k$, $\boldsymbol{\mu}_k^{(0)}$ is the prior mean vector of a component k, and $\boldsymbol{\Lambda}_k^{(0)}$ and $\nu_k^{(0)}$ are

the prior scale matrix and prior degrees of freedom, respectively, of $\mathbf{\Sigma}_k$ (Gelman et al., 2004, pp. 87). Therefore, $\nu_k^{(0)}$, $\kappa_k^{(0)}$, $\lambda_k^{(0)}$, $\boldsymbol{\mu}_k^{(0)}$ and $\mathbf{\Sigma}_k^{(0)}$ must be specified by the user for each $k \in \{1, \ldots, k\}$.

For ease of computer coding and simulation, these hyperparameters were chosen to be equal for each component in the current study. This choice limited the types of targets that could be studied effectively. The effect of such a choice is explored in the application to spin glass simulation in Chapter 5. With more careful coding, this limitation may be overcome, allowing BAIS+L to be applied to a larger number of target distributions.

3.1.3 The Posterior Model of the Proposal Parameters

The choice of prior distribution given in Section 3.1.2 results in the posterior model described by Equations (3.5), (3.6) and (3.7) for the vector of all component weights, and the variance-covariance matrix and mean of component k, respectively (Gelman et al., 2004, pp. 87):

$$\mathbf{d}|\mathbf{x}, \mathbf{z} \sim \text{Dirichlet}\left[\cdot \left| o_1 + \alpha_1^{(0)}, \ldots, o_K + \alpha_K^{(0)} \right.\right], \tag{3.5}$$

$$\mathbf{\Sigma}_k|\mathbf{x}, \mathbf{z} \sim \text{Inv-W}_{\nu_k^{(0)}+o_k}(\cdot|\mathbf{\Lambda}_k), \tag{3.6}$$

$$\boldsymbol{\mu}_k|\mathbf{\Sigma}_k, \mathbf{x}, \mathbf{z} \sim \mathcal{N}\left[\cdot \left| \frac{\kappa_k^{(0)}}{\kappa_k^{(0)} + o_k}\boldsymbol{\mu}_k^{(0)} + \frac{o_k}{\kappa_k^{(0)} + o_k}\bar{\mathbf{x}}_k, \frac{\mathbf{\Sigma}_k}{\kappa_k^{(0)} + o_k} \right.\right]. \tag{3.7}$$

In the preceding equations, o_k is the number of sampling chains currently associated with component k, $\bar{\mathbf{x}}_k$ is the sample mean of the states associated with component k and $\mathbf{\Lambda}_k$ incorporates the sample sum of squared errors of the states associated with component k, as expressed by Equations (3.8), (3.9) and (3.10),

$$o_k = \sum_{n=1}^{N} I_k(z_n), \tag{3.8}$$

$$\bar{\mathbf{x}}_k = \frac{1}{o_k}\sum_{n=1}^{N} I_k(z_n)\mathbf{x}_n, \tag{3.9}$$

$$\begin{aligned}\mathbf{\Lambda}_k = \mathbf{\Lambda}_k^{(0)} &+ \frac{\kappa_k^{(0)} o_k}{\kappa_k^{(0)} + o_k}\left[\bar{\mathbf{x}}_k - \boldsymbol{\mu}_k^{(0)}\right]\left[\bar{\mathbf{x}}_k - \boldsymbol{\mu}_k^{(0)}\right]^{\mathrm{T}} \\ &+ \sum_{n=1}^{N} I_k(z_n)(\mathbf{x}_n - \bar{\mathbf{x}}_k)(\mathbf{x}_n - \bar{\mathbf{x}}_k)^{\mathrm{T}}.\end{aligned} \tag{3.10}$$

Here $I_a(b)$ is the identity function, which is equal to 1 when $a = b$ and 0 otherwise.

3.1.4 Acceptance Ratio

Construction of the BAIS+L acceptance ratio follows the same approach as in BAIS, where the parameter update distribution is incorporated into the

Metropolis-Hastings acceptance ratio for the nth sampling chain, as in Equation (3.11),

$$\alpha_{\boldsymbol{\theta}}\left[(\mathbf{x}_n, z_n), (\mathbf{y}, w)\right] = \min\left\{1, \frac{\pi(\mathbf{y})}{\pi(\mathbf{x}_n)} \frac{p(\mathbf{x}_n, z_n|\boldsymbol{\theta})}{p(\mathbf{y}, w|\boldsymbol{\theta})} \frac{p(\boldsymbol{\theta}, \mathbf{z}^*|\mathbf{x}^*)}{p(\boldsymbol{\theta}, \mathbf{z}|\mathbf{x})}\right\}. \tag{3.11}$$

Here $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)$ is the vector of current states, $\mathbf{z} = (z_1, \ldots, z_N)$ is the vector of current component memberships, and $\mathbf{x}^*$ and $\mathbf{z}^*$ are the same as $\mathbf{x}$ and $\mathbf{z}$, respectively, except that the nth element has been swapped with its proposed value ($\mathbf{y}$ and w, respectively). $\boldsymbol{\theta} = (\mathbf{d}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) \in \mathcal{T}$ represents the vector of proposal parameters, where the parameter vector $\boldsymbol{\theta} = (\mathbf{d}, \boldsymbol{\mu}, \boldsymbol{\Sigma})$, which consists of the component weights $\mathbf{d} = (d_1, \ldots, d_K)$, component means $\boldsymbol{\mu} = (\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_K)$ and component variance-covariance matrices $\boldsymbol{\Sigma} = (\boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_K)$, takes values in the parameter space $\mathcal{T} = [0, 1]^K \times \mathbb{R}^{pK} \times [\mathbb{M}_{p\times p}(\mathbb{R})]^K$, where $\mathbb{M}_{p\times p}(\mathbb{R})$ is the space of $p \times p$ symmetric positive-definite variance-covariance matrices with real entries and p is the dimension of the state space $\mathcal{X}$.

A straightforward application of Bayes' Theorem $p(A|B) = p(A)p(B|A)/p(B)$ with $A = (\boldsymbol{\theta}, \mathbf{z})$ and $B = \mathbf{x}$ gives $p(\boldsymbol{\theta}, \mathbf{z}|\mathbf{x}) = p(\boldsymbol{\theta})p(\mathbf{x}, \mathbf{z}|\theta)/p(\mathbf{x})$. Substituting this into Equation (3.11) gives Equation (3.12),

$$\alpha_{\boldsymbol{\theta}}\left[(\mathbf{x}_n, z_n), (\mathbf{y}, w)\right] = \min\left\{1, \frac{\pi(\mathbf{y})}{\pi(\mathbf{x}_n)} \frac{p(\mathbf{x})}{p(\mathbf{x}^*)} \frac{p(\mathbf{x}_n, z_n|\boldsymbol{\theta})}{p(\mathbf{y}, w|\boldsymbol{\theta})} \frac{p(\boldsymbol{\theta})}{p(\boldsymbol{\theta})} \frac{p(\mathbf{x}^*, \mathbf{z}^*|\boldsymbol{\theta})}{p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta})}\right\}. \tag{3.12}$$

Cancelling like factors in the numerator and denominator, produces the expression for the acceptance probability for sampling chain n given in Equation (3.13)

$$\alpha_{\boldsymbol{\theta}}\left(\mathbf{x}_n, \mathbf{y}\right) = \min\left\{1, \frac{\pi(\mathbf{y})}{\pi(\mathbf{x}_n)} \frac{p(\mathbf{x}_n|\mathbf{x}_{-n})}{p(\mathbf{y}|\mathbf{x}_{-n})}\right\}, \tag{3.13}$$

where $\mathbf{x}_{-n}$ is $\mathbf{x}$ without the element corresponding to sampling chain n.

Unfortunately, $p(\mathbf{x}_n|\mathbf{x}_{-n})$ and $p(\mathbf{y}|\mathbf{x}_{-n})$ are computationally infeasible. This is due to the fact that each $\mathbf{x}_n$ is updated using *all* sampling chain states and latent allocations at the previous iteration, while these densities ignore the conditioning on the previous state of sampling chain in question. However, note that,

$$\begin{aligned} p(\mathbf{x}_n|\mathbf{x}_{-n}) &= \int_{\mathcal{T}} p(\mathbf{x}_n, \boldsymbol{\theta}|\mathbf{x}_{-n}) d\boldsymbol{\theta} \\ &= \int_{\mathcal{T}} p(\mathbf{x}_n|\boldsymbol{\theta}, \mathbf{x}_{-n}) p(\boldsymbol{\theta}|\mathbf{x}_{-n}) d\boldsymbol{\theta}. \\ &= \int_{\mathcal{T}} p(\mathbf{x}_n|\boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathbf{x}_{-n}) d\boldsymbol{\theta}, \end{aligned}$$

where the last line takes note of the fact that $p(\mathbf{x}_n|\boldsymbol{\theta}, \mathbf{x}_{-n}) = p(\mathbf{x}_n|\boldsymbol{\theta})$, by design.

As the number of sampling chains increases it is anticipated that $p(\boldsymbol{\theta}|\mathbf{x}_{-n})$ will approach a Dirac delta function, which motivates the use of the approximation given in Equation (3.14),

$$p(\mathbf{x}_n|\mathbf{x}_{-n}) \approx p(\mathbf{x}_n|\boldsymbol{\theta}) = \sum_{k=1}^{K} d_k \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k). \tag{3.14}$$

A similar approach is used to take care of $p(\mathbf{y}|\mathbf{x}_{-n})$.

This gives the approximate acceptance ratio in Equation (3.15),

$$\alpha_{\boldsymbol{\theta}}(\mathbf{x}_n, \mathbf{y}) \approx \min\left\{1, \frac{\pi(\mathbf{y}) \sum_{k=1}^{K} d_k \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\pi(\mathbf{x}_n) \sum_{n=1}^{K} d_k \mathcal{N}(\mathbf{y}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}\right\}. \tag{3.15}$$

There is no direct dependence in Equation (3.15) on the states or memberships of sampling chains other than the one of interest. This allows all sampling chains to be updated in parallel, unlike in BAIS, which required sequential updates. However, as the acceptance ratio is an approximation, the stationary distribution of the sampler (if there is one) will only be an approximation to the true target.

BAIS+L with a normal mixture proposal is summarized in Algorithm 3.1.

3.2 The Novelty of BAIS+L

Unlike the MC methods that utilise adaptive proposal distributions with possibly multiple local maxima, which were discussed in Section 1.7, BAIS+L does not refine the parameters of the proposal distribution relative to their current values, or use a clustering approach to generate new values. Instead, the novelty of BAIS+L lies in its extension of the Bayesian proposal parameter estimation of BAIS to mixture proposals. Through this extension, BAIS+L differs from other adaptive techniques that use proposal distributions with multiple local maxima, in that is *samples* its new proposal parameters directly from (an approximation to) their posterior model, given the current population of sampling chain states.

Similarly to Pooley et al. (2019), BAIS+L uses stochastic sampling to generate latent variables in its approach. However, the proposal distribution is not derived from the target. Instead, it is of a fixed form, with its own parameters, which are updated from *their own* posterior distribution, given the current population of samples from the target. While a proposal distribution constructed by the method of Pooley et al. (2019) is tailored to a specific target, making it more suitable than a general proposal, it also requires that such a proposal be tractable, so that it may be computed *before* sampling takes place. However, while the proposal distribution used by BAIS+L is not specifically tailored to any one target, it is more general, meaning it can cover a larger variety of target distributions with relatively minimal effort on the part of the MCMC practitioner.

While BAIS+L does have the obvious drawback of sampling from an approximation to the target, due to the difficulties introduced by the necessary use of latent variables, its novelty merits its development and investigation. In particular, the explicit Bayesian relationship between the samples at one iteration and the form of the posterior distribution at the next, explores a natural updating paradigm that does not require the additional complexity usually seen in practice.

3.3 Ensuring Ergodicity

While conditions have been given by Pompe et al. (2018) for the ergodicity of an adaptive MCMC sampler that utilises an auxilliary or latent variable, it is

Algorithm 3.1 BAIS+L with a normal mixture proposal.

Require:

1. $K \in \mathbb{Z}^+$ proposal components.
2. Initial chain states $\left[\mathbf{x}_n^{(0)}\right]_{n=1}^N$ and their allocations $\left[z_n^{(0)}\right]_{k=1}^K$.
3. Prior scales $\mathbf{\Lambda}_k^{(0)}$ and degrees of freedom $\nu_k^{(0)}$ of the distributions on $\mathbf{\Sigma}_k, k \in \{1, \ldots, K\}$.
4. Prior means $\boldsymbol{\mu}_k^{(0)}, k \in \{1, \ldots, K\}$.
5. Prior numbers of observations $\kappa_k^{(0)}$ of the scales of $\mathbf{\Sigma}_k, k \in \{1, \ldots, K\}$.
6. Prior component weights $\boldsymbol{\alpha}^{(0)} = \left[\alpha_k^{(0)}\right]_{k=1}^K$.

Ensure: N chains of samples $\left\{\left[\mathbf{x}_n^{(t)}\right]_{n=1}^N\right\}_{t=1}^\infty$ from an approximation to the target distribution π.

for Iteration $t \in \mathbb{Z}^+$ **do**
  for Component $k = 1$ to $k = K$, in parallel **do**
    Set $o_k = \sum_{n=1}^N I_k\left[z_n^{(t-1)}\right]$.
    Set $\bar{\mathbf{x}}_k = \sum_{n=1}^N I_k\left[z_n^{(t-1)}\right]\mathbf{x}_n^{(t-1)} \Big/ o_k$.
    Set $\mathbf{\Lambda}_k = \mathbf{\Lambda}_k^{(0)} + \frac{\kappa_k^{(0)} o_k}{\kappa_k^{(0)} + o_k}\left[\bar{\mathbf{x}}_k - \boldsymbol{\mu}_k^{(0)}\right]\left[\bar{\mathbf{x}}_k - \boldsymbol{\mu}_k^{(0)}\right]^{\mathrm{T}} + \sum_{n=1}^N I_k\left[z_n^{(t-1)}\middle|(\mathbf{x}_n - \bar{\mathbf{x}}_k)(\mathbf{x}_n - \bar{\mathbf{x}}_k)^{\mathrm{T}}\right]$.
  end for
  Generate $\mathbf{d} \sim \text{Dirichlet}\left[\cdot\middle|o_1 + \alpha_1^{(0)}, \ldots, o_K + \alpha_K^{(0)}\right]$.
  for Component $k = 1$ to $k = K$, in parallel **do**
    Generate $\mathbf{\Sigma}_k \sim \text{Inv-W}_{\nu_k^{(0)} + o_k}(\cdot|\mathbf{\Lambda}_k)$.
    Generate $\boldsymbol{\mu}_k \sim \mathcal{N}\left[\cdot\middle|\frac{\kappa_k^{(0)}}{\kappa_k^{(0)} + o_k}\boldsymbol{\mu}_k^{(0)} + \frac{o_k}{\kappa_k^{(0)} + o_k}\bar{\mathbf{x}}_k, \frac{\mathbf{\Sigma}_k}{\kappa_k^{(0)} + o_k}\right]$
  end for
  for Sampling chain $n = 1$ to $n = N$, in parallel **do**
    Generate $w \sim \text{Categorical}\left(\cdot\,|d_1, \ldots, d_K\right)$.
    Generate $\mathbf{y} \sim \mathcal{N}\left(\cdot\,|\boldsymbol{\mu}_w, \mathbf{\Sigma}_w\right)$.
    Set $\alpha = \min\left\{1, \frac{\pi(\mathbf{y})\sum_{k=1}^K d_k \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_k, \mathbf{\Sigma}_k)}{\pi(\mathbf{x}_n)\sum_{n=1}^K d_k \mathcal{N}(\mathbf{y}|\boldsymbol{\mu}_k, \mathbf{\Sigma}_k)}\right\}$.
    Generate $u \sim \mathcal{U}(0, 1)$.
    if $u \leq \alpha$ **then**
      Set $\mathbf{x}^{(t)} = \mathbf{y}$ and $z^{(t)} = w$.
    else
      Set $\mathbf{x}^{(t)} = \mathbf{x}^{(t-1)}$ and $z^{(t)} = z^{(t-1)}$.
    end if
  end for
end for

not clear that BAIS+L satisfies them.", which I added to the paper based on this chapter.

Given that BAIS+L uses an approximation in its acceptance ratio, detailed balance cannot be used to prove the existence of a stationary distribution. However, this section provides two sets of conditions that guarantee that the overall process of BAIS+L is ergodic. In fact, these two sets of conditions guarantee uniform ergodicity, enabling the provision of upper bounds on the convergence rate to stationarity.

Section 3.3.1 summarises these conditions, which are then proven, in Section 3.3.2, to be sufficient for uniform ergodicity.

3.3.1 Sufficient Conditions to Ensure Uniform Ergodicity

Ergodicity of the process induced by BAIS+L may not hold in general, so sufficient conditions, under which it does hold, must first be found. This section considers two such sets of sufficient conditions.

Both conditions involve the same parameter space $\mathcal{T}$. Recall the parameters $\boldsymbol{\theta} = (\mathbf{d}, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ of the normal mixture version of BAIS+L, where

- $\mathbf{d} = (d_1, \ldots, d_K)$ is the K-vector of mixture proportions,
- $\boldsymbol{\mu} = (\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_K)$ is the K-vector of p-dimensional component means,
- $\boldsymbol{\Sigma} = (\boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_K)$ is the K-vector of $p \times p$-dimensional strictly positive-definite symmetric component variance-covariance matrices.

Note that each scalar quantity in $\boldsymbol{\theta}$ is real-valued and that the parameter space $\mathcal{T}$ is trivially isomorphic to a proper subset of $\mathbb{R}^D$, where D is given by Equation (3.16),

$$D = K - 1 + Kp + \frac{Kp(p+1)}{2}. \tag{3.16}$$

The following additional conditions are imposed on $\mathcal{T}$ in both cases in which ergodicity is proven

Assumption 1 (Common conditions)**.** *The following conditions hold:*

1. *For each $k \in \{1, \ldots, K\}$ mixture proportion k is strictly positive,*

$$0 < d_k < 1$$

and all proportions add to 1*,*

$$\sum_{k=1}^{K} d_k = 1.$$

2. *For each $k \in \{1, \ldots, K\}$ component variance-covariance matrix $\boldsymbol{\Sigma}_k$ has positive determinant,*

$$0 < \det(\boldsymbol{\Sigma}_k) < \infty.$$

3. *$\mathcal{T}$ is compact in $\mathbb{R}^D$.*

4. *$\mathcal{T}$ has non-zero Lebesgue measure in $\mathbb{R}^D$.*

5. *The two-step proposal distribution* $p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta})$ *is continuous in its parameters* $\boldsymbol{\theta}$.

The purpose of the fourth condition is to ensure that the prior and posterior distributions for the parameters have finite densities in $\mathbb{R}^D$. The third assumption, that $\mathcal{T}$ is compact in $\mathbb{R}^D$, strengthens the first and second assumptions, by ensuring that the infimum and supremum of each d_k are finite and strictly positive. It also ensures that each variance-covariance matrix $\boldsymbol{\Sigma}_k$ and each mean $\boldsymbol{\mu}_k$ has a determinant and magnitude, respectively, that is bounded above and below by finite and strictly positive constants.

Recall that the state space for a single sampling chain is $\mathcal{X} = \mathbb{R}^p$, so the *sampling space* for all N chains is $\mathcal{X}^N \cong \mathbb{R}^{Np}$. Similarly take the N-tuple of $\mathcal{Z}$ to define the *latent variable space* to be $\mathcal{Z}^N = \{1, \ldots, K\}^N$.

Before discussing the sufficient conditions for uniform ergodicity in detail, an important property of $\mathcal{X} \times \mathcal{Z}$ and $\mathcal{T}$ needs to be highlighted. This property is summarised by Lemma 1.

Lemma 1. *The BAIS+L parameter space* $\mathcal{T}$ *and the Cartesian product* $\mathcal{X} \times \mathcal{Z}$ *of the state space* $\mathcal{X}$ *and latent variable space* $\mathcal{Z}$ *are Polish spaces.*

Proof. First note that $\mathcal{X}$ is trivially homeomorphic to $\mathbb{R}^{pN}$, which is a separable completely metrisable space with the Euclidean metric, for any $pN \in \mathbb{N}$ (Kechris, 1995, pp. 13). That is, it is Polish (Kechris, 1995, pp. 13). Since separability and topological completeness/complete metrisability are topological invariants (Steen and Seebach, 1978, pp. 8, 37), they are preserved under homeomorphism (Steen and Seebach, 1978, pp. 8). Hence, $\mathcal{X}$ is also Polish.

Now, $\mathcal{Z}$ is finite and, hence, compact (Kechris, 1995, pp. 18), when endowed with the taxi-cab metric (Tao, 2006, pp. 392). It then follows, from the definition of compactness, that it is also complete with this metric (Tao, 2006, pp. 413). The finiteness of $\mathcal{Z}$ further means that it is a countable dense subset of itself, making it separable (Steen and Seebach, 1978, pp. 7). Therefore, $\mathcal{Z}$ is Polish.

Being the finite product of two Polish spaces, $\mathcal{X} \times \mathcal{Z}$ is, therefore, also Polish (Kechris, 1995, pp. 13).

Now consider the parameter space $\mathcal{T}$. By assumption, it is compact in $\mathbb{R}^D$, hence, it is closed (Tao, 2006, pp. 413). Being a closed subspace of a Polish space, it is, therefore, also Polish (Kechris, 1995, pp. 13). □

The foundations are now in place to state the two cases under which uniform ergodicity is guaranteed.

Case One

Assumption 2 (Finite positive densities)**.** *The target density* $\pi(\mathbf{x})$ *and prior density on the proposal parameters* $p(\boldsymbol{\theta})$ *are finite and strictly positive on the state space* $\mathbb{R}^p$ *and parameters space* $\mathcal{T}$*, respectively. That is*

$$0 < \pi(\mathbf{x}) < \infty, \, \mathbf{x} \in \mathbb{R}^p$$

and

$$0 < p(\boldsymbol{\theta}) < \infty, \, \boldsymbol{\theta} \in \mathcal{T}.$$

Additionally, the support of $p(\boldsymbol{\theta})$ *only contains parameter values* $\boldsymbol{\theta}$ *for which the corresponding proposal density of the chain states* $p(\mathbf{x}|\boldsymbol{\theta})$ *is bounded below by some constant multiple* $\beta \in \mathbb{R}^+$ *of the target density* π*,*

$$p(\boldsymbol{\theta}) > 0 \implies p(\mathbf{x}|\boldsymbol{\theta}) \geq \beta\pi(\mathbf{x}),\ \beta > 0, \forall \mathbf{x} \in \mathcal{X}.$$

Note that $p(\boldsymbol{\theta})$ is a density with respect to Lebesgue measure on $\mathbb{R}^D$, which can only be finite under the assumption that $\mathcal{T}$ has non-zero Lebesgue measure.

Case Two

Assumption 3. *There exists a finite, positive density* $h(\mathbf{x}, z)$ *such that,*

$$\frac{1}{\zeta}h(\mathbf{x}, z) \leq p(\mathbf{x}, z|\boldsymbol{\theta}) \leq \zeta h(\mathbf{x}, z),$$
$$\frac{1}{\zeta}h(\mathbf{x}) \leq \pi(\mathbf{x}) \leq \zeta h(\mathbf{x}),$$

for all $\boldsymbol{\theta}$ *in the support of* $p(\boldsymbol{\theta})$*,* $\mathbf{x}$ *in the support of* $\pi(\mathbf{x})$ *and* z *in* $\{1, \ldots, K\}$*, where* $\zeta \in \mathbb{R}^+$ *and*

$$h(\mathbf{x}) = \sum_{k=1}^{K} h(\mathbf{x}, z = k).$$

3.3.2 Proofs of Uniform Ergodicity

Case One

Theorem 13 (Uniform Ergodicity under Assumption 2)**.** *With the conditions given in Assumption 2 the Markov chain* $\left\{\boldsymbol{\Theta}^{(t)}, \mathbf{X}^{(t)}\right\}_{t=1}^{\infty}$ *of random variables representing the parameters and states produced by the* N*-chain BAIS+L algorithm is uniformly ergodic.*

The proof of Theorem 13 will make use of the following lemmas.

Lemma 2 (Finite and Positive Proposal under Assumption 2)**.** *By Condition 2 of Assumption 2 the proposal density* $p(\boldsymbol{\theta}|\mathbf{x}, \mathbf{z})$ *is finite and strictly positive for all* $(\mathbf{x}, \mathbf{z}, \boldsymbol{\theta}) \in \mathcal{X}^N \times \mathcal{Z}^N \times \mathcal{T}$*.*

Proof. By Bayes' Theorem

$$p(\boldsymbol{\theta}|\mathbf{x}, \mathbf{z}) = \frac{p(\boldsymbol{\theta})p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta})}{p(\mathbf{x}, \mathbf{z})}. \tag{3.17}$$

It is therefore sufficient to show that each density on the right-hand side of Equation (3.17) is finite and strictly positive for all $(\mathbf{x}, \mathbf{z}, \boldsymbol{\theta}) \in \mathcal{X}^N \times \mathcal{Z}^N \times \mathcal{T}$. It immediately follows from Assumption 2 that $p(\boldsymbol{\theta})$ is finite and strictly positive. Recall, also, that the combined proposal density of states and latent variables is a weighted sum of normal densities

$$p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta}) = \sum_{n=1}^{N} d_{z_n} \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_{z_n}, \boldsymbol{\Sigma}_{z_n}). \tag{3.18}$$

Assumption 2 restricts the space of variance-covariance matrices to include only those for which $0 < \det(\boldsymbol{\Sigma}_k) < \infty$ for each $k \in \{1, \ldots, K\}$. Therefore the

normal density for each $n \in \{1, \ldots, N\}$ in (3.18) is restricted to its respective finitely-bounded range

$$\left(0, \frac{1}{\sqrt{2\pi \det(\boldsymbol{\Sigma}_{z_k})}}\right].$$

Combining this fact with the assumption that $0 < d_k < 1$ for each $k \in \{1, \ldots, K\}$, it is clear that $p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta})$ is the product of a finite number of strictly positive and finite values and is, therefore, itself strictly positive and finite.

Note that $p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta})$ is a continuous function of $\boldsymbol{\theta}$ for fixed $(\mathbf{x}, \mathbf{z})$. Hence, by the Extreme Value Theorem and the assumed compactness of $\mathcal{T}$, there exists $0 < M < \infty$ such that $1/M < p(\mathbf{x}, \mathbf{z}|\boldsymbol{\theta}) < M$ for all $\boldsymbol{\theta} \in \mathcal{T}$ (Abbott, 2001, pp. 115). Multiplying this inequality by $p(\boldsymbol{\theta})$ and invoking the assumption that $\mathcal{T}$ has non-zero Lebesgue measure, to integrate over $\mathcal{T}$, gives $1/M < p(\mathbf{x}, \mathbf{z}) < M$.

Therefore all densities on the right-hand side of (3.17) are finite and strictly positive and, hence, so is $p(\boldsymbol{\theta}|\mathbf{x}, \mathbf{z})$. □

The result given by Lemma 3, as well as its proof, may be considered to be obvious to most readers. Indeed, possibly due to a combination of this "obviousness", as well as its specialised nature, the author of this dissertation was unable to find a version of Lemma 3 or its proof in the literature. Therefore, a formal version is presented as follows.

Lemma 3. *Consider an arbitrary topological (state) space $\mathcal{X}$, with Borel σ-algebra $\mathscr{B}(\mathcal{X})$, and two transition functions $\mu_1(x, A)$ and $\mu_2(x, A)$ defined on $\mathcal{X}$. Let ν_1 and ν_2 be two more transition functions on $\mathcal{X}$, obtained by combining μ_1 and μ_2, according to Equations (3.19) and (3.20), respectively,*

$$\nu_1(x, A) = \int_{\mathcal{X}} \mu_2(y, A)\mu_1(x, dy) \tag{3.19}$$

$$\nu_2(x, A) = \int_{\mathcal{X}} \mu_1(y, A)\mu_2(x, dy). \tag{3.20}$$

Then the Markov chain induced by ν_1 is uniformly ergodic if and only if the one induced by ν_2 is.

Proof. Assuming that ν_1 induces a uniformly ergodic Markov chain, by Theorem 1 (cf. Section 1.2.2) the whole of $\mathcal{X}$ is small with respect to ν_1, with there being some non-trivial measure ϕ on $\mathcal{X}$, bounding the m-step ν_1-transition from below, for some $m \in \mathbb{Z}^+$, according to Equation (3.21),

$$\nu_1^m(x, A) \geq \phi(A), \forall x \in \mathcal{X}, \forall A \in \mathscr{B}(\mathcal{X}). \tag{3.21}$$

Consider the $(m+1)$-step ν_2-transition, which can expressed by combining Equations (3.19) and (3.20), according to Equation (3.22),

$$\nu_2^{m+1}(x, A) = \int_{\mathcal{X}} \int_{\mathcal{X}} \mu_1(z, A)\nu_1^m(y, dz)\mu_2(x, dy). \tag{3.22}$$

Substituting in the assumption of the uniform ergodicity induced by ν_1, leads to a lower-bounding measure ψ that depends only on A,

$$\nu_2^{m+1}(x, A) \geq \int_{\mathcal{X}} \int_{\mathcal{X}} \mu_1(z, A)\phi(dz)\mu_2(x, dy) = \int_{\mathcal{X}} \mu_1(z, A)\phi(dz) =: \psi(A).$$

Consider $\psi(\mathcal{X})$ and note that $\mu_1(x, \mathcal{X}) = 1$ for all $x \in \mathcal{X}$, by definition. Then observe that ψ is non-trivial, since it assigns the same measure to all of $\mathcal{X}$ that ϕ assigns to it, as demonstrated in Equation (3.23),

$$\psi(\mathcal{X}) = \int_{\mathcal{X}} \mu_1(z, \mathcal{X})\phi(dz) = \int_{\mathcal{X}} \phi(dz) = \phi(\mathcal{X}) \tag{3.23}$$

Invoking Theorem 1 once again, leads to the deduction that the Markov chain induced by ν_2 is uniformly ergodic.

Swapping the roles of ν_1 and ν_2 in the preceding argument, proves the converse claim.

□

The argument now returns to the proof of Theorem 13, which is motivated in part by the proof of Mengersen and Tweedie (1996) for the uniform ergodicity of independence samplers.

Proof of Theorem 13. To begin note that Algorithm 3.1 updates the chain states, latent memberships and proposal parameters in three steps. It first updates the proposal parameters, followed by the latent variables and finally, the states. For this proof, however, the order is permuted, performing the latent membership updates first, followed by the chain states and lastly, the proposal parameters. If the order of Algorithm 3.1 starting at $\left[\boldsymbol{\theta}^{(0)}, \mathbf{x}^{(0)}, \mathbf{z}^{(0)}\right]$ produces the sequence

$$\left\{ \left[\boldsymbol{\theta}^{(0)}, \mathbf{x}^{(0)}, \mathbf{z}^{(0)}\right], \left[\boldsymbol{\theta}^{(1)}, \mathbf{x}^{(0)}, \mathbf{z}^{(0)}\right], \left[\boldsymbol{\theta}^{(1)}, \mathbf{x}^{(0)}, \mathbf{z}^{(1)}\right], \left[\boldsymbol{\theta}^{(1)}, \mathbf{x}^{(1)}, \mathbf{z}^{(1)}\right], \ldots \right\},$$

then the permuted algorithm, starting at $\left[\boldsymbol{\theta}^{(1)}, \mathbf{x}^{(0)}, \mathbf{z}^{(0)}\right]$ produces the sequence

$$\left\{ \left[\boldsymbol{\theta}^{(1)}, \mathbf{x}^{(0)}, \mathbf{z}^{(0)}\right], \left[\boldsymbol{\theta}^{(1)}, \mathbf{x}^{(0)}, \mathbf{z}^{(1)}\right], \left[\boldsymbol{\theta}^{(1)}, \mathbf{x}^{(1)}, \mathbf{z}^{(1)}\right], \left[\boldsymbol{\theta}^{(2)}, \mathbf{x}^{(1)}, \mathbf{z}^{(1)}\right], \ldots \right\}.$$

Dropping the very first element of the first sequence (i.e. $\left[\boldsymbol{\theta}^{(0)}, \mathbf{x}^{(0)}, \mathbf{z}^{(0)}\right]$) leaves exactly the second sequence. Therefore, if this sequence is uniformly ergodic, then by Lemma 3 so is the one produced by Algorithm 3.1. Since the latter approach "swaps" the parameter update step and the latent memberships/chain states update step, it shall be referred to as *swapped BAIS+L*.

Let $\mathbb{P}[(\mathbf{x}, \mathbf{z}, \boldsymbol{\theta}), S]$ denote the probability of a swapped BAIS+L transition from $(\mathbf{x}, \mathbf{z}, \boldsymbol{\theta}) \in \mathcal{X}^N \times \mathcal{Z}^N \times \mathcal{T}$ into a set $S \in \mathscr{B}(\mathcal{X}^N \times \mathcal{Z}^N \times \mathcal{T})$ in the Borel σ-algebra of $\mathcal{X}^N \times \mathcal{Z}^N \times \mathcal{T}$. By Theorem 1, it is sufficient to show that $\mathcal{X}^N \times \mathcal{Z}^N \times \mathcal{T}$ is *small*. Recall that by "small" Mengersen and Tweedie (1996) meant that for all sets $S \in \mathscr{B}(\mathcal{X}^N \times \mathcal{Z}^N \times \mathcal{T})$ in the Borel σ-algebra of $\mathcal{X}^N \times \mathcal{Z}^N \times \mathcal{T}$, there exists a common $\delta > 0$ and common probability measure ν over $\mathcal{X}^N \times \mathcal{Z}^N \times \mathcal{T}$, whose product bounds $\mathbb{P}\left[(\mathbf{x}, \mathbf{z}, \boldsymbol{\theta}), S\right]$ from below (cf. Section 1.2.2). That is, that

$$\mathbb{P}\left[(\mathbf{x}, \mathbf{z}, \boldsymbol{\theta}), S\right] \geq \delta\nu(S), \ \forall S \in \mathscr{B}(\mathcal{X}^N \times \mathcal{Z}^N \times \mathcal{T}).$$

First note that a transition into S can take place in one of two ways. Either all proposed moves are into S and they are all accepted or at least one of the proposed moves is rejected but the result is nonetheless in S. Denote the probability of the first case as $\mathbb{P}_A[(\mathbf{x}, \mathbf{z}, \boldsymbol{\theta}), S]$ and the probability of the second as $\mathbb{P}_R[(\mathbf{x}, \mathbf{z}, \boldsymbol{\theta}), S]$. Then

$$\mathbb{P}\left[(\mathbf{x}, \mathbf{z}, \boldsymbol{\theta}), S\right] = \mathbb{P}_A\left[(\mathbf{x}, \mathbf{z}, \boldsymbol{\theta}), S\right] + \mathbb{P}_R\left[(\mathbf{x}, \mathbf{z}, \boldsymbol{\theta}), S\right] \geq \mathbb{P}_A\left[(\mathbf{x}, \mathbf{z}, \boldsymbol{\theta}), S\right].$$

By the design of swapped BAIS+L the density associated with $\mathbb{P}_A[(\mathbf{x}, \mathbf{z}, \boldsymbol{\theta}), S]$ is given by Equation (3.24),

$$\begin{aligned} p(\mathbf{x}', \mathbf{z}', \boldsymbol{\theta}'|\mathbf{x}, \mathbf{z}, \boldsymbol{\theta}) &= p(\boldsymbol{\theta}'|\mathbf{x}', \mathbf{z}') \prod_{n=1}^{N} p(\mathbf{x}'_n, z'_n|\boldsymbol{\theta}) \alpha_{\boldsymbol{\theta}}(\mathbf{x}_n, \mathbf{x}'_n) \\ &= p(\boldsymbol{\theta}'|\mathbf{x}', \mathbf{z}') \prod_{n=1}^{N} p(z'_n|\mathbf{x}'_n, \boldsymbol{\theta}) p(\mathbf{x}'_n|\boldsymbol{\theta}) \alpha_{\boldsymbol{\theta}}(\mathbf{x}_n, \mathbf{x}'_n), \quad (3.24) \end{aligned}$$

where $\alpha_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{y})$ is the BAIS+L acceptance probability.

By Assumption 2, $p(\mathbf{x}|\boldsymbol{\theta}) \geq \beta\pi(\mathbf{x})$ so the product of the last two factors inside the product is bounded below according to Equation (3.25),

$$p(\mathbf{x}'_n|\boldsymbol{\theta}) \alpha_{\boldsymbol{\theta}}(\mathbf{x}_n, \mathbf{x}'_n) = \min \left\{ p(\mathbf{x}'_n|\boldsymbol{\theta}), \pi(\mathbf{x}'_n) \frac{p(\mathbf{x}_n|\boldsymbol{\theta})}{\pi(\mathbf{x}_n)} \right\} \geq \beta\pi(\mathbf{x}'_n), \quad (3.25)$$

for each $n \in \{1, \ldots, N\}$.

Substituting Equation (3.25) into Equation (3.24) and taking the infimum over $\boldsymbol{\theta}$ gives Equation (3.26),

$$p(\mathbf{x}', \mathbf{z}', \boldsymbol{\theta}'|\mathbf{x}, \mathbf{z}, \boldsymbol{\theta}) \geq p(\boldsymbol{\theta}'|\mathbf{x}', \mathbf{z}') \inf_{\boldsymbol{\theta} \in \mathcal{T}} \left[\prod_{n=1}^{N} p(z'_n|\mathbf{x}'_n, \boldsymbol{\theta}) \right] \beta^N \prod_{n=1}^{N} \pi(\mathbf{x}'_n). \quad (3.26)$$

By Lemma 2

$$0 < p(\boldsymbol{\theta}|\mathbf{x}, \mathbf{z}) < \infty$$

and by Assumption 2

$$0 < \beta\pi(\mathbf{x}) < \infty.$$

Note also that

$$p(z|\mathbf{x}, \boldsymbol{\theta}) = \frac{p(\mathbf{x}, z|\boldsymbol{\theta})}{p(\mathbf{x}|\boldsymbol{\theta})} \quad (3.27)$$

Since the right-hand side (3.27) consists of the ratio of finite and strictly positive functions of $\boldsymbol{\theta}$, the left-hand side is also finite and strictly positive for all $\boldsymbol{\theta}$. Furthermore, as $\mathcal{T}$ is compact, by the Extreme Value Theorem its minimum is attained. That is

$$0 < \min_{\boldsymbol{\theta} \in \mathcal{T}} \left[\prod_{n=1}^{N} p(z'_n|\mathbf{x}'_n, \boldsymbol{\theta}) \right] < \infty.$$

Therefore, all factors on the right-hand side of Equation (3.26) are finite and strictly positive. Furthermore, as the right-hand side of Equation (3.26) is bounded above by $p(\mathbf{x}', \mathbf{z}', \boldsymbol{\theta}'|\mathbf{x}, \mathbf{z}, \boldsymbol{\theta})$, which is integrable over $\mathcal{X}^N \times \mathcal{Z}^N \times \mathcal{T}$, it too is integrable over $\mathcal{X}^N \times \mathcal{Z}^N \times \mathcal{T}$. Note also that, as it is bounded above by a probability density, the integral of the right-hand side of Equation (3.26) is no greater than 1. Let δ be this integral.

Integrating both sides of (3.26) over any set $S \in \mathscr{B}(\mathcal{X}^N \times \mathcal{Z}^N \times \mathcal{T})$ in the Borel σ-algebra of $\mathcal{X}^N \times \mathcal{Z}^N \times \mathcal{T}$ produces the inequality

$$\mathbb{P}_A\left[(\mathbf{x}, \mathbf{z}, \boldsymbol{\theta}), S\right] \geq \delta\nu(S),$$

where ν is the measure with probability density

$$\frac{1}{\delta}\beta^N p(\boldsymbol{\theta}'|\mathbf{x}',\mathbf{z}') \min_{\boldsymbol{\theta}\in\mathcal{T}} [p(z'_n|\mathbf{x}'_n,\boldsymbol{\theta})] \prod_{n=1}^{N} \pi(\mathbf{x}'_n),$$

as required.

Therefore, the Markov chain generated by BAIS+L with swapped transitions is uniformly ergodic.

It then follows, by Lemma 3, that the Markov chain generated by BAIS+L (without swapped) steps is uniformly ergodic. □

Case Two

Theorem 14 (Uniform Ergodicity under Assumption 3)**.** *Under Assumption 3 the Markov chain* $\left\{\boldsymbol{\Theta}^{(t)}, \mathbf{X}^{(t)}, \mathbf{Z}^{(t)}\right\}_{t=1}^{\infty}$ *of random variables representing the parameters, states and latent variables, produced by the N-chain BAIS+L sampler is uniformly ergodic.*

Before proving Theorem 14, a lower bound on the acceptance ratio is needed. This lower bound is given in Lemma 4.

Lemma 4 (Lower Bound on Acceptance Ratio by Assumption 3)**.** *The acceptance ratio of BAIS+L with the conditions listed in Assumption 3 is bounded below by a strictly positive constant.*

Proof. Consider the first condition in Assumption 3. Taking the sum over all possible values of the latent variable gives a relationship that depends only on the chain state,

$$\frac{1}{\zeta}h(\mathbf{x}_n) \leq p(\mathbf{x}_n|\boldsymbol{\theta}) \leq \zeta h(\mathbf{x}_n).$$

This produces a bound on the second argument of the BAIS+L acceptance ratio (cf. Equation (3.15)) according to Equation (3.28),

$$\frac{\pi(\mathbf{y})}{\pi(\mathbf{x}_n)} \cdot \frac{p(\mathbf{x}_n|\boldsymbol{\theta})}{p(\mathbf{y}|\boldsymbol{\theta})} \geq \frac{\frac{1}{\zeta}h(\mathbf{y})}{\zeta h(\mathbf{x}_n)} \cdot \frac{\frac{1}{\zeta}h(\mathbf{x}_n)}{\zeta h(\mathbf{y})} = \frac{1}{\zeta^4} \tag{3.28}$$

as required. □

Theorem 14 may now be proven.

Proof of Theorem 14. The argument to show that BAIS+L is uniformly ergodic under the given conditions follows the same idea as in Theorem 13, by showing that the whole space in which each $(\boldsymbol{\theta}, \mathbf{x}, \mathbf{z})$ lies is small.

As in the proof of Theorem 13, only the case in which all proposals are accepted $\mathbb{P}_A[(\boldsymbol{\theta}, \mathbf{x}, \mathbf{z}), S]$ needs to be considered in order to obtain a lower bound on $\mathbb{P}[(\boldsymbol{\theta}, \mathbf{x}, \mathbf{z}), S]$.

Let $S^{\boldsymbol{\theta},\mathbf{x}} \subset S$ be the projection of S onto $\mathcal{T} \times \mathcal{X}^N$ and let $S^{\mathbf{z}}_{\boldsymbol{\theta},\mathbf{x}}$ be the lower-dimensional subspace of S containing the possible values of $\mathbf{z}|\boldsymbol{\theta}, \mathbf{x}$, as defined in Equations (3.29) and (3.30),

$$S^{\boldsymbol{\theta},\mathbf{x}} := \{(\boldsymbol{\theta}, \mathbf{x}) : (\boldsymbol{\theta}, \mathbf{x}, \mathbf{z}) \in S \text{ for some } \mathbf{z} \in \{1, \ldots, K\}^N\}, \tag{3.29}$$

$$S^{\mathbf{z}}_{\boldsymbol{\theta},\mathbf{x}} := \{\mathbf{z} : (\boldsymbol{\theta}, \mathbf{x}, \mathbf{z}) \in S\}. \tag{3.30}$$

Unlike in the proof of Theorem 13, BAIS+L is considered to consist of parameter updates followed by chain updates. Thus, the joint density of $\boldsymbol{\theta}', \mathbf{x}', \mathbf{z}' | \boldsymbol{\theta}, \mathbf{x}, \mathbf{z}$ may be factored according to Equation (3.31),

$$p(\boldsymbol{\theta}', \mathbf{x}', \mathbf{z}' | \mathbf{x}, \mathbf{z}, \boldsymbol{\theta}) = p\left(\boldsymbol{\theta}' | \mathbf{x}, \mathbf{z}\right) \prod_{n=1}^{N} p\left(\mathbf{x}'_n, z'_n | \boldsymbol{\theta}'\right) \alpha\left(\mathbf{x}'_n | \mathbf{x}_n, \boldsymbol{\theta}'\right). \tag{3.31}$$

Since all proposed states are accepted, $\pi(\mathbf{x}_n) > 0$ and $\pi(\mathbf{x}'_n) > 0$ for all $n \in \{1, \ldots, N\}$, so the acceptance ratio may be replaced with the bound provided in Lemma 4. Integrating over $S^{\boldsymbol{\theta},\mathbf{x}}$ and summing over $S^{\mathbf{z}}_{\boldsymbol{\theta},\mathbf{x}}$ gives the lower bound of Equation (3.32),

$$\begin{aligned} \mathbb{P}_A[(\boldsymbol{\theta}, \mathbf{x}, \mathbf{z}), S] \geq \frac{1}{\zeta^{4N}} \int_{S^{\boldsymbol{\theta},\mathbf{x}}} p(\boldsymbol{\theta}' | \mathbf{x}, \mathbf{z}) \\ \times \sum_{\mathbf{z}' \in S^{\mathbf{z}}_{\boldsymbol{\theta},\mathbf{x}}} \left[\prod_{n=1}^{N} p(\mathbf{x}'_n, z'_n | \boldsymbol{\theta}') \right] d(\boldsymbol{\theta}', \mathbf{x}'). \end{aligned} \tag{3.32}$$

Now only one remaining factor depends on the current point, namely $p(\boldsymbol{\theta}' | \mathbf{x}, \mathbf{z})$. First use Bayes' Theorem to rearrange it, as in Equation (3.33)

$$p(\boldsymbol{\theta}' | \mathbf{x}, \mathbf{z}) = \frac{p(\boldsymbol{\theta}') \prod_{n=1}^{N} p(\mathbf{x}_n, z_n | \boldsymbol{\theta}')}{p(\mathbf{x}, \mathbf{z})} \geq \frac{p(\boldsymbol{\theta}') \zeta^{-N} \prod_{n=1}^{N} h(\mathbf{x}_n, z_n)}{p(\mathbf{x}, \mathbf{z})}. \tag{3.33}$$

Multiplying $p(\boldsymbol{\theta}')$ by the expressions in the first condition of Assumption 3 for all sampling chains gives the inequality presented in Equation (3.34),

$$p(\boldsymbol{\theta}) \prod_{n=1}^{N} \frac{1}{\zeta} h(\mathbf{x}_n, z_n) \leq p(\boldsymbol{\theta}) \prod_{n=1}^{N} p(\mathbf{x}_n, z_n | \boldsymbol{\theta}) \leq p(\boldsymbol{\theta}) \prod_{n=1}^{N} \zeta h(\mathbf{x}_n, z_n), \tag{3.34}$$

which, after integrating over $\boldsymbol{\theta}$, gives the inequality of Equation (3.35),

$$\frac{1}{\zeta^N} \prod_{n=1}^{N} h(\mathbf{x}_n, z_n) \leq p(\mathbf{x}, \mathbf{z}) \leq \zeta^N \prod_{n=1}^{N} h(\mathbf{x}_n, z_n). \tag{3.35}$$

Using this result in Equation (3.33), a bound is obtained, which does not depend on the current states or latent variables of the sampling chains,

$$p(\boldsymbol{\theta}' | \mathbf{x}, \mathbf{z}) \geq \frac{p(\boldsymbol{\theta}') \zeta^{-N} \prod_{n=1}^{N} h(\mathbf{x}_n, z_n)}{\zeta^N \prod_{n=1}^{N} h(\mathbf{x}_n, z_n)} = \frac{p(\boldsymbol{\theta}')}{\zeta^{2N}}.$$

Replacing the factor $p(\boldsymbol{\theta}' | \mathbf{x}, \mathbf{z})$ with this bound in the expression for $\mathbb{P}[(\boldsymbol{\theta}, \mathbf{x}, \mathbf{z}), S]$ and taking the factor of $1/\zeta^{2N}$ outside the integral gives Equation (3.36),

$$\mathbb{P}[(\boldsymbol{\theta}, \mathbf{x}, \mathbf{z}), S] \geq \frac{1}{\zeta^{6N}} \int_{S^{\boldsymbol{\theta},\mathbf{x}}} p(\boldsymbol{\theta}') \sum_{\mathbf{z}' \in S^{\mathbf{z}}_{\boldsymbol{\theta},\mathbf{x}}} \left[\prod_{n=1}^{N} p(\mathbf{x}'_n, z'_n | \boldsymbol{\theta}') \right] d(\boldsymbol{\theta}', \mathbf{x}'). \tag{3.36}$$

Define the probability measure ν according to Equation (3.37),

$$\nu(S) := \int_{S^{\boldsymbol{\theta},\mathbf{x}}} p(\boldsymbol{\theta}') \sum_{\mathbf{z}' \in S^{\mathbf{z}}_{\boldsymbol{\theta},\mathbf{x}}} \left[\prod_{n=1}^{N} p(\mathbf{x}'_n, z'_n | \boldsymbol{\theta}') \right] d(\boldsymbol{\theta}', \mathbf{x}') \tag{3.37}$$

Then

$$\mathbb{P}[(\boldsymbol{\theta}, \mathbf{x}, \mathbf{z}), S] \geq \frac{1}{\zeta^{6N}} \nu(S),$$

demonstrating that the entire state space is small. □

3.3.3 Promoting Adaptation of the Proposal Distribution

The conditions to ensure ergodicity that were common to both assumptions just discussed, required that the mixture weight of each component of the proposal distribution be bounded below by a strictly positive number η. However, the design of BAIS+L does not explicitly prohibit the case that none of the recorded samples come from a given component. That is, the situation given by Equation (3.38) is still possible for some $k \in \{1, \ldots, K\}$,

$$z_n \neq k, \forall n \in \{1, \ldots, N\}. \tag{3.38}$$

The first problem with such a situation arises when sampling new component weights. While the Dirichlet posterior used to generate them guarantees that each component will be given a weight on the interval $(0, 1)$, provided the prior scales α_0 of the components are strictly positive, in practice this may not be sufficient. Since BAIS+L only uses the samples and latent allocations from a single iteration of the algorithm, it is reasonable to require that there be at least one sample from the population at that iteration belonging to each component of the proposal distribution. This means that, as a general rule in a simulation with N sampling chains, no component should ever have a weight less than $1/N$. The effect of such a requirement is that every component is, on average, given the opportunity to be updated at each iteration of the sampler.

A second problem with the situation of Equation (3.38) arises because BAIS+L updates the proposal distribution parameters of an individual component using the samples that originated from it in the current population. If there are no such samples, then the posterior densities of $\boldsymbol{\Sigma}_k$ and $\boldsymbol{\mu}_k$ given in Equations (3.6) and (3.7) reduce to Equations (3.39) and (3.39),

$$\begin{aligned} \boldsymbol{\Sigma}_k | \mathbf{x}, \mathbf{z} &\sim \text{Inv-W}_{\nu_k^{(0)} + o_k}(\boldsymbol{\Lambda}_k), \\ \boldsymbol{\mu}_k | \mathbf{x}, \mathbf{z} &\sim \mathcal{N}\left[\frac{\kappa_k^{(0)}}{\kappa_k^{(0)} + o_k} \boldsymbol{\mu}_k^{(0)} + \frac{o_k}{\kappa_k^{(0)} + o_k} \bar{\mathbf{x}}_k, \frac{\boldsymbol{\Sigma}_k}{\kappa_k^{(0)} + o_k} \right]. \end{aligned} \tag{3.39}$$

Therefore, an empty component will be updated only using prior information, making the method sensitive to any assumptions made about the problem to which BAIS+L is applied.

The following two subsections describe two attempts to guarantee that the component weights are bounded below by a positive constant. Their effectiveness and practical implications are then assessed in Section 3.3.4.

Forcing a Minimum Component Weight

The first approach to ensure that component weights remain strictly positive throughout a simulation is to explicitly set a minimum weight $d_{\min}$. To do so, split each component weight into two parts: one fixed part, equal to $d_{\min}/K$; and a variable second part, which is updated based on the current latent allocations.

Then implement a fixed number of iterations of a Gibbs sampler to update the variable part.

The precise procedure involved begins by initialising each component $k \in \{1, \ldots, K\}$ to have equal weight of $d'_k = 1/K$. At each pass of the sampler the method first counts the number of allocations to each component, excluding some counts at random. This random exclusion accounts for the forced minimum weight, which requires that an expected number of $d_{\min}N$ allocations be ignored at each iteration. To prevent less-represented components from being disproportionately penalised, the individual probability of ignoring a count for component k needs to be proportional to its weight d_k. However, this is the value that is to be updated, hence a Gibbs sampler is used to infer it. The explicit probability $\mathbb{P}^{(k)}_{\text{ignore}}$ of ignoring the count of a sampling chain allocated to component k is given by Equation (3.40),

$$\mathbb{P}^{(k)}_{\text{ignore}} = \frac{d_{\min}}{K d_k}. \tag{3.40}$$

Therefore, the combined weight of a component is given by Equation (3.41),

$$d_k = \frac{d_{\min}}{K} + d'_k(1 - d_{\min}). \tag{3.41}$$

$\mathbf{d}$ is updated following the preceding prescription until some stopping criterion is met, such as a sufficient number of sweeps of the Gibbs sampler or once the change $\|\bar{\mathbf{d}}^{(t+1)} - \bar{\mathbf{d}}^{(t)}\|$ in the running average $\bar{\mathbf{d}}^{(t)} = \sum_{i=1}^{t} \mathbf{d}^{(i)}/t$, from one sweep t of the Gibbs sampler to the next, drops below a given threshold change Δ_t.

The procedure to enforce a minimum component weight is summarised by Algorithm 3.2.

In theory, any value of $d_{\min}$ greater than zero will ensure that the conditions for uniform ergodicity are met. However, while guaranteeing that each component will be available to propose new states throughout a simulation, the method of the current chapter does not guarantee that samples from every component will be represented in the final population. This is because the method only addresses the proposal distribution and not the acceptance ratio, meaning that proposed states may still be rejected. As such, the mean and variance-covariance matrix of an unrepresented component will still have to be updated using only prior information.

The result of some components only being updated infrequently is that the sampler may converge slowly. Therefore, care must be taken in the choice of $d_{\min}$. A sensible minimum value for $d_{\min}$ is one that results in at least two proposed states being drawn from each component at each iteration. If at least two unique proposed states from a component are accepted then the component will have enough information to be adapted, thereby not relying only on potentially imprecise prior information.

The next subsection discusses an approach that guarantees a minimum number of sampling chains allocated to each component, thereby forcing each component to explore the state space.

Fixing the Component Memberships of Some Sampling Chains

An alternative approach to setting a minimum component weight is to fix the component memberships of a fixed number of sampling chains. This approach

Algorithm 3.2 Algorithm to enforce a minimum component weight.

Require:

1. At least one of:
   (a) A sufficient number of iterations I_d to ensure convergence has been achieved.
   (b) A threshold change Δ_t in the running average in $\bar{\mathbf{d}}$.
2. A vector of prior concentrations $\boldsymbol{\alpha}^{(0)} = \left[\alpha_1^{(0)}, \ldots, \alpha_K^{(0)}\right]$.

Ensure: A vector of new component weights $d = (d_1, \ldots, d_K)$.

Set $d = (d'_1, \ldots, d'_K) = (1/K)$.
Set $i = 1$.
Set $\Delta = \infty$.
Set $\bar{\mathbf{d}} = \mathbf{0}$.
while $i \leq I_d$ and $\Delta > \Delta_t$ **do**
    Set $\mathbf{a} = (a_1, \ldots, a_K) = \boldsymbol{\alpha}^{(0)}$.
    Set $\bar{\mathbf{d}}_{\text{old}} = \bar{\mathbf{d}}$.
    for $n \in \{1, \ldots, N\}$ **do**
        Generate $u \sim \mathcal{U}(0, 1)$.
        Set $\mathbb{P}^{(k)}_{\text{ignore}} = \dfrac{d_{\min}}{K d_k}$.
        if $u > \mathbb{P}^{(k)}_{\text{ignore}}$ **then**
            Set $a_k = a + 1$.
        end if
    end for
    Generate $\mathbf{d}' \sim \text{Dirichlet}(\mathbf{a})$.
    for $k \in \{1, \ldots, K\}$ **do**
        Set $d_k = \dfrac{d_{\min}}{K} + d'_k(1 - d_{\min})$.
    end for
    Set $\Delta = \|\bar{\mathbf{d}} - \mathbf{d}_{\text{old}}\|$.
    Set $i = i + 1$.
end while

artificially inflates the weights of components corresponding to smaller local maxima of the target, thereby giving them more opportunity to adapt to a local maximum of the target than they would have had with their lower weights. However, these artificially inflated weights introduce a problem when making inference about the target distribution. Fortunately, it is solved by simply discarding the chains with fixed allocations and only using the variable allocation chains for inference. Therefore, a balance needs to be struck between guaranteeing reliable results and the extra computational burden entailed in keeping some latent allocations fixed.

To implement a fixed portion of memberships is more straightforward than the minimum component weight method of the previous subsection. Firstly, the first portion of $d_{\min}N$ fixed allocations needs to be set only once, according to Equation (3.42),

$$z_n = \left\lfloor \frac{nK}{d_{\min}N} \right\rfloor, \tag{3.42}$$

where $n \in \{1, \ldots, d_{\min}N\}$. The remainder of the chains are then initialised uniformly and updated at each iteration following the standard BAIS+L prescription of Section 3.1. To update the mixture proportions $\mathbf{d}$, only the allocations of the variable membership chains are used as input to the posterior distribution of Section 3.1.3.

It is obvious that the sampling chains drawn from a fixed component have a different proposal distribution from those with variable allocations; namely, they are drawn from a single component. Therefore, they must use a different acceptance ratio that only considers the involved component. Explicitly, the acceptance ratio of the nth fixed chain is given by Equation (3.43),

$$\alpha(\mathbf{x}_n, \mathbf{y}_n) = \frac{\pi(\mathbf{y}_n)}{\pi(\mathbf{x}_n)} \cdot \frac{\mathcal{N}\left(\mathbf{x}_n \,|\boldsymbol{\mu}_{z_n}, \boldsymbol{\Sigma}_{z_n}\right)}{\mathcal{N}\left(\mathbf{y}_n \,|\boldsymbol{\mu}_{z_n}, \boldsymbol{\Sigma}_{z_n}\right)}. \tag{3.43}$$

On one hand, the fixed membership method of the current subsection *never* updates component means and variance-covariance matrices only using prior information, unlike the method of the previous subsection, which may occasionally do so. On the other hand, unlike the method of the previous subsection, it does not explicitly guarantee a non-zero probability that a proposal will be drawn from each component of the proposal distribution. The effect of these criteria is that all components will start on an equal footing and no part of the parameter space will be overlooked *a priori*. These properties, coupled with the guaranteed adaptation of each component at each iteration using sampled points, will result in a proposal distribution with local maxima that better approximate those of the target. As such, if a component empties out then it will be more likely due to it being insignificant than from being poorly positioned in the state space.

Of course, the approach only accelerates the adaptation of the mean and variance-covariance matrix. It is possible that any improvement in the speed of adaptation may still be insufficient to overcome potential emptying of components if the number of fixed allocation chains is insufficient.

Both the methods of the previous and current subsections have their own advantages and disadvantages. Therefore, Section 3.3.4 explores the effectiveness of combinations of the two methods for various choices of their operating parameters.

3.3.4 Assessing the Effectiveness of Promoting Adaptation

This section implements the methods of Section 3.3.3 on a target distribution with four local maxima. The goal is to provide preliminary guidance on how to select both the minimum component weight d_{min} and the number of fixed-allocation sampling chains N_{fixed}.

Methodology

To study the effect of different combinations of d_{min} and N_{fixed}, BAIS+L was employed on a four-component mixture of bivariate normal distributions with density given by Equation (3.44),

$$\pi(\mathbf{x}) = \sum_{k=1}^{4} \frac{1}{4} \mathcal{N}\left(\mathbf{x} \middle| \begin{bmatrix} (-1)^k 4 \\ -4 + \lfloor \frac{k}{3} \rfloor \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right). \tag{3.44}$$

Ten independent simulations were run for each combination of

$$d_{\text{min}} \in \{0, 0.1, 0.2, 0.3, 0.4, 0.5\}$$

and

$$N_{\text{fixed}} \in \{0, 8, 16, 24, 32, 40\}.$$

Each simulation used eight components in the proposal distribution (twice as many as there were local maxima in the target) and 160 sampling chains, to provide, on average 20 sampling chains per component.

The prior distribution was chosen by considering the true form of the target. To make the prior form of each component overdispersed, $\boldsymbol{\Sigma}^{(0)}$ was set to $1.2\mathbf{I}_2$, where $\mathbf{I}_2$ is the two-dimensional identity matrix. The prior mean $\boldsymbol{\mu}^{(0)}$ was set to $(0, 0)^{\text{T}}$, the mean of the target. Finally, to make the entire prior distribution overdisperse, $\kappa^{(0)}$ was set to 0.046875 for each component. The chosen value represents the ratio of the variance of a single component in the target's mixture to that of the entire target.

With these settings, the 360 independent simulations were run for 3000 iterations each, initialising each dimension of each sampling chain from $\mathcal{N}(\mu = 0, \sigma = 6)$. A further 360 simulations with a different initialisation scheme were also run. In these simulations the initial states of the sampling chains were generated uniformly on the square $(0, 6) \times (0, 6)$. The reason for this different choice of initialisation was to investigate the effect of ignoring all but one component of the target.

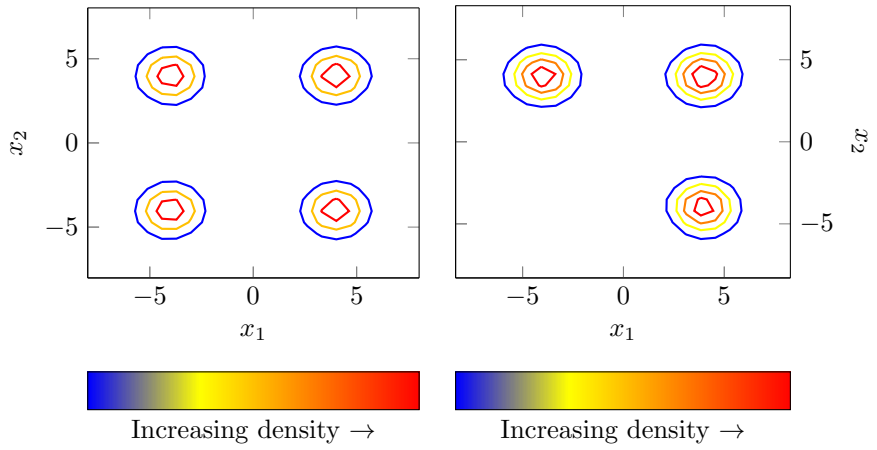
To ensure that convergence could not be ruled out, Gelman and Rubin's diagnostic was applied to the sampler output using the `gelman.plot` function from the `coda` library (Plummer et al., 2006) of the statistical computing package `R` (R Core Team, 2015), employing a PSRF threshold of 1.01.

In all simulations for which $d_{\text{min}} > 0$, I_d was taken to be 10 and Δ_t to be 0.01.

Results and Discussion

Gelman and Rubin's diagnostic indicated that all 720 simulations maintained a PSRF below 1.01 for the last halves of their runs. This number was deemed to

Figure 3.1: Examples of simulation output from the four-component mixture. The left plot shows the shape of the contours of the inferred stationary distribution from a successful run with $d_{\min} = 0$ and $N_{\text{fixed}} = 0$, while the right plot shows an example of a failed run, with $d_{\min} = 0$ and $N_{\text{fixed}} = 40$, which missed a component.



be sufficiently low in order to assume convergence had been met. Therefore, all simulations are considered in the following discussion.

With the overdispersed initial chain states from $\mathcal{N}(0, 6)$ it was found that there was no discernible difference between the various combinations of $d_{\min}$ and N_{fixed}. However, this was not the case for the simulations initialised on the $(0, 6) \times (0, 6)$ square. Of all combinations of $d_{\min}$ and N_{fixed}, it was found that $N_{\text{fixed}} = 0$ was the only setting that consistently reproduced the correct form of the target distribution, with four, clearly-separated regions of significant mass, centred at the true locations of the target components and with the true scales (see Figure 3.1 for examples). This result shows that, for this particular target, incorporating fixed-allocation sampling chains hindered the sampler's ability to detect all local maxima. Furthermore, all runs for $N_{\text{fixed}} = 0$ with any choice of $d_{\min}$ converged rapidly to an acceptance rate of around 0.8 (within 100 iterations in each case).

One possible reason for the simulations with $N_{\text{fixed}} > 0$ resulting in non-detection of some local maxima is that they force the component locations and scales to adapt at each iteration. Such action may be beneficial when components of the proposal distribution explore the state-space slowly (for example, due to irregularly-shaped or very well-separated local maxima in the target), thereby preventing the component from emptying before it has had a chance to converge to a region encompassing a true local maximum of the target. The observations of this section indicate that the local maxima of the target studied were not sufficiently-spaced to necessitate a method to accelerate the finding of local maxima. By using fixed-allocation sampling chains to force each component of the proposal distribution to focus on a particular part of the state space, BAIS+L was discouraged from further exploration once a component of the proposal distribution had found a local maximum, even if that local maximum was the same as one identified by another component.

Considering the requirement that each component weight be bounded below by a positive constant, it is recommended that every simulation run with BAIS+L set a non-zero $d_{\min}$. As the observations indicated, even though the convergence diagnostic used did not detect non-convergence of any of the simulation runs that were employed, it was not guaranteed that $N_{\text{fixed}} > 0$ would result in detection of all the local maxima of the target. Therefore, it is advised against using extra sampling chains with fixed allocations unless the acceptance rate of a simulation with $d_{\min}$ converges to an unsuitably low value. One such situation that may lead to a low acceptance rate is when the target has heavy tails, in which the probability of proposing a state in a low density region of the target is significant.

While a heavy-tailed target was not considered in these simulations, it is theoretically possible that such a situation may lead to sampled points within the tails that do not move. This is because such extreme points are more likely with a heavy-tailed target. The proposal distribution used does not use heavy tails, so once all local maxima of the target have been detected the proposal density in the tails of the target will be lower than the target density. Recall the standard MH acceptance ratio α,

$$\alpha(x, y) = \frac{\pi(y)}{\pi(x)} \cdot \frac{g(x)}{g(y)}.$$

It is evident that a high ratio of target density at a proposed state y to that at the current state x is lower than the ratio of the proposal density at the proposed state to that at the current one. The result is that the acceptance ratio will be very low, resulting in the state becoming stuck at the state in the tail of the target.

Furthermore, the states stuck in the tail inflate the variance-covariance matrix of the component to which they have been assigned, thereby lowering the proposal density of significant states within the region of the target that the proposal component represents. This, in turn, causes low density states to be more likely to be proposed than they would otherwise be, again resulting in a low acceptance ratio and, hence, a low acceptance rate.

With a fixed number of sampling chains in a component, it is encouraged to be updated more often, increasing the speed at which it explores the state space. This property means that the sampler should find and approximate the region about a true local maximum of the target sooner, although such a claim needs to be verified by finding a suitable target distribution for testing.

An alternative to using a fixed minimum number of sampling chains in each component is to use a heavy-tailed kernel, such as a Student's t-distribution. Part I has already reviewed the use of such a kernel in the mixture proposal methods introduced by Frühwirth-Schnatter and Pyne (2010) and Pompe et al. (2018), suggesting Student's t-distributions as a useful kernel to consider in future work.

3.4 Conjectured Properties of BAIS+L

The design of BAIS+L poses a number of important questions regarding the ergodicity and convergence properties of its resulting Markov chain. It has already been proven in Section 3.3 that BAIS+L is uniformly ergodic under

two given sets of conditions. However, it is not immediately obvious that the stationary distribution is equal to the designed target. Indeed, as noted in Section 3.1.4, the approximate acceptance ratio used in Equation (3.15) results in the stationary distribution being different from the target when BAIS+L is implemented with a finite number of sampling chains.

Recall that this approximation replaces $p(\cdot|\mathbf{x}_{-n})$ with $p(\cdot|\boldsymbol{\theta})$. The dependence structure of these two densities is different, with the former depending only on the population of sampling chain states *excluding* the current one being updated, and the latter depending on *all* of them. This difference in dependence motivates the following hypothesised properties of BAIS+L, the most important of which, is given by Conjecture 1.

Conjecture 1. *Let π be the target density and let π_N be the invariant measure of a Markov chain $\left\{\boldsymbol{\Theta}^{(t)}, \mathbf{X}^{(t)}, \mathbf{Z}^{(t)}\right\}_{t=1}^{\infty}$ produced using an N-sampling chain implementation of BAIS+L to sample from π. Then, as N is increased, π_N converges weakly to π. That is,*

$$\lim_{N\to\infty} \int f(\mathbf{x}) d\pi_N(\mathbf{x}) = \int f(\mathbf{x}) d\pi(\mathbf{x})$$

for any bounded and continuous function f.

Intuition suggests that a possible avenue towards the truth of Conjecture 1 is if the limiting density (with respect to N) of the proposal parameters is concentrated on just one value $\theta_\infty \in \mathcal{T}$. This belief is summarised in Conjecture 2.

Conjecture 2. *The marginal distribution $p(\boldsymbol{\theta})$ of the BAIS+L proposal parameters $\boldsymbol{\theta} \in \mathcal{T}$ approaches a point mass on some $\boldsymbol{\theta}_\infty \in \mathcal{T}$ as the number of sampling chains N is increased.*

If Conjecture 2 is true, then a natural consequence is that, in the limit of an infinite number of sampling chains, $p(\cdot|\boldsymbol{\theta} = \boldsymbol{\theta}_\infty)$ will be time-invariant. With the conditions outlined in Section 3.3.1, Theorems 13 and 14 guarantee that the Markov chain $\left\{\boldsymbol{\Theta}^{(t)}, \mathbf{X}^{(t)}, \mathbf{Z}^{(t)}\right\}_{t=1}^{\infty}$ of parameters, states and latent variables is uniformly ergodic. So, for BAIS+L with an infinite number of sampling chains, given enough time, $p(\boldsymbol{\theta})$ will converge. This hypothesis is summarised by Conjecture 3.

Conjecture 3. *The parameter vector $\boldsymbol{\theta}$ of BAIS+L with an infinite number of sampling chains, will converge to a constant $\boldsymbol{\theta}_\infty$. That is, the BAIS+L process with $N = \infty$ exhibits diminishing adaptation.*

A result of Conjecture 3 is that the sampling chains will decouple with time, resulting in some maximum acceptance rate for a given number K of mixture components. Intuition dictates that, in order for this acceptance rate to be 1, a sufficient condition is that $K = \infty$. Furthermore, as a BAIS+L simulation with an infinite number of sampling chains progresses, it will increasingly resemble a collection of simultaneous independent MH samplers, each of which will, therefore, converge to the target distribution.

The preceding conjectures provide important theoretical focus for future work, and Chapter 4 explores some of them empirically.

Chapter 4

Comparing BAIS+L to the Equi-Energy Sampler

This chapter compares BAIS+L to the Equi-Energy sampler (EES) of Kou et al. (2006). By applying BAIS+L to three of the continuous state space targets that Kou et al. (2006) used to study the performance of EES, it is demonstrated that BAIS+L is a viable alternative that is able to efficiently sample from a target distribution supported on a non-denumerable state space, without the need for an energy ladder. By doing so, it uses all sampling chains for inference, without modification, avoiding waste of high-temperature samples or the extra computational effort required to transform them. It also allows all post-burn-in samples to be used for inference of the target directly. This chapter also discusses some of the pitfalls associated with BAIS+L and suggests avenues for further research in order to address them.

4.1 Simulation from a Mixture Target

This section compares BAIS+L to EES by simulating from the mixture target of Liang and Wong (2001), which was used by Kou et al. (2006) to evaluate EES. For convenience, the probability density function of this target is restated in Equation (4.1),

$$f(\mathbf{x}) = \sum_{i=1}^{20} \frac{w_i}{2\pi\sigma_i^2} \exp\left[-\frac{1}{2\sigma_i^2}\left(\mathbf{x}-\boldsymbol{\mu}_i\right)^{\mathrm{T}}\left(\mathbf{x}-\boldsymbol{\mu}_i\right)\right], \tag{4.1}$$

where $\mathbf{x} = (x_1, x_2)^{\mathrm{T}}$ and

$$\boldsymbol{\mu}_1 = \begin{pmatrix} 2.18 \\ 5.76 \end{pmatrix}, \quad \boldsymbol{\mu}_2 = \begin{pmatrix} 8.67 \\ 9.59 \end{pmatrix}, \quad \boldsymbol{\mu}_3 = \begin{pmatrix} 4.24 \\ 8.48 \end{pmatrix}, \quad \boldsymbol{\mu}_4 = \begin{pmatrix} 8.41 \\ 1.68 \end{pmatrix},$$
$$\boldsymbol{\mu}_5 = \begin{pmatrix} 3.93 \\ 8.82 \end{pmatrix}, \quad \boldsymbol{\mu}_6 = \begin{pmatrix} 3.25 \\ 3.47 \end{pmatrix}, \quad \boldsymbol{\mu}_7 = \begin{pmatrix} 1.70 \\ 0.50 \end{pmatrix}, \quad \boldsymbol{\mu}_8 = \begin{pmatrix} 4.59 \\ 5.60 \end{pmatrix},$$
$$\boldsymbol{\mu}_9 = \begin{pmatrix} 6.91 \\ 5.81 \end{pmatrix}, \quad \boldsymbol{\mu}_{10} = \begin{pmatrix} 6.87 \\ 5.40 \end{pmatrix}, \quad \boldsymbol{\mu}_{11} = \begin{pmatrix} 5.41 \\ 2.65 \end{pmatrix}, \quad \boldsymbol{\mu}_{12} = \begin{pmatrix} 2.70 \\ 7.88 \end{pmatrix},$$
$$\boldsymbol{\mu}_{13} = \begin{pmatrix} 4.98 \\ 3.70 \end{pmatrix}, \quad \boldsymbol{\mu}_{14} = \begin{pmatrix} 1.14 \\ 2.39 \end{pmatrix}, \quad \boldsymbol{\mu}_{15} = \begin{pmatrix} 8.33 \\ 9.50 \end{pmatrix}, \quad \boldsymbol{\mu}_{16} = \begin{pmatrix} 4.93 \\ 1.50 \end{pmatrix},$$
$$\boldsymbol{\mu}_{17} = \begin{pmatrix} 1.83 \\ 0.09 \end{pmatrix}, \quad \boldsymbol{\mu}_{18} = \begin{pmatrix} 2.26 \\ 0.31 \end{pmatrix}, \quad \boldsymbol{\mu}_{19} = \begin{pmatrix} 5.54 \\ 6.86 \end{pmatrix}, \quad \boldsymbol{\mu}_{20} = \begin{pmatrix} 1.69 \\ 8.11 \end{pmatrix}.$$

For a direct comparison to EES, BAIS+L was applied the same two cases studied by Kou et al. (2006). In the first of these cases each mixture component i in the target has the same weight $w_i = 0.05$ and variance $\sigma_i^2 = 0.1$. In the second case the weight is inversely proportional to the distance between the component mean and $(5, 5)^{\mathrm{T}}$,

$$w_i \propto \left\| \boldsymbol{\mu}_i - \begin{pmatrix} 5 \\ 5 \end{pmatrix} \right\|^{-1}, \tag{4.2}$$

and the variance in each dimension is directly proportional to it,

$$\sigma_i^2 = \frac{1}{20} \left\| \boldsymbol{\mu}_i - \begin{pmatrix} 5 \\ 5 \end{pmatrix} \right\|. \tag{4.3}$$

In Equations (4.2) and (4.3) $\|\cdot\|$ represents the Euclidean norm.

4.1.1 Methodology

The comparison considered five different values of the number of mixture components K in the proposal distribution: $K = 20$, $K = 30$, $K = 40$, $K = 50$ and $K = 60$. The number of sampling chains N was also varied for each number of mixture components. In the simulations, BAIS+L was run with $N = 1000$, $N = 1500$ and $N = 2000$. Each parameter setting was repeated, for a total of 20 independent simulations, each with 1000 iterations. As Section 4.1.2 will show, this number of iterations was sufficient as all simulations used for inference converged within 500 iterations.

Finally, to ensure that the conditions for ergodicity discussed in Chapter 3, were met, a minimum component weight of $d_{\min} = 0.1$ was enforced by following the prescription given in Section 3.3.3, with $I_d = 10$ and $\Delta_t = 0.01$ (cf. Section 3.3.3 for the definitions of I_d and Δ_t).

Selecting the Parameters of the Prior Distribution

As the proposal parameter update process is Bayesian by design, a prior model must first be specified on the proposal parameters. The development of BAIS+L, in Chapter 3, gave the general form of the prior distribution on Euclidean state-spaces, whose hyperparameters $\boldsymbol{\alpha}^{(0)}$, $\nu^{(0)}$, $\boldsymbol{\kappa}^{(0)}$, $\boldsymbol{\mu}^{(0)}$ and $\boldsymbol{\Sigma}^{(0)}$ were chosen as follows for the mixture target.

In practice, it is not expected that the weights or numbers of components will be known beforehand, so the prior parameter of the weight distribution $\boldsymbol{\alpha}^{(0)}$ was set to the vector of 1s of length K.

In the mixture target of Liang and Wong (2001), the shape of the target distribution is known *a priori*, as is the fact that the centres of all 20 of its mixture components lie within the 10×10 square $[0, 10] \times [0, 10]$. However, following Kou et al. (2006), a challenge was posed for the sampler, by setting $\boldsymbol{\mu}_k^{(0)} = (0, 0)^{\mathrm{T}}$ for each component k. This choice also considerably simplified the computer code used to implement BAIS+L compared to that for a non-zero prior mean.

Pilot runs implemented 60 components in the proposal distribution and 2000 sampling chains. Initially, an arbitrary selection of $\nu_k^{(0)} = 3$, $\kappa_k^{(0)} = 1$ and $\boldsymbol{\Sigma}_k^{(0)} = \mathbf{I}_2$ (the two-dimensional identity matrix) was used for each component k. The choice of a diagonal matrix for $\boldsymbol{\Sigma}_k^{(0)}$ with the same value $\boldsymbol{\Sigma}^{(0)}$ in each diagonal position simplified the computer code used to implement the sampler.

For the equal weight and variance case, pilot runs indicated that these prior parameter settings were inappropriate, leading to inconsistent acceptance rates over multiple replications of the same simulation with the same input parameters. The sampler also did not detect all components of the target distribution in every run, thereby exhibiting the quasi-ergodic problem (cf. Section 1.2.2). This observation indicated that the prior distributions on the proposal parameters did not place sufficient mass near all of the target's component centres to explore them effectively.

Given enough time, however, even with this naïve choice of proposal parameters it is expected that the generated Markov chain will eventually converge to the target distribution. The validity of this claim is supported by the fact that some independent runs found more local maxima than others and by the proofs of ergodicity provided in Section 3.3. Nevertheless, these observations highlight why it is a bad idea to use a naïve choice of hyperparameters, even with an adaptive mixture proposal, as in the case of BAIS+L.

To make it easier for the sampler to find all local maxima of the target, $\kappa^{(0)}$ was lowered, thereby making the distribution of a component mean, given a variance-covariance matrix, more disperse. The scale of the distribution on the variance-covariance matrices $\boldsymbol{\Sigma}_k$ was also adjusted, by setting $\boldsymbol{\Sigma}_k^{(0)}$ to a smaller multiple of the two-dimensional identity matrix $\mathbf{I}_2$ than 1. The specific values settled on were $\boldsymbol{\Sigma}_k^{(0)} = 0.1\mathbf{I}_2$ and $\kappa_k^{(0)} = 0.001$, which resulted in the sampler consistently having a high acceptance rate of at least 0.6 and finding all local maxima of the target in almost all runs. Therefore, these values were fixed for all subsequent simulations.

Note that for this target it was known *a priori* that all of the target's mixture component centres were in the square $[0, 10]^2$ and that each had a true variance of 0.1 in each dimension, as specified by Kou et al. (2006). This prior knowledge is reflected in the choices of $\boldsymbol{\Sigma}_k^{(0)}$ and $\kappa_k^{(0)}$. By both reducing the diagonal elements of $\boldsymbol{\Sigma}_k^{(0)}$ to the true variance and reducing $\kappa_k^{(0)}$ so that the diagonal elements of $\boldsymbol{\Sigma}_k^{(0)}/\kappa_k^{(0)}$ were each equal to 100 (the square of the edge length of the $[0, 10]^2$ square) a higher, more repeatable acceptance rate was achieved, while nearly always finding all local maxima. On one hand, reducing the size of each diagonal element of $\boldsymbol{\Sigma}^{(0)}$ from 1 to 0.1 allowed a more faithful reflection of the true variance of an individual mixture component of the target, reducing the chances of the target's local maxima becoming conflated. On the other hand, reducing $\kappa_k^{(0)}$ from 1 (with $\boldsymbol{\Sigma}_k^{(0)} = \mathbf{I}_2$) to 0.001 (with $\boldsymbol{\Sigma}_k^{(0)} = 0.1\mathbf{I}_2$) increased

the size of the search area of the state space to one that better represented the true variance of the target distribution.

Ideally one will use prior information about the variance of the target to guide the choice of the hyperparameters. In practice, however, when neither the number nor the locations of the target's local maxima, nor the variances in their corresponding regions are known, some experimentation may be required in order to find suitable values. In such cases it is suggested that the practitioner record the acceptance rates and sampling chain allocations of several short runs with different values of the hyperparameters. One approach to using the acceptance rates of short pilot runs will be discussed in detail in Section 4.1.2.

In the pilot simulations, it was found that the acceptance rate of successful runs (those in which all of the target's mixture components were detected) settled rapidly to an approximately constant value, usually within 100 iterations. Therefore, it appears reasonable to use short pilot runs to aid in hyperparameter selection. Future work, however, should seek to develop a more systematic and robust methodology for hyperparameter selection.

In the unequal weight and variance case, pilot runs with 60 proposal components and 2000 sampling chains demonstrated that the same choice of hyperparameters as the unequal weight and variance case consistently produced the correct stationary distribution with a sufficiently high acceptance rate.

For neither target was it deemed necessary to adjust $\nu_k^{(0)}$ or $\alpha^{(0)}$ for any component k. A constant $\alpha_k^{(0)} = 1$, and $\nu_k^{(0)} = d+1 = 3$ were easy to implement and did not require careful consideration of the shape of the target.

Therefore, in all long simulations, the prior parameters were set according to $\alpha_k^{(0)} = 1$, $\nu_k^{(0)} = 3$, $\kappa_k^{(0)} = 0.001$ $\boldsymbol{\mu}_k^{(0)} = (0,0)^{\mathrm{T}}$ and $\boldsymbol{\Sigma}_k^{(0)} = 0.1\mathbf{I}_2$ for each component k of the proposal distribution.

Assessing Efficiency

For each full-length simulation, the run time was measured using the `time` built-in command from version 4.3 of `bash` (The Free Software Foundation, 2014). A simulation's run time was taken to be the total processor time used in its execution, given by the sum of the `user` and `sys` times reported by `time`.

In order to assess convergence, Gelman and Rubin's convergence diagnostic (Gelman and Rubin, 1992) (cf. Section 1.2.5) was applied to the simulation output, using the `gelman.plot` function from R's (R Core Team, 2015) `coda` library (Plummer et al., 2006) with the settings `confidence = 0.95`, `transform = FALSE`, `autoburnin = FALSE` and `multivariate = FALSE`, on the N sampling chains. With these settings the `gelman.plot` function computed the diagnostic using all samples up to iteration $51 + 19m$, for $m \in \mathbb{N}$.

While Gelman and Rubin's convergence diagnostic assumes that the sampling chains are independent, which is not the case for BAIS+L, if BAIS+L is implemented with a large-enough number of sampling chains, then the sampling chains are approximately independent. Hence, the required independence was assumed to hold approximately. As the number of sampling chains increases, the variance of the marginal distribution of $\boldsymbol{\theta}$ is reduced. A fixed value of $\boldsymbol{\theta}$ must produce independent chains because it has no dependence on them. If the distribution of $\boldsymbol{\theta}$ approaches a point mass, as conjectured, then the sampling chains should approach independence (cf. Section 3.4). The larger number of

sampling chains also results in the distributions of the sample statistics and, hence, the distribution of the family of posterior distributions of the proposal parameters, being more concentrated due to greater precision inherent in the larger sample. Furthermore, the sampling chain states within a single iteration are updated independently of one another, given the proposal parameters.

For each simulation, the convergence time τ was inferred to be the earliest iteration after which the maximum 97.5% confidence PSRF (cf. Section 1.2.5) remained below 1.01 for the remainder of the simulation. An estimate of the mean convergence time $\hat{\tau}$ was then computed using the sample mean of the convergence times of the simulations, as well as the maximum encountered 97.5% PSRF over the 20 simulations at each parameter setting.

To compute the variance of the estimated mean convergence time, it was first necessary to overestimate the variance of the value computed for a single run to be one quarter of the square of the length of the time interval by the end of which it converged. That is, if a simulation converged in the first 51 iterations then the variance of its observed convergence time was $51^2/4$ iterations and $19^2/4$ if it converged in any of the subsequent intervals. This variance corresponds to the conservative assumption that the variance of an individual estimate is maximal on the interval and is obtained by Popoviciu's Inequality (Popoviciu, 1935),

$$\mathrm{Var}[\hat{\tau}^{(r)}] \leq \frac{(b-a)^2}{4},$$

where the variance is denoted by $\mathrm{Var}[\hat{\tau}^{(r)}]$ and $\hat{\tau} \in [a, b]$.

The overall variance of the estimated mean convergence time $\hat{\tau}$ was then given by Equation (4.4),

$$\mathrm{Var}(\hat{\tau}) = \frac{1}{R}\sum_{r=1}^{R}\left\{\frac{1}{R-1}\left[\hat{\tau}^{(r)} - \hat{\tau}\right]^2 + \mathrm{Var}\left[\hat{\tau}^{(r)}\right]\right\}, \tag{4.4}$$

In Equation (4.4) the first term inside the sum represents the normalised squared deviation of the rth run's convergence time estimate from the estimated mean $\hat{\tau}$ over all $R = 20$ runs and the second term represents the variance of the rth estimate.

To determine the effective number of samples of each reported statistic generated by BAIS+L, the `effectiveSize` function in R's `coda` package, with its default settings, was applied to the samples from the last half of each simulation run. Like Gelman and Rubin's diagnostic, this function also assumes that the sampling chains are independent. However, the same reasoning that was used to justify the use of this diagnostic also justifies the use of the `effectiveSize` function. The computed values were used to correct the variances of the reported statistics in Section 4.1.2.

Since Kou et al. (2006) reported the autocorrelations of the lowest-temperature chain that they simulated, the integrated autocorrelation time (IAT, cf. Section 1.2.5) of each coordinate of each sampling chain was computed. A minimum estimate of the effective proportion of samples was then estimated as the reciprocal of the maximum IAT computed. For each parameter setting, the mean effective proportion over the 20 repeated simulations was estimated as the overall estimate $\hat{p}_{\mathrm{eff}}$ for that parameter setting.

To compute the variance of the minimum effective proportion for each simulation it was assumed that the effective number of samples followed a binomial

distribution. This gave the variance of the estimator of the effective proportion by a normal approximation to a binomial distribution (Bertsekas and Tsitsiklis, 2002, pp. 114), as given in Equation (4.5),

$$\mathrm{Var}(\hat{p}_{\mathrm{eff}}) = \frac{2\hat{p}_{\mathrm{eff}}(1-\hat{p}_{\mathrm{eff}})}{T}. \tag{4.5}$$

Here T represents the total number of iterations in a single simulation and the factor of 2 takes into account that only half of the iterations were used for inference.

For comparison, the effective number of samples produced by EES was also estimated, by referring to the correlograms depicted in Figures 3(c) and 4(a) of Kou et al. (2006). To extract the autocorrelation at each time-lag from their plots, they were passed through the online `WebPlotDigitizer` tool of Rohatgi (2011). In order to avoid under-reporting the effective numbers of samples produced by EES, a conservative approach was adopted when reading the autocorrelations from the plots of Kou et al. (2006). That is, the top of the column at each time-lag was taken to occur at the lowest possible correlation value that could reasonably be considered to be height of the column. The result is that the true effective number of samples from their results should be no more than the derived values that are reported in the figures of this chapter. Once the correlations of the samples of X_1 and X_2 obtained by Kou et al. (2006) using EES had been estimated, their corresponding IAT were then computed. Recall the expression for the IAT, which is restated in Equation (4.6),

$$\mathrm{IAT} = \frac{1}{2} + \sum_{t=1}^{\infty} \mathrm{Corr}_t(\mathbf{x}) \approx \frac{1}{2} + \sum_{t=1}^{T_{\max}} \mathrm{Corr}_t(\mathbf{x}), \tag{4.6}$$

In Equation (4.6), $T_{\max} = 100$ is the maximum lag considered by Kou et al. (2006) and $\mathrm{Corr}_t(\mathbf{x})$ is the normalised lag-t autocorrelation of time series $\mathbf{x}$, as read directly from Figures 3(c) and 4(a) of Kou et al. (2006).

Finally, the reciprocal of twice this number was taken, to infer the estimated proportion of effective samples produced by Kou et al. (2006).

The last mixing statistic that was considered was the acceptance rate. In each simulation there was only one acceptance rate per iteration. From each of these time series, its IAT in the second half of the run was computed using the `IAT` function from `R`'s `LaplacesDemon` package (Statisticat, 2017). Its mean and standard deviation were also computed over the same portion of the chain. The overall mean acceptance rate was again estimated by the sample mean of the acceptance rates $\hat{a}$ over the 20 simulations, giving an estimation error according to Equation (4.7),

$$\mathrm{Var}(\hat{a}) = \frac{1}{R}\sum_{r=1}^{R}\left\{\frac{1}{R-1}\left[\hat{a}^{(r)} - \hat{a}\right]^2 + \frac{4\mathrm{IAT}\left[a^{(r)}\right]}{T}\mathrm{Var}\left[\hat{a}^{(r)}\right]\right\}, \tag{4.7}$$

where $\hat{a}^{(r)}$ is its estimate for the rth simulation, $R = 20$, is the number of repeated simulations, $\mathrm{IAT}[a^{(r)}]$ is the integrated autocorrelation time of the acceptance rate time series of the rth simulation and T is the number of iterations. As in Equation (4.4), the first term inside the sum of Equation (4.7) represents the contribution of the squared deviation of acceptance rate $\hat{a}^{(r)}$ of the rth run

from the mean estimated acceptance rate $\hat{a}$, and the second term represents the variance of the mean acceptance rate of the rth run. The second term also takes into account the number of iterations used to compute the mean acceptance rate, namely $T/2$, where $T = 1000$ iterations, as well as any autocorrelation in the time series, by the factor of 2IAT.

Assessing Accuracy of Simulated Output

Before assessing the inferential power of BAIS+L, the stationary distribution of each simulation output was first checked in order to determine if it had converged to the correct form. This was achieved by first visually inspecting the scatter plots of the samples. In order to objectively quantify the number N_{correct} of simulations that had converged to stationary distributions that resembled the target, the `kmeans` function from `R` was applied to the last half of the samples of each simulation. By this time all reported simulations no longer failed the employed convergence diagnostic. When calling the function, the known means of the target's mixture components were provided as clustering centres. The number of simulations, for which the `kmeans` function assigned at least one sample to each cluster, was then counted. Variances of the counts were obtained by a normal approximation to a binomial distribution (Bertsekas and Tsitsiklis, 2002, pp. 114), according to Equation (4.8),

$$\text{Var}[N_{\text{correct}}] = 20N_{\text{correct}}(1 - N_{\text{correct}}). \tag{4.8}$$

To assess the inferential power of BAIS+L, estimates of the same statistics reported by Kou et al. (2006) were computed. These quantities were the marginal first central moments in each dimension $\mathbb{E}[X_1]$ and $\mathbb{E}[X_2]$, the marginal second central moments in each dimension $\mathbb{E}[X_1^2]$ and $\mathbb{E}[X_2^2]$, the mean exponential functions $\mathbb{E}[\exp(-10X_1)]$ and $\mathbb{E}[\exp(-10X_2)]$, and the tail probabilities p_1 and p_2. Kou et al. (2006) gave these tail probabilities according to Equations (4.9) and (4.10),

$$p_1 = \mathbb{P}[X_1 > 8.41, X_2 < 1.68, \sqrt{(X_1 - 8.41)^2 + (X_2 - 1.68)^2} > 4\sigma] \tag{4.9}$$

$$p_2 = \mathbb{P}(X_1^2 + X_2^2 > 175). \tag{4.10}$$

The preceding expectations were estimated by their respective sample averages $\overline{x_1}$, $\overline{x_2}$, $\overline{x_1^2}$, $\overline{x_2^2}$, $\overline{\exp(-10x_1)}$ and $\overline{\exp(-10x_2)}$. The estimates of the tail probabilities p_1 and p_2 were computed using the proportions of samples that satisfied the respective criteria in Equations (4.9) and (4.10). The error bars in the figures reported in Section 4.1.2 correspond to the standard deviations of these estimators. That is, their general form is given by Equation (4.11),

$$\begin{aligned}\text{Var}(\bar{y}) = \frac{1}{R}\sum_{r=1}^{R}\Bigg\{ &\frac{1}{R-1}\left[\bar{y}^{(r)} - \bar{y}\right]^2 + \frac{1}{N_{\text{eff},y}^{(r)}[N_{\text{eff},y}^{(r)} - 1]} \\ &\times \sum_{n=1}^{N}\sum_{t=T/2+1}^{T}\left[y^{(r,n,t)} - \bar{y}^{(r)}\right]^2\Bigg\}\end{aligned} \tag{4.11}$$

where R is the number of simulations that detected all of the target's mixture components, N is the number of sampling chains, T is the number of iterations,

$N_{\text{eff},y}^{(r)}$ is the effective number of samples of y in the second half of the rth simulation, the superscripts (r) and (r,n,t) are the index and the tth iterate of the nth sampling chain, respectively, of the rth simulation, and y is one of x_1, x_2, x_1^2, x_2^2, $\exp(-10x_1)$ or $\exp(-10x_2)$.

As in Equation (4.7), the first term inside the outer sum is the contribution of the deviation of the rth run's estimate of the statistic from the overall mean, while the second term is the contribution of the variance of the rth run's estimate. Just like in Equation (4.7) dependence between the samples from a single run was accounted for by rescaling each individual variance by its effective number of samples. Since the reported estimate is the mean of the sample mean, it was also divided through by this number.

p_1 and p_2 were estimated by their sample probabilities $\hat{p}_1$ and $\hat{p}_2$, respectively. The normal approximation of a binomial random variable was again used to estimate the variances of these estimators, giving an overall variance according to Equation (4.12),

$$\text{Var}(\hat{p}) = \frac{1}{R}\sum_{r=1}^{R}\left\{\frac{1}{R-1}\left[\hat{p}^{(r)} - \hat{p}\right]^2 + \frac{\hat{p}^{(r)}\left[1-\hat{p}^{(r)}\right]}{N_{\text{eff},p}^{(r)}}\right\}, \tag{4.12}$$

where the superscript (r) again represents the index of the rth simulation, $\hat{p}$ is one of $\hat{p}_1$ or $\hat{p}_2$ and $N_{\text{eff},p}^{(r)}$ is the effective number of samples used to compute $\hat{p}^{(r)}$. As in Equation (4.11), the first term inside the summation is the contribution of the squared deviation of a single estimate from the overall estimate, while the second term is once again the variance of the rth estimate. The second term reflects the use of the approximation in estimating the variance of the mean estimated tail probability from a single run.

Unlike x_1, x_2, x_1^2, x_2^2, $\exp(-10x_1)$ and $\exp(-10x_2)$, the effective numbers of samples of p_1 and p_2 were not computed following the prescription of Section 4.1.1. Instead,the effective numbers of samples of

$$\sqrt{(x_1 - 8.41)^2 + (x_2 - 1.68)^2}$$

and

$$x_1^2 + x_2^2$$

were computed, and the effective number of p_1 and p_2 determined, according to Equations (4.13) and (4.14), respectively,

$$N_{\text{eff},p_1} = \min\left[TR, N_{\text{eff},x_1}, N_{\text{eff},x_2}, N_{\text{eff},\sqrt{(x_1-8.41)^2+(x_2-1.68)^2}}\right], \tag{4.13}$$

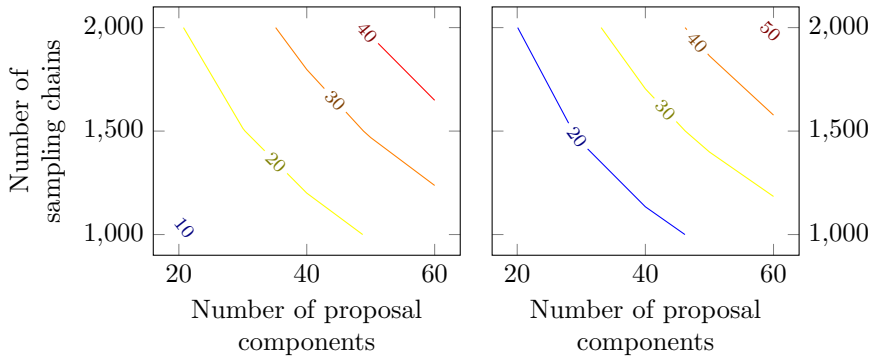
$$N_{\text{eff},p_2} = \min\left[TR, N_{\text{eff},x_1^2+x_2^2}\right]. \tag{4.14}$$

The reason for using this different approach for computing the effective numbers of samples of the tail probabilities stems from their expressions in Equations (4.9) and (4.10). That is, each equation involves tests for truth that involve combinations of the following quantities,

$$X_1, X_2, (X_1 - 8.41)^2 + (x_2 - 1.68)^2, X_1^2 + X_2^2.$$

As such, there can only be as many tests as there are samples of each quantity. To be conservative, the effective number of samplers was chosen to be the lowest

Figure 4.1: Contour plots of the run times of the mixture target with equal weights and variances (left) and unequal weights and variances (right). The numbers on the contours indicate the run time, in seconds.



effective number of any of the quantities used to infer a tail probability, as demonstrated in Equations (4.13) and (4.14). The inclusion of TR inside the minimum function indicates a conservative approach of excluding the possibility of superefficient sampling, following the practice of Gelman et al. (2004, pp. 298–299). That is, it was assumed that the effective number of samples did not exceed the true number of samples.

4.1.2 Results and Discussion

Convergence and Mixing

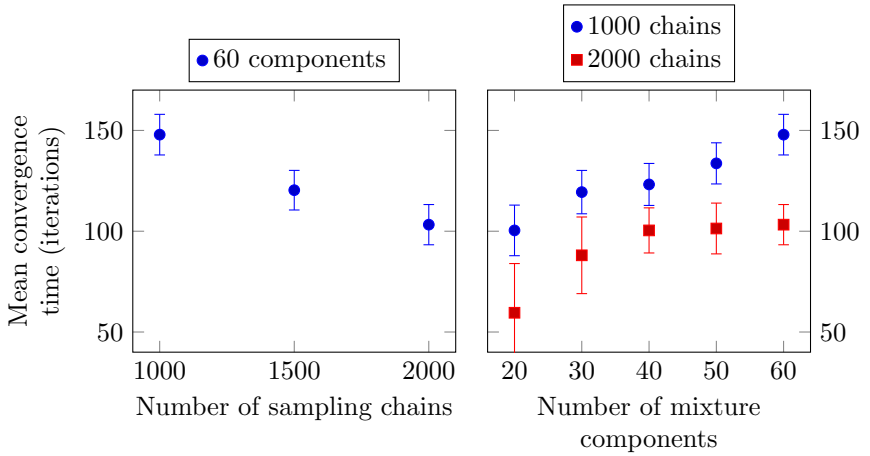
Figure 4.1 presents the mean run times, in seconds, of the equal and unequal weight and variance cases. The plot on the left represents the equal weight and variance case, while the one on the right represents the unequal weight and variance case. In both cases the standard deviation of the run times for any combination of N and K was less than 0.7.

The run time increased with both the number of mixture components in the proposal distribution and with the number of sampling chains (cf. Figure 4.2). This result is not to be confused with the convergence time (in iterations), which actually decreased with the number of sampling chains.

All test simulations of the equal weight and variance target passed the chosen convergence test within 500 iterations, with the PSRF not exceeding 1.004707 in the second half of any of the simulations. This was less than half of the time allotted to them. In the case of unequal weights and variances, however, some of the 20-component simulations did not pass the convergence test within the 1000 iterations, while all of the simulations with more components passed it within the first 500 iterations. Therefore, the mean convergence time of the 20-component simulations reported in Figure 4.3 is an underestimate of the true mean convergence time for that number of components, since only the converged simulations were used to compute it. This result indicates that it is important to overfit the proposal distribution with more components than there are clusters or components in the target.

In all reported figures of the current chapter, either the number K of com-

Figure 4.2: Mean convergence times of the mixture target simulations with equal weights and variances, according to the Gelman and Rubin convergence diagnostic with a threshold potential scale reduction factor of 1.01.



ponents in the proposal distribution or the number N of sampling chains are indicated. In some cases, more than one choice of N or K is indicated on the same plot, in order to highlight the qualitatively different observed trends in the reported statistic at different settings.

Figure 4.2 illustrates the variation of the estimated convergence time $\hat{\tau}$ with respect to the number of sampling chains for the equal weight and variance case at a PSRF threshold of 1.01. With respect to the number of sampling chains, it was observed that for any of the choices of K, the results were qualitatively the same. The left plot in Figure 4.2 illustrates this trend with the results for 60 components, where a clear downward trend is observed in the convergence time with the number of sampling chains.

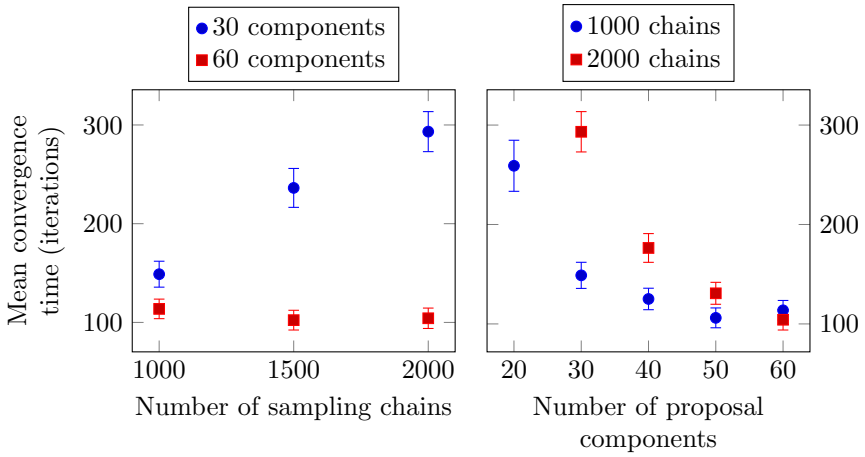
The right plot in Figure 4.2 shows that as the number of components in the proposal distribution increased so too did the convergence time. However, as N was increased the rate of increase in the convergence time with respect to K appeared to slow down, as indicated by the smaller gradient for 2000 sampling chains than with 1000.

Figure 4.3 illustrates the variation of the estimated convergence time $\hat{\tau}$ with respect to the number of sampling chains for the unequal weight and variance case at a PSRF threshold of 1.01.

Unlike the equal weight and variance case, with a smaller number of components in the proposal distribution, as the number of sampling chains is increased the convergence time increases, as indicated by the blue circles. However, with a larger number of components, there is once again a downward trend with respect to the number of sampling chains, just as in the equal weight and variance case. This result suggests that for this target, more components are required in the proposal distribution for greater sampling efficiency.

The right-hand plot of Figure 4.3 again shows a difference in behaviour from the equal weight and variance case. This time, as the number of components in the proposal distribution is increased, there appears to be a downward trend in convergence time and its rate appears to increase in magnitude with the

Figure 4.3: Mean convergence times of the mixture target simulations with unequal weights and variances, according to the Gelman and Rubin convergence diagnostic with a threshold potential scale reduction factor of 1.01. Note that the mean convergence time of the 20-component simulations is an underestimate of its true value, since not all 20-component simulations with 1500 or 2000 sampling chains converged.



number of sampling chains. Interestingly, at 60 components, the 2000 sampling chain simulations appear to converge faster than the 1000 sampling chain ones, suggesting a possible crossover point at this setting, although, this conjecture needs to be tested further with larger numbers of components in the proposal distribution. If such a crossover point does exist then it may be possible that the trends evident in Figure 4.2 may emerge for larger K.

One possible reason for faster convergence, when there is a sufficient number of proposal distribution components, is that the larger number of sampling chains provides more information and, hence, greater precision in inferred statistics. This greater precision, in turn, leads to lower sampling variability. However, as evidenced by the unequal weight and variance results, this extra precision has the opposite effect when the number of proposal distribution components is not sufficient. It may also have an unfavourable impact on parameter inferences, a matter that will be discussed in Section 4.1.1. However, the increasing trend suggests that the larger number of sampling chains also results in greater accuracy and, therefore, faster convergence of the proposal distribution to its limiting case.

Note that in the unequal weight and variance case, each target component has a greater variance, making the regions between them more dense than in the equal weight and variance case. This has the effect of making the task of finding the target's components less difficult, as will be seen in Section 4.1.2. However, while the components of the equal weight and variance case are similar in weight and variance, in the unequal weight and variance case they are quite different. It appeared that, for the same number of mixture components in the proposal distribution, the unequal weight and variance case took longer to converge than the equal weight and variance one. A possible cause for this difference is that the smaller variance of individual target components in the equal

Figure 4.4: Mean acceptance rate of the mixture target simulations, with equal weights and variances, for 60 components in the proposal distribution (left) and 2000 sampling chains (right).

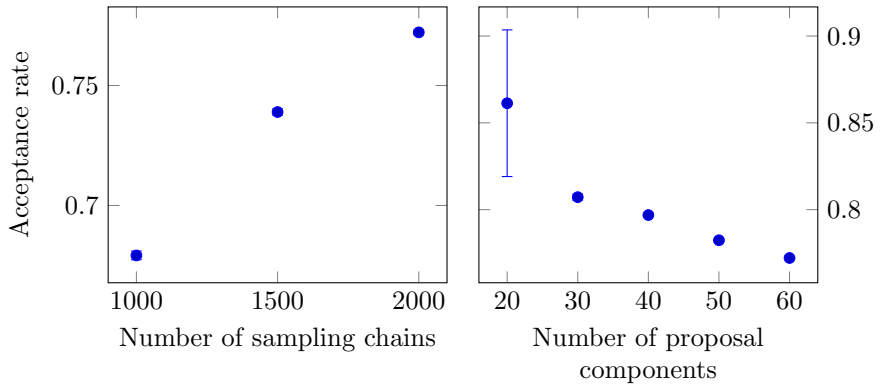
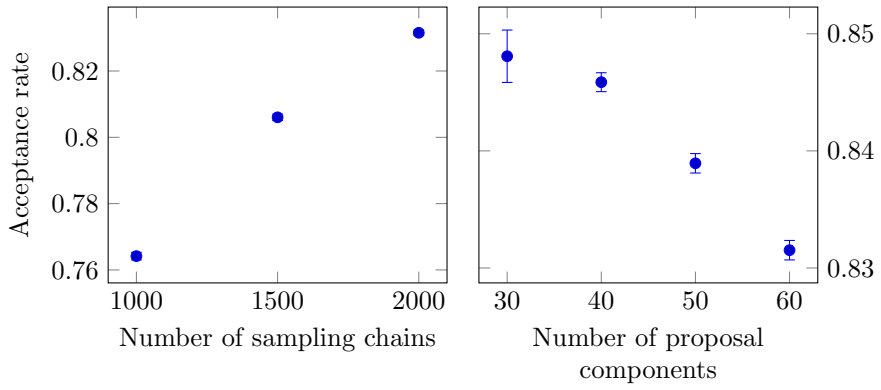


Figure 4.5: Mean acceptance rate of the mixture target simulations, with unequal weights and variances, for 60 components in the proposal distribution (left) and 2000 sampling chains (right).



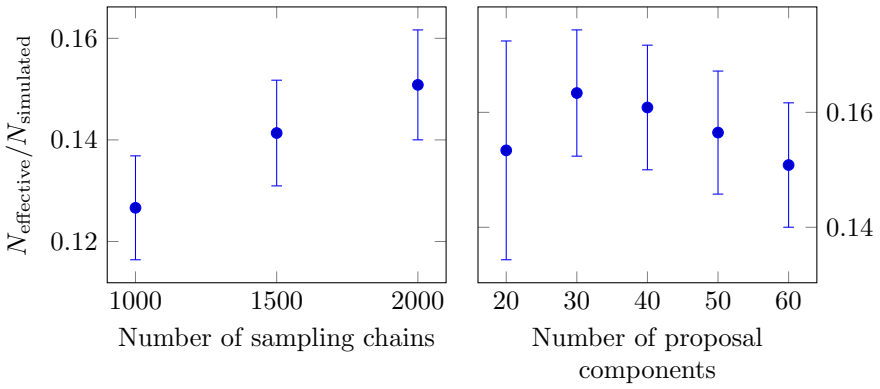
weight and variance case makes them more clearly-defined, which increases the sampling precision of a given component's parameter inferences. Furthermore, the assumption of equal weights and variances in the choice of hyperparameters is better suited to the equal weight and variance target.

Figures 4.4 and 4.5 illustrate the acceptance rates for the equal weight and variance and the unequal weight and variance cases, respectively.

Both cases appear to suggest an increase in acceptance rate with the number of sampling chains. This result supports the conjecture that the approximation of the proposal distribution to the target improves as more chains are incorporated into the proposal distribution because a closer approximation results in a higher acceptance ratio.

As the number of mixture components in the proposal distribution is increased, an opposite trend is observed, with a decreasing acceptance rate. One possible reason for this result is that there are fewer chains on average used to infer new parameters of each proposal component, thereby leading to greater

Figure 4.6: Effective proportion of samples from the mixture target simulations, with equal weights and variances, for 60 components in the proposal distribution (left) and 2000 sampling chains (right).



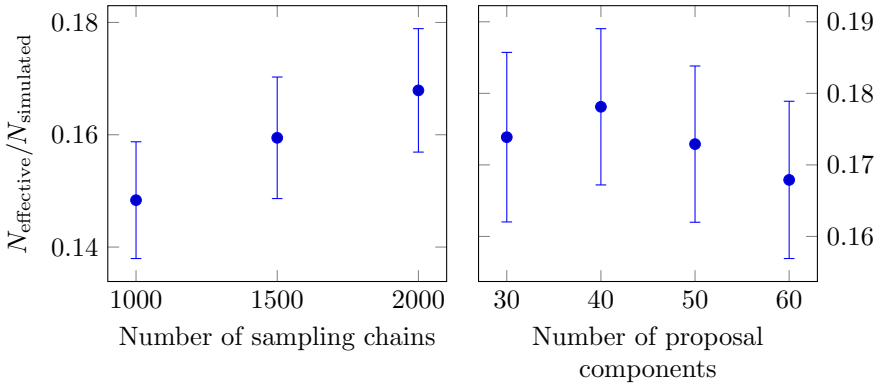
uncertainties in the parameters and a greater chance for the approximation to be sub-optimal.

Figures 4.6 and 4.7 illustrate the minimum ratio of the effective number of samples to the simulated number for the equal weight and variance and the unequal weight and variance cases, respectively. The trends are the same as those observed for the acceptance rates but less pronounced, with greater uncertainty in individual measurements and a more gradual gradient. This result is to be expected, since an improvement in the approximation between proposal and target leads to the proposed states more closely resembling i.i.d. samples from the target. As such, between-sample correlations are reduced, increasing the effective number of samples. With respect to the number of sampling chains, the increasing trend is more obvious but with respect to the number of proposal components, it is only noticeable in the unequal weight and variance case. Nevertheless, the results of both the acceptance rate and effective proportion of samples highlight that there is a possible efficiency penalty when using more mixture components, even before considering that more computer time is inherently required to update more components.

In comparison to the effective proportion of samples estimated for EES from Figures 3(c) and 4(a) of Kou et al. (2006), all effective proportions of samples resulting from the use of BAIS+L were higher, with the proportions from Kou et al. (2006) being smaller than the vertical axis minima reported here. Specifically, using the approach outlined in Section 4.1.1, the effective proportions of samples generated by Kou et al. (2006), using the EES, were inferred to be in the ranges 0.02–0.03 and 0.04–0.05 for the equal weight and variance and the unequal weight and variance cases, respectively. It should be noted that using the minimum integrated autocorrelation time over each dimension of the target tended to result in a much lower inference of the effective proportion of samples than that inferred using `R`'s `effectiveSize` function. This possibly explains why the effective proportion achieved using BAIS+L was only around one third, even though the acceptance rate was consistently much higher.

The comparatively higher proportion of effective samples of BAIS+L is to be expected, given the within-iteration independence between sampling chains.

Figure 4.7: Effective proportion of samples from the mixture target simulations, with unequal weights and variances, for 60 components in the proposal distribution (left) and 2000 sampling chains (right).



Contrast this property of BAIS+L with that of EES, which uses all samples collected post burn-in throughout a simulation. It should still be noted, however, that the effective proportion of samples is still only about one third. This means that the samples overall are still not truly independent, indicating that there is still temporal dependence between iterations, resulting in significant autocorrelation. The effect of this autocorrelation leads to the sampling chains still being coupled with one another, thereby increasing their overall correlation.

The closer that the steady state approximation of the proposal distribution is to the stationary distribution of the Markov chain induced by the sampler, the higher the acceptance rate will be. Future research will consider this matter theoretically, including answering the question of whether a unique limiting proposal distribution exists and, if so, how to guarantee and accelerate convergence towards it.

Accuracy of Simulated Output

When reviewing the empirical distributions of the generated samples, it was observed that not all simulations inferred stationary distributions with the correct shapes, with some components of the target missing. Since these particular simulations did not detect all components, they had clearly not converged, despite them not failing the convergence test. This result demonstrates why relying only on a single convergence diagnostic (Gelman and Rubin's in this case) is not enough to satisfy the MCMC practitioner that a BAIS+L simulation has converged, at least for small numbers of components in the proposal distribution (cf. Table 4.1). Of course, in the current study, it was possible to compare against the true form of the target, which effectively constituted a second diagnostic.

Table 4.1 counts the number of simulations for each parameter setting that found all target components in the equal weight and variance case. It is seen that as the number of components in the proposal distribution is increased, the chances of BAIS+L detecting all target components increases, suggesting that having sufficiently more proposal components than there are components

Table 4.1: The number of simulations that passed the stationary distribution check for each combination of the number of sampling chains N and number of components K in the proposal distribution. The table on the left represents the equal weight and variance simulations and the one on the right, the unequal weight and variance simulations. Note that some 20-component simulations did not converge in the allotted time, as indicated by an asterisk (*) next to the count.

		N		
		1000	**1500**	**2000**
K	**20**	3	0	0
	30	18	19	9
	40	20	20	20
	50	20	20	20
	60	20	20	20

		N		
		1000	**1500**	**2000**
K	**20**	20	17*	16*
	30	20	20	20
	40	20	20	20
	50	20	20	20
	60	20	20	20

or clusters in the target distribution prevents an excessive number of them from being prematurely emptied. This result highlights the importance of having a sampler that can fit a mixture proposal, such as BAIS+L, as it allows the MCMC practitioner to deliberately overfit the proposal model, making it easier to find all clusters or components of the target and, hence, all local maxima.

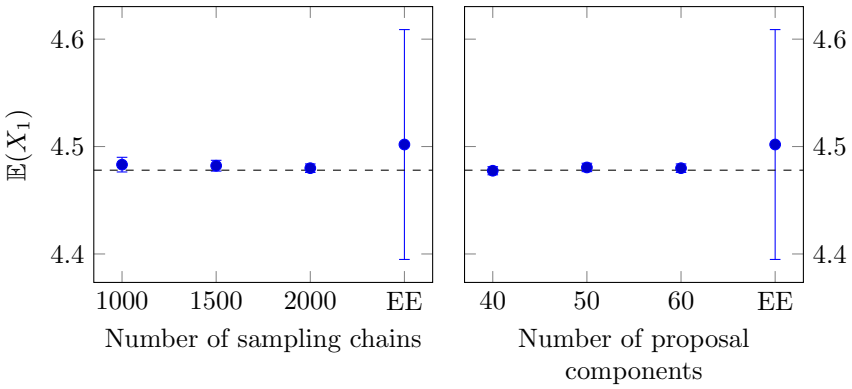
Table 4.1 also suggests that with an insufficient number of components in the proposal distribution, increasing the number of sampling chains produces the opposite effect, reducing the chances of finding all clusters or local maxima of the target.

Thus, the first preliminary guideline suggested by these results, is to run pilot simulations with as large a number of proposal components as possible.

Due to the inherent increase in the computational burden with an increased number of components, this approach is only recommended for short pilot runs. To determine the number of components required for longer simulations, the MCMC practitioner should assess a histogram of the component counts (cf. Section 4.2) at select iterations to determine if it converges to one on a smaller number of components. If the histogram does not clearly demonstrate that some components of the proposal distribution are receiving much less weight than others, then it may be necessary to increase the number of components used in further pilot runs. Additionally, the time series of the acceptance rate must also have converged before making any inference. This approach is demonstrated in the application to mixture exponential regression in Section 4.2.

Provided that a sufficiently high steady-state acceptance rate is achieved in a pilot run and the histogram of component counts appears to have converged, the MCMC practitioner may apply information criteria, such as those discussed in Section 1.5.3, to the output of pilot runs. Alternatively, they may simply count the number of components from the stationary histogram of component proportions. If the resulting number of components suggested by the criteria are much less than the number used in a pilot run then it may be possible to reduce the number in subsequent runs. Of course, how much the number may be reduced is still a question that requires further study, as Table 4.1 shows that too much of a reduction introduces the risk of the sampler exhibiting the quasi-ergodic problem, due to insufficient overfitting.

Figure 4.8: Inferred value of the first moment of the equal weight and variance target in the first dimension, for 60 components in the proposal distribution (left) and 2000 sampling chains (right). In each plot, the dotted line represents the true value and the far-right indicates the value inferred by Kou et al. (2006) using EES. Similar results were obtained in the second dimension and for the second moments in each dimension.



Once the simulations that detected all components of the target had been identified, the same statistics as reported by Kou et al. (2006) were computed. All figures derived from the equal weight and variance mixture target present the results for 60 proposal components on the left and for 2000 sampling chains on the right. These two settings were chosen for exposition because they represent the largest number of proposal components and the largest number of sampling chains, respectively, considered in the current study. Furthermore, these choices follow the preliminary guidance obtained from the findings in Table 4.1 and from Conjecture 1 of Chapter 3. Since not all 20- and 30-component simulations detected all components of the target, inferences made from them were not considered.

The statistic presented in Figure 4.8 is the first central moment in the first dimension X_1. Its results are encouraging. Despite the fact that BAIS+L uses an approximate acceptance ratio, and hence generates samples from an approximation to the true target, the inferred first moment is comparable to that inferred by Kou et al. (2006) using EES. The relatively small uncertainties of BAIS+L compared to those of EES are attributed to the substantially larger number of samples generated using it, due to the larger numbers of sampling chains and iterations considered in the current study.

$\mathbb{E}(X_2)$, $\mathbb{E}(X_1^2)$ and $\mathbb{E}(X_2^2)$ produced qualitatively similar results (not shown).

The results for $\mathbb{E}\left[\exp(-10X_1)\right]$ (Figure 4.9) and $\mathbb{E}\left[\exp(-10X_2)\right]$ (Figure 4.10) also highlight the success of BAIS+L at sampling the target, provided it has converged to the true stationary distribution. In the first dimension, however, BAIS+L consistently underestimated the true values, possibly illustrating the approximate nature of the stationary distribution. Finally, Figures 4.11 and 4.12 present the tail probabilities p_1 and p_2.

It is observed, once more, that the estimates of the tail probabilities p_1 and p_2 are comparable to those inferred by Kou et al. (2006) using EES. However, unlike the first two moments and the exponential quantities, the uncertainties of

Figure 4.9: Inferred values of $\mathbb{E}\left[\exp\left(-10X_1\right)\right]$ of the equal weight and variance target, for 60 components in the proposal distribution (left) and 2000 sampling chains (right). In each plot, the dotted line represents the true value and the far-right indicates the value inferred by Kou et al. (2006) using EES.

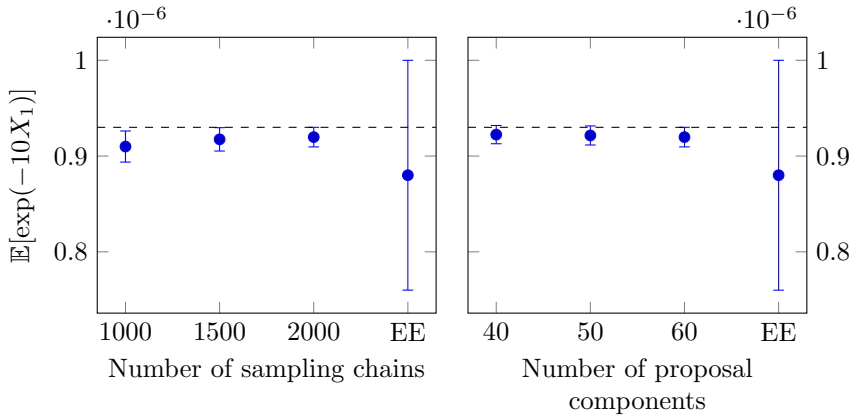


Figure 4.10: Inferred values of $\mathbb{E}\left[\exp\left(-10X_2\right)\right]$ of the equal weight and variance target, for 60 components in the proposal distribution (left) and 2000 sampling chains (right). In each plot, the dotted line represents the true value and the far-right indicates the value inferred by Kou et al. (2006) using EES.

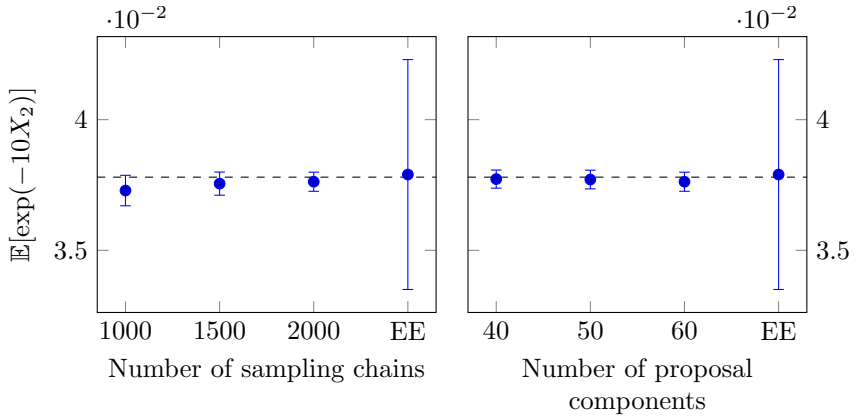


Figure 4.11: Inferred value of p_1 of the equal weight and variance target, for 60 components in the proposal distribution (left) and 2000 sampling chains (right). In each plot, the dotted line represents the true value and the far-right indicates the value inferred by Kou et al. (2006) using EES.

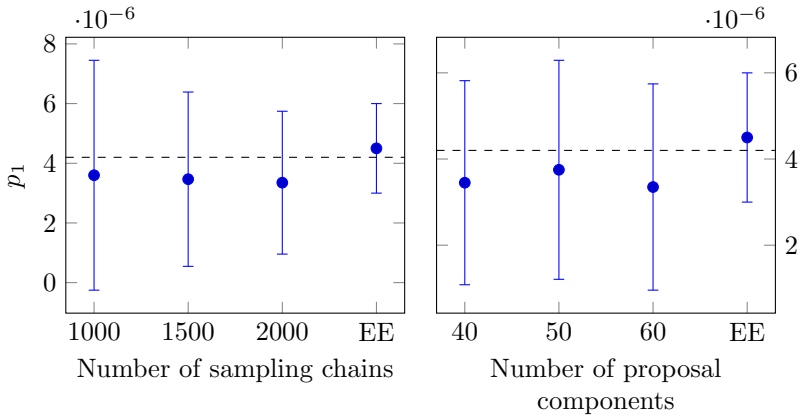


Figure 4.12: Inferred value of p_2 of the equal weight and variance target, for 60 components in the proposal distribution (left) and 2000 sampling chains (right). In each plot, the dotted line represents the true value and the far-right indicates the value inferred by Kou et al. (2006) using EES.

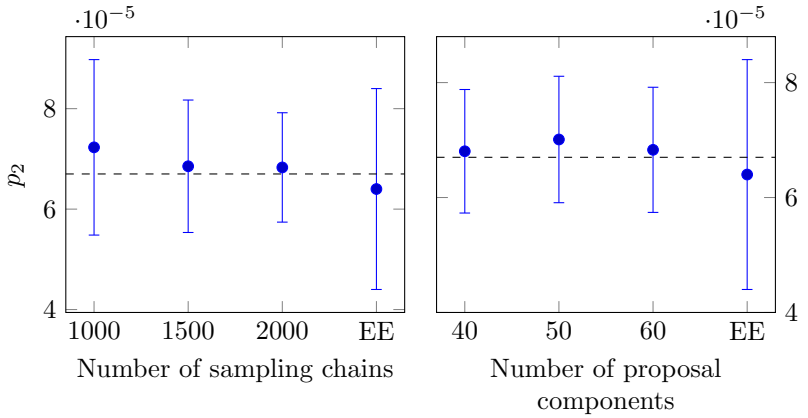
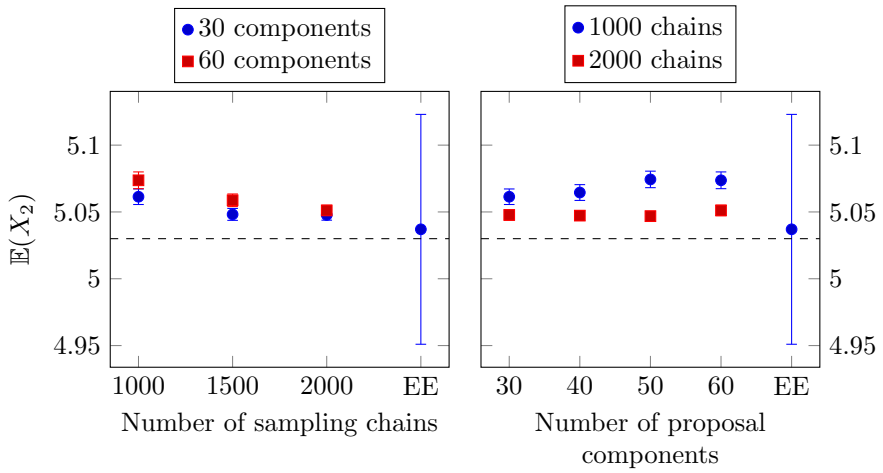


Figure 4.13: Inferred value of the first moment of the unequal weight and variance target in the second dimension. In each plot, the dotted line represents the true value and the far-right indicates the value inferred by Kou et al. (2006) using EES.



the estimated tail probabilities are larger than those of the estimates produced by Kou et al. (2006). This lower precision is in spite of the fact that the current study involved longer runs, suggesting that the approximation that results from using BAIS+L suffers most in the tails of the target distribution, at least for the mixture target with equal weights and variances. This poor performance in the tails of the target may be attributed to the use of a common prior on each of the components. It also highlights a limitation of the normal mixture proposal, which implicitly assumes that the target density has a tail decay that is at least as fast the tail decay of the component used to propose a new state. The relationship between the tails of the target and the components of the proposal distribution is an important one. Therefore, future work should consider it more closely.

The estimated first and second moments inferred using BAIS+L in the unequal weight and variance case were also of the same order of magnitude as those inferred by Kou et al. (2006) using EES, although not as accurate.

An important difference from the equal weight and variance results is the demonstration of the improvement of BAIS+L's approximation as the number of sampling chains is increased. This improvement is evident in Figure 4.13, where, for either 30 or 60 mixture components in the proposal distribution the estimates of the first moment in the second dimension approaches the true value from above. This result supports Conjecture 1, further motivating a rigorous theoretical treatment of its claim that in the limit of an infinite number of sampling chains the stationary distribution sampled by BAIS+L is exact.

The first moment in the first dimension, as well as the second moments in either dimension, (all not shown) were qualitatively the same as those for the equal weight and variance case.

The preceding results demonstrate that the approximate inferences made from a BAIS+L simulation can differ somewhat from true values but, at least

for the mixture target studied here, this deviation from the truth is at least partially offset by the efficiency of the sampler, as reported in Section 4.1.2. They also demonstrate that the approximate sampling nature becomes less relevant as the number of sampling chains is increased, provided there are sufficiently many components in the proposal distribution to guarantee that all components, clusters or local maxima of the target are found.

Guidance

From the preceding results it is suggested that N and K *initially* be chosen to be as large as practically possible, given the computing resources available. These large choices are not suitable for long runs and should only be used for selection of the parameters of the prior distribution. That is, for selecting $\alpha_k^{(0)}$, $\nu_k^{(0)}$, $\kappa_k^{(0)}$, $\Sigma_k^{(0)}$ and $\mu_k^{(0)}$ in the case of the mixture target.

In particular, it is suggested that the MCMC practitioner simulate short pilot runs with the large choices of N and K and by adjusting the hyperparameters $\kappa_k^{(0)}$ and $\mathbf{\Sigma}_k^{(0)}$ of each component k, so as to achieve a sufficiently high acceptance rate. Such adjustments may be based on any prior information available regarding the overall variance of the target or, failing this, running a few pilot runs with different combinations of $\kappa_k^{(0)}$ and $\mathbf{\Sigma}_k^{(0)}$.

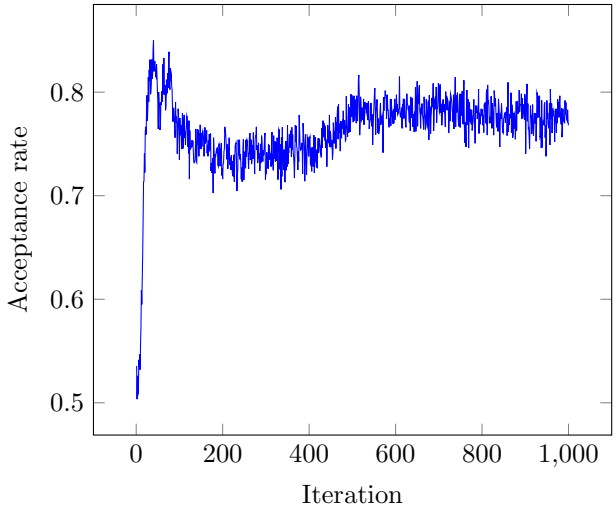
Since the ratio of $\kappa_k^{(0)}$ to $\mathbf{\Sigma}_k^{(0)}$ approximates the prior ratio of the variance of a region about a single local maximum, to the overall variance of the target distribution, one may start by fixing $\mathbf{\Sigma}_k^{(0)}$ to be reasonably disperse and then reducing $\kappa_k^{(0)}$ until the acceptance rate time series produced ends in a plateau greater than the minimum observed acceptance rate (see Figure 4.14, which illustrates the acceptance rate of one of the 60-component 2000-sampling chain simulations).

To adjust $\mathbf{\Sigma}_k^{(0)}$, one may use the height of the plateau as a guide. A suggestion is to progressively decrease $\mathbf{\Sigma}_k^{(0)}$ from a suitably large initial value until the height of the plateau in the acceptance rate does not increase any further. At the same time $\kappa_k^{(0)}$ must be adjusted in order to maintain the overall prior variance at a constant level. The reasoning behind this suggestion is that, once the prior on the mean considers a large-enough portion of the parameter space to cover all local maxima, reducing the variance of a single component should theoretically reduce the proposal rate of low-weight samples under the target. The example studied in Section 4.2 considers this approach.

As suggested in Section 4.1.2, it is also possible to visually inspect histograms of the latent allocations of all samples from a simulation resulting in a plateau and counting the largest number of significant components. As a general guideline, any parameter setting that *consistently* results in the maximal number of significant components should be satisfactory for long runs. Although to make such a determination will require repeated simulations, as seen in Section 4.1.2. This should not be a problem if the number of components used is significantly larger than the number suggested as significant by the histogram.

Of course, this experimental approach is not ideal, as it can potentially involve a substantial amount of preliminary work. One possible pathway to reduce this extra effort is to use an automated scheme that takes into account the acceptance rate as a statistic for updating the hyperparameters in an adaptive

Figure 4.14: An example of a plateau in the acceptance rate time series of a BAIS+L simulation from the mixture target of Liang and Wong (2001) with 60 components in the proposal distribution and 2000 sampling chains. Observe that the acceptance rate quickly reached a plateau of a high value. While the acceptance rate appeared to have increased later in the simulation, the consistently high acceptance rate over a long time period indicated a suitable choice of simulation parameters.



prior distribution. It will be interesting to consider such an approach in future work.

Unfortunately, automating the choice of the number of proposal components is not as clear. As observed, overfitting the proposal distribution appeared to result in a greater chance of finding all components of the target and, hence, all local maxima. However, increasing the number of components resulted in a lower acceptance rate. A larger number of components also requires more computational effort, as there are inherently more parameters that need to be updated. Therefore, it is recommended that the number of components be chosen larger than the prior believed number of clusters or local maxima in the target, as indicated by the preliminary runs, but not prohibitively so, given computing resources available to the MCMC practitioner. A precise method for determining a sufficient number is a matter that is left for future investigations.

Alternatively, one could incorporate the number of mixture components into the vector of parameters to be inferred and use a reversible jump (Green, 1995) or generalized Gibbs sampler (Keith et al., 2004) approach to update them. Care must be taken, however, to avoid underestimating the number of components, given the problems associated with convergence of the 20- and 30-component simulations identified in Table 4.1. This is also an avenue for future work, which would make BAIS+L more automatic.

4.2 An Application to Mixture Exponential Regression

In this section, BAIS+L is applied to the mixture exponential regression problem studied by Kou et al. (2006) using the guidance developed in its application to the mixture target of Liang and Wong (2001). The section begins with a restatement of the regression model before proceeding to a discussion of some pilot runs, which are used to determine the numbers of sampling chains and mixture components, as well as the prior parameters of the proposal distribution. The sampler is then implemented on the target using the results of the pilot runs.

4.2.1 The Problem

Recall the mixture exponential regression problem (cf. Section 2.2) studied by Kou et al. (2006) (cf. Section 1.4.3), where one has a collection of M values $\mathbf{y} = (y_m)_{m=1}^M$ sampled according to Equation (4.15).

$$y_m \sim \begin{cases} \text{Exp}\left[\exp\left(\beta_{11} + \beta_{12} x_m\right)\right] & \text{with probability } \alpha \\ \text{Exp}\left[\exp\left(\beta_{21} + \beta_{22} x_m\right)\right] & \text{with probability } 1 - \alpha. \end{cases}, \tag{4.15}$$

where $\text{Exp}(\lambda)$ represents an exponential distribution with parameter λ.

As in Kou et al. (2006), the goal was to infer the model parameters α, β_{11}, β_{12}, β_{21} and β_{22}, given known values of $(y_m)_{m=1}^M$ and $(x_m)_{m=1}^M$.

In order to do so, the procedure of Kou et al. (2006) was followed, by first generating 400 variable pairs (y_m, x_m). That is, α was set to 0.3, β_1 to $(1, 2)^{\text{T}}$, β_2 to $(4, 5)^{\text{T}}$ and each of the 400 x_m values was sampled from a uniform distribution on the interval $(0, 2)$,

$$x_{m2} \sim \mathcal{U}(0, 2).$$

With these parameters, $\mathbf{y} = (y_1, \ldots, y_{400})$ were sampled from the same distribution as in Kou et al. (2006), which is restated in Equation (4.16),

$$y_m \sim \exp\left(\beta_{\delta_m}^{\text{T}} x_m\right). \tag{4.16}$$

Here δ_m is latent variable with distribution

$$\delta_m - 1 \sim \text{Bernoulli}\left(\alpha\right).$$

Each δ_m was sampled before the corresponding y_m, giving the raw data that were used to perform the regression. These raw data are provided in Appendix B.

Unlike Kou et al. (2006), 400 (y_m, x_m) pairs were generated, as multiple independent sets of only 200 pairs resulted in inconsistent estimates of α, β_{11}, β_{12}, β_{21} and β_{22}. With 400 pairs this problem was not observed.

4.2.2 Methodology

Following the guidance that was suggested in Section 4.1.2, pilot runs were initially simulated, using proposal distributions with a large value of K.

Since any possible distribution of the mixing parameter α in the regression problem is supported on the bounded interval $[0, 1]$, while the other regression parameters are not, a normal mixture proposal over all five regression parameters is not a suitable choice. However, for convenience and to investigate its performance in such a situation, the study was continued with this proposal distribution.

In order to determine the hyperparameters $\kappa_k^{(0)} = \kappa^{(0)}$ and $\mathbf{\Sigma}_k^{(0)} = \Sigma^{(0)}\mathbf{I}_2$ (where $\mathbf{I}_2$ is the two-dimensional identity matrix), 1000 sampling chains were simulated, with 10 components in the proposal distribution, making sure to also record the latent allocations for the determination of the number of components in the target.

As with the mixture example, the prior proposal means were set to $\boldsymbol{\mu}_k^{(0)} = (0, 0)^{\mathrm{T}}$ for convenience and $\nu_k^{(0)} = p + 1 = 6$ for each component k. The same value of $d_{\min} = 0.1$ used for the mixture target was also implemented for this regression problem, by following the prescription of Section 3.3.3 with $I_d = 10$.

Five short 200 iteration simulations were run, each with 10 mixture components in the proposal and $\Sigma^{(0)}$ fixed at 1, while $\kappa^{(0)}$ was varied in $\{1, 0.1, 0.01, 0.001, 0.0001\}$ for each component k. In these short runs it was found that the acceptance rate time series of the smallest four of these choices of $\kappa^{(0)}$ exhibited the plateau that was discussed in Section 4.1.2. Therefore, ratios of $\kappa^{(0)}/\Sigma^{(0)}$ in $\{0.1, 0.01, 0.001, 0.0001\}$ and values of $\Sigma^{(0)}$ in $\{10, 1, 0.1, 0.01, 0.001\}$ were investigated, in order to determine a suitable $\Sigma^{(0)}$. Each simulation was again run with 200 iterations, 1000 sampling chains and 10 components.

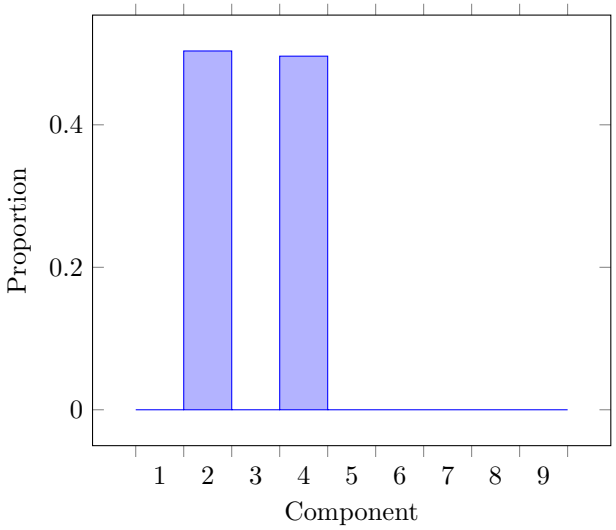
The resulting 20 short runs produced plateaus in the acceptance rate that were consistently around 0.75 for $\Sigma^{(0)} \in \{0.1, 0.01, 0.001\}$ and $\kappa^{(0)}/\Sigma^{(0)} \in \{0.01, 0.001, 0.001\}$, so the largest of each of these was selected. That is, $\Sigma^{(0)}$ was set to 0.1 and $\kappa^{(0)} = 0.001$ in the longer simulations. In addition, the number of proposal components, from which samples were generated, always quickly reduced to two or three, indicating that 10 mixture components in the proposal distribution were sufficient. Figure 4.15 illustrates the steady component proportions after 75% of the short simulation with $\Sigma^{(0)} = 0.1$ and $\kappa^{(0)} = 0.001$ was complete. It demonstrates that two of the ten components retained all weight, indicating that no more than ten components were required in the proposal distribution.

The choice of the number of iterations was made by visually inspecting the time series of the corresponding acceptance rate, which appeared to have converged before 100 iterations. The burn-in period was chosen to be more than twice this number, selecting a value of 500 iterations, with a further 500 iterations for inference.

Thus, to investigate the robustness of the chosen hyperparameters, 20 independent simulations from the regression target were run, each with 1000 sampling chains, 1000 iterations and 10 mixture components in the proposal distribution.

To make sure that convergence had been achieved within this time, the maximum PSRF of the five regression parameters in the last half of the run was computed using the `gelman.diag` function from `R`'s `coda` library, with the settings `confidence = 0.95`, `transform = FALSE`, `autoburnin = FALSE` and `multivariate = FALSE`.

Figure 4.15: The proportion of samples originating from each component of the proposal distribution in the last 25% of the short $\Sigma^{(0)} = 0.1$, $\kappa^{(0)} = 0.001$ run.



From the 20 simulations, the number that produced a maximal number of local maxima was counted by a visual inspection of the resulting marginal posterior densities of α, β_{11}, β_{12}, β_{21} and β_{22}. The mean of the minimum effective number of samples was computed as the sample mean over the 20 simulations of the reciprocal of twice the maximum IAT observed in the last 500 iterations over all dimensions and sampling chains in each simulation, as reported by the `IAT` function from R's `LaplacesDemon` package (Statisticat, 2017). The results report the mean values of all simulations whose stationary distributions exhibited the maximum number of local maxima. The variance of this effective proportion was again estimated by assuming that the effective number of samples followed a binomial distribution.

To estimate the effective proportion of samples produced by Kou et al. (2006) using the equi-energy sampler, Figure 7(d) of Kou et al. (2006) was loaded into the `WebPlotDigitizer` of Rohatgi (2011) to extract their autocorrelations at the reported lags. The lags were then combined with Equation (4.6), just as was done for the mixture target, and the reciprocal of the result was taken, to give the effective proportion of samples.

4.2.3 Results and Discussion

By observing the 20 full length runs, it was observed that all of them produced two local maxima in each marginal stationary distribution, in agreement with the results of Kou et al. (2006). The maximum PSRF encountered in the last 500 iterations of these 20 simulations was no more than 1.002694, which was deemed to be low enough to assume that they had all converged. They also produced a mean minimum ratio of the effective number of samples to the true number of samples of around 0.171 with a standard deviation of approximately 0.012. This proportion was greater than the approximately 0.03 inferred from Figure 7(d) of Kou et al. (2006) for EES.

Figure 4.16 reports the posterior densities of α, β_{11} and β_{12} from one of the successful runs, along with vertical lines, representing the true values. The density estimates for β_{21} or β_{22} are not shown here, as they were very similar to β_{11} and β_{12}, respectively, as expected and as noted by Kou et al. (2006).

Qualitatively, the marginal distributions of the runs performed in the current study runs were the same as those inferred by Kou et al. (2006) but with the local maxima slightly shifted.

In the case of α, its marginal posterior distribution either overestimates (lower α local maximum) or underestimates (upper α local maximum) the mixture parameter.

Similarly, the local maxima of the marginal posterior distribution of β_{11} and β_{12} are not centred on the true values. These discrepancies may either be due to the approximate nature of the BAIS+L acceptance ratio or the simulated the 400 input data pairs, which were different from those used by Kou et al. (2006).

These results indicate that the particular choice of parameter settings used to implement the sampler on the input data can approximately find the correct stationary distribution with greater efficiency than the equi-energy sampler. However, as with the mixture target, there was some waste in the need to use extra components in the proposal distribution to ensure that both local maxima of the posterior distribution were found.

4.3 Conclusion

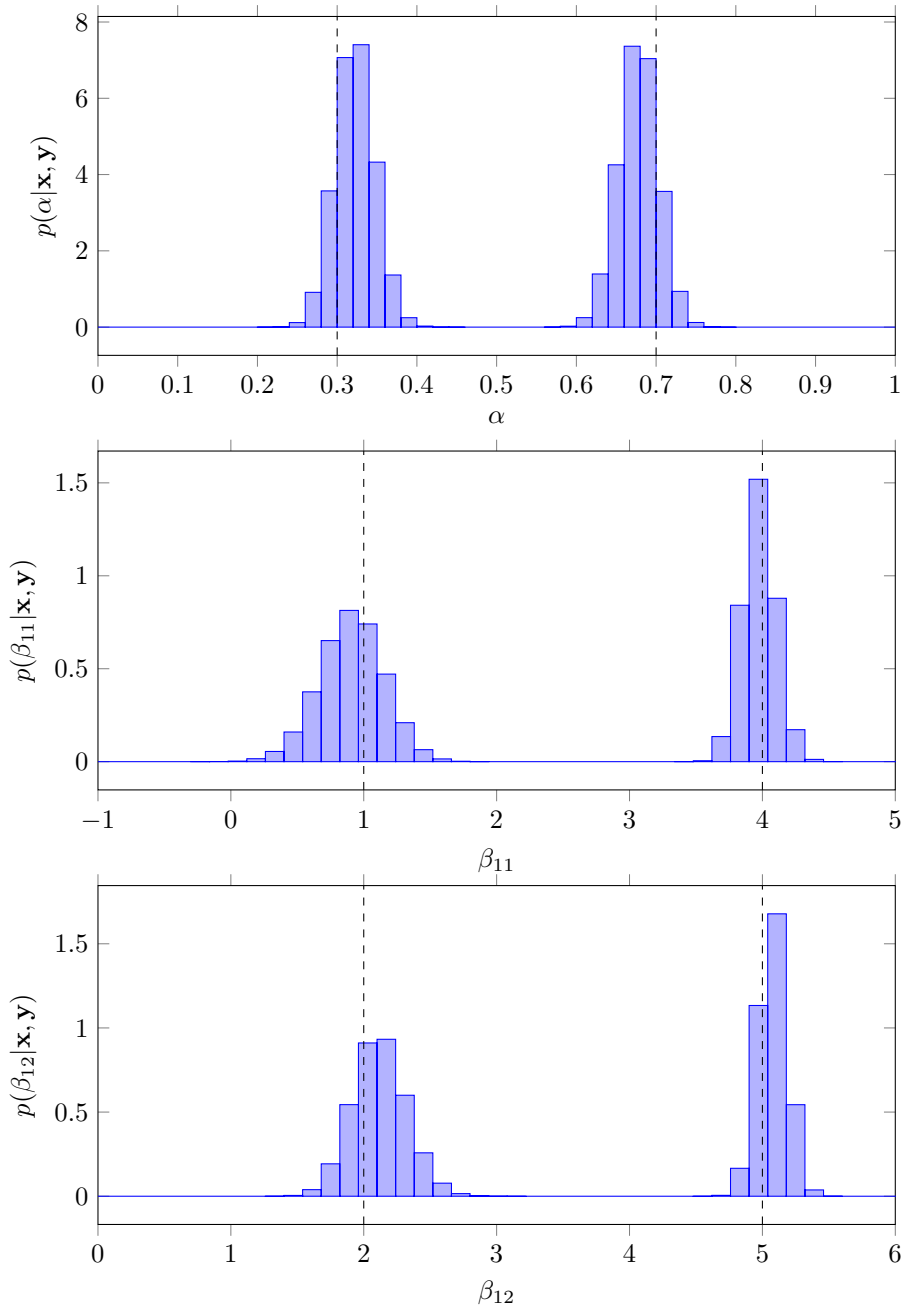
The preceding examples have shown that the normal mixture proposal version of BAIS+L can efficiently sample from difficult target distributions supported on $\mathbb{R}^p$, with the efficiency increasing as the number of sampling chains is increased. However, as seen in the application to the mixture target, there is still a risk of the sampler becoming stuck and not sampling all components or clusters of the target and, therefore, missing local maxima.

The results of simulating from the mixture target of Kou et al. (2006) demonstrated that increasing the number of mixture components in the proposal distribution can increase the chances of finding all clusters, components or local maxima in the target distribution, although this change comes at the expense of extra computational cost to update the extra components. It also leads to a lower acceptance rate, as evidenced by the results of the unequal weight and variance case.

The results also demonstrated that, for the simulation settings studied, the inferred statistics of the mixture target were comparable to those inferred by Kou et al. (2006) using EES. The results of the first moment in the unequal weight and variance case also suggest that the stationary distribution sampled by BAIS+L improves with the number of sampling chains, in support of Conjecture 1. Of course, such a claim needs to be proven rigorously, which is anticipated in future work.

In the current work, the parameters of the prior distribution on the proposal parameters were chosen on the basis of sampling efficiency. Future studies will investigate their choice further, with a focus not only on sampling efficiency but their effect on the resulting stationary distribution of the samples. In particular, given the fact that most of the 20-component mixture target runs did not detect all target components, the issue of premature emptying of mixture components

Figure 4.16: Histograms of the marginal posterior densities of α (top), β_{11} (middle) and β_{12} (bottom) of the mixture exponential regression problem as inferred using BAIS+L. The dashed vertical lines represent the true values of the parameters, taking into account the non-identifiability of the problem.



needs to be investigated in future work. One possibility is to enforce a minimum number of sampling chains to be allocated to each component.

The efficiency of BAIS+L was also promising when it detected all components of the target. However, the results, having effective proportions of samples of less than 1, indicated that there was considerable autocorrelation between samples at different iterations. This is another point that will be interesting to investigate in future work, in hopes of further increasing the efficiency of BAIS+L.

As noted in the results of Section 4.1.2, the future work should also consider what effect the chosen kernel in the mixture proposal has on BAIS+L's performance in the tails of a target distribution.

Finally, while the study of this chapter has provided some guidance on the selection of hyperparameters when the number and location of local maxima, or the scales of their corresponding components or clusters, are unknown, it should be stressed that it is not necessarily optimal. Indeed, it does not even guarantee that all local maxima will be found. The detection of local maxima is an important and non-trivial task, so it is recommended that future studies also consider systematic approaches to hyperparameter selection.

As stated in Section 4.1.2, one possibility is that of adaptively updating the prior information using suitable performance criteria and proposal distributions of previous iterations of the sampler. With this approach, the effects of an incorrect prior distribution could be mitigated, leading to a better approximation of the target and better mixing. Also suggested was the inclusion of the number of components in the parameter update procedure, which, when combined with the aforementioned adaptive updating of prior information, will automate the implementation of BAIS+L.

In conclusion, while BAIS+L is not without its limitations, it appears to have a competitive performance to EES, justifying further exploration of its properties and potential applications.

Chapter 5

Tailoring BAIS+L to Spin Glass Simulation

As discussed in Section 2.3, spin glasses are an interesting problem in condensed matter physics due to their complex energy distributions, which are challenging to sample from using standard MC or MCMC approaches. This chapter demonstrates a possible avenue for incorporating BAIS+L into spin glass simulations and highlights some hurdles that still need to be overcome.

The first portion of this chapter is devoted to describing two approaches to using BAIS+L as the MH component in a spin glass sampler, such as PT or PA. It starts by restating the target distribution for an Ising spin glass simulation and the choices of BAIS+L proposal distributions. It then outlines the two potential proposal mechanisms for new configurations, before describing prior knowledge of their parameters and identifying their posterior distributions.

The second portion of this chapter is devoted to a numerical study of one of these proposal distributions, to assess its viability and to compare it to to the more traditional single-spin-flip MH approach discussed in Section 2.3.4. The opportunity is taken to study the effectiveness of using BAIS+L to sample from some two-dimensional lattices and its relative efficiency is compared to single-spin-flip Metropolis.

The results discussed in this chapter illustrate some of the pitfalls in implementing BAIS+L, for which possible remedies are suggested for study in future work.

5.1 Motivation and Goals

As discussed in Section 1.4.3, a common tool for studying spin glasses is PT with replica exchange. Section 1.4.3 also discussed that Wang et al. (2015) demonstrated the comparable performance of PA for the study of spin glasses, proposing it as a competitive alternative to PT. At the time of this writing, neither method has been able to be used to definitively determine the low-temperature behaviour of the Edwards-Anderson spin glass with Ising spins. Both PT and PA also require some assumptions to be made before simulation, such as a cooling schedule and the interval between attempted replica exchanges

(cf. Section 1.4.3). Neither of these quantities can be chosen easily *a priori* but both affect the performance of their respective algorithms.

To implement either approach, an MH algorithm may be used to propose new configurations. As discussed in Chapter 2, two alternative ways to use MH are to successively propose to flip single spins or clusters of spins before accepting or rejecting the resulting proposed configurations. It is these two approaches that this chapter seeks to extend using BAIS+L.

While more components give greater flexibility to the proposal distribution, the results of Chapter 4 showed that the number of sampling chains (referred to as "replicas" in condensed matter physics simulations) must be considered, as it must be sufficiently larger than the number of proposal components for efficiency of sampling, both in terms of convergence rate and in terms of mixing rate. Additionally, the more sampling chains that are used in a BAIS+L simulation, the better the approximation of the resulting stationary distribution to the target, given a sufficient number of proposal distribution components to ensure that all local maxima of the target distribution are detected.

5.2 Sampling Approaches

This section identifies and discusses two proposal distributions for implementing BAIS+L as the MH method for simulating an Ising spin glass and it discusses the associated posterior distributions of the proposal parameters, given samples of spin configurations.

5.2.1 Multiple Spin Updating with BAIS+L

The first approach considered involves a modification of the single-spin-flip Metropolis method (cf. Section 2.3.4). BAIS+L offers two avenues for modifying this procedure, which may be applied together or individually.

The first and most obvious of these avenues uses BAIS+L to adapt the spin-up probability at each lattice site. When only one spin is updated at a time, there are only two possible results after the update: the current configuration; or the current configuration with the spin in question flipped. While single-spin-flip always uses the second of these configurations as the proposed configuration, BAIS+L assigns a probability to each configuration under the proposal distribution, with the probabilities not necessarily being equal. It is this probability that is updated using BAIS+L's posterior update mechanism (see below).

The second avenue offered by BAIS+L extends single-spin-flip to *multiple*-spin-flip. This approach requires an appropriate proposal distribution to investigate the Ising spin glass using BAIS+L. An obvious choice, which is adopted herein, is a mixture of independent binary choices of an up spin at each of the lattice sites, where the probability of this choice depends on each site of each component. For each sampling chain $n \in \{1, \ldots, N\}$, a component z_n is first proposed from a categorical distribution,

$$z_n | \, \mathbf{d} \sim \text{Categorical}(\mathbf{d}),$$

where $\mathbf{d} = (d_1, \ldots, d_K)$ is the vector of mixture weights.

A proposed spin state is then drawn independently for each vertex $i \in \{1, \ldots, L^p\}$ of the p-dimensional lattice of side length L, given z_n, according

to Equation (5.1),

$$\frac{1}{2}\left[s_n^{(i)}+1\right]\Big| z_n, q_{z_n}^{(i)} \sim \text{Bernoulli}\left[q_{z_n}^{(i)}\right], \tag{5.1}$$

where $q_k^{(i)}$ is the probability of an up spin at lattice site i under component k of the proposal.

The overall proposal distribution is, thus, given by Equation (5.2),

$$p(\mathbf{s}_n|\mathbf{d},\{q_k\}_{k=1}^K) = \sum_{k=1}^{K} d_k \prod_{i=1}^{L^p} \left[q_k^{(i)}\right]^{\left[s_n^{(i)}+1\right]/2} \left[1-q_k^{(i)}\right]^{1-\left[s_n^{(i)}+1\right]/2}. \tag{5.2}$$

It is clear that if the proposal distribution has as many components as there are possible configurations in the state space ($K = 2^{L^p}$) and if all spins are updated at once, then there exists a choice of proposal distribution that is equal to the target. This is true because each component can correspond to exactly one configuration by being a vector of only 1 and 0 probabilities of up spins, thereby making exactly one configuration possible under each component. The mixture proportions $\mathbf{d}$ will then (approximately) correspond to the probabilities of the configurations under the true target.

Of course, such a scheme is infeasible in practice, due to the sheer size of the configuration space, so in practice the number of components in the proposal distribution will always be less than the number of possible configurations of the spin glass.

An alternative to updating *all* spins at once is to update just m of them (but still more than just the one offered by single-spin-flip). This approach reduces the support of the proposal distribution to 2^m different configurations. Unlike the Swendsen-Wang Algorithm, the form of multiple-spin-flip outlined above does not look for clusters of satisfied interactions. To apply such an approach will require a component membership for each block of simultaneously-updated spins for each replica, since an accept/reject step will be attempted for each block being updated. This approach, however, would introduce a random walk aspect to sampling, which is not in the spirit of the current work.

While it has been chosen to construct the proposal distribution of a multiple spin updating approach in such a manner, no claim is made that it is the optimum prescription. It is, however, conceptually easy to understand and implement, making it a natural candidate for investigation.

Prior Distribution of Proposal Parameters

In order to update the spin up probabilities at each iteration, an independent Beta(α_k, β_k) prior is assumed on each of them in each component k. The prior shape parameters α_k and β_k of the distribution are unknown, so, in the current study, they were both assumed to be equal to 1 for each component k. This represents a uniform distribution (Gelman et al., 2004, pp. 581), reflecting the lack of prior knowledge about the spin up probabilities, as indicated by Equation (5.3) for the kth component,

$$q_k^{(i)} \sim \text{Beta}(1,1) = \mathcal{U}(0,1). \tag{5.3}$$

In the case of the mixture weights $\mathbf{d}$, the same prior distribution that was used in the normal mixture proposal case was also adopted. That is, they were assumed to have a Dirichlet distribution with common prior concentration of 1 on each component, as given by Equation (5.4),

$$\mathbf{d} \sim \text{Dirichlet}(\mathbf{1}_K), \tag{5.4}$$

where $\mathbf{1}_K$ is the K-dimensional vector of 1s.

Posterior Distribution of Proposal Parameters

As in the normal mixture proposal case, the posterior distribution of the mixture weights is again the Dirichlet distribution of Equation (5.5),

$$\mathbf{d}|\mathbf{s}, \mathbf{z} = \mathbf{d}|\mathbf{z} \sim \text{Dirichlet}(\mathbf{1}_K + \mathbf{o}), \tag{5.5}$$

where $\mathbf{o} = (o_1, \ldots, o_K)$ is the vector of component counts.

To obtain the posterior distribution of the spin-up probabilities, apply Bayes' Theorem to obtain

$$p\left[q_k^{(i)} \middle| \mathbf{s}, \mathbf{z}\right] \propto p\left[q_k^{(i)}\right] \prod_{n \in \mathcal{I}_k} p\left[s_n^{(i)} \middle| z_n = k, q_k^{(i)}\right]$$

where $\mathcal{I}_k \subseteq \{1, \ldots, N\}$ is the set of all sampling chain/replica indices for which $z_n = k$. This further simplifies as follows.

$$\begin{aligned}
p\left[q_k^{(i)} \middle| \mathbf{s}, \mathbf{z}\right] &= \text{Beta}\left[q_k^{(i)} \middle| \alpha_k, \beta_k\right] \prod_{n \in \mathcal{I}_k} \text{Bernoulli}\left\{\frac{1}{2}\left[s_n^{(i)} + 1\right] \middle| q_k^{(i)}\right\} \\
&\propto \left[q_k^{(i)}\right]^{\alpha_k - 1} \left[1 - q_k^{(i)}\right]^{\beta_k - 1} \prod_{n \in \mathcal{I}_k} \left[q_k^{(i)}\right]^{\frac{1}{2}\left[s_n^{(i)}+1\right]} \left[1 - q_k^{(i)}\right]^{1 - \frac{1}{2}\left[s_n^{(i)}+1\right]} \\
&= \left[q_k^{(i)}\right]^{\alpha_k - 1} \left[1 - q_k^{(i)}\right]^{\beta_k - 1} \prod_{n \in \mathcal{I}_k} \left[q_k^{(i)}\right]^{\frac{1}{2}\left[s_n^{(i)}+1\right]} \left[1 - q_k^{(i)}\right]^{\frac{1}{2}\left[1 - s_n^{(i)}\right]} \\
&= \left[q_k^{(i)}\right]^{\alpha_k - 1} \left[1 - q_k^{(i)}\right]^{\beta_k - 1} \left[q_k^{(i)}\right]^{\sum_{n \in \mathcal{I}_k}\left[s_n^{(i)}+1\right]/2} \\
&\quad \times \left[1 - q_k^{(i)}\right]^{\sum_{n \in \mathcal{I}_k}\left[1 - s_n^{(i)}\right]/2} \\
&= \left[q_k^{(i)}\right]^{\alpha_k - 1} \left[1 - q_k^{(i)}\right]^{\beta_k - 1} \left[q_k^{(i)}\right]^{\frac{o_k}{2}\left[\bar{s}_k^{(i)}+1\right]} \\
&\quad \times \left[1 - q_k^{(i)}\right]^{\frac{o_k}{2}\left[1 - \bar{s}_k^{(i)}\right]} \\
&= \left[q_k^{(i)}\right]^{\alpha_k - 1 + \frac{o_k}{2}\left[\bar{s}_k^{(i)}+1\right]} \left[1 - q_k^{(i)}\right]^{\beta_k - 1 + \frac{o_k}{2}\left[1 - \bar{s}_k^{(i)}\right]} \\
&\propto \text{Beta}\left\{q_k^{(i)} \middle| \alpha_k + \frac{o_k}{2}\left[\bar{s}_k^{(i)} + 1\right], \beta_k + \frac{o_k}{2}\left[1 - \bar{s}_k^{(i)}\right]\right\},
\end{aligned}$$

where $\bar{s}_k^{(i)}$ is the average spin at the ith lattice site in component k.

This gives

$$q_k^{(i)} \middle| \mathbf{s}, \mathbf{z} \sim \text{Beta}\left\{\alpha_k + \frac{o_k}{2}\left[\bar{s}_k^{(i)} + 1\right], \beta_k + \frac{o_k}{2}\left[1 - \bar{s}_k^{(i)}\right]\right\}.$$

5.2.2 Cluster Updating with BAIS+L

Another approach that could provide an independence-sampling alternative to the Swendsen-Wang algorithm, is to use an adaptive proposal that allows multiple clusters of spins to be updated simultaneously.

The idea introduced here is to propose new configurations from a histogram constructed in a different manner from Section 5.2.1. This histogram is built up using a clustering approach, and places most mass on a subset of $M \leq 2^{L^p}$ configurations of the configuration space. The parameter vector of this histogram consists of bonds $\mathbf{B}_k \in \{0,1\}^{pL^p}$, where $k \in \{1,\ldots,K\}$ and K is the fixed number of BAIS+L proposal distribution components, L is the lattice side length and p is the dimension of the lattice. Each $\mathbf{B}_k$ partitions the canonical lattice into C_k uniquely-determined clusters, each of whose spin preferences are perfectly satisfied by one of exactly two possible cluster configurations. These cluster configurations $\mathbf{U}_k^{(c)}$ and $\mathbf{V}_k^{(c)}$ $(c \in \{1,\ldots,C_k\})$ differ by total spin reversal and have probability of proposal $p_k^{(c)}$ and $q_k^{(c)}$, respectively, given their component k has been selected. Let $D_k^{(c)}$ be the dimension (i.e. the number of lattice sites) in cluster c of component k. Then the cluster has in total $2^{D_k^{(c)}}$ possible configurations, exactly two of which perfectly satisfy the interactions. The probability of any other frustrated cluster configuration is given a nominal probability $r_k^{(c)}$, such that $p_k^{(c)} + q_k^{(c)} + \left[2^{D_k^{(c)}} - 2\right] r_k = 1$. This ensures that the probability of a transition to any configuration is bounded below by a strictly positive value at each iteration.

Let $\psi_k^{(c)}(\mathbf{s})$ be the natural projection of configuration $\mathbf{s}$ onto its subvector containing only those spins in cluster c of component k. Then the probability of $\psi_k^{(c)}(\mathbf{s})$, under component k of the proposal distribution, is given by Equation (5.6),

$$\begin{aligned}
\mathbb{P}\left[\psi_k^{(c)}(\mathbf{s}) \middle| \mathbf{B}_k\right] &= \begin{cases} p_k^{(c)} & \text{if } \psi_k^{(c)}(\mathbf{s}) = \mathbf{U}_k^{(c)}, \\ q_k^{(c)} & \text{if } \psi_k^{(c)}(\mathbf{s}) = \mathbf{V}_k^{(c)}, \\ r_k^{(c)} & \text{otherwise.} \end{cases} \\
&= \left[p_k^{(c)}\right]^{I_{\mathbf{U}_k^{(c)}}\left[\psi_k^{(c)}(\mathbf{s})\right]} \left[q_k^{(c)}\right]^{I_{\mathbf{V}_k^{(c)}}\left[\psi_k^{(c)}(\mathbf{s})\right]} \\
&\quad \times \left[r_k^{(c)}\right]^{1 - I_{\mathbf{U}_k^{(c)}}\left[\psi_k^{(c)}(\mathbf{s})\right] - I_{\mathbf{U}_k^{(c)}}\left[\psi_k^{(c)}(\mathbf{s})\right]},
\end{aligned} \tag{5.6}$$

where $I_A(B) = 1$ if $A = B$ and 0 otherwise.

Therefore, the proposal distribution is given by Equation (5.7),

$$\mathbb{P}\left[\mathbf{s} \middle| \left\{d_k, \mathbf{B}_k, \left\{p_k^{(c)}, q_k^{(c)}, r_k^{(c)}\right\}_{c=1}^{C_k}\right\}_{k=1}^{K}\right] = \sum_{k=1}^{K} d_k \prod_{c=1}^{C_k} \mathbb{P}\left[\psi_k^{(c)}(\mathbf{s}) \middle| \mathbf{B}_k\right]. \tag{5.7}$$

Just as with the normal mixture proposal of Chapter 3, updating of a configuration begins with proposing a component k of the proposal distribution to use for sampling. The proposal distribution then proposes which of the $2^{D_k{}^{(c)}}$ configurations each cluster takes, independently of all other clusters.

When no bonds are added, which is entirely possible under the preceding scheme, the method resembles the multi-spin updating of Section 5.2.1.

With such a proposal mechanism in place, the question arises of how to specify the bonds $\mathbf{B}_k$ and the cluster state probabilities $p_k^{(c)}$, $q_k^{(c)}$ and $r_k^{(c)}$. Their derivation is discussed in the following two subsections.

Prior Distribution of Proposal Parameters

Before deriving the posterior distribution of the proposal parameters the prior knowledge of their distributions must first be outlined. To do so, first factor the prior distribution according to the dependence between the variables.

As in the normal mixture proposal case studied up to this point, the vector of mixture proportions $\mathbf{d} = (d_1, \ldots, d_K)$ is assumed to be *a priori* independent of all other proposal parameters. Since they have no dependence on any of the other parameters, they are assumed to have the same prior form as in the normal mixture proposal cases studied earlier. That is, they have a Dirichlet prior distribution, with equal concentration of α on each component.

$$p_{\mathbf{J}}\left(\{d_k, \mathbf{B}_k\}_{k=1}^K\right) \propto \text{Dirichlet}\left(\{d_k\}_{k=1}^K \middle| \left\{\alpha_k^{(0)}\right\}_{k=1}^K\right).$$

To keep things simple, consider the vector of bonds to be built up by a simple raster scan, which randomly place bonds between neighbouring spins. Starting at the lowest index spin, this approach iterates through each spin and each of its neighbouring lattice sites of higher index value, first determining if including a bond will introduce frustration to the spin's cluster or not. The probability of there being a bond $\mathbf{B}_k$ at index i in component k, given the addition of a new bond does not introduce frustration, is $b_k^{(i)}$. Otherwise, the probability of such a bond is zero, as given by Equation (5.8),

$$\begin{aligned}\mathbb{P}\left(\mathbf{B}_k\right) &= \prod_{i=1}^{pL^p} \mathbb{P}\left(B_k^{(i)} \middle| \left\{B_k^{(j)}\right\}_{j=1}^{i-1}\right) \\ &= \prod_{i=1}^{pL^p} \left[b_k^{(i)}\right]^{g\left[B_k^{(i)} \middle| \left\{B_k^{(j)}\right\}_{j=1}^{i-1}\right]} 0^{1-g\left[B_k^{(i)} \middle| \left\{B_k^{(j)}\right\}_{j=1}^{i-1}\right]},\end{aligned} \tag{5.8}$$

where $g\left[B_k^{(i)} \middle| \left\{B_k^{(j)}\right\}_{j=1}^{i-1}\right]$ is 1 if the presence of a bond at edge i of the lattice introduces frustration and 0 otherwise. Here the convention $0^0 = 1$ is adopted. Each $b_k^{(i)}$ is considered *a priori* i.i.d. Beta$(1, 1)$. After each spin pair has been considered, the result is a collection of $C \leq L^p$ clusters of spins, completely determined by the bonds.

The probability vector $\left[p_k^{(c)}, q_k^{(c)}, r_k^{(c)}\right]$ is assumed *a priori* Dirichlet $\left[\delta_k^{(c)}, \eta_k^{(c)}, \zeta_k^{(c)}\right]$.

Posterior Distribution of Proposal Parameters

Since the weights are *a priori* the same as in the normal mixture and multi-spin cases, and they are independent of the other proposal parameters, their joint posterior density is once again a Dirichlet distribution,

$$\mathbf{d} \left| \{z_n, \mathbf{S}_n\}_{n=1}^N \right. \sim \text{Dirichlet}\left[\mathbf{o} + \alpha^{(0)}\right],$$

where $\mathbf{o} = (o_1, \ldots, o_K)$ is the vector of observed counts of each component of the proposal distribution and $z_n \in \{1, \ldots, K\}$ is the component membership of replica n.

The method to determine bonds $\mathbf{B}_k$ and cluster configuration probabilities $p_k^{(c)}$, $q_k^{(c)}$ and $r_k^{(c)}$ is more complex.

$$\begin{aligned}
&\mathbb{P}\left(\mathbf{B}_k, \mathbf{p}_k, \mathbf{q}_k, \mathbf{r}_k \,\middle|\, \{z_n, \mathbf{S}_n\}_{n=1}^N\right) \\
&\quad \propto \mathbb{P}\left(\{\mathbf{S}_n\}_{n=1}^N \middle|\, \{z_n\}_{n=1}^N \mathbf{B}_k, \mathbf{p}_k, \mathbf{q}_k, \mathbf{r}_k\right) \mathbb{P}(\mathbf{p}_k, \mathbf{q}_k, \mathbf{r}_k | \mathbf{B}_k) \mathbb{P}(\mathbf{B}_k) \\
&\quad \propto \prod_{n \in \mathcal{I}_k} \prod_{c=1}^{C_k} \mathbb{P}_k^{(c)}(\mathbf{S}_n) \left[p_k^{(c)}\right]^{\delta_k^{(c)}-1} \left[q_k^{(c)}\right]^{\eta_k^{(c)}-1} \left[r_k^{(c)}\right]^{\xi_k^{(c)}-1} \\
&\qquad \times \prod_{i=1}^{pL^p} \mathbb{P}\left[B_k^{(i)} \,\middle|\, \left\{B_k^{(j)}\right\}_{j=1}^{i-1}\right] \\
&\quad = \prod_{n \in \mathcal{I}_k} \prod_{c=1}^{C_k} \left[p_k^{(c)}\right]^{\delta_k^{(c)}-1} \left[q_k^{(c)}\right]^{\eta_k^{(c)}-1} \left[r_k^{(c)}\right]^{\xi_k^{(c)}-1} \\
&\qquad \times \left[p_k^{(c)}\right]^{I_{\mathbf{U}_k^{(c)}}\left[\psi_k^{(c)}(\mathbf{S}_n)\right]} \left[q_k^{(c)}\right]^{I_{\mathbf{V}_k^{(c)}}\left[\psi_k^{(c)}(\mathbf{S}_n)\right]} \\
&\qquad \times \left[r_k^{(c)}\right]^{1-I_{\mathbf{U}_k^{(c)}}\left[\psi_k^{(c)}(\mathbf{S}_n)\right]-I_{\mathbf{U}_k^{(c)}}\left[\psi_k^{(c)}(\mathbf{S}_n)\right]} \\
&\qquad \times \prod_{i=1}^{pL^p} \left[b_k^{(i)}\right]^{g\left[B_k^{(i)} \middle| \left\{B_k^{(j)}\right\}_{j=1}^{i-1}\right]} 0^{1-g\left[B_k^{(i)} \middle| \left\{B_k^{(j)}\right\}_{j=1}^{i-1}\right]}.
\end{aligned}$$

Let $N_k^{(c)}(\mathbf{U})$ be the number of occurrences of $\mathbf{U}_k^{(c)}$, let $N_k^{(c)}(\mathbf{V})$ be the number of occurrences of $\mathbf{V}_k^{(c)}$ and let $N_k^{(c)}(\emptyset)$ be $o_k - N_k^{(c)}(\mathbf{U}) - N_k^{(c)}(\mathbf{V})$. By this preceding design, each cluster will have the property that all of the bonds between its member spins can be simultaneously satisfied by one of exactly two possible states $\mathbf{U}_c$ and $\mathbf{V}_c$, differing by total spin reversal. Then the posterior distribution is given as Equation (5.9),

$$\begin{aligned}
&\mathbb{P}\left(\mathbf{B}_k, \mathbf{p}_k, \mathbf{q}_k, \mathbf{r}_k \,\middle|\, \{z_n, \mathbf{S}_n\}_{n=1}^N\right) \\
&\quad = \prod_{n \in \mathcal{I}_k} \prod_{c=1}^{C_k} \left[p_k^{(c)}\right]^{\delta_k^{(c)}+N_k^{(c)}(\mathbf{U})-1} \left[q_k^{(c)}\right]^{\eta_k^{(c)}+N_k^{(c)}(\mathbf{V})-1} \left[r_k^{(c)}\right]^{\xi_k^{(c)}+N_k^{(c)}(\emptyset)-1} \\
&\qquad \times \prod_{i=1}^{pL^p} \left[b_k^{(i)}\right]^{g\left[B_k^{(i)} \middle| \left\{B_k^{(j)}\right\}_{j=1}^{i-1}\right]} 0^{1-g\left[B_k^{(i)} \middle| \left\{B_k^{(j)}\right\}_{j=1}^{i-1}\right]}.
\end{aligned} \tag{5.9}$$

Therefore, $\left[p_k^{(c)}, q_k^{(c)}, r_k^{(c)}\right]$ is Dirichlet with parameter,

$$\left[\delta_k^{(c)} + N_k^{(c)}(\mathbf{U}), \eta_k^{(c)} + N_k^{(c)}(\mathbf{V}), \xi_k^{(c)} + N_k^{(c)}(\emptyset)\right].$$

However, the question still remains of how to update the bonds $B_k^{(i)}$ using information obtained from the replica configurations $\{\mathbf{S}_n\}_{n=1}^N$. Tailoring g to determine frustration based on a set of random couplings $\mathcal{J}$ that is dual to the fixed couplings $\mathbf{J}$ of the disorder sample is a possibility. Given the already much

more complicated forms of the equations for this approach, however, exploration of the form of g and its consequences on the posterior distribution of the bonds is left for future work.

5.3 Investigating Multiple Spin Updating with BAIS+L

This section implements the design for the Ising spin glass described in Section 5.2.1. The results provided in the current section initiate the study of the viability of BAIS+L as a new tool for studying spin glasses, by considering its efficiency and the quality of the simulated targets, compared to traditional single-spin-flip dynamics. More importantly, it addresses the shortcomings of the approach taken to implementing BAIS+L thus far, while identifying aspects of the sampler that require further attention.

As this is an initial study of a new method for investigating a very difficult problem, no attempt is made in the current work to answer any questions regarding the physics of spin glasses.

5.3.1 Aims

The primary goal of the current chapter is to qualitatively explore the efficiency and effectiveness of BAIS+L when applied to a single disorder sample, without the assistance of a metaheuristic, such at PT. It is intended that the results of the current chapter identify problem areas with the approach outlined in Section 5.2.1, in the hopes of guiding future modification of the method.

To explore the efficiency of the sampler, the shape of the acceptance rate time series was considered. As highlighted in Chapter 4, a successful BAIS+L simulation results in a high-value plateau in the acceptance rate time series. By varying the number of proposal components, the acceptance rate time series were monitored, to see if such a plateau was formed.

Since BAIS+L is an approximate technique, the stationary distribution will be an approximation of the true target. Therefore, empirical histograms of the configurations were also considered for a comparison to their known true values.

Finally, the results from the BAIS+L simulations were compared to those of a corresponding single-spin flip MCMC approach.

Given the results of Chapter 4, which suggested that the limiting distribution of BAIS+L approaches the target distribution as $N \to \infty$, the effect of N on sampler performance was not studied. Instead, N was kept fixed at a value of 1000.

5.3.2 Methodology

To understand the potential role of BAIS+L in spin glass simulation, five independent disorder samples (cf. Section 2.3) were simulated on a 3×3 lattice (which has $2^{3\times 3} = 512$ possible configurations) at an inverse temperature of $\beta = 1.8$ for $T = 1000$ iterations with $N = 1000$ replicas. For each simulation BAIS+L was run with $K = 20, 40, 60, 80$ and 100 components in the proposal distribution.

The single-spin-flip Metropolis method reviewed in Section 2.3.4 was also applied to the same set of disorder samples, for comparison, also with $N = 1000$ simultaneous replicas. Note that in this case, the replicas were completely independent of each other. For both samplers, the proportion of accepted moves at each iteration was recorded.

In the case of single-spin-flip Metropolis, one iteration was considered to be one complete sweep through all lattice sites, with the acceptance rate taken to be the average number of accepted moves over all sites in all replicas.

Since a spin glass has support on a discrete set of configurations, not $\mathbb{R}^p$, it was not deemed appropriate to use Gelman and Rubin's diagnostic to assess convergence of their simulations. Hence, to satisfy that convergence had been achieved, the acceptance rate time series was inspected visually. For a given simulation, once the time series appeared approximately constant, convergence was assumed.

The run times of the simulations using the two samplers were also recorded. As in the simulations of Chapter 4, this was achieved using the built-in `time` command from version 4.3 of `bash` (The Free Software Foundation, 2014), with a simulation's total run time given by the sum of the `user` and `sys` times reported by it.

5.3.3 Results and Discussion

The first results considered were the stationary distributions of the configurations inferred by both BAIS+L and single-spin-flip using the samples in the last half of their runs. Figure 5.1 demonstrates the inferred distributions for one of the disorder samples, where top plot illustrates the histogram inferred using single-spin-flip, the middle one represents the histogram inferred using BAIS+L and the bottom plot illustrates the true histogram.

Observe that both samplers successfully identified the most significant configurations under this choice of couplings, with their histograms having similar shapes to the true one. This result demonstrates that both samplers have similar inferential power, at least for the small disorder samples studied.

The acceptance rate, however, highlights the difference in performance of the two approaches. Visual inspection of its time series for each simulation performed with BAIS+L illustrated that none of them exhibited the plateau described in Chapter 4. Instead, all demonstrated a downward trend towards a minimum acceptance rate, similar to the traditional single-spin-flip simulations. Figure 5.2 shows one example of the observations for the same disorder sample. The left plot represents traditional single-spin-flip sampling, while the right represents BAIS+L.

Notice that both samplers appear to have a very low acceptance rate, with the rate for traditional single-spin-flip being higher than that for BAIS+L.

Such poor performance of BAIS+L is disappointing but not surprising, since the support of its proposal distribution is much larger than that of single-spin-flip (512 versus 1 in the example studied), leading to an increased chance of proposing insignificant configurations. Furthermore, while there are multiple chances for a configuration to change at each iteration in a single-spin-flip sampler, there is only one in BAIS+L with the proposal distribution used in the current study. This results in slower mixing through the configuration space with BAIS+L.

Figure 5.1: The histograms over indexed configurations from one of the disorder samples. The top plot illustrates the histogram inferred using single-spin-flip, the middle one represents the histogram inferred using BAIS+L and the bottom plot illustrates the true histogram.

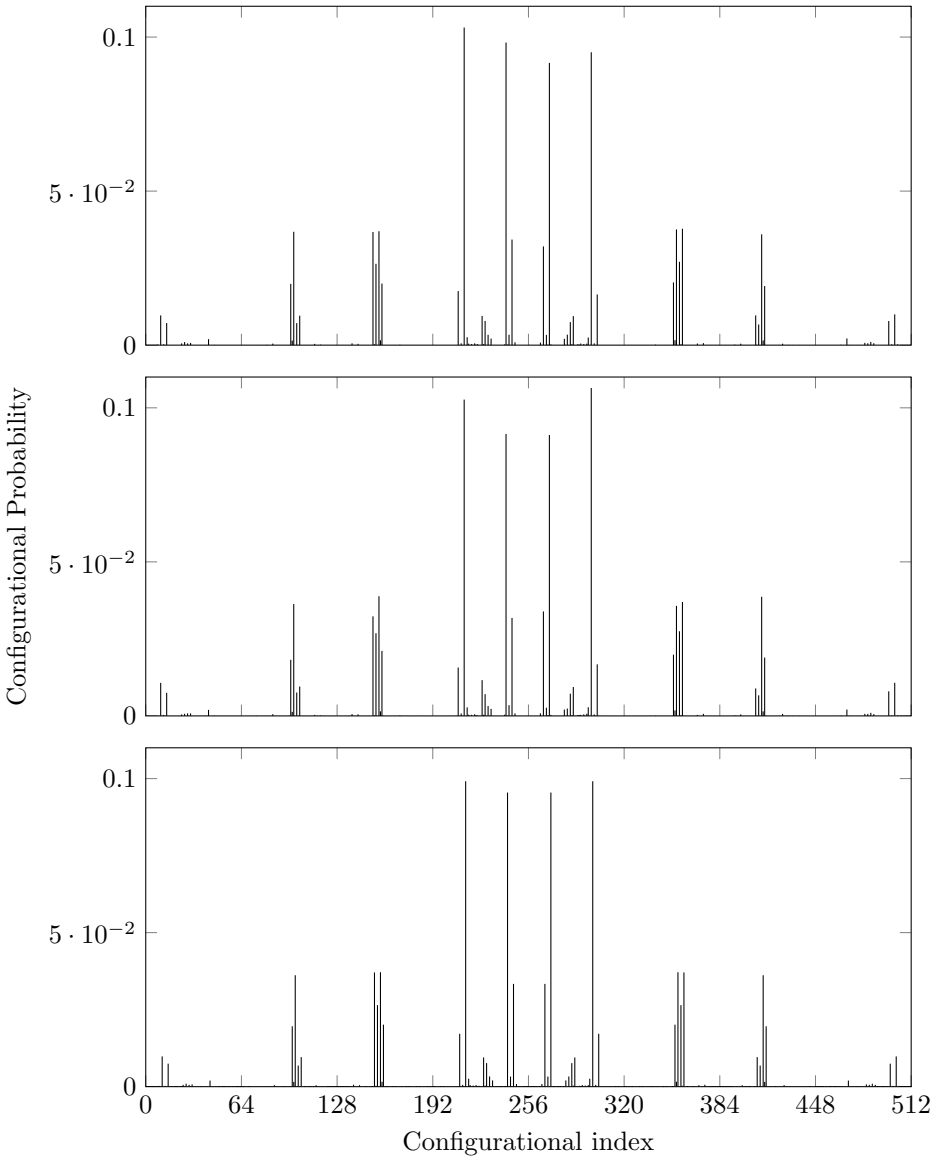
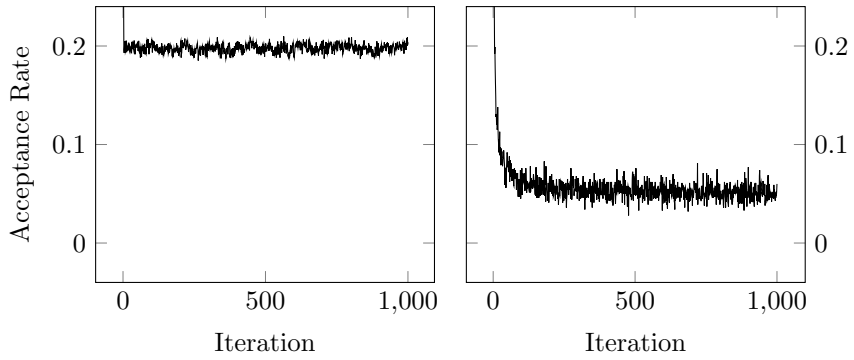


Figure 5.2: A typical set of acceptance rate time series for one of the spin-glass disorder samples. The left plot displays the acceptance rate time series for single-spin-flip and the right one displays the one for BAIS+L. Note that neither exhibits the plateau that was identified in Chapter 4 as a sign of a well-tuned BAIS+L simulation, indicating that the naïvely chosen hyperparameters used in the simulation were not appropriate for the spin glass simulated.



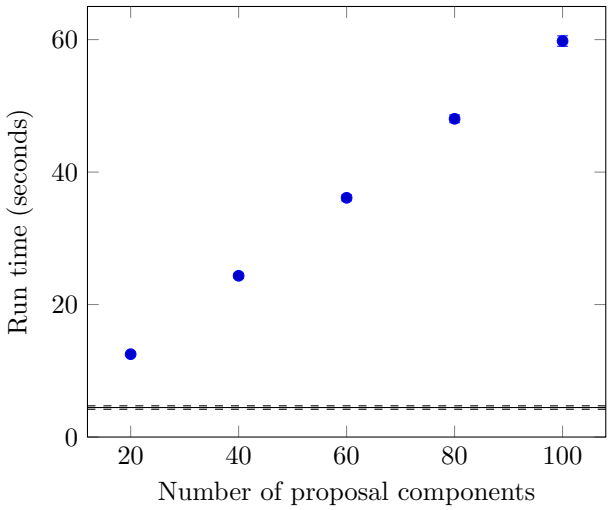
In addition to the lower acceptance rate, the computational complexity of BAIS+L, over that of single-spin-flip, was evident in the run time results. The mean run time of single-spin-flip was around 4.456 seconds, with an estimated standard deviation of 0.1149274 over the five disorder samples, while the fastest mean run time for BAIS+L was around 12.511 seconds, with a standard deviation of approximately 0.276 seconds. This minimum corresponded to the case with 20 mixture components in the proposal distribution. Figure 5.3 summarises the run time results.

Furthermore, the single-spin-flip approach runs all replicas independently of each other, while BAIS+L does not. Hence, estimates produced using single-spin-flip are at least as precise as those produced using BAIS+L, due to lower correlation. Even so, BAIS+L is, by design, more flexible, due its various tunable parameters, and the studied proposal distribution is only one of many possibilities.

As stated in Section 5.3.2, it was not expected that such a naïve choice of proposal distribution would be effective. Note that a key property of BAIS+L is the implicit clustering provided by having multiple components in the proposal distribution. For the normal mixture proposal introduced in Chapter 3 clusters may be defined by the distance of their constituent members from the centre of the component. For the multi-spin-flip proposal distribution studied in this chapter there was no analogue. The relative density at the component's mean and another configuration in the configuration space of the spin glass therefore, is not determined by the distance between them. In fact, as demonstrated by Figure 5.1, unless a component is degenerate on a single configuration, it is entirely possible that neighbouring configurations will have vastly different densities under the target.

However, there is still tremendous scope for further improvement. While there is no clear guidance to select the hyperparameters (the prior shapes of the beta distributions on the up-spin probabilities) as there was for the nor-

Figure 5.3: A comparison of the mean spin glass simulation run times of BAIS+L (blue dots) to that of the single-spin-flip approach (sold line). The standard deviations of the sample means for BAIS+L were similar to that of single-spin-flip (± one standard deviation illustrated by the dashed lines) and, therefore, smaller than the size of the dots. Notice the stark contrast in the time to complete a simulation, demonstrating the much greater computational complexity of BAIS+L.



mal mixture proposal studied in Chapter 4, further study into the automated approach mentioned there may help to alleviate such a burden on the MCMC practitioner.

5.3.4 Conclusion

The results presented in this chapter demonstrate that a naïve choice of fixed hyperparameters in a multi-spin-flip approach with BAIS+L is currently not viable for practical use. However, a possible avenue for alleviating such a problem has been identified in Chapter 4. This approach involved adaptation of the hyperparameters, given performance of the sampler at earlier iterations. Development of such an approach has the potential to overcome the problem with identifying suitable hyperparameters, which was identified in Section 5.2.1.

Furthermore, two other approaches to using BAIS+L in spin glass simulation in have been outlined in the current chapter. One of these approaches involved updating the spins in blocks (cf. the end of Section 5.2.1). As stated there, this approach will require multiple accept/reject steps within each iteration of each replica, much like traditional spin-flip. To accommodate this, a component membership will be required for each block of spins of each replica and the up-spin probabilities of spins not being updated will have to be ignored when computing proposal probabilities. Additionally, this approach would be a random walk, rather than an indepenence sampler.

The second approach considered in Section 5.2 was an alternative to the Swendsen-Wang algorithm. This approach was not explored numerically in

the current work, due to its incomplete formulation and much greater level of complexity, compared to the multiple spin update approach. However, given the faster mixing of Swendsen-Wang compared to single-spin-flip (cf. Section 2.3.4), it is worthwhile to consider if the technical aspects of a clustering approach of the configuration space may be overcome to produce a method that takes into account the dependence between adjacent spin sites.

As mentioned in Section 5.3.3, it is desirable that BAIS+L's proposal distribution impose an implicit clustering, as it did in Chapters 3 and 4 with a normal mixture proposal. Note that this clustering is of the *configuration space* and not of the lattice, as it is in the case of Swendsen-Wang. While the current study did not provide a proposal distribution with this important property for spin glass simulations, future work should consider how such clustering may be achieved.

Finally, the theoretical results of Chapter 3 were developed for BAIS+L with a normal mixture proposal distribution. As the spin glass application of the current chapter used a different proposal distribution, the negative results for spin glasses may indicate that BAIS+L is better suited to sampling $\mathbb{R}^d$-valued random variables than it is to discrete-valued ones. Future work should study this issue more closely.

Therefore, despite the underwhelming performance of BAIS+L observed in the current chapter, the potential for further improvement inherent in the flexibility that it provides means that there is still hope for it in the world of spin glass simulation.

Chapter 6

An Exact Approach

This chapter is joint work with Christian Robert of Paris Dauphine University and the University of Warwick. It considers a sampling method of a similar nature to BAIS+L that does not resort to sampling from an approximation to the target distribution. This new approach is named *Exact Bayesian Adaptive Independence Sampling with Latent variables* (EBAIS+L). The new sampler requires more computational effort than BAIS+L for the same number of samples but has the advantage of sampling exactly from the target. This extra computational effort leads to a tradeoff between quality of the simulated output and efficiency of the sampler. Furthermore, the sampling chains must be updated sequentially. This is in contrast to BAIS+L, which has the ability to update the sampling chains in parallel once the proposal distribution has been updated.

To explore this tradeoff, the performance of EBAIS+L is compared to that of BAIS+L by applying both samplers to a selection of target distributions from the optimisation test beds of Storn and Price (1997). As the results of the current chapter will demonstrate, the results of this comparison suggest that in the limit of an infinite number of sampling chains, the stationary distribution of the process induced by BAIS+L converges to the target and, hence, the stationary distribution induced by EBAIS+L. They also illustrate the greater efficiency of BAIS+L, suggesting it as a preferable approach when exactness of the target is not paramount.

Nevertheless, the construction of the exact sampler developed in this chapter is an important step towards an efficient exact sampler for target distributions with many local maxima.

This chapter begins with a description of the theoretical basis of the key modification to BAIS+L in Section 6.1, along with the description of EBAIS+L in Section 6.1.1. Section 6.2 explores the differences in performance of EBAIS+L and BAIS+L both in terms of efficiency and accuracy of simulated output. The section concludes with a discussion of the consequences of these results and provides suggestions for future work.

6.1 Going from an Approximate Approach to an Exact One

Recall from Chapter 3 that BAIS+L approximates the density of the state of sampling chain n under the target distribution, given those of all other sampling chains, according to the approximation restated in Equation (6.1),

$$p(\mathbf{x}_n|\mathbf{x}_{-n}) \approx p(\mathbf{x}_n|\boldsymbol{\theta}). \tag{6.1}$$

This approximation was introduced in BAIS+L's acceptance ratio, as a ratio of the densities of the current and proposed states of chain n, as in Equation (6.2)

$$\frac{p(\mathbf{x}_n|\mathbf{x}_{-n})}{p(\mathbf{y}|\mathbf{x}_{-n})} \approx \frac{p(\mathbf{x}_n|\boldsymbol{\theta})}{p(\mathbf{y}|\boldsymbol{\theta})}. \tag{6.2}$$

To provide an exact sampler, this approximation must be avoided and, instead, the left-hand side of Equation (6.1) must be used exactly.

Consider the modification of BAIS+L's proposal mechanism so that its parameter vector $\boldsymbol{\theta}$ is updated before each sampling chain update, resulting in an individual parameter vector $\boldsymbol{\theta}_n$ for updating each sampling chain n. Furthermore, do not make $\boldsymbol{\theta}_n$ dependent on the sampling chain n currently being updated, but only on the remaining $N-1$ sampling chains. Then, to update sampling chain n, while holding all the others fixed, corresponds to a single iteration of a standard MH algorithm, with acceptance ratio given by Equation (6.3)

$$\alpha(\mathbf{x}_n, \mathbf{y}) = \min\left\{1, \frac{\pi(\mathbf{y})p(\mathbf{x}_n|\boldsymbol{\theta}_n)}{\pi(\mathbf{x})p(\mathbf{y}_n|\boldsymbol{\theta}_n)}\right\}. \tag{6.3}$$

Recall from Section 1.2.3 of Chapter 1 that Besag et al. (1995) showed in their first appendix that under such a scheme, if the update mechanism for a particular sampling chain n satisfies detailed balance for any random choice of $\boldsymbol{\theta}$ (provided that it depends only on $\mathbf{x}_{-n}$), then the mixture over all $\boldsymbol{\theta}$ also satisfies detailed balance with the original target as its stationary distribution.

In fact, Besag et al. (1995) explicitly gave the example of a random MH update as one such method to which their result applies. This result follows directly from the fact that MH sampling satisfies detailed balance (cf. Section 1.3.1).

Since the preceding modification of BAIS+L's update mechanism for a single sampling chain is just an MH update, it immediately follows that it satisfies detailed balance with the original target as its stationary distribution.

Thus, if the proposal parameters of BAIS+L for each chain are updated sequentially while keeping all other chains fixed, instead of all at once before making MH moves, then the standard MH acceptance ratio may be used and the stationary distribution will be exactly equal to the target.

6.1.1 Exact BAIS+L (EBAIS+L)

The assertion of Besag et al. (1995) that was invoked in the preceding section only requires that the transition kernels used to update each sampling chain preserve detailed balance. By sampling the proposal parameters $\{\boldsymbol{\theta}_n\}_{n=1}^N$ conditionally on $\mathbf{x}_{-n}$ and $\mathbf{z}_{-n}$ but not the state $\mathbf{x}_n$ or the latent allocation z_n being updated, a sampler may still exploit the result of Besag et al. (1995).

Therefore, this section proposes a modified version of BAIS+L that takes advantage of this result. Following on from the theoretical basis discussed in Section 6.1, the modified approach, which shall be called the *Exact Bayesian Adaptive Independence Sampler with Latent variables* (EBAIS+L) requires sampling new proposal parameters for each sampling chain, given all other sampling chain states and latent allocations.

When updating sampling chain n, the proposed vectors of chain states $\mathbf{x}^*$ and latent allocations $\mathbf{z}^*$ are the same as the current vectors of chain states $\mathbf{x}$ and latent allocations $\mathbf{z}$, respectively, except in the nth entry.

Denote the proposed state of the nth sampling chain by $\mathbf{y}$ and the proposed latent allocation of the nth sampling chain by w. For each sampling chain n and mixture component k, the sampler requires the observed counts o_k of the component, the component mean $\bar{\mathbf{x}}_k$ and a term $\mathbf{\Lambda}_k$ containing the component sum of squares, all excluding sampling chain n. These summary statistics are given by Equations (6.4), (6.5) and (6.6), respectively,

$$o_k = \sum_{i=1}^{n-1} I_k\left[z_i^{(t)}\right] + \sum_{i=n+1}^{N} I_k\left[z_i^{(t-1)}\right], \tag{6.4}$$

$$\bar{\mathbf{x}}_k = \frac{1}{o_k}\left[\sum_{i=1}^{n-1} I_k\left[z_i^{(t)}\right]\mathbf{x}_i^{(t)} + \sum_{i=n+1}^{N} I_k\left[z_i^{(t-1)}\right]\mathbf{x}_i^{(t-1)}\right], \tag{6.5}$$

$$\begin{aligned}\mathbf{\Lambda}_k = \mathbf{\Lambda}_k^{(0)} &+ \frac{\kappa_k^{(0)} o_k}{\kappa_k^{(0)} + o_k}\left[\bar{\mathbf{x}}_k - \boldsymbol{\mu}_k^{(0)}\right]\left[\bar{\mathbf{x}}_k - \boldsymbol{\mu}_k^{(0)}\right]^{\mathrm{T}} \\ &+ \sum_{i=1}^{n-1} I_k\left[z_i^{(t)}\right](\mathbf{x}_i - \bar{\mathbf{x}}_k)(\mathbf{x}_i - \bar{\mathbf{x}}_k)^{\mathrm{T}}. \\ &+ \sum_{i=n+1}^{N} I_k\left[z_i^{(t-1)}\right](\mathbf{x}_i - \bar{\mathbf{x}}_k)(\mathbf{x}_i - \bar{\mathbf{x}}_k)^{\mathrm{T}},\end{aligned} \tag{6.6}$$

where, just as in Chapter 3, $\kappa_k^{(0)}$ is the prior number of observations of the scale of $\mathbf{\Sigma}_k$, $\boldsymbol{\mu}_k^{(0)}$ is the prior mean vector of a component k, and $\mathbf{\Lambda}_k^{(0)}$ is the prior scale matrix of $\mathbf{\Sigma}_k$ (Gelman et al., 2004, pp. 87). Note that, by the sequential nature of EBAIS+L, the states and allocations of the $n-1$ sampling chains that have been updated are those at time step t, while the $N-n$ sampling chains of higher index are those from the previous iteration $t-1$.

Given these statistics, the distributions used to update the parameters are the same as in BAIS+L. For convenience, they are restated below,

$$\begin{aligned}(d_1, \ldots, d_K)|\mathbf{x}, \mathbf{z} &\sim \text{Dirichlet}\left[\cdot\,\middle|\, o_1 + \alpha_1^{(0)}, \ldots, o_K + \alpha_K^{(0)}\right], \\ \mathbf{\Sigma}_k|\mathbf{x}, \mathbf{z} &\sim \text{Inv-W}_{\nu_k^{(0)} + o_k}(\cdot|\mathbf{\Lambda}_k); \\ \boldsymbol{\mu}_k|\mathbf{\Sigma}_k, \mathbf{x}, \mathbf{z} &\sim \mathcal{N}\left[\cdot\,\middle|\, \frac{\kappa_k^{(0)}}{\kappa_k^{(0)} + o_k}\boldsymbol{\mu}_k^{(0)} + \frac{o_k}{\kappa_k^{(0)} + o_k}\bar{\mathbf{x}}_k, \frac{\mathbf{\Sigma}_k}{\kappa_k^{(0)} + o_k}\right],\end{aligned}$$

where $\nu_k^{(0)}$ is prior degrees of freedom of $\mathbf{\Sigma}_k$ (Gelman et al., 2004, pp. 87) and $\alpha_k^{(0)}$ is the prior (unnormalised) weight of component k.

Note that, given the forms of $\mathbf{o}_k$, $\bar{\mathbf{x}}_k$ and $\mathbf{\Lambda}_k$ given in Equations (6.4), (6.5) and (6.6), respectively, these posterior update distributions have a different dependence on the current states of the sampling chains. As stated at the start of the current section, the parameters must be sampled for each chain individually, dependent on all chains other than the one being updated. This key difference increases the computational effort required by EBAIS+L over BAIS+L, making the sampler of order $n(n-1)$, compared to order n for BAIS+L. However, using a dynamic updating approach for the sample statistics, it is possible to reduce the computational effort required by EBAIS+L back down to an order of n, with only marginal extra effort over BAIS+L required, due to the extra updating of proposal parameters for each sampling chain. Such an approach would compute the statistics o_k, $\bar{\mathbf{x}}_k$ and $\mathbf{\Lambda}_k$ "on-line", computing each in full at the start of a simulation using all relevant initial samples, then updating them as each sampling chain is updated.

While EBAIS+L is inherently more computationally expensive than BAIS+L, the fact that its output has the correct stationary distribution motivates a comparison of it to BAIS+L, a matter that will be addressed in Section 6.2.

Algorithm 6.1 summarises the procedure for implementing EBAIS+L.

6.2 Comparing EBAIS+L to BAIS+L

Having identified an exact modification of BAIS+L, the focus of this chapter now turns to comparing its performance to the approximate sampler, BAIS+L. As the following comparison will show, despite its approximate nature, the performance of BAIS+L is only slightly worse than that of EBAIS+L, while being more efficient.

6.2.1 Methodology

To compare EBAIS+L and BAIS+L, both samplers were applied to a selection of targets from the test beds of Storn and Price (1997), with the same settings. Since the purpose of these targets was to test optimisation algorithms, Storn and Price (1997) used them in the context of finding their global minima. The needs of the current chapter were somewhat different, as the goal was to test algorithms for sampling from probability distributions. To this end the selected test functions were modified into probability distributions that were suitable for sampling.

In order to do so, their supports were first restricted to suitable domains. In this manner they were guaranteed to have finite total mass. The specific domain of each target distribution is given with its description in the following subsection.

Target Distributions

Both EBAIS+L and BAIS+L were run on a selection of one-, two- and three-dimensional targets from the test-beds of Storn and Price (1997). The specific functions chosen were Shekel's foxholes (a two-dimensional target), one-, two- and three-dimensional Rastrigin's and Ackley's functions, and Goldstein's function (a one-dimensional target). These targets were chosen for their multiple

Algorithm 6.1 EBAIS+L: An exact Bayesian Adaptive Mixture Independence Sampler.

Require:

1. $K \in \mathbb{Z}^+$ proposal components.
2. Initial chain states $\left[\mathbf{x}_n^{(0)}\right]_{n=1}^N$ and their allocations $\left[z_n^{(0)}\right]_{n=1}^N$.
3. Prior scales $\mathbf{\Lambda}_k^{(0)}$ and degrees of freedom $\nu_k^{(0)}$ of the distributions on $\mathbf{\Sigma}_k, k \in \{1, \ldots, K\}$.
4. Prior means $\boldsymbol{\mu}_k^{(0)}, k \in \{1, \ldots, K\}$.
5. Prior numbers of observations $\kappa_k^{(0)}$ of the scales of $\mathbf{\Sigma}_k, k \in \{1, \ldots, K\}$.
6. Prior component weights $\boldsymbol{\alpha}^{(0)} = \left[\alpha_k^{(0)}\right]_{k=1}^K$.

Ensure: N chains of samples $\left\{\left[\mathbf{x}^{(t)}\right]_{n=1}^N\right\}_{t=1}^\infty$ from the target distribution π.

for Iteration $t \in \mathbb{Z}^+$ **do**
  for Chain $n = 1$ to $n = N$ **do**
    for Mixture component $k = 1$ to $k = K$ **do**
      Set $o_k = \sum_{i=1}^{n-1} I_k\left[z_i^{(t)}\right] + \sum_{i=n+1}^{N} I_k\left[z_i^{(t-1)}\right]$.
      Set $\bar{\mathbf{x}}_k = \left\{\sum_{i=1}^{n-1} I_k\left[z_i^{(t)}\right]\mathbf{x}_i^{(t)} + \sum_{i=n+1}^{N} I_k\left[z_i^{(t-1)}\right]\mathbf{x}_i^{(t-1)}\right\} \Big/ o_k$.
      Set $\mathbf{\Lambda}_k = \mathbf{\Lambda}_k^{(0)} + \frac{\kappa_k^{(0)} o_k}{\kappa_k^{(0)} + o_k}\left[\bar{\mathbf{x}}_k - \boldsymbol{\mu}_k^{(0)}\right]\left[\bar{\mathbf{x}}_k - \boldsymbol{\mu}_k^{(0)}\right]^{\mathrm{T}} + \sum_{i=1}^{n-1} I_k\left[z_i^{(t)}\right](\mathbf{x}_i - \bar{\mathbf{x}}_k)(\mathbf{x}_i - \bar{\mathbf{x}}_k)^{\mathrm{T}} + \sum_{i=n+1}^{N} I_k\left[z_i^{(t-1)}\right](\mathbf{x}_i - \bar{\mathbf{x}}_k)(\mathbf{x}_i - \bar{\mathbf{x}}_k)^{\mathrm{T}}$.
      Generate $\mathbf{\Sigma}_k \sim \text{Inv-W}_{\nu_k^{(0)} + o_k}(\cdot|\mathbf{\Lambda}_k)$.
      Generate $\boldsymbol{\mu}_k \sim \mathcal{N}\left[\cdot \left| \frac{\kappa_k^{(0)}}{\kappa_k^{(0)} + o_k}\boldsymbol{\mu}_k^{(0)} + \frac{o_k}{\kappa_k^{(0)} + o_k}\bar{\mathbf{x}}_k, \frac{\mathbf{\Sigma}_k}{\kappa_k^{(0)} + o_k}\right.\right]$.
    end for
    Generate $(d_1, \ldots, d_K) \sim \text{Dirichlet}\left[\cdot \left| o_1 + \alpha_1^{(0)}, \ldots, o_K + \alpha_K^{(0)}\right.\right]$.
    Generate $w \sim \text{Categorical}(\cdot|d_1, \ldots, d_K)$.
    Generate $\mathbf{y} \sim \mathcal{N}(\cdot|\boldsymbol{\mu}_w, \mathbf{\Sigma}_w)$.
    Set $\alpha = \min\left\{1, \frac{\pi(\mathbf{y})p(\mathbf{x}_n|\mathbf{d}, \boldsymbol{\mu}, \mathbf{\Sigma})}{\pi(\mathbf{x}_n)p(\mathbf{y}|\mathbf{d}, \boldsymbol{\mu}, \mathbf{\Sigma})}\right\}$.
    Generate $u \sim \mathcal{U}(0, 1)$.
    if $u \leq \alpha$ **then**
      Set $\mathbf{x}_n^{(t)} = \mathbf{y}$.
      Set $z_n^{(t)} = w$.
    else
      Set $\mathbf{x}_n^{(t)} = \mathbf{x}_n^{(t-1)}$.
      Set $z_n^{(t)} = z_n^{(t-1)}$.
    end if
  end for
end for

or non-trivial local maxima, which make them suitable for a comparison of samplers with mixture proposal distributions.

As noted in Section 6.2.1, the target functions were converted into densities by restricting their domains and normalising them. Outside a target's domain, the density was set to zero, to ensure that the total mass was finite.

Since the version of Shekel's foxholes used had a fixed number of 25 foxholes, the domain was restricted to a suitable region that included all of them. This region was $\{(x, y) : x, y \in [-40, 40]\}$.

For Rastrigin's and Ackley's functions, the same domains were chosen, due to the similarity between the two targets. In one dimension the domain was restricted to $\{x : x \in [-4, 4]\}$, in two dimensions it was restricted to $\{(x, y) : x, y \in [-2, 2]\}$ and in three dimensions it was restricted to $\{(x, y, z) : x, y, z \in [-1, 1]\}$. These choices resulted in targets with 8, 16 and 8 local maxima, respectively.

Finally, Goldstein's function was restricted to $[-4, 4]$, to give its two local maxima comparable density to that at the end-points of the considered support.

As each of the preceding functions is non-negative on its chosen domain, as noted by Storn and Price (1997), no further adjustments beyond the domain restrictions and normalisations were required. In fact, even the normalisation constant was not necessary, as it cancels out in all calculations.

Selecting Simulation Parameters

Having defined the target densities, the number of components in the proposal distribution for each target was chosen, by following the considerations suggested in Section 4.1.2 of Chapter 4. Once again, common hyperparameters $\alpha_k^{(0)} = \alpha^{(0)}$, $\boldsymbol{\Sigma}_k^{(0)} = \sigma^{(0)}\mathbf{I}_p$, $\boldsymbol{\mu}_k^{(0)} = \boldsymbol{\mu}^{(0)}$, $\kappa_k^{(0)} = \kappa^{(0)}$ and $\nu_k^{(0)} = \nu^{(0)}$ were used for convenience and ease of implementation. Here, $\mathbf{I}_p$ represents the p-dimensional identity matrix.

Starting with the initial parameter setting of $\alpha^{(0)} = \sigma^{(0)} = 1$ and $\nu^{(0)} = p + 1$, where p is the dimension of the target, one pilot run of each target density was simulated, with $K = 100$ components in the proposal distribution, $N = 2000$ sampling chains and $T = 100$ iterations for each choice of $\kappa^{(0)} \in \{10, 1, 0.1, 0.01, 0.001\}$. As in Chapter 4, the prior component means were centred at the origin $\boldsymbol{\mu}^{(0)} = (0, 0)^{\mathrm{T}}$.

The acceptance rate time series of the results of the various choices of $\kappa^{(0)}$ were inspected, to determine which exhibited a plateau, as discussed in Section 4.1.2. Once the choices of $\kappa^{(0)}$ that produced the required result were identified, further pilot runs were simulated, with the identified values of $\kappa^{(0)}$ as the ratio $\kappa^{(0)}/\sigma^{(0)}$. One pilot run was simulated for each such choice of $\kappa^{(0)}$ and $\sigma^{(0)} \in \{10, 1, 0.1, 0.01, 0.001\}$.

Once all pilot runs were complete, the smallest $\kappa^{(0)}$ and $\sigma^{(0)}$ that resulted in the maximum observed plateau for both EBAIS+L and BAIS+L were selected.

After settling on choices of $\kappa^{(0)}$ and $\boldsymbol{\Sigma}^{(0)}$ for each target distribution (cf. Table 6.1), the `multmixmodel.sel` function from the `mixtools` package of Benaglia et al. (2009) was applied to the latent allocations of the last half of the samples from each sampler. This function takes as input a collection of multinomial samples (in the case of EBAIS+L or BAIS+L, the vector at each iteration of latent allocations) and reports the corresponding Akaike, Bayesian and consistent Akaike information criteria, as well as the integrated completed

Table 6.1: Simulation parameter settings for each target in the comparison of EBAIS+L to BAIS+L. p is the dimension of the target, a is the maximum of the absolute value in any axis direction of the support, K is the number of mixture components in the proposal and N is the number of sampling chains.

Target	p	a	K	N			$\kappa^{(0)}$	$\Sigma^{(0)}$
Shekel's foxholes	2	40	26	520	780	1040	1	10
Rastrigin	1	4	32	640	960	1280	0.001	0.1
	2	2	30	600	900	1200	0.01	
	3	1	28	560	840	1120		
Ackley	1	4	27	540	810	1080		
	2	2	31	620	930	1240		
	3	1	26	520	780	1040		
Goldstein	1	4	32	640	960	1280	0.1	0.001

likelihood and log-likelihood (cf. Section 1.5.3). It also reports a "winning" number of components for each computed criterion, indicating the inferred optimum number.

The `multmixmodel.sel` function was run on the latent allocations in the last 50 iterations, by which time all acceptance rates had converged, as evidenced by an approximately constant acceptance rate time series from this time. This was done with the optional `comps` argument of the function set to each value from 1 to K. The reason for running the function with the K different `comps` values was due to the fact that with too many components to test the `multimixmodel.sel` function may fail with an error. By testing each possible value of K the function was given more opportunity to succeed. Once the function failed for the first time or all `comps` values up to 100 had been simulated, the analysis of the number of components was determined to be complete.

The log-likelihood tended to exaggerate the number of components more than the other statistics did, consistently reporting the "winning" number of components to be greater than those reported by the other criteria. Knowing the true forms of the target distributions, such numbers of components were considered to be excessive. Therefore, the largest reported number from the remaining statistics was used as the required number of components. Finally, the number of sampling chains was set to 20, 30 and 40 times the chosen fixed number of mixture components for longer simulations.

By the preceding approach, the parameter settings listed in Table 6.1 were determined. These settings were employed for both EBAIS+L and BAIS+L.

Finally, a $d_{\min} = 0.1$ was enforced, following the prescription given in Section 3.3.3, with $I_d = 10$ and $\Delta_t = 0.01$.

Observed Quantities

To compare the sampling efficiency of the two samplers, their mean convergence times, their mean acceptance rates and their mean effective proportions of samples were computed. To compute these quantities, the approach of Chapter 4 was followed.

For the mean convergence time this means that for each of the ten independent simulations from the same target, with the same number of sampling

chains, the PSRF was computed at iterations $51 + 19m$ for $m \in \mathbb{N}$ and the convergence time reported as the earliest of these iterations from which the 97.5% PSRF remained below 1.01. Maximal variance of each estimate was assumed, depending on the width of the time interval to which it corresponded. That is, if the convergence time of a simulation was 51 then the reported variance was $51^2/4$, otherwise it was $19^2/4$, as in Chapter 4. The convergence times of the ten independent repeated simulations were combined to give the mean convergence times and their corresponding overall variances.

Computation of the acceptance rate was more straightforward, simply being the number of accepted proposed states at each iteration, divided by the number of sampling chains. The result was a single chain of 1000 acceptance rate estimates for each simulation. The mean and standard deviation of the acceptance rate of a single simulation were estimated as the geometric mean and standard deviation over the last half of the samples, respectively. To account for the autocorrelation between the acceptance rates at successive iterations, their effective number of samples were computed as the reciprocal of twice the integrated autocorrelation time of the last half of the acceptance rate estimates. These estimates were used to rescale the standard deviations accordingly.

Finally, to compute the effective number of samples, the `effectiveSize` function from `R`'s `MASS` library was applied to each dimension of the last half of the samples from a simulation. The effective proportion of samples was then given by dividing the result by the total number of samples, and its variance was estimated using the standard variance approximation of a binomial random variable used in Chapter 4.

To compare the accuracy of inference, the inferred values of the means of the first four central moments in each dimension for each sampler were compared to their true values. The true marginal moments were numerically approximated using the `adaptIntegrate` function from `R`'s `cubature` (Narasimhan et al., 2018) package.

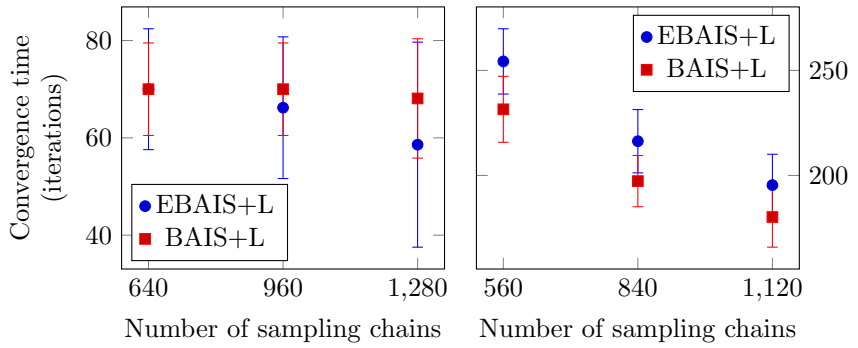
Naïve kernel density estimates from the simulated output were also computed for the one-dimensional and two-dimensional targets. R's (R Core Team, 2015) built-in `density` function was used to generate one-dimensional kernel density estimates, while the `kde2d` function from R's `MASS` library (Venables and Ripley, 2002) was used to generate two-dimensional kernel density estimates. Each function was called with its default settings. For all targets, the last half of samples from a simulation were used for inference. For each target with each number of sampling chains, the individual kernel density estimates were combined to give a mean kernel density estimate and the corresponding standard deviation at each state.

6.2.2 Results

Sampling Efficiency

The simulations demonstrated that BAIS+L typically converged faster than EBAIS+L in terms of the number of iterations. However, this was not a hard-and-fast rule. Figure 6.1 presents the mean observed convergence times of two of the simulated targets for each number of sampling chains simulated, using both EBAIS+L and BAIS+L. The horizontal axis in each case indicates the number of sampling chains, while the vertical axis reports the convergence time

Figure 6.1: The mean convergence times for the one- and three-dimensional Rastrigin function. The blue circles represent the convergence times obtained using EBAIS+L, while the red squares represent those obtained using BAIS+L.



in iterations.

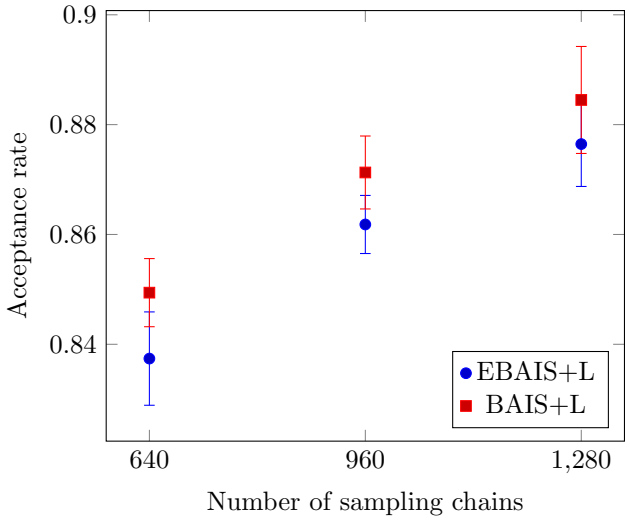
The left plot of Figure 6.1 represents the convergence times of the one-dimensional Rastrigin function, for which EBAIS+L converged slightly faster than BAIS+L, while the right plot represents the convergence times of the three-dimensional Rastrigin function, whose Markov chain converged slightly faster using BAIS+L. The situation observed for the three-dimensional Rastrigin was the most commonly-observed in the simulations of this chapter, with BAIS+L converging slightly faster with respect to the number of iterations. This result suggests that the sacrifice in exactness of the stationary distribution associated with BAIS+L is slightly offset by faster sampling. Even though this increased efficiency of BAIS+L with respect to the convergence time is not very large, recall that BAIS+L has a lower computational burden than EBAIS+L, since it only updates proposal parameters once per iteration, unlike EBAIS+L, which does so N times per iteration.

Regardless of the sampler or the target, the convergence time always decreased with the number of sampling chains for both samplers. This result is consistent with that observed in the comparison of BAIS+L to the equi-energy sampler in Chapter 4. Since EBAIS+L also exhibits the same trend, it is conjectured that, as in the case of BAIS+L, the reduction in sampling variance obtained with the use of more sampling chains leads to faster convergence of the proposal distribution.

However, as the number of sampling chains was increased, the typical case, exemplified by the three-dimensional Rastrigin function results in Figure 6.1, also suggested that the convergence times of the two samplers converge. Such a result is not surprising, as it agrees with the Conjecture 1, which postulated that as the number of sampling chains is increased, the stationary distribution approaches the target. In other words, BAIS+L becomes exact.

The acceptance rate results further highlight the greater efficiency of BAIS+L over that of EBAIS+L. Figure 6.2 once again shows that BAIS+L demonstrates slightly better mixing, as evidenced by its acceptance rate being greater than that of EBAIS+L in most cases studied. As with the convergence times, the acceptance rates of the two samplers converge, lending further support to the conjecture that the two samplers are the same in the limit of an infinite number

Figure 6.2: The mean acceptance for the one-dimensional Rastrigin target. The blue circles represent the acceptance rates obtained using EBAIS+L, while the red squares represent those obtained using BAIS+L.



of sampling chains. With respect to the acceptance rate, BAIS+L was always at least as efficient as EBAIS+L for the same hyperparameter settings.

Figure 6.3 illustrates the effective proportions of samples in each case studied. The results are similar to those for the acceptance rates, with BAIS+L being consistently more efficient than EBAIS+L but converging in efficiency with an increasing number of sampling chains.

Finally, consider the run times of the two samplers. Figure 6.4 presents a typical result observed in the simulations of this chapter, illustrated by those for the three-dimensional Rastrigin function. The first observation to note is that the run time of EBAIS+L was substantially longer than that of BAIS+L. Furthermore, the run time increases more rapidly with the number of sampling chains than it does in the case of BAIS+L. Such a result is to be expected, since EBAIS+L updates the parameters of its proposal distribution each time it updates a sampling chain's state, while BAIS+L only updates them once per iteration. However, the reported difference in run times should be considered to be a soft upper bound, given that the code used to implement EBAIS+L in the simulations of this chapter computed the sample mean and covariance matrix from scratch each time it updated the proposal parameters. By writing the computer code for the algorithm more carefully, in order to update the statistics dynamically, as discussed in Section 6.1.1, considerable speed improvements are possible.

Nevertheless, even with such changes, the run time of BAIS+L cannot theoretically exceed that of EBAIS+L, so long as the procedure to update the mixture proportions (cf. Section 3.3.3) uses a fixed number of iterations. Furthermore, as already noted, the design of EBAIS+L presented in this chapter does not lend itself to parallel updates of sampling chain states, while the design of BAIS+L does.

Figure 6.3: The mean effective proportion of samples for the two-dimensional Rastrigin function. The blue circles represent the effective proportions obtained using EBAIS+L, while the red squares represent those obtained using BAIS+L.

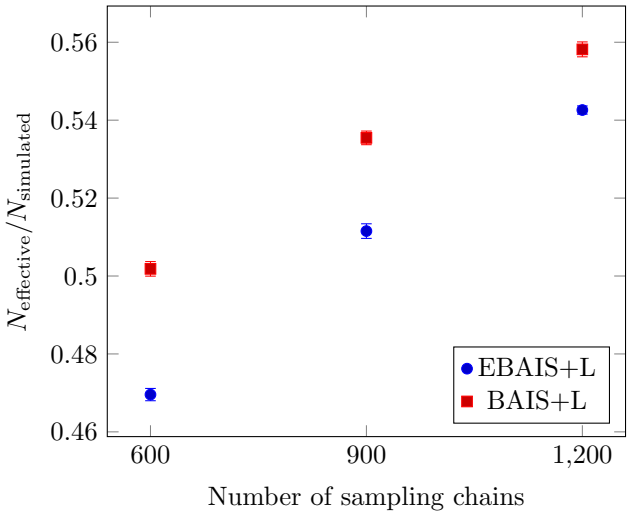


Figure 6.4: Run times of the three-dimensional Rastrigin function using EBAIS+L (blue circles) and BAIS+L (red squares).

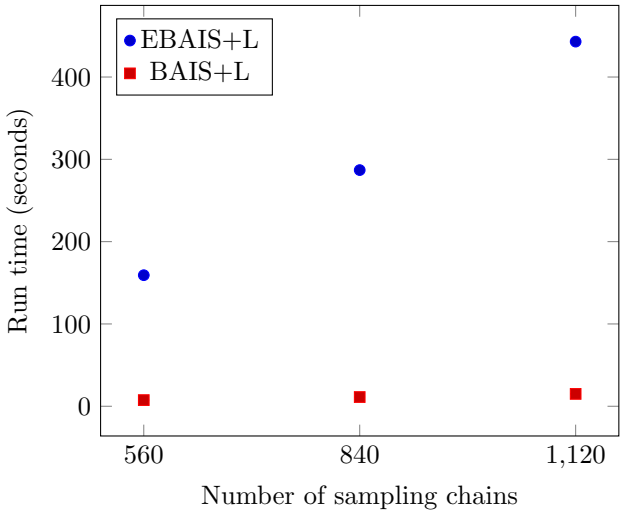
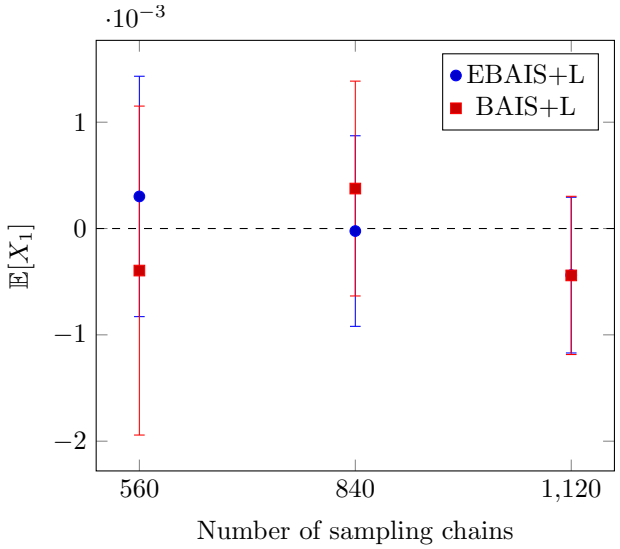


Figure 6.5: The first marginal central moment of the first dimension of the three-dimensional Rastrigin function. The blue circles represent the inferences obtained using EBAIS+L, while the red squares represent those obtained using BAIS+L.



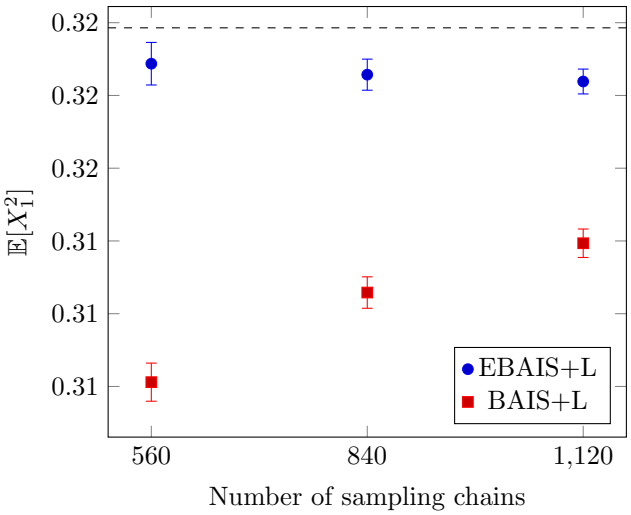
Sampler Accuracy

Figure 6.5 illustrates the first marginal central moment of the first dimension of the three-dimensional Rastrigin function. There appears to be little difference between the performance of EBAIS+L (blue circles) and BAIS+L (red squares) for any number of sampling chains, with both samplers producing comparable estimates of odd the moments of the target of interest. Since all target distributions studied were symmetric about the origin in each dimension, the true values of the odd central moments were zero, as indicated by the horizontal dashed line in each plot.

The second and fourth marginal central moments, however, demonstrated a stark contrast between EBAIS+L and BAIS+L. Figure 6.6 demonstrates the second moment in the first dimension of the three-dimensional Rastrigin function. The results illustrate the typical relationship that was observed, namely, that EBAIS+L produces better inferences of the even moments of the target than does BAIS+L. This is to be expected, since EBAIS+L is designed to sample exactly from the target, while BAIS+L only samples from an approximation to it. However, as was observed in the convergence and mixing results, the moments inferred by the two samplers appear to converge as the number of sampling chains is increased, which further supports the main conjecture of Chapter 3: that as the number of sampling chains is increased, the limiting distribution of BAIS+L converges towards the target and, hence, the limiting distribution of EBAIS+L.

Finally, the kernel density estimates of the one- and two-dimensional targets (not shown) were considered. The inferred estimates generated by the two samplers were qualitatively the same for each target and successfully produced

Figure 6.6: The second marginal central moment of the first dimension of the three-dimensional Rastrigin function. The blue circles represent the inferences obtained using EBAIS+L, while the red squares represent those obtained using BAIS+L.



shapes that were similar to the true forms of the densities (cf. Section 2.1 of Chapter 2). The variability in the estimates, in the form of the standard deviation of the density at each estimated state, were also of the same order of magnitude for both samplers.

6.2.3 Discussion

From the preceding results it is evident that, for the target distributions simulated in this study, BAIS+L is more efficient than the exact approach of EBAIS+L, while still producing comparable output, despite its approximate nature.

The convergence time and mixing results suggest the following conjecture.

Conjecture 4 (Convergence of BAIS+L and EBAIS+L)**.** *As the number of sampling chains N increases, the convergence times of BAIS+L and EBAIS+L with respect to the number of iterations performed, converge for the same target. Furthermore, the two samplers have the same stationary distribution, effective number of samples and acceptance rate in the limit as $N \to \infty$.*

However, it should be noted that BAIS+L's computational effort is much lower than that of EBAIS+L. This is evident because EBAIS+L requires that the proposal parameters be updated each time a sampling chain is updated, whereas BAIS+L only needs to perform proposal parameter updates once per iteration. It is also possible to directly perform BAIS+L sampling chain updates in parallel, while those in EBAIS+L, by design, must be performed sequentially.

Regarding the quality of the output it was also observed that, as the number of sampling chains was increased, the differences between the estimates of the first four moments in each axis direction, for each of the targets simulated, appeared to decrease. In terms of the first and third marginal central moments in each dimension for each target, both EBAIS+L and BAIS+L were indistinguishable, at least for the targets simulated in this study. It is possible that results may differ when sampling from target distributions that are not symmetric and this should be checked in future work. Furthermore, the qualitative output of the kernel density estimates was the same for both samplers, resembling their true forms.

One clear advantage of EBAIS+L over BAIS+L that was demonstrated by the results of the current chapter, was the estimation of the second and fourth marginal central moments. Unlike the first and third moments, EBAIS+L produced more accurate estimates. This greater success may be attributed to the exact nature of EBAIS+L, as opposed to the approximate one of BAIS+L. However, as the number of sampling chains was increased, BAIS+L's inferences tended towards the true estimates. This result supports the conjecture that as the number of sampling chains increases the stationary distributions of the samples generated by the two samplers converge.

These results highlight that BAIS+L is a viable method for simulating from distributions with multiple or non-trivial local maxima, using a mixture proposal. They also identify it as a method to decouple sampling chains within an iteration, while keeping approximation errors arbitrarily small, given a sufficient number of sampling chains.

Further work will consider the theoretical link between EBAIS+L and BAIS+L. In particular, it should ascertain the validity of the overarching conjecture put forth in this study: that the two samplers are identical in the limit of an infinite number of sampling chains.

Closing Remarks

This study has presented a new approach (BAIS+L) to adaptive MCMC that makes use of the posterior dependence of the parameters in the proposal distribution of an MH sampler on the samples generated. The new approach is flexible, using a mixture of normal distributions as its proposal distribution, allowing it to approximate a wide variety of targets to arbitrary accuracy.

While the stationary distribution of the samples generated by BAIS+L is only an approximation to the target, the results of Chapter 4 are encouraging. By having comparable performance to the equi-energy sampler, in terms of mixing, BAIS+L demonstrates that it has promise as a tool for the MCMC practitioner to use for sampling from targets on non-denumerable state spaces. The approximate nature of the stationary distribution is evident in the results, however, they support Conjecture 1, suggesting that as the number of sampling chains increases, the stationary distribution approaches the intended target.

Additionally, by increasing the number of components in the proposal distribution, the results of Chapter 4 demonstrated that BAIS+L had more success in finding the local maxima of the target. Given the primary motivation of developing a method for sampling from target distributions with many local maxima, this result is an illustration of the importance of being able to adjust the number of components in the proposal distribution. While there are other adaptive samplers that use mixture proposals (cf. Section 1.7), the method of adaptation provided by BAIS+L appears to be unique for such a problem.

Further supporting the role of BAIS+L are the theoretical results of Section 3.3. By design, the ergodicity of BAIS+L cannot be guaranteed by the same theory behind other adaptive samplers. However, easy-to-verify conditions, which do ensure ergodicity, have been provided in the course of this study and they have been proven to be sufficient. In particular, the provided conditions provide *uniform* ergodicity, ensuring geometric convergence to the stationary distribution. Some guidance has also been provided in Section 3.3.3 on how to enforce these conditions.

In Chapter 5 two possible proposal distributions for BAIS+L simulations of spin glasses were suggested. Each approach offers a mechanism that is tailored to spin glass simulation, since a normal mixture proposal is not suitable in such a situation. The details one of these proposal mechanisms, a clustering approach, were beyond the scope of the current study. The other mechanism, however, was tractable, eliciting a posterior distribution of a standard form.

Computer simulation with this tractable approach demonstrated results that contrasted the promising ones of Chapter 4, highlighting some shortcomings of of BAIS+L. In particular, it was noted that BAIS+L with a common prior on

the parameters of each proposal component, had difficulty in providing efficient sampling from a spin glass model. This difficulty was evidenced by the acceptance rate time series reducing to a minimum rate. It was posited that for such a problem to be overcome, it may be necessary to select the hyperparameters of the parameter model on an individual basis. However, it was not obvious how this may be done before the start of a simulation.

Chapter 6 introduced an exact version of BAIS+L, which was called EBAIS+L. By taking advantage of a theoretical result of Besag et al. (1995), it was possible able to modify BAIS+L to guarantee a stationary distributionthat was equal to the intended target. Computer simulations from a collection of targets illustrated the improvement in inferences from the stationary distribution over those produced using the approximate BAIS+L algorithm. This exactness, however, came at the cost of greater computational expense, as the exact approach requires more function evaluations. Furthermore, unlike BAIS+L, EBAIS+L must update sampling chains in sequence. Nevertheless, EBAIS+L provides an important step towards practical Bayesian independence sampling with a proposal distribution that can have multiple local maxima.

Future Directions

The current study has identified a number of open problems regarding BAIS+L, both from theoretical and practical perspectives. The most important of these is the unknown nature of the stationary distribution in the limit of an infinite number of sampling chains. It was posited in Conjecture 1 that this limiting stationary distribution is equal to the target. While the empirical results obtained in this study appear to suggest that this is the case, it is important to know the truth of this conjecture concretely.

Section 3.4 also provided two other conjectures with implications on the mixing rate of BAIS+L. These were in relation to possible diminishing adaptation of the proposal parameters with time and the consequential decoupling of an infinite number of sampling chains. While the empirical results for a finite number of sampling chains in Chapter 4 lent support to these claims, a more theoretical study will also be required. Thus, future work should look to determine sufficient and necessary conditions, if any, to ensure the truth of these conjectures, as well as bounds on the rates of convergence of the parameters.

Future work should also consider more closely, the effect that the chosen kernel in the mixture proposal has on sampling performance, both for BAIS+L and EBAIS+L. As noted in Chapter 4, particular focus should be placed on understanding the effect on sampling performance in the tails of the target distribution

Chapter 5 highlighted that the need for a tailored proposal distribution can be difficult to satisfy in practice. As such, future work should study the roles of different prior distributions in a variety of applications. In particular, in the case of spin glasses, an automated approach to selecting hyperparameters may be necessary for any practical application of BAIS+L, due to the need for their careful tailoring to the couplings, as well as the need to do so for large numbers of disorder samples. A possible avenue for such automation may come in the case of an adaptive prior distribution on the proposal parameters.

For example, the instantaneous acceptance rate at a given iteration or seg-

ment of the generated Markov chain may be used to guide adaptation of the prior. Alternatively, the prior from one iteration could be set to the posterior of the one preceding it. A combination of these two approaches is also worth considering, where the prior parameters are "shifted" towards the posterior parameters according to acceptance rate.

Section 5 also introduced an attempt at a cluster proposal for spin glasses. In the current study there was insufficient time to explore the details of the prior and posterior distributions of the proposal parameters of such an approach. However, future work should look into these more carefully, as such a proposal distribution would take into account the correlations between neighbouring spins, unlike the multi-spin flip approach that was simulated.

Chapter 6 presented an exact version of BAIS+L. Even though it was computationally more expensive than the approximate version of BAIS+L, its accuracy motivates future study into methods to make such a method more parallel.

Finally, a possible use of BAIS+L or EBAIS+L could be in inferring the parameters of a normal mixture approximation to a given target. While this point was not studied in this dissertation, it will be natural to consider it in future work. Given that both BAIS+L and EBAIS+L adapt a proposal distribution in the form of a normal mixture, if a high steady state acceptance rate can be achieved in a simulation, analysis of the resulting chains of parameters produced by BAIS+L may be able to suggest a good approximation to the target in terms of this standard mixture. Such a scenario will be particularly useful in the case of high-dimensional targets, as a mixture approximation will be more efficient to use than the large collection of samples that would otherwise be needed to capture the shape of the target.

Bibliography

S. Abbott. *Understanding analysis*. Springer, New York, 2001.

D. H. Ackley. *A connectionist machine for genetic hillclimbing*. Kluwer Academic Publishers, Norwell, Massachusetts, 1987.

H. Akaike. Information theory and an extension of the maximum likelihood principle. In *Second International Symposium on Information Theory*, pages 267–281, Budapest, 1987. Academiai Kado.

P. W. Anderson. Localisation theory and the Cu-Mn problem: Spin glasses. *Materials Research Bulletin*, 5(8):549–554, 1970.

C. Andrieu and É. Moulines. On the ergodicity properties of some adaptive MCMC algorithms. *The Annals of Applied Probability*, 16(3):1462–1505, 2006.

J. Ashkin and E. Teller. Statistics of two-dimensional lattices with four components. *Physical Review*, 64(5/6):178–184, 1943.

Y. F. Atchadé and J. S. Rosenthal. On adaptive Markov chain Monte Carlo algorithms. *Bernoulli*, 11(5):815–828, 2005.

Y. F. Atchadé, G. Fort, É. Moulines, and P. Priouret. Adaptive Markov chain Monte Carlo: Theory and methods. In D. Barber, A. T. Cemgil, and S. Chiappa, editors, *Bayesian Time Series Models*, pages 32–51. Cambridge University Press, Cambridge, UK, 2011.

A. Azzalini and A. Capitanio. Distributions generated by perturbation of symmetry with emphasis on a multivariate skew t-distribution. *Journal of the Royal Statistical Society, Series B (Statistical Methodology)*, 65(2):367–389, 2003.

A. Azzalini and A. Dalla Valle. The multivariate skew-normal distribution. *Biometrika*, 83(4):715–726, 1996.

J. D. Banfield and A. E. Raftery. Model-based Gaussian and non-Gaussian clustering. *Biometrics*, 49(3):803–821, 1993.

L. E. Baum. An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes. In O. Shisha, editor, *Inequalities III: Proceedings of the third symposium on inequalities*, pages 1–8, Los Angeles, 1972. Academic Press.

L. E. Baum and J. A. Eagon. An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology. *Bulletin of the American Mathematical Society*, 73(3):360–363, 1967.

L. E. Baum, T. Petrie, G. Soules, and N. Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *The Annals of Mathematical Statistics*, 41(1):164–171, 1970.

R. J. Baxter. *Exactly solved models in statistical mechanics*. Academic Press, London, 1982.

P. A. Beck. Some recent results on magnetism in alloys. *Metallurgical Transactions*, 2:2015–2024, 1971.

T. Benaglia, D. Chauveau, D. R. Hunter, and D. Young. mixtools: An R package for analyzing finite mixture models. *Journal of Statistical Software*, 32(6):1–29, 2009.

D. P. Bertsekas and J. N. Tsitsiklis. *Introduction to probability*. Athena Scientific, Belmont, Massachusetts, 2002.

J. Besag, P. Green, D. Higdon, and K. Mengersen. Bayesian computation and stochastic systems. *Statistical Science*, 10(1):3–66, 1995.

M. Betancourt. The convergence of Markov chain Monte Carlo methods: From the Metropolis method to Hamiltonian Monte Carlo. *Annalen der Physik*, 531(3):1700214, 2019.

C. Biernacki. Assessing a mixture model for clustering with the integrated completed likelihood. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(7):719–725, 2000.

P. Billingsley. *Probability and Measure*. John Wiley & Sons, 1995.

G. E. P. Box and M. E. Muller. A note on the generation of random normal deviates. *The Annals of Mathematical Statistics*, 29(2):610–611, 1958.

H. Bozdogan. Model selection and Akaike's information criterion (AIC): The general theory and its analytical extensions. *Psychometrika*, 52(3):345–370, 1987.

S. P. Brooks and A. Gelman. General methods for monitoring convergence of iterative simulations. *Journal of Computational and Graphical Statistics*, 7 (4):434–455, 1998.

V. Cannella and J. A. Mydosh. Magnetic ordering in gold-iron alloys. *Physical Review B*, 6(11):4220–4237, 1972.

O. Cappé, A. Guillin, J. M. Marin, and C. P. Robert. Population Monte Carlo. *Journal of Computational and Graphical Statistics*, 13(4):907–929, 2004.

G. Celeux, F. Forbes, C. P. Robert, and D. M. Titterington. Deviance information criteria for missing data models. *Bayesian Analysis*, 1(4):651–674, 2006.

K. S. Chan. Asymptotic behaviour of the Gibbs sampler. *Journal of the American Statistical Association*, 88(421):320–326, 1993.

S. Chib and E. Greenberg. Understanding the Metropolis-Hastings algorithm. *The American Statistician*, 49(4):327–335, 1995.

N. Chopin. A sequential particle filter method for static models. *Biometrika*, 89(3):539–551, 2002.

P. Contucci and C. Giardinà. *Perspectives on spin glasses*. Cambridge University Press, Cambridge, UK, 2013.

J. B. Conway. *Functions of one complex variable*. Springer-Verlag, New York, second edition, 1978.

R. V. Craiu and X. L. Meng. Multiprocess parallel antithetic coupling for backward and forward markov chain monte carlo. *The Annals of Statistics*, 33(2):661–697, 2005.

R. V. Craiu, J. Rosenthal, and C. Yang. Learn from thy neighbor: Parallel-chain and regional adaptive mcmc. *Journal of the American Statistical Association*, 104(488):1454–1466, 2009.

K. A. De Jong. *An analysis of the behaviour of a class of genetic adaptive systems*. PhD thesis, The University of Michigan, 1975.

P. Del Moral, J. C. Noyer, G. Rigal, and G. Salut. Traitement non-linéaire du signal par réseau particulaire. In *Quatorzième colloque GRETSI*, Juan-Les-Pins, 1993. GRETSI.

P. Del Moral, J. C. Noyer, G. Rigal, and G. Salut. Résolution particulaire et traitement non-linéaire du signal: Applications RADAR/SONAR. *Traitement du signal*, 12(4):287–301, 1995.

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B (Methodological)*, 39(1):1–38, 1977.

L. Devroye. *Non-uniform random variate generation*. Springer Science+Business Media, New York, 1986.

J. Diebolt and C. P. Robert. Estimation of finite mixture distributions through Bayesian sampling. *Journal of the Royal Statistical Society, Series B (Methodological)*, 56(2):363–375, 1994.

G. L. Dirichlet. Über die Reduction der positiven quadratischen Formen mit drei unbestimmten ganzen Zahlen. *Journal für die reine und angewandte Mathematik*, 40:209–227, 1850.

J. Dixmier. *General topology*. Springer Science+Business Media, New York, 1984.

A. Doucet, S. Godsill, and C. Andrieu. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and Computing*, 10(3):197–208, 2000.

S. Duane, A. D. Kennedy, B. J. Pendleton, and D. Roweth. Hybrid monte carlo. *Physics letters B*, 195(2):216–222, 1987.

R. Eckhardt. Stan Ulam, John von Neumann and the Monte Carlo method. *Los Alamos Science*, 15:131–141, 1987.

S. F. Edwards and P. W. Anderson. Theory of spin glasses. *Journal of Physics F: Metal Physics*, 5(5):965–974, 1975.

Z. Feng and J. Li. An adaptive independence sampler MCMC algorithm for Bayesian inferences of functions. *SIAM Journal on Scientific Computing*, 40 (3):A1301–A1321, 2018.

K. H. Fischer and J. A. Hertz. *Spin glasses*. Cambridge University Press, Cambridge, UK, 1991.

E. Forgy. Cluster analysis of multivariate data: Efficiency versus interpretability of classifications. *Biometrics*, 21(3):768–769, 1965.

G. E. Forsythe. Von Neumann's comparison method for random sampling from the normal and other distributions. *Mathematics of Computation*, 26(120): 817–826, 1972.

C. M. Fortuin and P. W. Kasteleyn. On the random-cluster model: I. Introduction and relation to other models. *Physica*, 57(4):536–564, 1972.

S. Frühwirth-Schnatter and S. Pyne. Bayesian inference for finite mixtures of univariate and multivariate skew-normal and skew-t distributions. *Biostatistics*, 11(2):317–336, 2010.

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.

A. Gelman and D. B. Rubin. Inference from iterative simulation using multiple sequences. *Statistical Science*, 7(4):457–472, 1992.

A. Gelman, G. O. Roberts, and W. R. Gilks. Efficient Metropolis jumping rules. *Bayesian statistics*, 5(42):599–608, 1996.

A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin. *Bayesian data analysis*. Chapman & Hall/CRC, Boca Raton, Florida, second edition, 2004.

S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–741, 1984.

J. Geweke. Bayesian inference in econometric models using Monte Carlo integration. *Econometrica*, 57(6):1317–1339, 1989.

J. Geweke. Evaluating the accuracy of sampling-based approaches to the calculation of posterior moments. In J. M. Bernardo, A. P. Dawid, and A. F. M Smith, editors, *Bayesian statistics 4: Proceedings of the fourth Valencia international meeting, April 15–20, 1991*, pages 169–193, Valencia, Spain, 1991. Clarendon Press, Oxford, UK.

C. J. Geyer. Markov chain Monte Carlo maximum likelihood. In *Computing Science and Statistics, Proceedings of the 23rd Symposium on the Interface*, pages 156–163, Seattle, Washington, 1991. Interface Foundation of North America.

W. R. Gilks, G. O. Roberts, and E. I. George. Adaptive direction sampling. *Journal of the Royal Statistical Society. Series D (The Statistician)*, 43(1): 179–189, 1994.

W. R. Gilks, S. Richardson, and D. J. Spiegelhalter. Introducing Markov chain Monte Carlo. In W. R. Gilks, S. Richardson, and D. J. Spiegelhalter, editors, *Markov chain Monte Carlo in practice*, pages 1–16. Chapman & Hall/CRC, Boca Raton, Florida, 1996.

P. Giordani and R. Kohn. Adaptive independent Metropolis-Hastings by fast estimation of mixtures of normals. *Journal of Computational and Graphical Statistics*, 19(2):243–259, 2010.

R. J. Glauber. Time-dependent statistics of the Ising model. *Journal of Mathematical Physics*, 4(2):294–307, 1963.

N. J. Gordon, D. J. Salmond, and A. F. M. Smith. Novel approach to nonlinear/non-Gaussian Bayesian estimation. *IEE Proceedings-F*, 140(2):107–113, 1993.

J. Gorham and L. Mackey. Measuring sample quality with stein's method. In *Advances in Neural Information Processing Systems*, pages 226–234, 2015.

P. J. Green. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82(4):711–732, 1995.

H. Haario, E. Saksman, and J. Tamminen. Adaptive proposal distribution for random walk Metropolis algorithm. *Computational Statistics*, 14(3):375–396, 1999.

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

G. Hamerly and C. Elkan. Alternatives to the k-means algorithm that find better clusterings. In *Proceedings of the Eleventh International Conference on Information and Knowledge Management*, pages 600–607, McLean, Virginia, 2002. ACM.

J. M. Hammersley and D. C Handscomb. *Monte Carlo methods*. Chapman and Hall, London, 1964.

J. M. Hammersley and K. W. Morton. Poor man's Monte Carlo. *Journal of the Royal Statistical Society. Series B (Methodological)*, 16(1):23–38, 1954.

J. E. Handschin. Monte Carlo techniques for prediction and filtering of non-linear stochastic processes. *Automatica*, 6:555–563, 1970.

J. E. Handschin and D. Q. Mayne. Monte Carlo techniques to estimate the conditional expectation in multi-stage non-linear filtering. *International Journal of Control*, 9(5):547–559, 1969.

V. Hasselblad. Estimation of parameters for a mixture of normal distributions. *Technometrics*, 8(3):431–444, 1966.

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

W. Heisenberg. Zur Theorie des Ferromagnetismus. *Zeitschrift für Physik*, 49 (9–10):619–636, 1928.

D. A. Hendrix and C. Jarzynski. A "fast growth" method of computing free energy differences. *The Journal of Chemical Physics*, 114(14):5974–5981, 2001.

L. Holden, R. Hauge, and M. Holden. Adaptive independent Metropolis-Hastings. *The Annals of Applied Probability*, 19(1):395–413, 2009.

K. Hukushima and Y. Iba. Population annealing and its application to a spin glass. *AIP Conference Proceedings*, 690(1):200–206, 2003.

K. Hukushima and K. Nemoto. Exchange Monte Carlo method and application to spin glass simulations. *Journal of the Physical Society of Japan*, 65(6): 1604–1608, 1996.

E. Ising. Beitrag zur Theorie des Ferromagnetismus. *Zeitschrift fü Physik*, 31: 253–258, 1925.

C. Jarzynski. Equilibrium free-energy differences from nonequilibrium measurements: A master-equation approach. *Physical Review E*, 56(5):5018–5035, 1997a.

C. Jarzynski. Nonequilibrium equality for free energy differences. *Physical Review Letters*, 78(14):2690–2693, 1997b.

C. Ji and S. C. Schmidler. Adaptive Markov chain Monte Carlo for Bayesian variable selection. *Journal of Computational and Graphical Statistics*, 22(3): 708–728, 2013.

A. Kechris. *Classical descriptive set theory*. Springer-Verlag, New York, 1995.

J. M. Keith and C. M. Davey. Bayesian approaches to the design of Markov chain Monte Carlo samplers. In J. Dick, F. Y. Kuo, G. W. Peters, and I. H. Sloan, editors, *Monte Carlo and quasi-Monte Carlo methods 2012*, pages 455–466. Springer, Berlin, Heidelberg, Sydney, 2013.

J. M. Keith, D. P. Kroese, and D. Bryant. A generalized Markov sampler. *Methodology and Computing in Applied Probability*, 6(1):29–53, 2004.

J. M. Keith, D. P. Kroese, and G. Y. Sofronov. Adaptive independence samplers. *Statistics and Computing*, 18(4):409–420, 2008.

C. Keribin. Consistent estimation of the order of mixture models. *Sankhyā: The Indian Journal of Statistics, Series A (1961–2002)*, 62(1):49–66, 2000.

S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

A. Kong, J. S. Liu, and W. H. Wong. Sequential imputations and Bayesian missing data problems. *Journal of the American Statistical Association*, 89 (425):278–288, 1994.

S.C. Kou, Q. Zhou, and W. H. Wong. Equi-energy sampler with applications in statistical inference and statistical mechanics. *The Annals of Statistics*, 34 (4):1581–1619, 2006.

H. A. Kramers and G. H. Wannier. Statistics of the two-dimensional ferromagnet. Part I. *Physical Review*, 60(3):252–262, 1941.

E. L. Lehmann and G. Casella. *Theory of point estimation*. Springer-Verlag, New York, second edition, 1998.

W. Lenz. Beiträge zum Verstädnis der Magnetischen. *Physikalische Zeitschrift*, 21:613–615, 1920.

W. Li and G. Lin. An adaptive importance sampling algorithm for Bayesian inversion with multimodal distributions. *Journal of Computational Physics*, 294:173–190, 2015.

F. Liang and W. H. Wong. Real-parameter evolutionary Monte Carlo with applications to Bayesian mixture models. *Journal of the American Statistical Association*, 96(454):653–666, 2001.

J. S. Liu and R. Chen. Sequential Monte Carlo methods for dynamic systems. *Journal of the American Statistical Association*, 93(443):1032–1044, 1998.

F. Llorente, L. Martino, and D. Delgado. Parallel Metropolis-Hastings coupler. *IEEE Signal Processing Letters*, 26(6):953–957, 2019.

S. P. Lloyd. Least squares quantization in PCM, 1957.

S. P. Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, IT-28(2):129–137, 1982.

J. MacQueen. Some methods for classification and analysis of multivariate observations. In L. M. Le Cam and J. Neyman, editors, *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297, Berkeley, California, 1967. University of California Press.

N. Madras and G. Slade. *The self-avoiding walk*. Birkhäuser, Boston, 1996.

E. Marinari and G. Parisi. Simulated tempering: A new Monte Carlo scheme. *Europhysics Letters*, 19(6):451–458, 1992.

L. Martino, H. Yang, D. Luengo, J. Kanniainen, and J. Corander. The FUSS algorithm: A fast universal self-tuned sampler within Gibbs, 2014.

M. Matsumoto and T. Nishimura. Mersenne twister: A 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 8(1):3–30, 1998.

G. J. McLachlan and D. Peel. *Finite mixture models*. John Wiley & Sons, New York, 2000.

K. L. Mengersen and C. P. Robert. IID sampling using self-avoiding population Monte Carlo: The pinball sampler. *Bayesian Statistics*, 7:277–292, 2003.

K. L. Mengersen and R. L. Tweedie. Rates of convergence of the Hastings and Metropolis algorithms. *The Annals of Statistics*, 24(1):101–121, 1996.

N. Metropolis. The beginning of the Monte Carlo method. *Los Alamos Science*, 15(584):125–130, 1987.

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

S. Meyn and R. L. Tweedie. *Markov chains and stochastic stability*. Springer-Verlag, London, 1993.

P. Müller. A generic approach to posterior integration and Gibbs sampling. Technical report, Purdue University, 1991.

P. Müller. Alternatives to the Gibbs sampling scheme. Technical report, Duke University, 1993.

D. E. Murnick, A. T. Fiory, and W. J. Kossler. Muon-spin depolarization in spin-glasses. *Physical Review Letters*, 36(2):100–104, 1976.

B. Narasimhan, S. G. Johnson, T. Hahn, A. Bouvier, and K. Kiêu. *cubature: Adaptive Multivariate Integration over Hypercubes*, 2018. URL https://CRAN.R-project.org/package=cubature. R package version 2.0.3.

R. M. Neal. Annealed importance sampling. *Statistics and Computing*, 11(2): 125–139, 2001.

R. M. Neal. Mcmc using hamiltonian dynamics. *Handbook of markov chain monte carlo*, 2(11):2, 2011.

L. Onsager. Crystal statistics. I. A two-dimensional model with an order-disorder transition. *Physical Review*, 65(3/4):117–149, 1944.

J. N. Onuchic, P. G. Wolynes, Z. Luthey-Schulten, and N. D. Socci. Toward an outline of the topography of a realistic protein-folding funnel. *Proceedings of the National Academy of Sciences of the United States of America*, 92(8): 3626–3630, 1995.

R. Peierls. On Ising's model of ferromagnetism. *Mathematical Proceedings of the Cambridge Philosophical Society*, 32(3):477–481, 1936.

M. Plummer, N. Best, K. Cowles, and K. Vines. CODA: Convergence diagnosis and output analysis for MCMC. *R News*, 6(1):7–11, 2006. URL http://CRAN.R-project.org/doc/Rnews/.

E. Pompe, C. Holmes, and K. Łatuszyński. A framework for adaptive MCMC targeting multimodal distributions, 2018.

C. M. Pooley, S. C. Bishop, A. Doeschl-Wilson, and G. Marion. Posterior-based proposals for speeding up Markov chain Monte Carlo. *Royal Society Open Science*, 6:190619, 2019.

T. Popoviciu. Sur les équations algébriques ayant toutes leurs racines réelles. *Mathematica (Cluj)*, 9:129–145, 1935.

R. B. Potts. Some generalized order-disorder transformations. *Mathematical Proceedings*, 48(1):106–109, 1952.

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2015. URL https://www.R-project.org/.

C. P. Robert and G. Casella. *Monte Carlo statistical methods*. Springer Science+Business Media, New York, second edition, 2004.

G. O. Roberts and W. R. Gilks. Convergence of adaptive direction sampling. *Journal of Multivariate Analysis*, 49:287–298, 1994.

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

G. O. Roberts and A. F. M Smith. Simple conditions for the convergence of the Gibbs sampler and Metropolis-Hastings algorithms. *Stochastic Processes and their Applications*, 49:207–216, 1994.

A. Rohatgi. WebPlotDigitizer, 2011. URL https://automeris.io/WebPlotDigitizer/.

J. S. Rosenthal. Convergence rates for Markov chains. *SIAM Review*, 37(3): 387–405, 1995.

D. B. Rubin. *Multiple imputation for nonresponse in surveys*. John Wiley & Sons, New York, 1987.

D. B. Rubin. Using the SIR algorithm to simulate posterior distributions. *Bayesian Statistics*, 3:395–402, 1988.

G. Schwarz. Estimating the dimension of a model. *The Annals of Statistics*, 6 (2):461–464, 1978.

D. Sejdinovic, H. Strathmann, M. L. Garcia, C. Andrieu, and A. Gretton. Kernel adaptive metropolis-hastings. In *International conference on machine learning*, pages 1665–1673, 2014.

D. Sherrington and S. Kirkpatrick. Solvable model of a spin-glass. *Physical review letters*, 35(26):1792–1795, 1975.

D. J. Spiegelhalter, N. G. Best, B. P. Carlin, and A. van der Linde. Bayesian measures of model complexity and fit. *Journal of the Royal Statistical Society, Series B (Statistical Methodology)*, 64(4):583–639, 2002.

H. E. Stanley. Dependence of critical properties on dimensionality of spins. *Physical Review Letters*, 20(12):589–592, 1968.

LLC Statisticat. LaplacesDemon: Complete environment for Bayesian inference. Bayesian-Inference.com. R package version 16.1.0, 2017. URL https://web.archive.org/web/20150206004624/http://www.bayesian-inference.com/software.

L. A. Steen and J. A. Seebach. *Counterexamples in topology*. Springer-Verlag, New York, second edition, 1978.

C. Stein. A bound for the error in the normal approximation to the distribution of a sum of dependent random variables. In *Proceedings of the Sixth Berkeley Symposium on Mathematical Statistics and Probability, Volume 2: Probability Theory*. The Regents of the University of California, 1972.

D. L. Stein and C. M. Newman. *Spin glasses and complexity*. Princeton University Press, Princeton, New Jersey, 2013.

H. Steinhaus. Sur la division des corps materiels en parties. *Bulletin de l'Académie Polonaise des Sciences, Classe III*, 4(12):801–804, 1956.

R. Storn and K. Price. Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11:341–359, 1997.

H. Strathmann, D. Sejdinovic, S. Livingstone, Z. Szabo, and A. Gretton. Gradient-free hamiltonian monte carlo with efficient kernel exponential families. In *Advances in Neural Information Processing Systems*, pages 955–963, 2015.

R. H. Swendsen and J. S. Wang. Replica Monte Carlo simulation of spin-glasses. *Physical Review Letters*, 57(21):2607–2609, 1986.

R. H. Swendsen and J. S. Wang. Nonuniversal critical dynamics in Monte Carlo simulations. *Physical Review Letters*, 58(2):86–88, 1987.

M. A. Tanner and W. H. Wong. The calculation of posterior distributions by data augmentation. *Journal of the American Statistical Association*, 82(398): 528–540, 1987.

T. Tao. *Analysis II*. Hindustan Book Agency, New Delhi, 2006.

The Free Software Foundation. Bash version 4.3, 2014. URL https://www.gnu.org/software/bash/.

G. Toulouse. Theory of the frustration effect in spin glasses: I. *Communications on Physics*, 2:115–119, 1977.

Z. van Havre, N. White, J. Rousseau, and K. Mengersen. Overfitting Bayesian mixture models with an unknown number of components. *PLoS ONE*, 10(7), 2015.

V. Černý. Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. *Journal of Optimization Theory and Applications*, 45(1):41–51, 1985.

W. N. Venables and B. D. Ripley. *Modern applied statistics with S*. Springer, New York, fourth edition, 2002.

G. Voronoi. Nouvelles applications des paramè continus à la theéorie des formes quadratiques. premier mémoire. Sur quelques propriétés des formes quadratiques positives parfaites. *Journal für die reine und angewandte Mathematik*, 133:97–178, 1908.

W. Wang, J. Machta, and H. G. Katzgraber. Population annealing: Theory and application in spin glasses. *Physical Review E*, 92(6):063307, 2015.

M. West. Bayesian kernel density estimation. Technical report, Duke University, 1990.

M. West. Approximating posterior distributions by mixture. *Journal of the Royal Statistical Society. Series B (Methodological)*, 55(2):409–422, 1993.

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

J. H. Wolfe. Pattern clustering by multivariate mixture analysis. *Multivariate Behavioural Research*, 5(3):329–350, 1970.

W. W. Wood and F. R. Parker. Monte Carlo equations of state of molecules interacting with the Lennard-Jones potential. I. A supercritical isotherm at about twice the critical temperature. *The Journal of Chemical Physics*, 27 (3):720–733, 1957.

B. Zhang. Generalized K-harmonic means – Boosting in unsupervised learning. Technical report, HP Laboratories, 2000.

B. Zhang, M. Hsu, and U. Dayal. K-harmonic means - A data clustering algorithm. Technical report, HP Laboratories, 1999.

Appendices

Appendix A

Algorithms Cited from the Literature

In all algorithms outlined in this appendix, π represents the target density or target/objective function, with support $\mathcal{X}$.

A.1 MC Algorithms

Algorithm A.1 A rejection sampler (Gelman et al., 2004, pp. 284-285)

Require:

1. Proposal density $g : \mathcal{X} \to \mathbb{R}^+$, such that:
   (a) $\int_{\mathcal{X}} g(x)dx < \infty$;
   (b) $g(x) \geq \pi(x)$ for all $x \in \mathcal{X}$;
   (c) $\pi(x)\ g(x) \leq M \in \mathbb{R}, \forall x \in \mathcal{X}$.
2. Desired number of samples $N \in \mathbb{N}$,

Ensure: N samples from the target density π.

Set $n = 0$.
while $n < N$ **do**
    Generate proposal $y \sim g$.
    Generate $u \sim \mathcal{U}(\cdot|0, 1)$.
    if $u \cdot g(y) < \pi(y)$ **then**
        Set $x^{(n)} = y$.
        Set $n = n + 1$.
    end if
end while

Algorithm A.2 Sequential Importance (Re)Sampling (Doucet et al., 2000)

Require:

1. A fixed population size N.
2. A likelihood function $\pi\left[\mathbf{y}^{(t)} \middle| \mathbf{x}^{(t)}\right]$ at time t.
3. Transition density $\pi\left[\cdot \middle| \left\{\mathbf{x}^{(i)}\right\}_{i=1}^{t}\right]$ at time t.
4. An importance sampling density $g\left[\mathbf{x}^{(t)} \middle| \left\{\mathbf{x}^{(i)}\right\}_{i=1}^{t-1}, \left\{\mathbf{y}^{(i)}\right\}_{i=1}^{t}\right]$.
5. A threshold effective number of samples N_{thresh} per time step.

Ensure:

A time series of populations and importance weights $\left[\left\{\mathbf{x}_n^{(t)}, w_n^{(t)}\right\}_{n=1}^{N}\right]_{t=1}^{\infty}$.

for Iteration $t \in \mathbb{Z}^+$ **do**
  for Sample $n \in \{1, \ldots, N\}$ **do**
    Generate $\mathbf{x}_n'^{(t)} \sim g\left[\cdot \middle| \left\{\mathbf{x}_n^{(i)}\right\}_{i=1}^{t-1}, \left\{\mathbf{y}^{(i)}\right\}_{i=1}^{t}\right]$.
    Set $w_n^{(t)} = \dfrac{w_n^{(t-1)} p\left[\mathbf{y}^{(t)} \middle| \mathbf{x}_n'^{(t)}\right] p\left[\mathbf{x}_n'^{(t)} \middle| x_{k-1}^{(t)}\right]}{g\left[\mathbf{x}_n'^{(t)} \middle| \left\{\mathbf{x}_n^{(i)}\right\}_{i=1}^{t-1}, \left\{\mathbf{y}_n^{(i)}\right\}_{i=1}^{t}\right]}$.
  end for
  for Sample $n \in \{1, \ldots, N\}$ **do**
    Set $\tilde{w}_n^{(t)} = w_n^{(t)} \Big/ \sum_{i=1}^{N} w_i^{(t)}$.
  end for
  Set $\widehat{N_{\text{eff}}^{(t)}} = N \Big/ \left\{\sum_{n=1}^{N} \left[\tilde{w}_n^{(t)}\right]^2\right\}$.
  if $N_{\text{eff}}^{(t)} \geq N_{\text{thresh}}$ **then**
    for Variable $n \in \{1, \ldots, N\}$ **do**
      Set $\mathbf{x}^{(t)} = \mathbf{x}'^{(t)}$.
    end for
  else
    for Variable $n \in \{1, \ldots, N\}$ **do**
      Generate $i \sim \text{Categorical}\left[\cdot \middle| \tilde{w}_1^{(t)}, \ldots, \tilde{w}_N^{(t)}\right]$.
      Set $\mathbf{x}_n^{(t)} = \mathbf{x}_i'^{(t)}$.
      Set $w_n^{(t)} = 1/N$.
    end for
  end if
end for

Algorithm A.3 Population Monte Carlo (Cappé et al., 2004)

Require:

1. Population size N.
2. An initial collection of K importance functions $\left\{g_1^{(0)}, \ldots, g_K^{(0)}\right\}$.
3. Initial importance function selection probabilities $\left\{p_1^{(0)}, \ldots, p_K^{(0)}\right\}$.

Ensure:

A time series of populations and importance weights $\left[\left\{\mathbf{x}_n^{(t)}, w_n^{(t)}\right\}_{n=1}^{N}\right]_{t=1}^{\infty}$.

1: **for** Iteration $t \in \mathbb{Z}^+$ **do**
2: **for** Population member $n \in \{1, \ldots, N\}$ **do**
3: Generate $z \sim$ Categorical $\left[\cdot \middle| p_1^{(t)}, \ldots, p_K^{(t)}\right]$
4: Generate $\mathbf{x}_n^{(t)} \sim g_z^{(t)}$.
5: Set $w_n^{(t)} = \pi\left[\mathbf{x}_n^{(t)}\right] \Big/ \left\{g_z^{(t)}\left[\mathbf{x}_n^{(t)}\right]\right\}$.
6: **end for**
7: Update $\left\{g_k^{(t+1)}\right\}_{k=1}^{K}$ and $\left\{p_k^{(t+1)}\right\}_{k=1}^{K}$.
8: Normalise $\left\{w_1^{(t)}, \ldots, w_N^{(t)}\right\}$.
9: Resample the population (cf. lines 15–19 of Algorithm A.2).
10: **end for**

Algorithm A.4 Sequential Monte Carlo (Liu and Chen, 1998).

Require:

1. An initial collection of N samples and their weights $\left\{x_n^{(0)}, w_n^{(0)}\right\}_{n=1}^{N}$.
2. Threshold squared coefficient of variation c_{thresh}^2.
3. Maximum resampling interval k.

Ensure:

$\left[\left\{x_n^{(t)}, w_n^{(t)}\right\}_{n=1}^{N}\right]_{t=1}^{\infty}$ satisfying $\mathbb{E}_{\pi^{(t)}}\left[f^{(t)}(x)\right] \approx \sum_{n=1}^{N} w_n^{(t)} f\left[x_n^{(t)}\right]$.

1: **for** Iteration $t \in \mathbb{Z}^+$ **do**
2: Set $\left[c_v^{(t)}\right]^2 = \text{Var}\left[\left\{w_n^{(t)}\right\}_{n=1}^{N}\right] \Big/ \left\{\mathbb{E}\left[\left\{w_n^{(t)}\right\}_{n=1}^{N}\right]\right\}^2$.
3: **if** $\left[c_v^{(t)}\right]^2 \geq c_{\text{thresh}}^2$ or $t \mod k = 0$ **then**
4: Resample the population (cf. lines 15–19 of Algorithm A.2).
5: **else**
6: Generate $x_n^{(t)} \sim g^{(t)}\left[\cdot \middle| \left\{x_n^{(i)}\right\}_{i=1}^{t-1}\right]$.
7: Set $u_n^{(t)} = \dfrac{\pi^{(t)}\left[\left\{x_n^{(t)}\right\}_{i=1}^{t}\right]}{\pi^{(t-1)}\left[\left\{x_n^{(i)}\right\}_{i=1}^{t-1}\right] g^{(t)}\left[x_n^{(t)} \middle| \left\{x_n^{(i)}\right\}_{i=1}^{t-1}\right]}$.
8: Set $w_n^{(t)} = u_n^{(t)} w_n^{(t-1)}$.
9: **end if**
10: **end for**

A.2 MCMC Algorithms

Algorithm A.5 The Metropolis-Hastings Sampler (Metropolis et al., 1953)

Require:
1. Proposal density g,
2. Initial state $x^{(0)}$.

Ensure: A chain of samples $\left[x^{(t)}\right]_{t=1}^{T}$ with stationary distribution π.

for $t \in \mathbb{N}$ **do**
  Generate $y \sim g$.
  Set $\alpha\left[x^{(t-1)}, y\right] = \min\left\{1, \dfrac{\pi(y)g(x|y,\theta)}{\pi(x)g(y|x,\theta)}\right\}$.
  Generate $u \sim \mathcal{U}(\cdot|0,1)$.
  if $u \leq \alpha\left[x^{(t-1)}, y\right]$ **then**
    Set $x^{(t)} = y$.
  else
    Set $x^{(t)} = x^{(t-1)}$.
  end if
end for

Algorithm A.6 The Gibbs Sampler (Glauber, 1963; Geman and Geman, 1984)

Require:
1. A partition $\mathbf{S} = \cup_{n=1}^{N} S_n$ of the state space indices $\{1, \ldots, p\}$, with $N \leq p$.
2. A conditional distribution $\pi\left[\{x_j\}_{j\in S_n} \middle| \{x_j\}_{j\in \mathbf{S}\setminus S_n}, \theta_n\right]$ for each $n \in \{1, \ldots, N\}$.
3. An initial state $\mathbf{x}^{(0)} = \left\{x_i^{(0)}\right\}_{i=1}^{p}$.

Ensure: A chain of samples $\left[\mathbf{x}^{(t)}\right]_{t=1}^{T}$ with stationary distribution π.

for Iteration $t \in \mathbb{Z}^+$ **do**
  Select a permutation $\mathcal{E}$ of the subvector indices $\{1, \ldots, N\}$.
  for $n \in \mathcal{E}$ **do**
    Sample $\{x_j\}_{j\in S_n}$ from $\pi\left(\cdot \middle| \{x_j\}_{j\in \mathbf{S}\setminus S_n}\right)$.
  end for
end for

A.3 Temperature-Based Algorithms

Algorithm A.7 Simulated Annealing (Kirkpatrick et al., 1983; Černý, 1985)

Require:

1. A function $\beta(T)$ decreasing monotonically with a sufficiently slow rate.
2. A stochastic optimsation method.
3. A number of steps $N(T)$ of optimisation to perform at temperature T.

Ensure: The global minimum $x_{\min}$ of π.

1: Set $T = \infty$.
2: Randomly initialise $x_{\min}$.
3: **while** $T > 1$ **do**
4: Update $x_{\min}$ via $N(T)$ optimisation steps on $\pi_T(x) = \exp[-\pi(x)/T]$.
5: Set $T = \beta(T)$.
6: **end while**

Algorithm A.8 Parallel Tempering (Geyer, 1991)/Exchange Monte Carlo (Hukushima and Nemoto, 1996).

Require:

1. A Monte Carlo method.
2. A temperature ladder $T_1 < \cdots < T_n < \cdots < T_N$.

Ensure: Sampling chains $\left\{\left[x_n^{(t)}\right]_{t=1}^{\infty}\right\}_{n=1}^{N}$, with stationary distributions $\{\exp[-\pi(x)/T_n]\}_{n=1}^{N}$.

1: **for** Iteration $t \in \mathbb{Z}^+$ **do**
2: **for** $n = 1$ to $n = N$ **do**
3: Generate $x_n^{(t)}$ using the Monte Carlo method on target $\exp[-\pi(x)/T_n]$.
4: **end for**
5: Randomly select two replicas $n, m \in \{1, \ldots, N\}$.
6: Set $p_{\text{swap}}(n, m) = \min\left\{1, \dfrac{\pi_n(x_m)\,\pi_m(x_n)}{\pi_n(x_n)\,\pi_m(x_m)}\right\}$.
7: Generate $u \sim \mathcal{U}(\cdot|0, 1)$.
8: **if** $u \leq p_{\text{swap}}(n, m)$ **then**
9: Swap states $x_n^{(t)}$ and $x_m^{(t)}$.
10: **end if**
11: **end for**

Algorithm A.9 Population Annealing (Hukushima and Iba, 2003)

Require:

1. A standard MCMC approach.
2. A function $f(x)$, whose average with respect to density $\exp[-\beta\pi(x)]$ is sought at various inverse temperatures β.
3. An annealing schedule of T increasing inverse temperatures $\beta_1 > \beta_2 > \cdots > \beta_T$.
4. $N \in \mathbb{N}$ replicas (parallel sampling chains).
5. An initial population $\left\{x_n^{(0)}\right\}_{n=1}^{N}$.
6. Initial weights $w_1^{(0)} = \cdots = w_N^{(0)} = 0$.
7. A resampling interval $M \in \mathbb{N}$.

Ensure: An average $\left\{\langle f \rangle_{\beta_t}\right\}_{t=1}^{T}$ at each β_t.

for Temperature index $t = 1$ to $t = T$ **do**
  for Replica $n = 1$ to $n = N$ **do**
    Set $w_n^{(t)} = w_n^{(i-1)} \exp\left\{-\left(\beta_t - \beta_{t-1}\right)\pi\left[x_n^{(t)}\right]\right\}$.
  end for
  for Replica $n = 1$ to $n = N$ **do**
    Set $p_n^{(t)} = w_n^{(t)} \Big/ \sum_{m=1}^{N} w_m^{(t)}$.
  end for
  if $t \mod M = 0$ **then**
    for Replica $n = 1$ to $n = N$ **do**
      Generate $m \sim$ Categorical $\left[\cdot \middle| p_1^{(t)}, \ldots, p_N^{(t)}\right]$.
      Set $x'_n = x_m^{(t)}$.
    end for
    Set $\left[x_1^{(t)}, \ldots, x_N^{(t)}\right] = (x'_n, \ldots, x'_N)$.
    Set $w_1^{(t)} = \cdots = w_N^{(t)} = 1$.
  end if
  for Replica $n = 1$ to $n = N$ **do**
    Update $x_n^{(t)}$ using the chosen MCMC approach.
  end for
end for

Algorithm A.10 The Equi-Energy Sampler (Kou et al., 2006)

Require:

1. A standard MCMC method.
2. $B \in \mathbb{N}$ burn-in iterations.
3. $R \in \mathbb{N}$ energy ring-initialisation iterations.
4. Energy truncation levels $H_0 \leq \inf_x h(x) < \cdots < H_N < H_{N+1} = \infty$.
5. Corresponding temperatures $1 = T_0 < \cdots < T_N < T_{N+1} = \infty$.
6. Initial sampling chain states $\left\{x_n^{(0)}\right\}_{n=1}^{N}$.
7. Initially empty empirical energy rings $\left[\left\{\hat{D}_n^{(k)}\right\}_{n=1}^{N}\right]_{k=1}^{N}$.
8. An equi-energy jump probability $p_{ee} \in (0,1)$.

Ensure: Sampling chains $\left[x_0^{(t)}\right]_{t=1}^{\infty}, \ldots, \left[x_N^{(t)}\right]_{t=1}^{\infty}$ with stationary distributions $\pi_0(x) = \exp\left[-\max\left\{h(x), H_0\right\}\right], \ldots, \pi_N(x) = \exp\left[-\max\left\{\frac{h(x)}{T_N}, H_N\right\}\right]$, respectively.

for Iteration $t \in \mathbb{Z}^+$ **do**
  for Energy truncation level $n = N$ to $n = 1$ **do**
    if $t > (N-n)(B+R)$ **then**
      Set $i = k$ such that $x_n^{(t-1)} \in [H_k, H_{k+1})$.
      if $n = N$ or $\hat{D}_i^{(n+1)} = \emptyset$ **then**
        Set $u = 0$.
      else
        Generate $u \sim \mathcal{U}(\cdot|0,1)$.
      end if
      if $u \geq p_{ee}$ **then**
        Perform a standard MCMC update from π_n to obtain $x_n^{(t)}$.
      else
        Generate $y \sim \mathcal{U}\left[\cdot \middle| \hat{D}_i^{(n+1)}\right]$.
        Set $\alpha = \min\left\{1, \frac{\pi_n(y)\pi_{n+1}\left[x_n^{(t-1)}\right]}{\pi_n\left[x_n^{(t-1)}\right]\pi_{n+1}(y)}\right\}$.
        Generate $u \sim \mathcal{U}(\cdot|0,1)$.
        if $u < \alpha$ **then**
          Set $x_n^{(t)} = y$.
        else
          Set $x_n^{(t)} = x_n^{(t-1)}$.
        end if
      end if
      if $t > (N-n)(B+R) + B$ **then**
        Set $i = k$ such that $x_n^{(t)} \in [H_k, H_{k+1})$.
        Set $\hat{D}_i^{(n)} = \hat{D}_i^{(n)} \cup \{x_n^{(t)}\}$.
      end if
    else
      Perform a standard MCMC update from π_n to obtain $x_n^{(t)}$.
    end if
  end for
end for

A.4 Mixture Approximation Algorithms

Algorithm A.11 Expectation maximisation (Dempster et al., 1977).

Require:

1. $K \in \mathbb{Z}^+$ mixture components.
2. Observed p-dimensional data $\mathbf{y} = (\mathbf{y}_1, \ldots, \mathbf{y}_N)$.
3. An initial set of parameters $\boldsymbol{\theta}^{(0)}$ at iteration $t = 0$.
4. A suitable metric $m(\cdot, \cdot)$ and threshold change ϵ_{thresh}.
5. A suitable maximisation method.

Ensure: A set of parameters $\boldsymbol{\theta}$ that (locally) maximises $\pi(\mathbf{y}, \mathbf{z}|\boldsymbol{\theta})$.

while $m[\theta^{(t)}, \theta^{(t+1)}] > \Delta_{\text{thresh}}$ **do**
  Set $\mathbf{z}' = \mathbb{E}[\mathbf{z}|\mathbf{y}, \boldsymbol{\theta}]$.
  Set $\boldsymbol{\theta}^{(t+1)} = \arg\max_{\boldsymbol{\theta} \in \mathcal{T}} \log \pi \left[\mathbf{y}, \mathbf{z}'|\theta^{(t)}\right]$.
  Set $t = t + 1$.
end while

Algorithm A.12 Lloyd's k-means algorithm (Lloyd, 1982)

Require:

1. Input data $\mathbf{y}_1, \ldots, \mathbf{y}_N \in \mathcal{X}$.
2. An initial partition $\left\{B_k^{(0)}\right\}_{k=1}^K$ of $\mathcal{X}$ at iteration $t = 0$.
3. A suitable metric $m(\cdot, \cdot)$ and threshold change Δ_{thresh}.
4. Initial maximum observed change $m^{(0)} = \infty$.

Ensure: A final partition $\{B_k\}_{k=1}^K$ of $\mathcal{X}$ that locally minimises $\sum_{k=1}^K I(B_k, \mathbf{q}_k)$, the sum of the moment of inertia of each B_k about its final centre $\mathbf{q}_k$.

while $m^{(t)} > \epsilon_{\text{thresh}}$ for any $i \in \{1, \ldots, K\}$ **do**
  Set $t = t + 1$.
  for Region $k = 1$ to $k = K$ **do**
    Set $\mathbf{q}_k = \int_{B_k} \mathbf{y} dF(\mathbf{y}) \Big/ \int_{\mathcal{X}} dF(\mathbf{y})$.
  end for
  for Data point $n = 1$ to $n = N$ **do**
    Assign $\mathbf{y}_n$ to region B_i such that $i = \arg\min_{k \in \{1, \ldots, K\}} m(\mathbf{y}_n, \mathbf{q}_k)$
  end for
  Set $m^{(t)} = \max_{k \in \{1, \ldots, K\}} m\left[\mathbf{q}_k^{(t)}, \mathbf{q}_k^{(t-1)}\right]$.
end while

Algorithm A.13 k-Harmonic Means Clustering (Zhang et al., 1999; Zhang, 2000)

Require:

1. Input data $\mathbf{y}_1, \ldots, \mathbf{y}_N \in \mathcal{X}$.
2. Initial cluster centres $\mathbf{q}_1^{(0)}, \ldots, \mathbf{q}_K^{(0)} \in \mathcal{X}$ at iteration $t = 0$.
3. A suitable metric $m(\cdot, \cdot)$ and threshold change Δ_{thresh}.
4. $\text{Perf}^{(0)} = \infty$ and $\Delta\text{Perf} = \infty$.

Ensure: Final cluster centres $\mathbf{q}_1, \ldots, \mathbf{q}_K \in \mathcal{X}$ that locally minimise $\sum_{n=1}^{N} K \Big/ \sum_{k=1}^{K} \left[m\left(\mathbf{y}_n, \mathbf{q}_k\right)\right]^{-1}$.

while $\Delta\text{Perf} > \Delta_{\text{thresh}}$ **do**
  Set $t = t + 1$.
  for Data point $n = 1$ to $n = N$ **do**
    Set $d_n^{(\min)} = \min_{k \in \{1,\ldots,K\}} \left\{ m\left[\mathbf{y}_n, \mathbf{q}_k^{(t-1)}\right] \right\}$.
    for Cluster $k = 1$ to $k = K$ **do**
      Set $b_k^{(n)} = \dfrac{\left[d_n^{(\min)}\right]^{p-2} \left\{ d_n^{(\min)} \Big/ m\left[\mathbf{y}_n, \mathbf{q}_k^{(t-1)}\right] \right\}^{p+2}}{\left(\sum_{k=1}^{K} \left\{ d_n^{(\min)} \Big/ m\left[\mathbf{y}_n, \mathbf{q}_k^{(t-1)}\right] \right\}^{p} \right)^2}$.
    end for
    Set $b_k = \sum_{n=1}^{N} b_k^{(n)}$.
    for Data point $n = 1$ to $n = N$ **do**
      Set $p_k^{(n)} = b_k^{(k)} \Big/ b_k$.
    end for
    for Cluster $k = 1$ to $k = K$ **do**
      Set $\mathbf{q}_k^{(t)} = \sum_{n=1}^{N} p_k^{(n)} \mathbf{y}_n$.
    end for
  end for
  Set $\text{Perf}^{(t)} = \sum_{n=1}^{N} K \Big/ \sum_{k=1} \left\{ m\left[\mathbf{y}_n, \mathbf{q}_k^{(t)}\right] \right\}^{-1}$
  Set $\Delta\text{Perf} = \left| \text{Perf}^{(t-1)} - \text{Perf}^{(t)} \right|$.
end while

Algorithm A.14 Adaptive mixture refinement for normal kernel density estimation with a common variance-covariance matrix (West, 1993).

Require:

1. An initial p-variate normal mixture approximation $g^{(0)}\left[\mathbf{x}\middle|\left\{w_k^{(0)},\mu_k^{(0)}\right\}_{k=1}^{K^{(0)}},\boldsymbol{\Sigma}^{(0)}\right] = \sum_{k=1}^{K^{(0)}} w_k^{(0)}\mathcal{N}\left[\mathbf{x}\middle|\mu_k^{(0)},\boldsymbol{\Sigma}^{(0)}\right]$ to $\pi(\mathbf{x})$ at refinement $t=0$.
2. Function $N(t)$, the required number of samples from refinement t.
3. A threshold entropy relative to uniformity $H_{\text{thresh}} > 0$.
4. Initial entropy relative to uniformity $H^{(0)} = -\infty$.

Ensure:

A refined p-variate normal mixture approximation $g\left[\mathbf{x}\middle|\{w_k,\mu_k\}_{k=1}^{K},\boldsymbol{\Sigma}\right] = \sum_{k=1}^{K} w_k\mathcal{N}\left(\mathbf{x}|\mu_k,\boldsymbol{\Sigma}\right)$ to $\pi(\mathbf{x})$.

while $\left|H^{(t)}\right| > H_{\text{thresh}}$ **do**
  Set $t = t+1$.
  Set $K^{(t)} = N(t)$.
  for $k=1$ to $k=K^{(t)}$ **do**
    Generate $\mathbf{x}_k \sim g^{(t-1)}\left[\cdot\middle|\left\{w_k,\mu_k^{(t-1)},\boldsymbol{\Sigma}_k^{(t-1)}\right\}_{k=1}^{K^{(t-1)}}\right]$.
  end for
  Set $Z = \sum_{k=1}^{K^{(t)}} \pi\left(\mathbf{x}_k\right)/\, g^{(t-1)}\left[\mathbf{x}_k\middle|\left\{w_l^{(t-1)},\mu_l^{(t-1)},\boldsymbol{\Sigma}_l^{(t-1)}\right\}_{l=1}^{K^{(t-1)}}\right]$.
  for $k=1$ to $k=K^{(t)}$ **do**
    Set $w_k^{(t)} = \dfrac{\pi\left(\mathbf{x}_k\right)}{Z\cdot g^{(t-1)}\left[\mathbf{x}_k\middle|\left\{w_l^{(t-1)},\mu_l^{(t-1)},\boldsymbol{\Sigma}_l^{(t-1)}\right\}_{l=1}^{K^{(t-1)}}\right]}$.
  end for
  Set $\bar{\mathbf{x}} = \sum_{k=1}^{K^{(t)}} w_k^{(t)}\mathbf{x}_k$.
  Set $h = \left[\dfrac{4}{K^{(t)}(1+2p)}\right]^{1/(1+4p)}$.
  Set $\boldsymbol{\Sigma}^{(t)} = h^2\sum_{k=1}^{K^{(t)}} w_k^{(t)}\left(\mathbf{x}_k-\bar{\mathbf{x}}\right)\left(\mathbf{x}_k-\bar{\mathbf{x}}\right)^{\mathrm{T}}$.
  for $k=1$ to $k=K_t$ **do**
    Set $\mu_k^{(t)} = \mathbf{x}_k^{(t)}\sqrt{1-h^2} + \bar{\mathbf{x}}^{(t)}\left(1-\sqrt{1-h^2}\right)$.
  end for
  Set $H^{(t)} = -\sum_{k=1}^{K^{(t)}} \dfrac{\log w_k^{(t)}}{\log K^{(t)}}$.
end while

Algorithm A.15 Bayesian estimation of the parameters of a skew-normal mixture distribution (Frühwirth-Schnatter and Pyne, 2010).

Require:

1. Input data $\mathbf{y} = (\mathbf{y}_1, \ldots, \mathbf{y}_N) \in \mathbb{R}^{Np}$.
2. $K \in \mathbb{Z}^+$ mixture components.
3. At iteration $t = 0$:
   (a) Initial latent allocations $\mathbf{S}^{(0)} = \left[S_1^{(0)}, \ldots, S_N^{(0)}\right] \in \{1, \ldots, K\}^N$.
   (b) Initial random effects $\mathbf{z}^{(0)} = \left[z_1^{(0)}, \ldots, z_N^{(0)}\right] \in [0, \infty)^N$.
4. A marginal component weight posterior distribution $p_{\mathcal{SNM}}(\mathbf{d} \,|\mathbf{S}, \mathbf{z}, \mathbf{y})$.
5. A marginal component posterior distribution $p_{\mathcal{SNM}}(\boldsymbol{\xi}, \boldsymbol{\Sigma}, \boldsymbol{\psi} \,|\mathbf{S}, \mathbf{z}, \mathbf{y})$.

Ensure: A Markov chain $\left[\mathbf{d}^{(t)}, \boldsymbol{\xi}^{(t)}, \boldsymbol{\Omega}^{(t)}, \boldsymbol{\alpha}^{(t)}\right]_{t=0}^{T}$ of inferred parameters of a skew-normal kernel density estimate of $\mathbf{y}$.

1: **for** Iteration $t \in \mathbb{Z}^+$. **do**
Generate $\mathbf{d}^{(t)} \sim p_{\mathcal{SNM}}\left[\cdot \,\middle|\, \mathbf{S}^{(t-1)}, \mathbf{z}^{(t-1)}, \mathbf{y}\right]$.
2: Generate $\boldsymbol{\xi}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \boldsymbol{\psi}^{(t)} \sim p_{\mathcal{SNM}}\left[\cdot \,\middle|\, \mathbf{z}^{(t-1)}, \mathbf{S}^{(t-1)}, \mathbf{y}\right]$.
3: **for** Component $k = 1$ to $k = K$ **do**
4: Set $\boldsymbol{\Omega}_k^{(t)} = \boldsymbol{\Sigma}_k^{(t)} + \boldsymbol{\psi}_k^{(t)} \left[\boldsymbol{\psi}_k^{(t)}\right]^{\mathrm{T}}$.
5: Set $\mathbf{W} = \sqrt{\mathrm{Diag}\left[\boldsymbol{\Omega}_k^{(t)}\right]}$.
6: Set $\boldsymbol{\alpha}_k^{(t)} = \dfrac{1}{\sqrt{1 - \left[\boldsymbol{\psi}_k^{(t)}\right]^{\mathrm{T}} \left[\boldsymbol{\Omega}_k^{(t)}\right]^{-1} \boldsymbol{\psi}_k^{(t)}}} \mathbf{W} \left[\boldsymbol{\Omega}_k^{(t)}\right]^{-1} \boldsymbol{\psi}_k^{(t)}$.
7: **end for**
8: Generate $\mathbf{S}^{(t)}, \mathbf{z}^{(t)} \sim p_{\mathcal{SNM}}\left[\cdot \,\middle|\, \boldsymbol{\xi}^{(t)}, \boldsymbol{\psi}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \mathbf{d}^{(t)}, \mathbf{y}\right]$.
9: **end for**

Algorithm A.16 Bayesian estimation of the parameters of a skew-t mixture distribution (Frühwirth-Schnatter and Pyne, 2010).

Require:

1. Input data $\mathbf{y} = (\mathbf{y}_1, \ldots, \mathbf{y}_N) \in \mathbb{R}^{Np}$.
2. $K \in \mathbb{Z}^+$ mixture components.
3. At iteration $t = 0$:
   (a) Initial latent allocations $\mathbf{S}^{(0)} = \left[S_1^{(0)}, \ldots, S_N^{(0)}\right] \in \{1, \ldots, K\}^N$.
   (b) Initial random effects:
      i. $\mathbf{z}^{(0)} = \left[z_1^{(0)}, \ldots, z_N^{(0)}\right] \in [0, \infty)^N$.
      ii. $\mathbf{b}^{(0)} = \left[b_1^{(0)}, \ldots, b_N^{(0)}\right] \in (0, \infty)^N$.
4. A marginal component weight posterior distribution $p_{\mathcal{STM}}(\mathbf{d} \,|\mathbf{S}, \mathbf{z}, \mathbf{b}, \mathbf{y})$.
5. A marginal component posterior distribution $p_{\mathcal{STM}}(\boldsymbol{\xi}, \boldsymbol{\Sigma}, \boldsymbol{\psi} \,|\mathbf{S}, \mathbf{z}, \mathbf{b}, \mathbf{y})$.
6. A suitable MCMC method for updating $\boldsymbol{\nu}$ and $\mathbf{b}$.

Ensure: A Markov chain $\left[\mathbf{d}^{(t)}, \boldsymbol{\xi}^{(t)}, \boldsymbol{\Omega}^{(t)}, \boldsymbol{\alpha}^{(t)}\boldsymbol{\nu}^{(t)}\right]_{t=0}^{T}$ of inferred parameters of a skew-t kernel density estimate of $\mathbf{y}$.

for Iteration $t \in \mathbb{Z}^+$. **do**
  Generate $\mathbf{d}^{(t)} \sim p_{\mathcal{STM}}\left[\cdot \,\middle|\, \mathbf{S}^{(t-1)}, \mathbf{z}^{(t-1)}, \mathbf{b}^{(t-1)}, \mathbf{y}\right]$.
  Generate $\boldsymbol{\xi}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \boldsymbol{\psi}^{(t)} \sim p_{\mathcal{STM}}\left[\cdot \,\middle|\, \mathbf{z}^{(t-1)}, \mathbf{S}^{(t-1)}, \mathbf{b}^{(t-1)}, \mathbf{y}\right]$.
  for Component $k = 1$ to $k = K$ **do**
    Set $\boldsymbol{\Omega}_k^{(t)} = \boldsymbol{\Sigma}_k^{(t)} + \boldsymbol{\psi}_k^{(t)}\left[\boldsymbol{\psi}_k^{(t)}\right]^{\mathrm{T}}$.
    Set $\mathbf{W} = \sqrt{\mathrm{Diag}\left[\boldsymbol{\Omega}_k^{(t)}\right]}$.
    Set $\boldsymbol{\alpha}_k^{(t)} = \dfrac{1}{\sqrt{1 - \left[\boldsymbol{\psi}_k^{(t)}\right]^{\mathrm{T}}\left[\boldsymbol{\Omega}_k^{(t)}\right]^{-1}\boldsymbol{\psi}_k^{(t)}}}\mathbf{W}\left[\boldsymbol{\Omega}_k^{(t)}\right]^{-1}\boldsymbol{\psi}_k^{(t)}$.
  end for
  Generate $\mathbf{S}^{(t)}, \mathbf{z}^{(t)} \sim p_{\mathcal{STM}}\left[\cdot \,\middle|\, \boldsymbol{\xi}^{(t)}, \boldsymbol{\psi}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \mathbf{d}^{(t)}, \mathbf{b}^{(t-1)}, \mathbf{y}\right]$.
  Generate $\boldsymbol{\nu}^{(t)}, \mathbf{b}^{(t)}$ using MCMC.
end for

Algorithm A.17 A method to collapse the number of mixture components in a mixture of normal distributions from N to M with common variance-covariance matrix (West, 1993).

Require:

1. Initial approximation, $g^{(0)}(\mathbf{x}) = \sum_{k=1}^{N} w_k^{(0)} \mathcal{N}\left[\mathbf{x} \middle| \mu_k^{(0)}, \mathbf{\Sigma}\right]$,.
2. Final number of mixture components $M < N$.

Ensure: Collapsed approximation,

$$g^{(N-M)}(\mathbf{x}) = \sum_{k=1}^{M} w_k^{(N-M)} \mathcal{N}\left[\mathbf{x} \middle| \mu_k^{(N-M)}, \mathbf{\Sigma}\right].$$

for $t = 1$ to $t = N - M$ **do**
  Sort $\left\{\left[w_k^{(t-1)}, \mu_k^{(t-1)}\right]\right\}_{k=1}^{(N-t+1)}$ in ascending order of $\mathbf{w}_k^{(N-t)}$.
  for $k = 2$ to $k = N - t + 1$ **do**
    Set $\Delta_k = \left\|\mu_1^{(t-1)} - \mu_k^{(t-1)}\right\|$.
  end for
  Set $i = \arg\min_{k \in \{1,\ldots,N-t+1\}} (\Delta_k)$.
  Set $w_1^{(t)} = w_1^{(t-1)} + w_i^{(t-1)}$.
  Set $\mu_1^{(t)} = \left[w_1^{(t-1)} \mu_1^{(t-1)} + w_i^{(t-1)} \mu_i^{(t-1)}\right] \Big/ w_1^{(t)}$.
  for $k = 2$ to $k = i - 1$ **do**
    Set $\left[w_k^{(t)}, \mu_k^{(t)}\right] = \left[w_k^{(t-1)}, \mu_k^{(t-1)}\right]$..
  end for
  for $k = i + 1$ to $k = N - t$ **do**
    Set $\left[w_k^{(t)}, \mu_k^{(t)}\right] = \left[w_{k+1}^{(t-1)}, \mu_{k+1}^{(t-1)}\right]$.
  end for
end for

A.5 Adaptive MCMC Algorithms

Algorithm A.18 The Adaptive Direction Sampler (Gilks et al., 1994; Roberts and Gilks, 1994).

Require:

1. An initial set of points $\mathbf{S}^{(0)} = \left\{\mathbf{x}_n^{(0)}\right\}_{n=1}^{N} \subset \mathbb{R}^p$ at iteration $t = 0$, with $N > p$.
2. A vector-valued distribution $D_{\mathbf{v}}(\mathbf{v}|\mathbf{S})$.
3. A scalar-valued distribution $D_u(u|\mathbf{S})$.
4. A method for sampling r.

Ensure: Sets of samples $\left\{\mathbf{S}^{(t)}\right\}_{t=1}^{\infty}$ from the target distribution π.

1: **for** Iteration $t \in \mathbb{Z}^+$ **do**
2: Generate $\mathbf{x}_c^{(t)} \sim \mathcal{U}\left[\cdot \left| \mathbf{S}^{(t-1)} \right.\right]$.
3: Generate $u \sim D_u\left[\cdot \left| \mathbf{S}^{(t-1)} \right.\right]$.
4: Generate $\mathbf{v} \sim D_{\mathbf{v}}\left[\cdot \left| \mathbf{S}^{(t-1)} \right.\right]$.
5: Set $J(r) = \dfrac{\pi\left\{\mathbf{x}_c^{(t-1)} + r \cdot \left[\mathbf{v} + u\mathbf{x}_c^{(t-1)}\right]^{p-1}\right\} |1 + ru|^{p-1}}{\int_{\mathbb{R}} \pi\left\{\mathbf{x}_c^{(t-1)} + s \cdot \left[\mathbf{v} + u\mathbf{x}_c^{(t-1)}\right]^{p-1}\right\} |1 + ru|^{p-1}\, ds}$.
6: Generate $r \sim J(\cdot)$.
7: Set $\mathbf{S}^{(t)} = \mathbf{S}^{(t-1)} \cup \left\{\mathbf{x}_c^{(t-1)} + r\left[u\mathbf{x}_c^{(t-1)} + \mathbf{v}\right]\right\} \setminus \left\{\mathbf{x}_c^{(t-1)}\right\}$.
8: **end for**

Algorithm A.19 Adaptive Proposal Sampler (Haario et al., 1999).

Require:

1. Memory parameter $H \in \mathbb{Z}^+$.
2. Initial state $\mathbf{x}^{(0)} \in \mathbb{R}^p$.
3. $c_p = 2.38/\sqrt{p}$.
4. Starting proposal density $g^{(0)}(\mathbf{x})$.

Ensure: A chain of samples $\left\{\mathbf{x}^{(t)}\right\}_{t=0}^{\infty}$ from an approximation to the target π.

for Iteration $t \in \mathbb{Z}^+$ **do**
  if $t < H$ **then**
    Generate $\mathbf{y} \sim g^{(0)}$.
  else
    Set $\bar{\mathbf{x}} = \dfrac{1}{H}\sum_{i=t-H}^{t-1} \mathbf{x}^{(i)}$.
    Set $\hat{\mathbf{\Sigma}} = \dfrac{1}{H-1}\sum_{i=t-H}^{t-1} \left[\mathbf{x}^{(i)} - \bar{\mathbf{x}}\right]\left[\mathbf{x}^{(i)} - \bar{\mathbf{x}}\right]^{\mathrm{T}}$.
    Generate $\mathbf{y} \sim \mathcal{N}\left[\cdot \left| \mathbf{x}^{(t-1)}, c_p^2 \hat{\mathbf{\Sigma}}^{(t)} \right.\right]$.
  end if
  Set $\alpha = \min\left\{1, \dfrac{\pi(\mathbf{y})}{\pi\left[\mathbf{x}^{(t-1)}\right]}\right\}$.
  Generate $u \sim \mathcal{U}(\cdot|0,1)$.
  if $u \leq \alpha$ **then**
    Set $\mathbf{x}^{(t)} = \mathbf{y}$.
  else
    Set $\mathbf{x}^{(t)} = \mathbf{x}^{(t-1)}$.
  end if
end for

Algorithm A.20 Adaptive Metropolis Sampler (Haario et al., 2001).

Require:

1. Memory parameter $H \in \mathbb{Z}^+$.
2. Initial sample length $t_0 \in \mathbb{Z}^+$.
3. $c_p = 2.38/\sqrt{p}$.
4. Starting variance-covariance matrix $\mathbf{\Sigma}^{(0)}$.

Ensure: A chain of samples $\left\{\mathbf{x}^{(t)}\right\}_{t=0}^{\infty}$, with stationary distribution equal to the target π.

for Iteration $t \in \mathbb{Z}^+$ **do**
  if $t < t_0$ **then**
    Generate $\mathbf{y} \sim \mathcal{N}\left[\cdot \left| \mathbf{x}^{(t-1)}, c_p^2 \mathbf{\Sigma}^{(0)} \right.\right]$.
  else
    if $t = t_0$ **then**
      Set $\bar{\mathbf{x}}^{(t-1)} = \frac{1}{t}\sum_{i=0}^{t-1} \mathbf{x}^{(i)}$.
      Set $\mathbf{\Sigma}^{(t-1)} = \frac{1}{t-1}\sum_{i=0}^{t-1} \left[\mathbf{x}^{(i)} - \bar{\mathbf{x}}\right]\left[\mathbf{x}^{(i)} - \bar{\mathbf{x}}\right]^{\mathrm{T}} + \epsilon \mathbf{I}_p$.
    else
      Set $\bar{\mathbf{x}}^{(t-1)} = \frac{t-2}{t-1}\mathbf{x}^{(t-2)} + \frac{1}{t-1}\mathbf{x}^{(t-1)}$.
      Set $\mathbf{\Sigma}^{(t-1)} = \frac{t-2}{t-1}\mathbf{\Sigma}^{(t-2)} + \frac{c_p}{t-1}\left\{(t-1)\mathbf{x}^{(t-2)} + \left[\mathbf{x}^{(t-2)}\right]^{\mathrm{T}} \cdot \right.$
      $\left. -t\bar{\mathbf{x}}^{(t-1)}\left[\bar{\mathbf{x}}^{(t-1)}\right]^{\mathrm{T}} + \mathbf{x}^{(t-1)}\left[\mathbf{x}^{(t-1)}\right]^{\mathrm{T}} + \epsilon \mathbb{I}_p \right\}$
    end if
    Generate $\mathbf{y} \sim \mathcal{N}\left[\cdot \left| \mathbf{x}^{(t-1)}, c_p^2 \mathbf{\Sigma}^{(t-1)} \right.\right]$.
  end if
  Set $\alpha = \min\left\{1, \frac{\pi(\mathbf{y})}{\pi\left[\mathbf{x}^{(t-1)}\right]}\right\}$.
  Generate $u \sim \mathcal{U}(\cdot|0,1)$.
  if $u \leq \alpha$ **then**
    Set $\mathbf{x}^{(t)} = \mathbf{y}$.
  else
    Set $\mathbf{x}^{(t)} = \mathbf{x}^{(t-1)}$.
  end if
end for

Algorithm A.21 AIMH state update procedure (Giordani and Kohn, 2010).

Require:

1. Latest Markov chain state $\mathbf{x}^{(t-1)}$.
2. Current proposal density $q^{(t)}\left[\cdot\left|\boldsymbol{\lambda}^{(t)}\right.\right]$ with parameters $\boldsymbol{\lambda}^{(t)}$.
3. Number of accepted proposed moves N_{accepted}.
4. A sequence of acceptance ratios $\left\{\alpha^{(i)}\right\}_{i=1}^{t-1}$.
5. A window M in which to compute the $\alpha_{\min}$.

Ensure:

1. An updated Markov chain state $\mathbf{x}^{(t)}$ with acceptance ratio $\alpha^{(t)}$.
2. An minimum acceptance rate $\alpha_{\min}$.
3. An updated number of accepted moves N_{accepted}.

Generate $\mathbf{x}' \sim q^{(t)}\left[\cdot\left|\boldsymbol{\lambda}^{(t)}\right.\right]$
Set $\alpha^{(t)} = \min\left[1, \dfrac{\pi\left(\mathbf{x}'\right) q_t\left[\mathbf{x}^{(t-1)}\left|\boldsymbol{\lambda}^{(t)}\right.\right]}{\pi\left(\mathbf{x}^{(t-1)}\right) q_t\left[\mathbf{x}'\left|\boldsymbol{\lambda}^{(t)}\right.\right]}\right]$.
Generate $u \sim \mathcal{U}(\cdot|0,1)$.
if $u < \alpha^{(t)}$ **then**
    Set $\mathbf{x}^{(t)} = \mathbf{x}'$.
    Set $N_{\text{accepted}} = N_{\text{accepted}} + 1$.
else
    Set $\mathbf{x}^{(t)} = \mathbf{x}^{(t-1)}$.
end if
Set $\alpha_{\min} = \min_{i=M+1}^{t} \alpha^{(i)}$.

Algorithm A.22 AIMH proposal adaptation procedure (Giordani and Kohn, 2010).

Require:

1. A chain of samples $\left\{\mathbf{x}^{(i)}\right\}_{i=1}^{t-1}$.
2. $\omega_1, \omega_2 \in (0,1)$.
3. $\omega_2' = \omega_2/\left(1-\omega_1\right)$.
4. $k \in \mathbb{R}^+$.

Ensure:

A new proposal distribution $q^{(t)}\left[\cdot\left|\boldsymbol{\lambda}^{(t)}\right.\right]$, with parameters $\boldsymbol{\lambda}^{(t)}$ and updated $\tilde{g}_\star^{(t)}\left[\cdot\left|\tilde{\boldsymbol{\lambda}}_\star^{(t)}\right.\right]$ and $g_\star^{(t)}\left[\cdot\left|\boldsymbol{\lambda}_\star^{(t)}\right.\right]$

Construct $g_\star^{(t)}\left[\cdot\left|\boldsymbol{\lambda}_\star^{(t)}\right.\right]$ via Algorithm A.13 with inputs $\left\{x^{(i)}\right\}_{i=1}^{t-1}$.
Set $\tilde{g}_\star^{(t)}\left[\cdot\left|\tilde{\boldsymbol{\lambda}}_\star^{(t)}\right.\right] = g_\star^{(t)}\left[\cdot\left|\boldsymbol{\lambda}_\star^{(t)}\right.\right]$.
Multiply the variance-covariance matrices of $\tilde{g}_\star^{(t)}\left[\cdot\left|\tilde{\boldsymbol{\lambda}}_\star^{(t)}\right.\right]$ by k.
Set $g\left[\cdot\left|\boldsymbol{\lambda}^{(t)}\right.\right] = \omega_2' \tilde{g}_\star^{(t)}\left[\cdot\left|\tilde{\boldsymbol{\lambda}}_\star^{(t)}\right.\right] + \left(1-\omega_2'\right) g_\star^{(t)}\left[\cdot\left|\boldsymbol{\lambda}_\star^{(t)}\right.\right]$, where $\boldsymbol{\lambda}^{(t)} = \left[\tilde{\boldsymbol{\lambda}^{(t)}}_\star, \boldsymbol{\lambda}_\star^{(t)}\right]$.
Set $q^{(t)}\left[\cdot\left|\boldsymbol{\lambda}^{(t)}\right.\right] = \omega_1 g^{(0)}(\cdot) + \left(1-\omega_1\right) g\left[\cdot\left|\boldsymbol{\lambda}^{(t)}\right.\right]$.

Algorithm A.23 Adaptive independent Metropolis-Hastings (Giordani and Kohn, 2010).

Require:

Inputs:

1. Dimension $p \in \mathbb{Z}^+$ of the state space.
2. A threshold minimum acceptance rate α_{thresh}.
3. An initial mixture of normal distributions $\phi^{(0)}(\mathbf{x})$.
4. A set of update times $\mathcal{T} \subset \mathbb{N}$.
5. A window size $M \in \mathbb{N}$ for determining α_{min}.
6. A window size $L \in \mathbb{N}$ for determining $\bar{\alpha}$.

Intialisations:

1. An initially empty set $\mathcal{N}^\star$ of times n such that $N_{\text{accepted}} = 5p$.
2. Minimum acceptance rate $\alpha_{\text{min}} = 0$ in the last M iterations.
3. The number of accepted propsed states $N_{\text{accepted}} = 0$.
4. An initial state $\mathbf{x}^{(0)}$.

Ensure: A sequence of samples $\left\{\mathbf{x}^{(t)}\right\}_{t=t^{(0)}+1}^{\infty}$, from the target distribution π.

while $N_{\text{accepted}} \leq 5p$ **do**
  Generate $\mathbf{x}^{(t)}$, $\alpha^{(t)}$, α_{min} and N_{accepted} via Algorithm A.21.
  if $N_{\text{accepted}} = 5d$ **then**
    Set $\mathcal{N}^\star = \mathcal{N}^\star \cup \{t\}$.
  end if
  Set $t = t + 1$.
end while
while $\alpha_{\text{min}} \leq \alpha_{\text{thresh}}$ **do**
  Set $\bar{\alpha} = \sum_{j=t-L}^{t-1} \alpha^{(j)}$.
  if $\bar{\alpha} < \alpha_{\text{thresh}}$ **then**
    Construct $g_\star^{(t)}$, $\tilde{g}_\star^{(t)}$ and $q^{(t)}$ via Algorithm A.22.
  else
    if $\{t - t^\star | t^\star \in \mathcal{N}^\star\} \cap \mathcal{T} \neq \emptyset$ **then**
      Construct $g_\star^{(t)}$, $\tilde{g}_\star^{(t)}$ and $q^{(t)}$ via Algorithm A.22.
    else
      Set $g_\star^{(t)} = g_\star^{(t-1)}$, $\tilde{g}_\star^{(t)} = \tilde{g}_\star^{(t-1)}$ and $q^{(t)} = q^{(t-1)}$.
    end if
  end if
  Generate $\mathbf{x}^{(t)}$, $\alpha^{(t)}$, α_{min} and N_{accepted} via Algorithm A.21.
  Set $t = t + 1$.
end while
Set $\phi^{(0)} = g_t^\star$.
Set $t^{(0)} = t$.
for Iteration $t \in \{t^{(0)} + 1, \ldots\}$ **do**
  if $\{t - t^\star | t^\star \in \mathcal{N}^\star\} \cap \mathcal{T} \neq \emptyset$ **then**
    Construct $g_t^\star$, $\tilde{g}_t^\star$ and q_t via Algorithm A.22.
  else
    Set $g_\star^{(t)} = g_\star^{(t-1)}$, $\tilde{g}_\star^{(t)} = \tilde{g}_\star^{(t-1)}$ and $q^{(t)} = q^{(t-1)}$.
  end if
  Generate $\mathbf{x}^{(t)}$, $\alpha^{(t)}$, α_{min} and N_{accepted} via Algorithm A.21.
end for

Algorithm A.24 The Bayesian Adaptive Metropolis-Hastings Sampler (Keith and Davey, 2013).

Require:

1. $\left[\mathbf{x}_1^{(0)}, \ldots, \mathbf{x}_N^{(0)}\right]$ initial chain states.
2. Prior (hyper)parameters on $\boldsymbol{\theta}$.

Ensure: N chains of samples $\left\{\left[\mathbf{x}_n^{(t)}\right]_{n=1}^{N}\right\}_{t=1}^{\infty}$ from the target distribution π.

for Iterations $t \in \mathbb{Z}^+$ **do**
  Set $\boldsymbol{\lambda}^{(t)} = \Lambda^{(t)}\left[\mathbf{x}_1^{(t-1)}, \ldots, \mathbf{x}_N^{(t)}\right]$.
  Generate $\boldsymbol{\theta}^{(t)} \sim p\left[\cdot \left| \boldsymbol{\lambda}^{(t)} \right.\right]$.
  for Sampling chain $n = 1$ to $n = N$ **do**
    Generate $\mathbf{y} \sim p\left[\mathbf{y} \left| \boldsymbol{\theta}^{(t)} \right.\right]$.
    Set $\boldsymbol{\lambda}_n^{(t)} = \Lambda\left[\mathbf{x}_1^{(t)}, \ldots, \mathbf{x}_{n-1}^{(t)}, \mathbf{y}, \mathbf{x}_{n+1}^{(t-1)}, \ldots, \mathbf{x}_N^{(t-1)}\right]$.
    Set $\boldsymbol{\lambda}_{n*}^{(t)} = \Lambda\left[\mathbf{x}_1^{(t)}, \ldots, \mathbf{x}_{n-1}^{(t)}, \mathbf{x}_n^{(t-1)}, \ldots, \mathbf{x}_N^{(t-1)}\right]$.
    Set $\alpha = \min\left\{1, \dfrac{\pi(\mathbf{y}) p\left[\boldsymbol{\theta}^{(t)} \left| \boldsymbol{\lambda}_n^{(t)} \right.\right] p\left[\mathbf{x}_n^{(t-1)} \left| \mathbf{y}, \boldsymbol{\theta}^{(t)} \right.\right]}{\pi\left[\mathbf{x}_n^{(t-1)}\right] p\left[\boldsymbol{\theta}^{(t)} \left| \boldsymbol{\lambda}_{n*}^{(t)} \right.\right] p\left[\mathbf{y} \left| \mathbf{x}_n^{(t-1)}, \boldsymbol{\theta}^{(t)} \right.\right]}\right\}$.
    Generate $u \sim \mathcal{U}(\cdot|0,1)$.
    if $u < \alpha$ **then**
      Set $\mathbf{x}_n^{(t)} = \mathbf{y}$.
    else
      Set $\mathbf{x}_n^{(t)} = \mathbf{x}_n^{(t-1)}$.
    end if
  end for
end for

A.6 Cluster Construction with the Swendsen-Wang Algorithm

Algorithm A.25 The Swendsen-Wang algorithm (Swendsen and Wang, 1987) for clustering a p-dimensional Ising spin glass configuration.

Require:
Input:

1. Spins $\mathbf{s} = (s_1, \ldots, s_N)$.
2. Neighbourhood system $\mathcal{V} = (\mathcal{V}_1, \ldots, \mathcal{V}_N)$.
3. A set of interactions $\mathbf{J} = \{J_{ij}\}_{\langle i,j \rangle}$ between neighbouring spins.

Initialisation: $\mathbf{B} = \{B_{ij}\}_{\langle i,j \rangle} = \{0\}^{pN}$ so that $\mathbf{C} = (1, \ldots, N)$.

Ensure:
Bond indicators $\mathbf{B} = \{B_{i,j}\}_{\langle i,j \rangle}$, defining cluster memberships $\mathbf{C}$.

for $i \in \{1, \ldots, N\}$ **do**
  for $j \in \{k : k \in \mathcal{V}_i, k > i\}$ **do**
    if $J_{ij} s_i s_j > 0$ **then**
      Set $q = 1 - \exp(-J_{ij} s_i s_j) / (k_B T)$.
      Generate $B_{i,j} \sim \text{Bernoulli}(\cdot | q)$.
      if $B_{ij} = 1$ **then**
        Set $a = \max\{C_i, C_j\}$ and $b = \min\{C_i, C_j\}$.
        for $k \in \{1, \ldots, N\}$ **do**
          if $C_k = a$ **then**
            Set $C_k = b$.
          end if
        end for
      end if
    end if
  end for
end for

Appendix B

Mixture Exponential Regression Problem Input

In the regression problem of Section 4.2 the observed values $\mathbf{y}$ that were used as input, correct to seven significant digits, were as follow:

1.309695e+01	2.430403e+03	3.261820e+03	8.636133e+05
4.466468e+00	1.783633e+04	3.534417e+00	5.663570e+02
1.863636e+00	2.036670e+03	2.202005e+02	2.196023e+04
1.575938e+01	7.928759e+01	5.939511e+03	1.143254e+01
5.394978e+00	2.401060e+03	2.699985e-01	3.190646e+00
5.627336e+04	1.638039e+01	5.576615e+01	5.560532e+03
6.985903e+03	1.150495e+01	1.978150e+02	1.777180e+02
2.289263e+05	1.201811e+01	1.028756e+02	1.761554e+03
8.231389e+04	2.293974e+06	1.041309e+03	5.940492e+00
8.427673e+00	2.580721e+02	1.014598e+04	1.329592e+03
3.671473e+02	1.752186e+00	6.615758e+05	2.961039e+03
8.065092e+03	2.436018e-02	2.083174e+02	2.268002e+04
6.682670e+04	1.314722e+04	6.712476e+04	1.889883e+02
7.711255e+03	8.109527e+00	4.662443e+02	3.429431e+02
3.677644e+01	5.264001e+05	1.247478e+02	4.836799e+01
3.491766e+01	1.626983e+04	2.923627e+02	1.832941e+02
8.834408e+01	7.045200e+05	3.112785e+01	3.272776e+03
7.195403e-01	2.418156e+02	3.306138e+00	4.401473e+00
2.318599e+03	2.572656e+04	7.118442e+01	2.246056e+00
5.210043e+02	1.629595e+02	1.019343e+02	5.487202e+00
1.138948e+02	1.514166e+01	1.003909e+01	1.360367e+03
1.747951e+03	1.377623e+02	1.648420e+06	3.393363e+02
1.272573e+01	8.224176e+00	2.879227e+01	1.714025e+04
1.280992e+04	1.824786e+03	2.855519e+05	2.790759e+04
3.484368e+01	8.519555e+04	2.049992e+02	3.894594e+00
1.617570e+04	5.543943e+04	1.059379e+06	6.198323e+01
3.340858e+05	5.570985e+02	9.234789e+05	1.220240e+03
2.446172e+04	7.467221e+00	3.930216e+00	1.421162e+00
3.746826e+03	5.782438e+02	1.046059e+04	4.162258e+04
4.771656e+05	2.196029e+04	4.031599e+02	3.428644e+01

```
1.882675e+01  1.946660e+03  8.803912e+03  1.087874e+02
8.448045e+02  2.391888e+00  7.558763e+02  1.346730e+02
4.366036e+05  7.948315e+03  5.153367e+05  4.974714e+04
7.953697e+02  2.495861e+05  1.881587e+02  2.606366e+02
6.288317e+01  1.338802e+05  2.826253e+01  2.477176e+02
1.806823e+05  2.135105e+04  4.022419e+04  2.183532e+06
6.125438e+00  1.396249e+02  1.264899e+04  5.240692e+00
1.647360e+01  4.715093e+01  4.623986e+01  2.393887e+03
1.212787e+02  5.113745e+03  5.810376e+00  1.120209e+02
1.306330e+01  2.190876e+04  6.618099e+02  8.189750e+01
4.413947e+01  4.704494e+03  1.977910e+06  1.898604e+01
1.130834e+02  4.276533e+05  3.606661e+03  5.481049e-01
9.466520e+03  7.979420e+04  1.638326e+05  3.221927e+04
1.327496e+03  4.983853e+00  5.519293e+05  1.876600e+01
4.046702e+04  2.376706e+06  3.676653e+02  1.163504e+06
3.410882e+03  2.967235e+04  3.481601e+05  1.606226e+05
8.046160e+01  1.297738e+02  5.980117e+00  2.284021e+05
1.057213e+02  6.907671e+02  9.683374e+03  2.101644e+00
9.366247e+03  4.109384e+01  1.673926e+05  1.788600e+04
3.075062e+05  1.121331e+03  4.294920e+03  1.258334e+03
2.342612e+01  1.047765e+02  9.819695e+02  1.076927e+03
8.634533e+04  6.390336e+02  1.491184e+05  9.380673e+01
7.637153e+00  1.506699e+04  2.800772e+02  1.055455e+04
1.290842e+04  3.998335e+01  6.483727e-01  3.036478e+05
8.225693e+02  1.431223e+02  1.250452e+01  3.893653e+03
1.876683e+04  4.964429e+02  1.654412e+02  1.034352e+05
1.233947e+05  1.520483e+03  1.074897e+04  1.471570e+06
2.414916e+05  3.808724e+03  1.503548e+04  1.163588e+03
6.837756e+01  6.258433e+00  1.665615e+02  1.348234e-01
1.876768e+01  5.264255e-01  6.679798e+02  8.072140e+02
3.304782e+00  1.281836e+02  2.368619e+02  9.768329e+04
1.670545e+02  1.039789e+01  6.545500e+03  8.897026e+05
1.224279e+05  6.990040e+00  1.010711e+02  1.013965e+01
4.119015e+00  1.062029e+01  5.917336e-01  8.370067e+00
2.479491e+04  6.772381e+04  9.665392e+01  1.958898e+00
3.434272e+02  7.040295e+03  4.781945e-01  7.395474e+02
2.013038e+02  8.271867e+03  9.101481e+00  9.082867e+04
1.898944e+00  3.194818e+04  1.082143e+01  5.846976e+01
1.554920e+02  9.331065e+05  1.045058e+05  4.133161e+04
1.117507e+02  1.786820e+01  7.710025e+05  2.757262e+05
5.879076e+03  1.754910e+04  7.967753e+04  8.063295e+04
5.800033e+00  1.787956e+03  2.148065e+03  2.271796e+05
1.284442e+03  1.750606e+01  4.095971e+01  3.105797e+00
9.597183e+01  1.356230e+03  5.158868e+04  9.068979e+03
1.923235e+01  5.072705e-01  1.387087e+01  1.782607e+02
2.070364e+00  6.151354e+05  2.414718e+05  3.145502e+00
7.506777e+05  6.969670e-01  4.985317e+02  5.378461e+04
2.957133e+03  3.186271e+00  1.400688e+02  5.024794e+02
1.457920e+05  8.629873e+05  4.354370e+01  2.830182e+01
3.155764e+02  8.598545e+00  4.935226e+04  1.806819e+01
```

```
1.959658e-01  4.174515e+02  2.659207e+05  9.547887e-01
1.604120e+00  2.075590e+02  3.436319e+00  1.689408e+05
2.074020e+04  3.776455e-01  2.725563e+00  1.452425e+02
6.357766e+03  1.130789e+05  4.476076e+00  3.073362e+03
1.074718e+01  2.705436e+02  7.570952e+04  2.273017e+02
6.630173e+01  2.986500e+00  1.020707e+02  2.376728e+01
2.774824e+04  9.605961e+04  3.970495e-01  4.756449e+00
9.047262e+01  7.643596e+02  4.113923e+00  3.380439e+02
2.970017e+02  1.162728e+03  1.212458e+01  1.287260e+05
3.136945e+04  1.240221e+01  3.528691e+01  1.947034e+05
5.950225e-01  2.129701e+06  1.584225e+00  2.639975e+05
8.917655e+00  1.162448e+04  2.457841e+00  1.121008e+01
5.818474e+04  4.946833e+00  2.712526e+02  8.320717e+04
3.127835e+02  3.029134e+01  8.093477e+01  5.598414e+01
9.390663e+02  1.270376e+05  4.842376e+03  7.233237e+00
1.066001e+03  3.546362e+03  5.415684e+05  2.297747e+02
1.400082e+02  1.033153e+02  4.678024e+04  1.116145e+01
8.521717e-02  2.132548e+05  9.078590e+04  5.790456e+01
1.093798e+04  2.693704e-01  3.723260e+01  6.601221e+01
2.648793e+01  1.775369e+01  6.986070e+03  3.193565e+01.
```

Their corresponing latent (unobserved) values **x** correct to seven significant digits were as follow:

```
1.183279e+00  7.063289e-01  5.572729e-01  1.847644e+00
4.842465e-01  9.674903e-01  6.041547e-01  2.427018e-01
5.339486e-01  6.755581e-01  4.359557e-01  1.252928e+00
7.493157e-01  1.283949e+00  1.271734e+00  1.894422e+00
1.034190e+00  5.977923e-01  8.226407e-01  5.077067e-01
1.500050e+00  1.814558e+00  2.127299e-01  7.911112e-01
7.398109e-01  7.555346e-01  4.030051e-01  3.070758e-01
1.398898e+00  1.040671e+00  1.193303e+00  8.879474e-01
1.745913e+00  1.950694e+00  7.480085e-01  1.643915e-01
1.126803e+00  2.011553e-02  9.334818e-01  5.362698e-01
3.001054e-01  5.658700e-01  1.986912e+00  1.238711e+00
1.049267e+00  8.883633e-01  7.762932e-01  1.578918e+00
1.438837e+00  1.401298e+00  1.514456e+00  7.865142e-01
1.021908e+00  6.631864e-03  5.064745e-01  4.719198e-01
1.500679e+00  1.594011e+00  1.945227e+00  3.267102e-02
1.584952e+00  1.376371e+00  5.398773e-01  8.347476e-02
1.552227e+00  1.769840e+00  1.350062e+00  7.548920e-01
4.080396e-01  4.369013e-01  7.955499e-01  9.138431e-01
8.247154e-01  8.575388e-01  1.846759e+00  9.368524e-02
1.053832e+00  1.768903e+00  3.546512e-01  2.510973e-01
2.343288e-01  1.690454e+00  1.174051e+00  5.330493e-01
6.350967e-01  5.559316e-02  1.891560e+00  5.651522e-01
1.026171e+00  6.704944e-01  1.038220e+00  1.106045e+00
1.459507e+00  6.056217e-01  1.734877e+00  1.290805e+00
8.718794e-01  1.552105e+00  5.442994e-01  8.899186e-02
1.081007e+00  1.670615e+00  1.956950e+00  1.317440e+00
```

```
1.749935e+00  2.803846e-01  1.792210e+00  6.745791e-01
1.254635e+00  1.258702e+00  1.182321e+00  1.850413e-01
6.488455e-01  4.110827e-01  9.429366e-01  1.179654e+00
1.765331e+00  1.232111e+00  5.747337e-01  1.248281e+00
1.236667e+00  6.884453e-01  1.643346e+00  3.742142e-01
1.146840e+00  4.171371e-01  7.330420e-01  1.482293e+00
1.559391e+00  1.235545e+00  1.556788e+00  1.631821e+00
3.121651e-01  1.459336e+00  3.129281e-01  5.491643e-01
7.976565e-01  1.354997e+00  1.088123e+00  8.863363e-01
1.649361e+00  1.229257e+00  1.564800e+00  1.985521e+00
1.250383e+00  7.622465e-01  1.031708e+00  6.313445e-01
1.203367e+00  1.783298e+00  4.672972e-01  1.416113e+00
1.848364e-01  6.382608e-01  6.797533e-02  4.833610e-01
8.572407e-02  1.266112e+00  3.698662e-01  1.932963e+00
4.367800e-01  8.941453e-01  1.848557e+00  6.037706e-02
1.318960e+00  1.990214e+00  9.386180e-01  1.672826e-01
1.255175e+00  1.440133e+00  1.828552e+00  1.232431e+00
8.856720e-01  7.137044e-01  1.833695e+00  1.345260e+00
1.435773e+00  1.970544e+00  4.550083e-01  1.954827e+00
7.910121e-01  1.044417e+00  1.723977e+00  1.636058e+00
3.660204e-01  4.030060e-01  1.282524e-01  1.633734e+00
4.196217e-01  4.951000e-01  1.294143e+00  5.544250e-01
1.309111e+00  1.667265e+00  1.781329e+00  1.234252e+00
1.766088e+00  6.483188e-01  9.596187e-01  8.379648e-01
1.972003e-02  8.840636e-01  6.652484e-01  5.614228e-01
1.449564e+00  5.481608e-01  1.499815e+00  1.244253e+00
8.449669e-03  1.493301e+00  3.526368e-01  1.554060e+00
1.235453e+00  9.796261e-01  6.257859e-02  1.501333e+00
8.056792e-01  1.515978e+00  1.781849e-01  9.470601e-01
1.202642e+00  2.627015e-01  6.497742e-01  1.771650e+00
1.724223e+00  1.013484e+00  1.373881e+00  1.930476e+00
1.733526e+00  8.833436e-01  9.035335e-01  8.468161e-01
6.342685e-03  1.813374e+00  1.671994e-02  6.663802e-01
7.717088e-01  3.520431e-01  4.781021e-01  4.065687e-01
3.643943e-01  1.681211e-02  2.364361e-01  1.535121e+00
1.966055e+00  1.170762e+00  1.083014e+00  1.875634e+00
1.577076e+00  1.284431e+00  1.499933e-01  1.251650e+00
3.760718e-01  1.233759e+00  1.810087e+00  1.249024e-01
1.261512e+00  1.339458e+00  1.146307e-01  4.141503e-02
1.894986e+00  1.002473e+00  3.417562e-01  7.981337e-01
6.207835e-01  1.014935e+00  1.100933e-01  1.781919e+00
8.528163e-01  1.351870e+00  3.463492e-01  7.896148e-01
1.285843e-01  1.780147e+00  1.698661e+00  1.479199e+00
1.863689e-01  1.196588e+00  1.919738e+00  1.592548e+00
1.060199e+00  1.321985e+00  1.526239e+00  1.531177e+00
6.182777e-01  8.593112e-01  7.898212e-01  1.527050e+00
3.641918e-01  1.613947e+00  1.104901e+00  1.568205e+00
3.125636e-01  5.078201e-01  1.461427e+00  1.570438e+00
1.243880e+00  7.199732e-01  7.034500e-01  3.721815e-01
7.407499e-01  1.656155e+00  1.822577e+00  9.002644e-01
```

```
1.806887e+00  2.399173e-01  5.798938e-01  1.603864e+00
6.989678e-01  3.537466e-01  3.329830e-01  7.264919e-01
1.540945e+00  1.860542e+00  2.179914e-01  6.072022e-01
6.310540e-01  8.343633e-02  1.576427e+00  4.193881e-02
2.733764e-01  7.212725e-01  1.554530e+00  1.624780e-01
4.653029e-01  1.944103e+00  1.182821e+00  1.358879e+00
1.183844e+00  4.876243e-01  3.103066e-01  4.008332e-01
1.112542e+00  1.433611e+00  1.002702e-01  7.718897e-01
8.997379e-01  1.709486e+00  1.379576e+00  1.939741e-01
9.718234e-01  1.308087e+00  2.971388e-01  6.734621e-01
1.335416e+00  1.356139e+00  2.545512e-01  2.254865e-01
2.895672e-01  5.075779e-01  6.374817e-01  1.868822e+00
1.883868e-01  6.566476e-01  7.603758e-01  1.447013e+00
1.256409e+00  2.197442e-01  1.062166e+00  1.623020e+00
3.527968e-01  1.977544e+00  1.468192e-01  1.860550e+00
1.166641e-01  1.745027e+00  2.465556e-01  1.762890e+00
1.762381e+00  8.716148e-01  1.910947e-01  1.932613e+00
5.218806e-01  8.463956e-01  4.040361e-01  1.044441e+00
1.577305e+00  1.266178e+00  1.010783e+00  2.638285e-01
4.919128e-01  6.737844e-01  1.649955e+00  1.244180e+00
1.243774e+00  7.676159e-01  1.126205e+00  8.619492e-01
3.120778e-01  1.906963e+00  1.584317e+00  9.164582e-01
1.042249e+00  1.183101e+00  7.939883e-01  1.783723e+00
1.257055e-01  8.055362e-01  1.003036e+00  1.007698e+00
```

Appendix C

Spin Glass Disorder Sample Couplings

The coupling strengths of the disorder samples used in the spin glass application of Chapter 5, correct to seven significant figures were as follow. Consecutively-index couplings follow each other along the same row. That is, the general form is given by the following table.

J_{12}	J_{14}	J_{23}	J_{25}
J_{31}	J_{36}	J_{45}	J_{47}
J_{56}	J_{58}	J_{64}	J_{69}
J_{78}	J_{71}	J_{89}	J_{82}
J_{97}	J_{93}		

Disorder sample 1:

4.2155540e-01	1.2453571e+00	3.8312985e-02	4.3871379e-01
-2.4332057e-01	3.8961054e-01	4.4355056e-01	1.6047072e+00
1.0809477e+00	-8.4039664e-01	1.5413163e+00	-3.7327924e-01
-1.6615210e-01	-2.9253610e-01	3.5758813e-01	1.1476499e-01
5.3943735e-01	-1.8404330e-01		

Disorder sample 2:

-7.1155727e-01	1.4977837e+00	-8.0870904e-02	-9.5194020e-01
-7.8408021e-01	4.6868541e-01	-6.4632994e-02	3.4509499e-01
1.0390932e+00	-9.3261663e-01	1.2319192e+00	7.2252520e-01
-9.1574694e-01	8.7309051e-01	1.1123359e+00	-1.4063241e+00
-5.4482539e-01	-1.0126993e-01		

Disorder sample 3:

```
 1.1419116e+00  -3.4788024e-01  -1.0856426e+00   1.7216597e-01
 1.3457303e+00  -8.9511705e-01  -1.5523596e-01  -7.6548778e-01
-4.3363387e-01   1.2644001e+00   9.9238509e-01   1.3995357e+00
 8.9541058e-01  -2.8997490e-01  -1.4878131e+00  -8.2927815e-01
 6.6655481e-01  -1.1491685e+00
```

Disorder sample 4:

```
-8.5795509e-01  -5.0879455e-01   9.6207049e-01  -9.6302296e-01
 1.2352635e+00   9.2447556e-01   2.1996264e-01   2.3396314e-01
-1.4437830e-01   7.0278832e-01  -4.9573368e-01  -1.5564514e+00
-1.2753092e+00   1.8175945e+00  -1.3287557e+00   3.1516281e-01
-1.7187183e+00  -1.2746759e+00
```

Disorder sample 5:

```
-9.9066336e-01   1.2335162e-01  -3.7065410e-01  -1.0702369e+00
 3.5338792e-01  -6.3331869e-01  -1.6036592e-01  -1.2397522e+00
 2.8336535e+00   1.0581820e+00   1.4963147e+00   1.4017751e+00
-4.5049878e-01   1.0497276e+00   1.7453192e+00  -1.8507570e+00
-9.2245706e-02  -1.5901527e+00
```