NTNU
Norwegian University of Science and Technology

Faculty of Information Technology, Mathematics and Electrical Engineering

MASTER'S THESIS

FOR

STUD.TECHN. Kristian Stormark

FACULTY OF INFORMATION TECHNOLOGY, MATHEMATICS AND ELECTRICAL ENGINEERING

NTNU

Date due: June 16, 2006

Discipline: Statistics

Title: "Multiple proposal strategies for Markov Chain Monte Carlo"

Purpose of the work: The candidate should study Markov Chain Monte Carlo methods with multiple proposals at each step of transition. In particular, the candidate should consider how ideas from Quasi Monte Carlo can be used in the design of multiple proposal methods to obtain Markov chains with improved mixing properties.

This diploma thesis is to be carried out at the Department of Mathematical Sciences under guidance of Associate Professor Håkon Tjelmeland.

Trondheim, January 20, 2006.

Trond Digernes
Instituttleder
Dept. of Mathematical Sciences

Håkon Tjelmeland
Associate Professor
Dept. of Mathematical Sciences

Preface

This Master's thesis completes my five year Master of Science program. It has been carried out at the Department of Mathematical Sciences of the Norwegian University of Science and Technology (NTNU) during the spring of 2006. The thesis represents 20 weeks of work.

I have attempted to make the presentation as plain and simple as possible, while at the same time self-contained and sufficiently precise. I have been especially thorough on the topics that I myself found the most challenging. It should be accessible for a student with some background in mathematics at the university level.

I would like to give thanks to my supervisor Associate Professor Håkon Tjelmeland for his generous guidance. His skilled, yet patient response has been of great value to my work, and his kindness towards me has been highly appreciated. I would also like to thank my fiancèe for taking care of me when I was all worn out, for her love, and for frequently taking my mind of the thesis work. Finally, I would like to thank my fellow students for constructive discussions, and the Lord for all his blessings.

This paper is inspired by the work of many authors. I have tried to give credit where credit is due. The book *Monte Carlo Statistical Methods* by Professor Christian P. Robert and Professor George Casella has been an especially useful source.

Trondheim, June 2006
Kristian Stormark

Abstract

The multiple proposal methods represent a recent simulation technique for Markov Chain Monte Carlo that allows several proposals to be considered at each step of transition. Motivated by the ideas of Quasi Monte Carlo integration, we examine how strongly correlated proposals can be employed to construct Markov chains with improved mixing properties. We proceed by giving a concise introduction to the Monte Carlo and Markov Chain Monte Carlo theory, and we supply a short discussion of the standard simulation algorithms and the difficulties of efficient sampling. We then examine two multiple proposal methods suggested in the literature, and we indicate the possibility of a unified formulation of the two methods. More essentially, we report some systematic exploration strategies for the two multiple proposals methods. In particular, we present schemes for the utilization of well-distributed point sets and maximally spread search directions. We also include a simple construction procedure for the latter type of point set. A numerical examination of the multiple proposal methods are performed on two simple test problems. We find that the systematic exploration approach may provide a significant improvement of the mixing, especially when the probability mass of the target distribution is "easy to miss" by independent sampling. For both test problems, we find that the best results are obtained with the QMC schemes. In particular, we find that the gain is most pronounced for a relatively moderate number of proposal. With fewer proposals, the properties of the well-distributed point sets will no be that relevant. For a large number of proposals, the independent sampling approach will be more competitive, since the coverage of the local neighborhood then will be better.

Contents

1 Introduction

One of the major gains of electronic information processing is the ability to solve complex computational problems by stochastic simulation. The numerical solution of a multitude of problems can then be approximated, often with adequate precision, in a relatively straightforward manner. While the concrete applications may be far from routine, the basic idea of simulation is agreeably simple: by recording an indefinite number of observations from a stochastic process, it is possible to learn everything about it, at least from a practical point of view. The methods that employ such techniques are commonly referred to as Monte Carlo (MC) methods. Catalyzed by the constant increase in computing power, the random sampling strategy has made a rather revolutionizing entry in a wide range of scientific disciplines.

In the recent decades, there has been vast development in the field of MC, and much of the progress has been with the *Markov Chain Monte Carlo* (MCMC) methods. The MCMC approach constitutes an almost universal framework for stochastic simulation, and the limitations of classical MC, with the employment of independent sample points, are therefore generally overcome. Further on, various cunning MCMC techniques have been designed to improve the mixing properties of the simulated Markov chain, in order to increase the span of feasible applications.

A somewhat different innovation is the *Quasi Monte Carlo* (QMC) methods, which has been developed to improve the accuracy of the estimates in the classical MC situation. As the manners of operation are rather dissimilar for QMC and MCMC, a deep unification of the two methodologies seems awkward. However, some of the QMC *principles* may readily be utilized for MCMC.

In this paper, we will examine how QMC ideas may be used within the realm of *multiple proposal methods*. The multiple proposal methods represent a novel MCMC strategy where several states are considered at each transition of the Markov chain. To our knowledge, there has been suggested two distinct multiple proposal methods in the literature: the method of Liu et al. (2000) and the method of Tjelmeland (2004). In addition, a modification of the method of Liu et al. (2000) has been proposed in Qin and Liu (2001). We will restrict our examination to the two original multiple proposal methods. The application of QMC point sets for the method of Liu et al. (2000) has been proposed in Craiu and Lemieux (2005).

The paper will be organized as follows. In Section 1 and 2 we will review some of the relevant background material by looking at the mathematical theory of MC and MCMC, respectively. Our presentation of the MCMC theory will in most parts follow that of Robert and Casella (2004). In Section 3, we will have a brief look at some of the standard MCMC algorithms, with a particular focus on the Metropolis-Hastings algorithm. In Section 4, we will present the challenge of sampling from a wider perspective, in order to make the possibilities and limitations of the multiple proposals methods more apparent. The actual multiple proposal methods will be introduced in Section 5, for which we will consider some suitable proposal strategies in Section 6. Numerical results for the methods on a few test cases will be presented in Section 7. In Section 8, we will provide a short summary.

We will occasionally make use of the mathematical notion of *measure*, mainly in Section 2, and some of related definitions will be given in Appendix A-1 for reference. For a more thorough treatment, we refer to Stroock (1999).

1.1 The general setting

Let X be a random variable on the probability space $(\Omega, \mathcal{F}, \mathbb{P})$ such that $X\colon \Omega \to \mathcal{S} \subseteq \mathbb{R}^d$, and let π denote the probability distribution of X over $\mathcal{S}$. That is, for each set $\mathcal{A}$ in the Borel σ-algebra over $\mathbb{R}^d$, which we will denote by $\mathcal{B}(\mathbb{R}^d)$, the probability of the event that corresponds to $X \in \mathcal{A}$ is given by

$$\pi(\mathcal{A}) = \Pr(X \in \mathcal{A}) = \mathbb{P}(\{\omega \in \Omega | X(\omega) \in \mathcal{A}\}) = \mathbb{P}(X^{-1}(\mathcal{A})).$$

In particular, we define the cumulative distribution function of X as

$$F_X(x) = \pi((-\infty, x]), \quad x \in \mathbb{R}^d,$$

where $(-\infty, x]$ denotes the minus infinity-anchored rectangles in $\mathbb{R}^d$ given by the Cartesian product of intervals bounded by the coordinates of x. That is, for $x = \left(x_{(1)}, \ldots, x_{(d)}\right)$,

$$(-\infty, x] = \prod_{i=1}^{d} \left(-\infty, x_{(i)}\right] = \left(-\infty, x_{(1)}\right] \times \cdots \times \left(-\infty, x_{(d)}\right].$$

The reason that we write $x_{(i)}$ for ith coordinate of x instead of the perhaps more natural x_i is that we want to reserve the latter for vector quantities. When $F_X(x)$ is an absolutely continuous function, the distribution π will have a density function $\pi_h(x)$ with respect to the Lebesgue measure on $\mathbb{R}^d$, in the sense that

$$\pi(\mathcal{A}) = \int_{\mathcal{A}} \pi_h(x) \mathrm{d}x.$$

Following standard convention, we will frequently use the short hand forms $X \sim \pi$ and $X \sim \pi_h$ in the meaning that X is a random variable with probability distribution π or probability density $\pi_h(x)$, respectively.

As our point of departure, we will assume that we have interest in some characteristics of the distribution π, e.g., the average vector, the marginal distributions or the probability of an associated event $\mathcal{A}$. The determination of such characteristics will typically correspond to evaluating integrals. In many practical situations, these integrals tend to be difficult or even impossible to evaluate by analytical means. Very often, the only available solution will be an empirical approach, most conveniently implemented by stochastic simulation on a computer. As with all numerical methods, it is important to asses the usefulness of such simulation methods in terms of consistency and convergence. The Law of Large Numbers (LLN) will provide the theoretical foundation for random samples from π that enjoy independence, and for samples with the special type of dependence known as *Markov dependence*, similar results may continue to hold by the Ergodic theorem. Definitions of the different types of convergence that we will refer to can be found in Appendix A-2.

1.2 The Monte Carlo approach

The term *Monte Carlo* originated in World War II as a code word for the stochastic simulations conducted at Los Almos during the development of the atomic bomb. The

code word was inspired by the famous casino in Monaco, reflecting the random nature of the methods (Metropolis, 1987). Today the term denotes various approaches for solving computational problems by means of "random sampling". Many real-life quantities of interest can be estimated in such a way, and the Monte Carlo methods have become important tools in a variety of applied fields, from computational physics, computer science and statistics to economics, medicine and sociology. In the absence of a generally recognized definition of term, we will adopt the one given by Anderson (1999).

Definition 1.2.1 (Monte Carlo)**.** *Monte Carlo is the art of approximating an expectation by the sample mean of a function of simulated random variables.*

The idea of estimating an expectation by the corresponding sample mean is intuitive, and by the law of large numbers it also has a sound mathematical basis. We will report some of the essential results below.

Let $X \colon \Omega \to \mathcal{S} \subseteq \mathbb{R}^d$ be a random variable with probability distribution π, and denote by $\pi_h(x)$ the density of π in the (absolutely) continuous case. Let $g(\cdot)$ be a mapping from $\mathbb{R}^d$ into $\mathbb{R}$. The expected value (or mean) of the random variable $g(X)$, denoted $\mathrm{E}\, g(X)$, is then given by

$$\mu = \mathrm{E}\, g(X) = \begin{cases} \int\limits_{\mathcal{S}} g(x)\pi_h(x)\mathrm{d}x & \text{in the continuous case,} \\ \sum\limits_{x \in \mathcal{S}} g(x)\pi(x) & \text{in the discrete case,} \end{cases} \tag{1}$$

provided that the left-hand side is finite. Given a sequence $X_1, X_2, \ldots$ of independent and identically distributed (i.i.d.) random variables with the same probability distribution as X, we define the (classical) Monte Carlo estimator of μ as

$$\hat{\mu}_n = \frac{1}{n} \sum_{i=1}^{n} g(X_i).$$

By the strong law of large numbers, $\hat{\mu}_n$ will converge to μ almost surely (a.s.), provided that $\mathrm{E}\,|g(X)| < \infty$. If in addition the variance of $g(X)$ is finite, then by the Central Limit Theorem (CLT), the asymptotic distribution of the approximation error, $\hat{\mu}_n - \mu$, is the normal distribution with mean 0 and standard deviation $\sqrt{\sigma^2/n}$, where σ^2 is the variance of $g(X)$. Due to the form of the limiting standard deviation of the error, the convergence rate of the MC estimator is said to be

$$O\left(\sqrt{\sigma^2/n}\right) = O\left(n^{-1/2}\right). \tag{2}$$

If the variance of $g(X)$ is unknown, it can be estimated by the sample variance,

$$\sigma^2 = \mathrm{E}\,(g(X) - \mu)^2 \approx \frac{1}{n-1} \sum_{i=1}^{n} \left(g(X_i) - \hat{\mu}_n\right)^2 = s_n^2,$$

and provided that s_n^2 converges to σ^2 in probability, the CLT continues to hold with the approximated variance, in the sense that the limiting distribution of $(\hat{\mu}_n - \mu)/\sqrt{s_n^2/n}$ is the standard normal distribution. Details on the asymptotic properties of the Monte Carlo estimator can be found in Casella and Berger (2002).

As can readily be seen from the definition (1), the connection between expectations and integrals is very strong. All integrals can be interpreted as taking the expectation of some continuous random variable and vice versa. For instance, if $f(x)$ is an integrable function on some bounded domain $\mathcal{X}$, then by definition

$$\int_{\mathcal{X}} f(x)\mathrm{d}x = \int_{\mathcal{X}} cf(x)\frac{1}{c}\,\mathrm{d}x = \mathrm{E}\,[cf(X)] = \mathrm{E}\,\varphi(X),$$

where $\varphi(X) = cf(X)$ and X is a random variable with the uniform density over $\mathcal{X}$, i.e., $c = \int_{\mathcal{X}} \mathrm{d}x$. Consequently, the MC methods can easily be used in the purpose of plain numerical integration. Such applications can be very successful, particularly in the high-dimensional cases, as the convergence rate (2) of the MC estimator is independent of the problem dimension. For this reason, the MC approach is sometimes referred to as *Monte Carlo integration*. The principle is illustrated in Figure 1 by a rather trivial example.

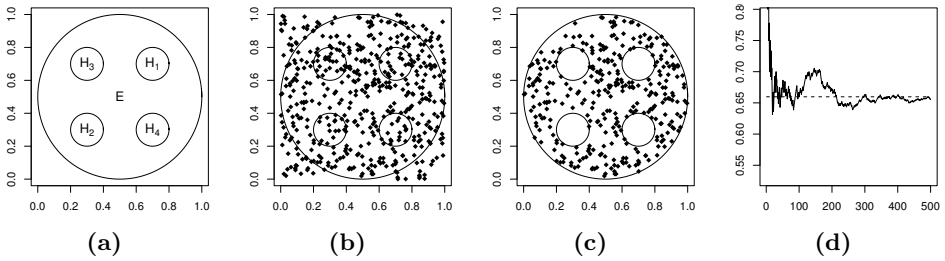


Figure 1: A simple Monte Carlo application: estimating the area of a button. **(a)** Button B consisting of the base disc E, minus the holes $H_1, \ldots, H_4$. The area of B is given by $\int_{[0,1)^2} \mathbb{I}_B(x)\mathrm{d}x$, where $\mathbb{I}_B(x)$ is the indicator function on $B = E \setminus \left(\cup_{i=1}^4 H_i\right)$. **(b)** Random sample of size $n = 500$ generated from $\mathrm{Unif}\left([0,1)^2\right)$. **(c)** Points of the random sample located on the button, that is, the points for which the indicator function is non-zero. **(d)** Area of button estimated with the MC estimator as a function of $n \in \{1, \ldots, 500\}$, that is, the cumulative mean of the random sample. The true value is indicated with a dotted line.

1.2.1 Importance sampling

For any density function $f\colon \mathbb{R}^d \to \mathbb{R}$ such that the support of $\pi_h(x)$ is a subset of the support of $f(x)$, the expectation of $g(X)$ as given by (1) can be reformulated as

$$\int_{\mathcal{S}} g(x)\pi_h(x)\mathrm{d}x = \int_{\mathcal{S}} g(x)\frac{\pi_h(x)}{f(x)}f(x)\mathrm{d}x = \int_{\mathcal{S}} g(x)w(x)f(x)\mathrm{d}x = \mathrm{E}\,\left[w(Y)g(Y)\right],$$

where Y is a random variable with distribution corresponding to the density f, and the so-called *weight function* is given as $w(x) = \pi_h(x)/f(x)$. Given an i.i.d. sample $\{Y_i\}$ from f, it is then possible to estimate $\mathrm{E}\,g(X)$ by the sample mean of $\{w(Y_i)g(Y_i)\}$, and such an estimator will converge for the same reason as the regular MC estimator. This method is known as *importance sampling* (or *weighted sampling*), and it can be useful when it is more convenient to sample from $f(x)$ than from $\pi_h(x)$. In addition, it can be used as an variance reduction technique if the function $f(x)$ is selected cleverly (Glasserman, 2003; Liu, 2001).

We have only made reference to the continuous case in order to maintain a plain presentation, but everything is analogous in the discrete case (with integrals replaced by

sums and so forth). However, it is usually straightforward to sample from the original distribution π in the first place when the sample space is discrete.

1.2.2 Quasi Monte Carlo

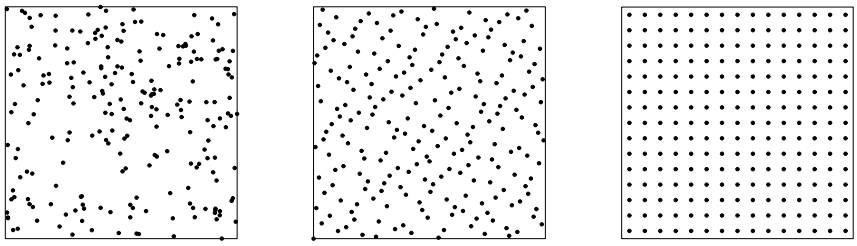
To improve on the convergence rate of classical MC, an alternative sampling methodology known as *Quasi Monte Carlo* has been developed. In the QMC setting, the problem under consideration is integration over the unit hypercube, so that the original problem has to be reformulated as an integral of the form

$$\mu = \int_{[0,1)^d} f(x)\mathrm{d}x. \tag{3}$$

The classical MC strategy would then be to sample $\{X_i\}$, $X_i \sim \text{Unif}\left([0,1)^d\right)$ and take the mean of $\{f(X_i)\}$ as the estimate of the integral. The QMC approach is seemingly quite similar, the only difference being that the random sample $\{X_i\}$ is replaced by a point set $\{y_i\}$ taken from a carefully chosen deterministic sequence in the unit hypercube. The normalized integral (3) is then approximated as

$$\int_{[0,1)^d} f(x)\mathrm{d}x \approx \frac{1}{n}\sum_{i=1}^{n} f(y_i). \tag{4}$$

The basic idea is that by employing point sets that are more uniformly distributed than the typical realizations of random samples, the distribution of the sample points will be less discrepant with the continuous uniform distribution, and the approximation correspondingly better. The principle of uniform representation is illustrated in Figure 2.



(a) **(b)** **(c)**

Figure 2: Plots of three different point sets in $[0,1)^2$, each of size 225. **(a)** points of a (pseudo-) random sample. **(b)** points from a low-discrepancy sequence. **(c)** the points of a grid. The clusters and gaps in the left plot is a result of the mutual independence of the sample points.

Even though the grid of Figure 2 may seem as an excellent configuration with respect to uniformity, it has a major drawback, namely that the size of the sample needs to be specified a priori to the computations. This inconvenience is due to the fact that if the grid structure is to be maintained throughout succeeding refinements, the number of points needed will grow exponentially. Fortunately, there exists sequences that preserve the high uniformity as the original sequence is extended with bounded-length segments.

By theoretical results related to the so-called *Koksma-Hlawka inequality*, the convergence rate of QMC approximations with such *low-discrepancy sequences* can be shown to be $O(n^{-1}\log^d(n))$, which for a fixed dimension d is $O(n^{-1+\epsilon})$ for any $\epsilon > 0$ (Niederreiter, 1992; Hickernell, 1998; Glasserman, 2003). However, the latter version of the asymptotic convergence rate may not be relevant in high dimensions, since for large d, the factor $\log^d(n)/n$ is considerably larger than $n^{-1/2}$ unless n is huge. Still, the QMC methods have been successfully applied to many high-dimensional problems with relatively moderate sample sizes, and substantial theory have been developed to account for the somewhat unexpected improvement over MC in such situations. In particular, the notion of *effective problem dimension* has been introduced (Caflisch et al., 1997).

The deterministic error bounds associated with the QMC approximations are usually useless in practice, and to be able to estimate the integration error statistically through confidence intervals and independent replications, different randomization procedures have been constructed. The principle behind such randomized QMC methods (RQMC) is to construct point sets $\{X_i\}$ where each individual point $X_i \sim \text{Unif}\left([0,1)^d\right)$, while the points collectively will remain well-distributed over the unit hypercube with probability one. A particularly simple type of randomization is the so-called *Cranley-Patterson rotation*. For a given well-distributed, deterministic point set $\{y_i\} \subseteq [0,1)^d$, the randomized version $\{X_i\}$ is then constructed by taking a rotation vector $U \sim \text{Unif}\left([0,1)^d\right)$ and take $X_i = y_i + U \mod 1$, where the modulus operator is applied element-wise. The procedure is illustrated in Figures 3 and 4.

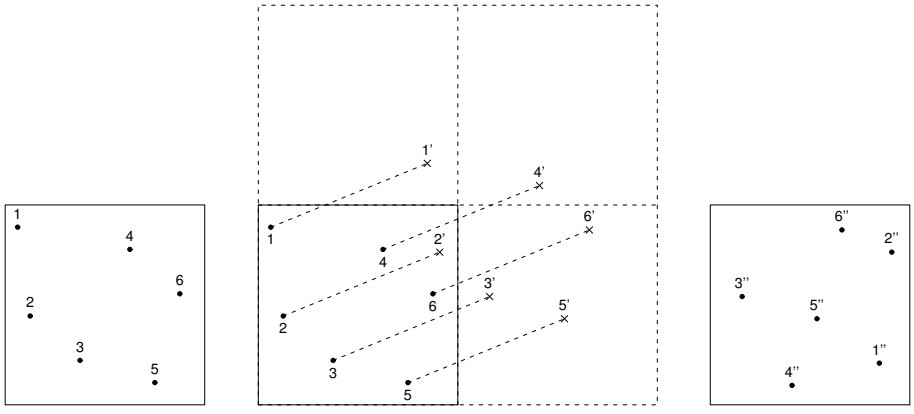
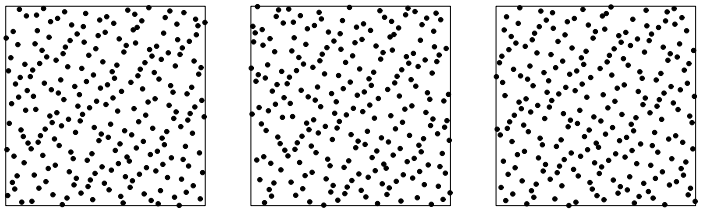


Figure 3: The principle of Cranley-Patterson rotation demonstrated on sample of size 6 in dimension 2. Each point is translated by adding the same rotation vector u, and all coordinates are then taken modulus 1, i.e., only the fractional part of each coordinate is kept. **(a)** Original points $\{y_i\} = \{1, \ldots, 6\}$. **(b)** Translated point set $\{y_i + u\} = \{1', \ldots, 6'\}$. **(c)** Rotated point set, $\{y_i + u \mod 1\} = \{1'', \ldots, 6''\}$.

To construct a suitable deterministic point set $\{y_i\}$ for the QMC approximation (4), different methods are available. The most natural setting for the analysis of such methods are number theory and abstract algebra, and a thorough treatment of the QMC theory is given by Niederreiter (1992). The two most popular families of methods are those based on *digital nets* and *lattice rules*, as described in Glasserman (2003). Examples of digital net constructions are the so-called *Halton*, *Sobol'* and *Faure sequences*, and a popular family

of lattice rule constructions is the so-called *Korobov rules*. Points take from the Halton sequence in dimension 2 is shown in Figure 4.



(a) **(b)** **(c)**

Figure 4: Randomization of a 225 points from the Halton sequence in dimension $d = 2$ by Cranley-Patterson rotation. The randomization preserves the collective distribution of the point set, while each individual point is random and uniformly distributed in the unit hypercube. **(a)** Original point set. **(b)** A randomization of the point set. **(c)** Another randomization of the point set.

We will not present any of the methods for QMC point set construction here, but refer instead to the excellent treatment given by Glasserman (2003). However, the main ideas of QMC should be accessible from the short introduction given above:

(i) it is possible to construct sequences for which certain segments will form point sets $\{x_i\}$ that, in some sense or another, will be well-distributed with respect to the uniform distribution on $[0,1)^d$,

(ii) if $f(x)$ is a transformation such that $\{f(x_i)\}$ is well-distributed with respect to the distribution π when $\{x_i\}$ is well-distributed with respect to Unif $\left([0,1)^d\right)$, then it may be beneficial to approximate the expectation of π by the sample mean of $\{f(x_i)\}$ with the x_i's taken from a low-discrepancy sequence instead of a pseudo-random sequence,

(iii) there exist simple randomization procedures for point sets $\{x_i\} \subseteq [0,1)^d$ that preserve the collective distribution property of the point set with probability one, while the marginal distribution of each individual randomized point will be Unif $\left([0,1)^d\right)$.

2 Markov Chain Monte Carlo

The classical Monte Carlo approach as described in Section 1.1 is based on the capability of generating independent realizations of the random variable X from the probability distribution π. Practically all such sampling algorithms start off with a random number generator that produces number sequences that mimic the uniform distribution on $(0, 1)$ and then either employ some sort of transformation (e.g., inversion of cumulative distribution functions) or random search strategy (e.g., rejection sampling) to generate samples from the distribution π. For non-standard distributions, this type of direct sampling is usually not possible, and at this point the MCMC methods come in handy.

In general terms, Markov Chain Monte Carlo a is device for sampling from a (multivariate) distribution π by simulating a Markov Chain (X_k) with π as its limiting distribution of states. The distribution of interest, π, is in the MCMC setting usually referred to as the *target distribution*. In Bayesian statistics, the random vector X will typically represent the unknowns and π the posterior distribution given the data. The Bayesian setting is indeed one of the major domains of MCMC, but various other applications exist. In this section, we will consider MCMC from a general point of view, briefly outlining the theory behind it. For a more thorough treatment, we refer to Robert and Casella (2004). The presentation of basic Markov chain theory in Section 2.1 may appear somewhat technical, but we will attempt to support the intuitive apprehension by supplying descriptive comments. The theory gives a formal mathematical foundation for the MCMC methods of the succeeding sections, and we will include it for completeness. However, it will not be necessary to get lost in the details.

As a shorthand notation, we will occasionally use $x \rightarrow y$ to denote transitions from x to y. In the same way as we write $\Pr(X_{k+1}|X_k)$ to emphasize the conditional relationship of the random variables X_{k+1} and X_k, we may write $f(x \rightarrow y)$ instead of $f(x, y)$ to simplify interpretation of the variables x and y as related to chain transitions. This notation should cause no confusion, and it will hopefully improve the readability.

2.1 Markov chain theory

A Markov chain is a sequence of random variables that can be thought of as a stochastic process evolving in time over a state space $\mathcal{S} \subseteq \mathbb{R}^d$. For the sequence $X_0, X_1, \ldots$ to be a Markov chain, which we will denote by (X_k), the states of the chain must enjoy the property of *Markov dependence*. That is, the distribution of the next state X_{k+1} given the present state x_k should always be independent of the prior history $(x_i)_{i<k}$ of the chain, in the sense that

$$\Pr(X_{k+1} \in \mathcal{A}|X_k = x_k, X_i = x_i, i < k) = \Pr(X_{k+1} \in \mathcal{A}|X_k = x_k), \quad k = 0, 1, 2, \ldots.$$

The transition of the chain from state X_k to the state X_{k+1} is governed by a so-called *transition kernel* K. The transition kernel is a function on $\mathcal{S} \times \mathcal{B}(\mathcal{S})$ that takes on values in $[0, 1]$, such that

$$K(x, \mathcal{A}) = \Pr(X_{k+1} \in \mathcal{A}|X_k = x) \qquad \forall x \in \mathcal{S}, \mathcal{A} \in \mathcal{B}(\mathcal{S}). \tag{5}$$

When the kernel is time-invariant in the sense that the probability of making a certain transition never depends on the position k in the chain, we say that the chain is (time)

homogeneous. More mathematically, we say that the chain (X_k) is homogeneous if the distribution of $(X_{t_1}, \ldots, X_{t_m})$ given X_{t_0} is the same as the distribution of $(X_{t_1-t_0}, \ldots, X_{t_m-t_0})$ given X_0 for every m and every $(m+1)$-tuplet $t_0 \leq t_1 \leq \ldots \leq t_m$. We will primarily consider homogeneous chains here, and the same kernel K will then govern each transition, as given by (5).

When the state space $\mathcal{S}$ is discrete, say $\mathcal{S} = \{s_1, s_2, \ldots, s_m\}$, the transition kernel can be represented as a transition matrix P with elements

$$P_{i,j} = \Pr(X_{k+1} = s_j | X_k = s_i), \quad i, j \in \{1, 2, \ldots, m\}.$$

In the continuous case, the transition kernel can be expressed in terms of a transition density. That is, there exists a function $\kappa(x, y) \equiv \kappa(x \to y)$, such that

$$K(x, \mathcal{A}) = \int_{\mathcal{A}} \kappa(x \to y) \mathrm{d}y \qquad \forall x \in \mathcal{S}, \mathcal{A} \in \mathcal{B}(\mathcal{S}).$$

In the more general setting, such measure-associated density functions are known as *Radon-Nikodym derivatives*. The so-called *iterated kernel*, K^n, which is the kernel for n transitions of the chain, is given by

$$K^n(x, \mathcal{A}) = \int_{\mathcal{S}} K^{n-1}(y, \mathcal{A}) K(x, \mathrm{d}y), \quad n > 1, \tag{6}$$

with $K^1(x, \mathcal{A}) = K(x, \mathcal{A})$, and it gives the probability that the chain, when in state x, will be in one of the states of $\mathcal{A}$ after n succeeding transitions.

The objective of all MCMC methods is to simulate a Markov chain (X_k) with a stationary distribution that allows the realizations of the chain to be treated as samples from the target distribution. To assess the topic of sample mean convergence for the Markov chains, we will need the the concepts of recurrence and irreducibility.

Definition 2.1.1 (φ-irreducible)**.** *Given a measure φ, the Markov chain (X_k) with transition kernel $K \colon \mathcal{S} \times \mathcal{B}(\mathcal{S}) \to [0, 1]$ is φ-irreducible if, for every $\mathcal{A} \in \mathcal{B}(\mathcal{S})$ with $\varphi(\mathcal{A}) > 0$, there exists an n such that $K^n(x, \mathcal{A}) > 0$ for all $x \in \mathcal{S}$.*

Qualitatively speaking, the chain is irreducible with respect to a measure φ when all the corresponding non-negligible sets have a positive probability of being visited in a finite number of transition. That is, the significant sets of the chain (in terms of φ) will remain active with positive probability. The effective sample space will therefore not be reduced as the chain evolves, and so the chain is classified as irreducible with respect to the measure φ. If the measure is a probability measure, we may simply say that the chain is irreducible.

The irreducibility property of a Markov chain (X_k) ensures that every "important" set $\mathcal{A}$ will be visited, but is not strong enough to ensure that the trajectory of the chain will enter $\mathcal{A}$ often enough to establish the necessary type of stability. We will therefor need to introduce the notion of *recurrence*. Denote by $\eta_{\mathcal{A}}$ the number of passages of (X_k) in $\mathcal{A}$, i.e.,

$$\eta_{\mathcal{A}} = \sum_{k=1}^{\infty} \mathbb{I}_{\mathcal{A}}(X_k).$$

where $\mathbb{I}_{\mathcal{A}}(x)$ is the indicator function on $\mathcal{A}$, and let E_x denote the expectation operator under the condition of fixed initial state $X_0 = x$. We then define recurrence as follows.

Definition 2.1.2 (Recurrence)**.** *A Markov Chain* (X_k) *is recurrent if*

(i) there exists a measure φ such that (X_k) is φ-irreducible, and

(ii) for every $\mathcal{A} \in \mathcal{B}(\mathcal{S})$ such that $\varphi(\mathcal{A}) > 0$, $\mathrm{E}_x[\eta_{\mathcal{A}}] = \infty$ for every $x \in \mathcal{A}$.

The recurrence property establishes that the chain *on average* will visit every significant set $\mathcal{A}$ (in terms of φ) infinitely often. An even stronger stability condition is the so-called *Harris recurrence*, which assures that *every path* of the chain will visit every such set $\mathcal{A}$ infinitely often. As with E_x, we will let $\Pr_x$ denote probability conditionally on the initial state $X_0 = x$.

Definition 2.1.3 (Harris recurrence)**.** *A set* $\mathcal{A}$ *is Harris recurrent if* $\Pr_x(\eta_{\mathcal{A}} = \infty) = 1$ *for all* $x \in \mathcal{S}$*. The chain* (X_k) *is Harris recurrent if*

(i) there exists a measure φ such that (X_k) is φ-irreducible, and

(ii) for every set $\mathcal{A}$ such that $\varphi(\mathcal{A}) > 0$, $\mathcal{A}$ is Harris recurrent.

If there exists a covering $\{S_i\}$ of $\mathcal{S}$ and a constant M such that for each i and all $x \in \mathcal{S}$ it holds that $\mathrm{E}_x[\eta_{S_i}] < M$, we say that the chain is *transient*. It can be shown that a φ-irreducible chain is either recurrent or transient (Robert and Casella, 2004). Informally speaking, we can sum up the last three definition by the following three descriptive statements.

- A irreducible chain has a non-diminishing state space.
- A recurrent chain will on average visit the different parts of the state space infinitely often.
- A Harris recurrent chain will always visit the different parts of the state space infinitely often.

For some transition kernels K there exist so-called invariant measures. More precisely, a σ-finite measure π is *invariant* for the transition kernel K (and for the associated chain) if

$$\pi(\mathcal{A}) = \int_{\mathcal{S}} K(x, \mathcal{A})\pi(\mathrm{d}x) \qquad \forall \mathcal{A} \in \mathcal{B}(\mathcal{S}).$$

When there exists an invariant *probability measure* for a φ-irreducible chain, we say that the chain is *positive*. It can be shown that any positive chain necessarily is recurrent (possibly even Harris recurrent), so that when we say that a chain is (Harris) positive it is always a positive and (Harris) recurrent chain. It can also be shown that for a recurrent chain with a σ-finite invariant measure, the measure is unique up to a multiplicative constant. Consequently, if a chain has an invariant probability measure we can speak of it in singular form. The probability measure of a positive chain is often referred to as the

stationary distribution of the chain, since $X_0 \sim \pi$ implies $X_k \sim \pi$ for every k. This can be seen from the so-called law of total probability, since

$$\begin{aligned}\Pr(X_1 \in \mathcal{A}) &= \int_{\mathcal{S}} \Pr(X_1 \in \mathcal{A}|X_0 = x) \Pr(X_0 \in \mathrm{d}x) \\ &= \int_{\mathcal{S}} \Pr(X_1 \in \mathcal{A}|X_0 = x)\pi(\mathrm{d}x) = \int_{\mathcal{S}} K(x, \mathcal{A})\pi(\mathrm{d}x) = \pi(\mathcal{A}),\end{aligned}$$

where the second equality follows from the assumption of $X_0 \sim \pi$, the third from the definition of the kernel K and the fourth from the fact that π is an invariant measure for K. We say that a Markov chain (X_k) is stationary if it has a stationary probability measure.

To establish that a probability distribution corresponds to the invariant probability measure of a Markov chain, it is usually convenient to refer to the additional stability property of *reversibility* and the so-called *detailed balance condition*.

Definition 2.1.4 (Reversibility)**.** *A stationary Markov chain (X_k) is reversible if the distribution of X_{k+1} conditionally on $X_{k+2} = x$ is the same as the distribution of X_{k+1} conditionally on $X_k = x$.*

Definition 2.1.5 (Detailed balance condition)**.** *A Markov chain with transition kernel K satisfies the detailed balance condition if there exists a measure π satisfying*

$$\int_{\mathcal{A}} K(x, \mathcal{D})\pi(dx) = \int_{\mathcal{D}} K(x, \mathcal{A})\pi(dx) \qquad \forall \mathcal{A}, \mathcal{D} \in \mathcal{B}(\mathcal{S}).$$

Simply stated, the detailed balance condition deals with the issue of whether the transitions of the chain is perfectly balanced over the sample space with respect to the distribution π. If the condition is satisfied, then for any two collections of states $\mathcal{A}$ and $\mathcal{D}$, the probability of being in $\mathcal{A}$ and making a transition to $\mathcal{D}$ is equal to the probability of being in $\mathcal{A}$ and making a transition to $\mathcal{D}$. The detailed balance is a sufficient, but not necessary, condition for π to be an invariant measure for the kernel K. The sufficient part of the previous statement follows immediately, since

$$\int_{\mathcal{S}} K(x, \mathcal{A})\pi(\mathrm{d}x) = \int_{\mathcal{A}} K(x, \mathcal{S})\pi(\mathrm{d}x) = \pi(\mathcal{A}),$$

where the last equality can be seen from the fact that $K(x, \mathcal{S}) = 1$ for all $x \in \mathcal{S}$. When K and π has densities with respect to the Lebesgue measure, i.e., $K(x, \mathrm{d}y) = \kappa(x \to y)\mathrm{d}y$ and $\pi(\mathrm{d}x) = \pi_h(x)\mathrm{d}x$, the detailed balance condition can be stated as

$$\kappa(x \to y)\pi_h(x) = \kappa(y \to x)\pi_h(y) \qquad \forall x, y \in \mathcal{S},$$

which then indicates that under π, the reverse action of any transition should be as likely as the transition itself. In other words, the dynamics of the chain should then accommodate a net "flow" of zero between any two points of the state space. It can be shown that if a Markov chain with transition density $\kappa(x \to y)$ satisfies the detailed balance with the

probability density function $\pi_h(x)$, then the chain is reversible and the density $\pi_h(x)$ is the invariant density of the chain.

We will complete this section with one more convenient classification criteria for Markov chains, namely that of periodicity.

Definition 2.1.6 (Periodicity)**.** *A φ-irreducible chain (X_k) with kernel K is periodic if there exists an integer $q \geq 2$ and a sequence of sets $S_0, S_1, \cdots, S_q \subseteq \mathcal{B}(\mathcal{S})$ with $S_q = S_0$, such that for each $i \in \{0, 1, \ldots, q-1\}$*

$$K(x, S_{i+1}) = 1 \qquad \forall x \in S_i,$$

and $\varphi(S_i) > 0$.

If a φ-irreducible chain is not periodic, we say that it is *aperiodic*. The property of aperiodicity will then assert that there are no cycles for the chain to get "trapped in".

2.2 Convergence

When considering a Markov chain (X_k), it will be highly relevant to establish the limiting distribution of its states. If we were to run the chain for a very long time, what would the probability distribution of the distant state X_k converge to? Given that the chain has a invariant probability measure π, it would seem reasonable that such a limiting distribution would coincide with π, but it is not necessarily so. There exist several results that make certain that the concurrence at least holds when some sufficient conditions are satisfied. We will report two basic results here, and refer to Robert and Casella (2004) and the references therein for a more extensive treatment.

Result 2.2.1. *If (X_k) is φ-irreducible and aperiodic with stationary distribution π, then for π-almost all $x \in \mathcal{S}$*

$$\lim_{n\to\infty} \|K^n(x, \cdot) - \pi(\cdot)\|_{TV} = 0. \tag{7}$$

In the above result, $\|\cdot\|_{TV}$ denotes the norm of total variation (see Appendix A-2), and when the limit (7) holds for a Markov chain with kernel K, we say that the chain is *ergodic*. For Harris recurrent chains, the above result takes the following, stronger form.

Result 2.2.2. *If (X_k) is Harris recurrent and aperiodic with stationary distribution π, then for every initial distributions μ,*

$$\lim_{n\to\infty} \left\| \int K^n(x, \cdot)\mu(\mathrm{d}x) - \pi(\cdot) \right\|_{TV} = 0.$$

2.3 Limit theorems

The result of the previous section establishes a condition for the limiting distribution of Markov chain (X_k) to be identical to the stationary distribution π in norm. In this section we will state a result that ensures that the realizations of a Markov chain can be employed for Monte Carlo approximation.

Given a Markov chain (X_k), let $S_n(g)$ denote be the partial sum

$$S_n(g) = \frac{1}{n}\sum_{i=1}^{n} g(X_i).$$

The next result, known as the *Ergodic theorem*, provides the theoretical basis for the convergence of $S_n(g)$ to the corresponding expectation when (X_k) is Harris positive.

Result 2.3.1 (Ergodic theorem)**.** *If (X_k) has a σ-finite invariant measure π, the following two statements are equivalent:*

(i) If $f, g \in L^1(\pi)$ with $\int g(x)\pi(\mathrm{d}x) \neq 0$, *then*

$$\lim_{n\to\infty} \frac{S_n(f)}{S_n(g)} = \frac{\int f(x)\pi(\mathrm{d}x)}{\int g(x)\pi(\mathrm{d}x)}$$

(ii) The Markov chain (X_k) is Harris recurrent.

2.4 Summary

Our treatment of Markov chain theory in this section is by no means complete. On the contrary, we have only presented a small selection of results and concepts from an extensive and complex mathematical field. In the subsequent sections we will consider MCMC from a more practical point of view, focusing on particular Markov chain simulation techniques. In that regard, we will wrap up this section by a short and practicable summary of the most relevant theory, following that of Roberts and Rosenthal (2003).

To use the states of a simulated Markov chain as a sample from π, we require that

- the chain has π as its stationary distribution (for which the satisfaction of the detailed balance condition is a sufficient, but not necessary criteria),
- the chain is φ-irreducible,
- the chain is aperiodic.

If these conditions hold, then

- the distribution of the chain will converge to π in the norm of total variation,
- $\Pr(X_k \in \mathcal{A}) \to \pi(\mathcal{A})$ for all $\mathcal{A} \in \mathcal{B}(\mathcal{S})$,
- $\mathrm{E}\, h(X_k) \to \int h(x)\pi(\mathrm{d}x)$, provided that $h \in L^1(\pi)$.

If the chain is Harris recurrent, then the above holds for all starting values $x \in \mathcal{S}$, otherwise it holds only for π-almost all.

3 Algorithm review

In this section we will introduce some general MCMC algorithms. We will start by adopting the definition of MCMC methods as given by Robert and Casella (2004).

Definition 3.0.1 (MCMC method)**.** *A Markov Chain Monte Carlo method for the simulation of a distribution π is any method producing an ergodic Markov chain (X_k) whose stationary distribution is π.*

Practically all conventional MCMC implementations are based on the following simple, but rather general algorithm. We will continue to use the short hand notation introduced in Section 2, so that $x \to y$ will denote a transition from x to y. Similarly, for a function $f(x, y)$ we may write $f(x \to y)$ to emphasize the interpretation of the variables with respect to chain transitions.

Algorithm 1 Markov chain simulation method

```
Initialize X_0 with some x_0 ∈ S
for i = 1 to n do
    x = X_{i-1}
    Generate y from proposal Q(x → y)
    Compute acceptance rate α(x → y)
    Generate u from Unif([0, 1))
    if u < α(x → y) then
        X_i = y
    else
        X_i = x
    end if
end for
```

For non-homogeneous chains, the proposal distribution Q and the acceptance rate α will in general differ from step to step, so that we could have written Q_i and α_i in the above algorithm. This would for instance be the case for the so-called *cyclic Gibb-sampler*. We will focus on homogeneous chains here, and we may therefor employ the simplified notation.

The proposal distribution $Q(x \to y)$ will roughly play the role that the trial distributions play when random variables are generated by so-called *rejection sampling* (see Liu (2001) for details), and it denotes the distribution of the proposal state Y given the present state x of the chain. For instance, $Q(x \to y)$ could be taken as a Gaussian distribution centered at x, or perhaps, simply as the uniform distribution over some domain $\mathcal{D} \supseteq \mathcal{S}$ if the sample space is bounded. The acceptance rate $\alpha(x \to y)$ gives the probability of accepting the proposed transition $x \to y$, and the whole acceptance/rejection step can thus be seen as a mechanism for weighting the importance of the transition according to the target and proposal distributions.

From this point on, we will assume that both the proposal distribution and the target distribution has an associated probability density function, which we then will denote by q and π, respectively. That is, $\pi(x)$ will denote the probability density function of X, and $q(x \to y)$ will denote the conditional density function of Y when in state x. In the previous sections, we have been careful to make a distinction between the probability

density function and the corresponding probability measure by employing the subscript π_h, but since we mainly will consider density functions from this point on, we will drop the distinction. The subsequent treatment would be analogous for the discrete case, except that the probability density functions would then be replaced by probability mass functions.

The Markov chain (X_k) generated by Algorithm 1 will have $\pi(x)$ as its limiting distribution when the detailed balance condition is satisfied. The condition

$$\pi(x)q(x\rightarrow y)\alpha(x\rightarrow y) = \pi(y)q(y\rightarrow x)\alpha(y\rightarrow x) \tag{8}$$

will then be necessary and sufficient (cf. Section 3.2.1). As can readily be seen from the listing, the sampling efficiency of the chain will exclusively be given by the specification of the proposal distribution, Q, and acceptance rate function, $\alpha(x\rightarrow y)$, and the actual design is usually of high importance. The full range of the techniques used in applied MCMC is enormous, but even so, most of the implemented methods are in fact modifications of a few generic methods designed to sample from practically any distribution. That being said, the actual computational efficiency will in general be highly dependent on problem-specific adjustments.

We will treat the so-called *Metropolis-Hastings* algorithm in more detail in Section 3.2, but we will anticipate somewhat by listing the three most popular MCMC methods (Jiang, 2003). Strictly speaking, all of them may be classified as Metropolis-Hastings algorithms, but it is common practice in the literature to make the distinction. It should also be noted that they all share a highly convenient property; the target density π needs only be known up to proportionality, since any normalizing constant will cancel out!

Metropolis-Hastings (1970) By maximization of the acceptance rate $\alpha(x \rightarrow y)$ under the restriction (8), we get

$$\alpha(x\rightarrow y) = \alpha(y\rightarrow x)\frac{\pi(y)q(y\rightarrow x)}{\pi(x)q(x\rightarrow y)} = \min\left\{1, \frac{\pi(y)q(y\rightarrow x)}{\pi(x)q(x\rightarrow y)}\right\}, \tag{9}$$

provided that $q(x\rightarrow y) > 0$ whenever $q(y\rightarrow x) > 0$.

Metropolis (1953) Taking a symmetric proposal density, i.e., $q(x\rightarrow y) = q(y\rightarrow x)$, we have the special case of (9) that

$$\alpha(x\rightarrow y) = \min\left\{1, \frac{\pi(y)}{\pi(x)}\right\}. \tag{10}$$

Clearly, the Metropolis-Hastings algorithm is a generalization of the Metropolis algorithm.

Gibbs sampler (1984) In the Gibbs sampler, the coordinates of (X_k) is updated separately. Usually, the proposed state y is identical to the current state x except at a single coordinate, say, the ith. We will write y_i to indicate that only the ith component will be updated at the given transition, and x_{-i} will denote all *but* the ith component of x. Similarly, π_{-i} will denote the marginal density of X_{-i}. Taking the proposal density to be the full conditional density such that $q(x\rightarrow y) = \pi(y_i|x_{-i}) = \pi(y)/\pi_{-i}(x_{-i})$, we then have that

$$\frac{\pi(y)q(y\rightarrow x)}{\pi(x)q(x\rightarrow y)} = \frac{\pi(y)\pi_{-i}(x_{-i})\pi(x)}{\pi(x)\pi(y)\pi_{-i}(y_{-i})} = \frac{\pi_{-i}(x_{-i})}{\pi_{-i}(y_{-i})} = 1,$$

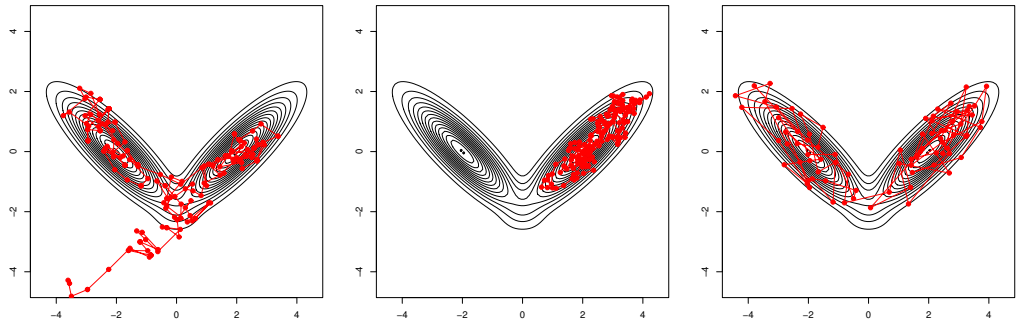
so that the acceptance rate (9) is always 1. The components of X is usually updated individually, as indicated above, or possibly in groups. The updating is either performed in a systematic manner or by random selection, with the latter case producing a homogeneous chain. We will not elaborate further, but refer instead to Barndorff-Nielsen et al. (2001).

3.1 General comments

All MCMC implementations are challenged by two generic issues, namely that of the *mixing rate* and the limited-run effect known as *burn-in*. While the mixing properties of a Markov chain relates to the way the chain explores the state space, the notion of a burn-in period refers to the fact that the simulated chain usually is started from some fixed, and then potentially improbable initial state.

If π is the stationary distribution of the chain, then $X_k \sim \pi$ implies that $X_{k+1} \sim \pi$ for all k. On the other hand, if the chain is started from some fixed state x_0 then we only have that $X_k \sim \pi$ in the limit. The amount of time the chain needs before it "hits" the stationary distribution is commonly referred to as the burn-in period, and the corresponding part of the chain is then usually discarded from the sample. Of course, the burn-in period is more of a symptomatic description than a mathematical term, and in fact, as pointed out by Geyer (2006), discarding the initial states is nothing more than a *"method, and not a particularly good method, of finding a good starting point"*. As an alternative, Geyer validates a more general heuristic, by stating that *"Any point you don't mind having in a sample is a good starting point"*. On the other hand, as Geyer is quick to point out, *"In a typical application, one has no mathematical analysis of the Markov chain that tells where the good starting points are"*. The advantage of selecting a good starting point is illustrated in Figure 5, where a rather unfortunate initial state is employed in the simulation of subfigure *(a)* . The particular state is by all means a legitimate state as it has non-zero density under the target distribution, and the chain would eventually have to visit it (or at least get arbitrarily close to it). However, in a sample of the given size, the state should not be seen. The standard burn-in solution would then be to discard the initial part of the chain. Alternatively, a better initial state could be picked, as illustrated in subfigure *(b)* . For instance, selection by local optimization from a random start point might perhaps be favorable in some situations.

The evolution of any simulated chain is governed by its transition kernel. If, for instance, the trajectory of the chain typically will make local transitions, in the sense that the next state often will be relatively close to the current state, or the state space have "bottlenecks" in the statistical sense, then it may take some time before the whole state space is visited in a balanced way. The way the chain explores the state space is commonly referred to as "the mixing" of the chain. If relatively many transitions are needed to cover the domain, we say that the chain has poor mixing, but if the chain relatively often makes radical transitions, we may say that the mixing is good. The property is illustrated in Figure 5, where the proposal of subfigure *(c)* gives better mixing than the the proposal of subfigure *(b)* . Of course, there will usually be a trade-off between a high acceptance rate and the ability to make large steps. In the simulation of subfigure *(b)* , approximately 10-20% of the proposed transitions were rejected, while the rejection percentage of subfigure *(c)* was as high as 60-70%.



(a) **(b)** **(c)**

Figure 5: Plots illustrating the concepts of the mixing and burn-in. The target density is indicated by contour plots, and the MCMC samples of size 200 is shown by dots and lines. The simulation was performed by means of Metropolis-Hastings with proposal distribution $Q(x \to y) = x + \text{Unif}\left((-\alpha, \alpha)^2\right)$. **(a)** Initial state $x_0 = (-4, -4)^T$, $\alpha = 0.75$. **(b)** Initial state $x_0 = (2, 0)^T$, $\alpha = 0.35$. **(c)** Initial state $x_0 = (2, 0)^T$, $\alpha = 1.75$.

3.2 Metropolis-Hastings algorithm

The Metropolis-Hastings algorithm introduced in the previous section is an almost universal algorithm for simulating a Markov chain with stationary distribution density π. The algorithm can be stated as follows, where we define $\alpha(x \to y) = 1$ for $\pi(x)q(x \to y) = 0$.

Algorithm 2 Metropolis-Hastings algorithm

Initialize X_0 with some $x_0 \in \mathcal{S}$
for $i = 1$ to n **do**
  $x = X_{i-1}$
  Generate y from proposal density $q(x \to y)$
  Compute acceptance rate $\alpha(x \to y) = \min\left\{1, \frac{\pi(y)q(y \to x)}{\pi(x)q(x \to y)}\right\}$
  Generate u from $\text{Unif}([0, 1))$
  if $u < \alpha(x \to y)$ **then**
    $X_i = y$
  else
    $X_i = x$
  end if
end for

Obviously, the Metropolis-Hastings algorithm will always accept transitions $x \to y$ for which the stationary "backward flow of proposals", $\pi(y)q(y \to x)$, is higher than the stationary "forward flow of proposals", $\pi(x)q(x \to y)$, since such a proposition will have acceptance probability $\alpha(x \to y) = 1$. The algorithm may also accept transitions that decreases the ratio, with an acceptance probability inversely proportional to the relative decrease.

Another way to think about the dynamics of the algorithm is to reformulate the acceptance rate as

$$\frac{\pi(y)q(y \to x)}{\pi(x)q(x \to y)} = \frac{\pi(y)/\pi(x)}{q(x \to y)/q(y \to x)}.$$

The ratio $\pi(y)/\pi(x)$ will then express how much more likely (density-wise) a realization y is than a realization x, and an increase in the "likelihood" will favor the corresponding transition. On the other hand, if the ratio $q(x \to y)/q(y \to x)$ is greater than 1, then the transition is not "that necessary" each time it is proposed, but if it less than 1 it is "even more necessary". The ratio $q(x \to y)/q(y \to x)$ will then express how much more likely (density-wise) the transition proposal $x \to y$ is compared to the opposite transition proposal, $y \to x$, and the transition probability is adjusted accordingly. The weighting function, $q(x \to y)/q(y \to x)$, is thus designed to balance the transitions, so that the influence of the proposal density on the distribution of states will be insignificant in the long run, with the limiting distribution of the chain hopefully given by π.

3.2.1 Validity

The Markov chains generated by the Metropolis-Hastings algorithm will be valid under rather mild conditions. We will report the essential requirements in this section. Details can be found in Robert and Casella (2004).

In practice, a MCMC simulation is usually conducted by be starting a single Markov chain from some fixed state x_0, and then run it for a sufficiently long time. For the chain to be able to explore the target density properly, it will be necessary that the proposal density has access to the operative area of the target distribution π. Thus, a minimal, necessary condition on $q(x \to y)$ will be that

$$\operatorname{supp}(\pi) \subseteq \bigcup_{x \in \operatorname{supp}(\pi)} \operatorname{supp}(q(x \to \cdot)).$$

The transition kernel density associated with the Metropolis-Hastings algorithm can be expressed as

$$\kappa(x \to y) = \alpha(x \to y) q(x \to y) + r(x)\delta_x(y), \tag{11}$$

where $r(x) = 1 - \int \alpha(x \to y) q(x \to y) \mathrm{d}y$ and $\delta_x(y)$ is the Dirac mass in x. The transition density from x to y is thus basically given by the density of making an accepted transition $x \to y$, with the additional "impulse" density of having all transitions rejected when $x = y$. A sufficient condition for the chain to be aperiodic is that $\Pr(X_{k+1} = X_k) > 0$, which holds when $\Pr(\alpha(X \to Y) < 1) > 0$, or equivalently,

$$\Pr(\pi(X) q(X \to Y) \leq \pi(Y) q(Y \to X)) < 1. \tag{12}$$

If the condition

$$q(x \to y) > 0 \quad \forall x, y \in \operatorname{supp}(\pi) \tag{13}$$

holds, then the kernel (11) of the Metropolis-Hastings chain satisfies the detailed balance

condition, since for $x, y \in \text{supp}\,(\pi)$, it then follows that

$$\begin{aligned}
\pi(x)q(x\rightarrow y)\alpha(x\rightarrow y) &= \pi(x)q(x\rightarrow y)\min\left\{1, \frac{\pi(y)q(y\rightarrow x)}{\pi(x)q(x\rightarrow y)}\right\} \\
&= \min\{\pi(x)q(x\rightarrow y), \pi(y)q(y\rightarrow x)\} \\
&= \min\{\pi(y)q(y\rightarrow x), \pi(x)q(x\rightarrow y)\} \\
&= \pi(y)q(y\rightarrow x)\min\left\{1, \frac{\pi(x)q(x\rightarrow y)}{\pi(y)q(y\rightarrow x)}\right\} \\
&= \pi(y)q(y\rightarrow x)\alpha(y\rightarrow x), \qquad (14)
\end{aligned}$$

and by the properties of the delta function, we have that

$$\pi(x)r(x)\delta_x(y) = \pi(y)r(y)\delta_y(x). \qquad (15)$$

For definiteness, we follow the convention that $\alpha(x \rightarrow y) = 0$ if $\pi(x) = \pi(y) = 0$. The detailed balance for the Metropolis-Hastings kernel (11) follows immediately from (14) and (15), and so π will be the stationary distribution of the chain. In addition, under the assumption (13) on the proposal density, the Metropolis-Hastings chain will be irreducible with respect to the Lebesgue measure, and positive (recurrent), as explained by Robert and Casella (2004), pp. 272-273. Robert and Casella also show the following result.

Result 3.2.1. *Suppose that the Metropolis-Hastings Markov chain (X_k) is π-irreducible.*

(i) If $g \in L^1(\pi)$, then

$$\lim_{n\rightarrow\infty}\frac{1}{n}\sum_{k=1}^{n} g(X_k) = \int g(x)\pi(x)\mathrm{d}x \qquad \pi\text{-a.e.}$$

(ii) If, in addition, (X_k) is aperiodic, then

$$\lim_{n\rightarrow\infty}\left\|\int K^n(x,\cdot)\mu(\mathrm{d}x) - \pi(\cdot)\right\|_{TV} = 0$$

for every initial distribution μ, where $K^n(x,\cdot)$ denotes the kernel for n transitions, as in (6).

The Metropolis-Hastings Markov chain (X_k) will be π-irreducible if the proposal density $q(x\rightarrow y)$ satisfy the conditions (12) and (13).

3.3 Proposal strategies

When it comes to the selection of proposal density for a particular MCMC implementation, there is usually considerable freedom of choice, at least from a theoretical point of view. The following proposal schemes are popular (see Liu (2001) for an elaborated treatment).

Independence sampler The employed proposal density does not depend on the current position, i.e.,

$$q(x\rightarrow y) = q(y),$$

resulting in a global exploration approach.

Random walk The proposal density is such that it only depends on the deviation from the current state, i.e.,

$$q(x \rightarrow y) = q(x - y),$$

resulting in a local exploration approach. For instance, $N(x, \Sigma)$ and $\text{Unif}(\mathcal{N}_x)$ for some matrix Σ and some neighborhood $\mathcal{N}_x$ of x represent two classes of random walk distributions.

Langevin algorithm The proposals are generated in accordance with a discrete approximation to a Langevin diffusion process, i.e.,

$$y \sim N\left(x + \frac{\delta}{2}\nabla \log \pi(x), \delta\right),$$

for some (typically small) δ.

The most common strategy to accommodate proper mixing properties in a Markov chain simulation is to first select an advantageous type of proposal density, and then set any parameters so as to be well suited for the target density. For instance, with random walk proposals, there is a natural trade-off between a high acceptance probability and large steps, so that it might be reasonable to select a proposal density that allows for "large enough" steps under a suitable restriction on the corresponding acceptance rate. Of course, such considerations will depend heavily on the target distribution. For instance, if the dominating areas of high probability under π are slanted with respect to the coordinate axis, it would not be statistically efficient to update the coordinates sequentially by a random walk proposal, since large steps would very likely be rejected. Simultaneously updating of all the coordinates would then work much better. In the general case, however, the topography of the target distribution is not known a priori. The only available guidance device will then be the density function, and such particular optimality considerations are therefore made difficult.

4 The sampling challenge

In this section we will illustrate some key features of random sample generation using a rather trivial example. In particular, we will consider the task of generating a sample from a bivariate normal distribution. The target density is then given by

$$\pi(x) = \frac{1}{(2\pi|\Sigma|)^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right),$$

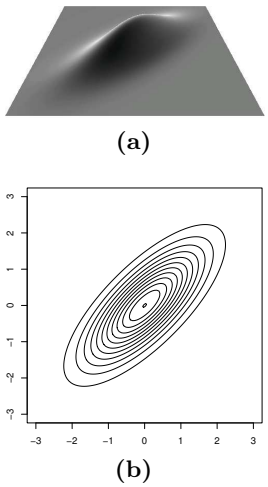
where μ is the expectation and Σ is the covariation matrix and $|\cdot|$ denotes the determinant operator. More specifically, we will consider the case where

$$\mu = (0,0)^T, \quad \Sigma = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix},$$

so that the variance of each component is 1 and the covariance between them is ρ. The matrix Σ can be represented in terms of its the so-called *Cholesky decomposition*, i.e.,

$$\Sigma = \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \rho & \sqrt{1-\rho^2} \end{bmatrix} \begin{bmatrix} 1 & \rho \\ 0 & \sqrt{1-\rho^2} \end{bmatrix} = LL^T.$$

We will take $\rho = 0.75$. Plots of the density is shown in Figure 6.

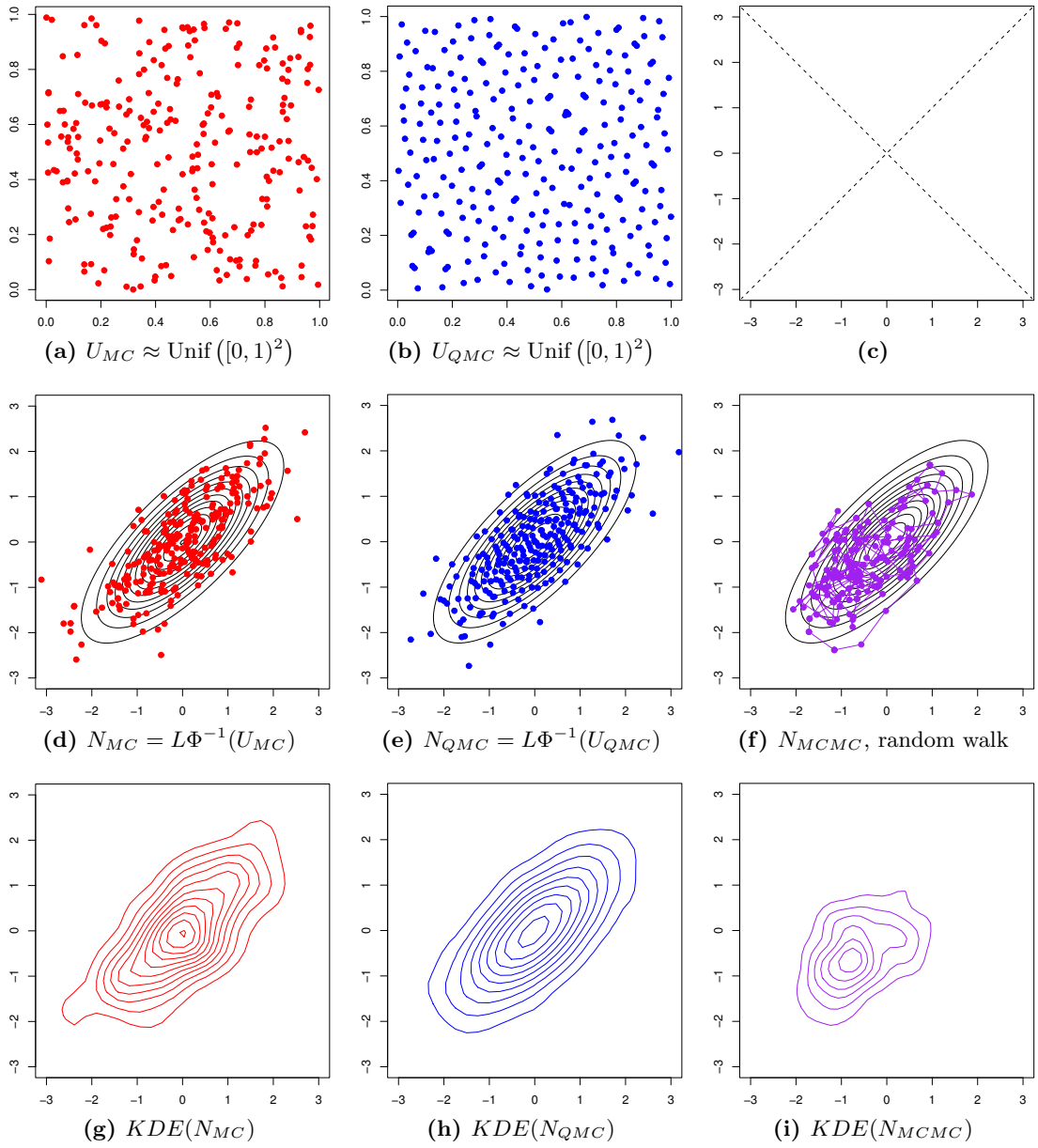


(a)

(b)

Figure 6: Plot of the bivariate normal distribution where each component has mean 0 and variance 1, with correlation $\rho = 0.75$. **(a)** Surface plot. **(b)** Contour plot.

The Monte Carlo approach for handling the distribution will be to approximate the continuous distribution by some sample of points. The challenge is then to generate point sets that are representative of the distribution, in the sense that each area of the domain should contain a number of points that corresponds to the probability measure of that area. In other words, the discrepancy between the empirical distribution and the exact distribution should be minimal. The classical MC solution is to take an i.i.d. sample from π, the QMC solution is to select some deterministic, low-discrepancy point set, and the MCMC solution is to take the states visited by a Markov chain with π as its stationary distribution. The three different procedures are illustrated in Figure 7.



(a) $U_{MC} \approx \text{Unif}\left([0,1)^2\right)$ **(b)** $U_{QMC} \approx \text{Unif}\left([0,1)^2\right)$ **(c)**

(d) $N_{MC} = L\Phi^{-1}(U_{MC})$ **(e)** $N_{QMC} = L\Phi^{-1}(U_{QMC})$ **(f)** N_{MCMC}, random walk

(g) $KDE(N_{MC})$ **(h)** $KDE(N_{QMC})$ **(i)** $KDE(N_{MCMC})$

Figure 7: Generation of samples from $N(0,\Sigma)$ of size $n = 256$ by MC (left column), QMC (center column) and MCMC (right column). The MC and QMC samples were generated by inversion of the cumulative distribution function, while the MCMC sample was generated with the Metropolis-Hastings algorithm using the random walk proposal $y = x + \text{Unif}\left((-0.75, 0.75)^2\right)$. **(a)** Sample U_{MC}, taken i.i.d. from $\text{Unif}\left([0,1)^2\right)$, **(b)** Sample U_{QMC}, taken from the Faure sequence in base 2 after a Cranley-Patterson rotation, **(d)** Sample U_{MC} transformed to normal random sample N_{MC}, **(e)** Sample U_{QMC} transformed to normal low-discrepant sample N_{QMC}, **(f)** Sample N_{MCMC} from $N(0,\Sigma)$, generated with MH with $x_0 = 0$, **(g)** Kernel density estimate of N_{MC}, **(h)** Kernel density estimate of N_{QMC}, **(i)** Kernel density estimate of N_{MCMC}.

A few comments should be made at this point. Firstly, the given case is completely trivial, and MCMC is not really necessary since direct i.i.d. sampling is possible. Secondly, each column of plots in Figure 7 depicts *a single realization* of the random samples,

and of course one could have gotten "better" or "worse" results for both the MC and MCMC case. The QMC situation is more stable, as the the collective distribution of the points are preserved in the randomization procedure (cf. Section 1.2.2). Nevertheless, the figure illustrates well the weakness of both MC and MCMC, namely the lack of long-term memory, in the sense that when generating the next sample point, the previously generated points are not considered. This property is of course introduced by construction, as MC uses independent sample points, and MCMC uses sample points with the minimal memory given by the Markov property. The highly momentary behavior will not be a problem in the long run, that is, as the sample size increases, since all areas on average will get the correct number of visits. In the short run, however, it may be a problem, since the sample trajectory may require a long time to cover the whole domain adequately. In the MC case, the problem may induce randomly distributed clusters and gaps, while in the MCMC case, the chain may have poor mixing properties, in the sense that the paths taken often will be unbalanced over the sample space (cf. Section 3.1). On the other hand, the lack of memory makes the theoretical aspects of the simulation procedure relatively simple.

The QMC sample N_{QMC} from the target distribution of Figure 7 was created by inversion of the cumulative distribution function using uniformly distributed points U_{QMC} in the unit hyper cube. As seen in the figure, the discrepancy between the empirical distribution of N_{QMC} and the target distribution is small, in the same way as the discrepancy between the empirical distribution of U_{QMC} and $\text{Unif}\left([0,1)^d\right)$ is small. The low-discrepancy property of the point set is thus conserved under that particular transformation. The same is is not true for a general transformation, but it always hold for the inversion of cumulative distribution function.

4.1 The big picture

The objective of random sample generation is, fundamentally, to assemble a non-deterministic collection of points such that discrepancy between the empirical distribution and the true distribution is small. In addition, the discrepancy should, in some relevant measure, tend to zero as the sample size goes to infinity, at least probabilistically. Of course, generation procedures that give rise to easily extensible samples will be attractive, in the sense that it should be possible to improve a previously generated sample by adding another sample of relatively moderate size.

In principle, there are then three apparent ways to generate random samples. The first is by regular random sampling (e.g., MC, MCMC), the second is by deterministic selection with the use of some kind of randomization procedure (e.g., QMC, modified integration quadratures), and the third is by some sort of guided exploration of the target distribution (perhaps by a hybridization of the regular sampling methods and some of the search algorithms of optimization theory). Qualitatively speaking, the three sampling strategies will then correspond to point set generation with "minimal memory" sampling (by independence or Markov dependence), "maximal memory" sampling (by deterministic relations) or "moderate memory" (by prominent dependence), respectively.

The deterministic approach will typically be rendered useless in high dimensions, due to the so-called *curse of dimensionality*. On the other hand, regular random sampling is in principle relatively easy, but of course, highly sub-optimal. The hybrid approach seems to be able to reap the best of both the others approaches, but the development of such methods is by far incomplete.

That being said, the deterministically-directed ideas of QMC is still alluring, and

different utilizations of low-discrepancy point sets for MCMC purposes have been proposed. A strategy proposed by Owen and Tribble (2005a,b) is to replace the pseudo-random driving sequence of a standard MCMC method with a so-called *completely uniformly distributed* (CUD) sequence, and the altered methods can be proven to yield consistent estimates for some problems. Further on, the proposed method is found to be considerably more accurate than the ordinary method, at least for a selection of numerical test cases.

In this paper, we are considering methods of the random sampling paradigm, and in particular, the MCMC methods. In the remaining sections, we will examine certain techniques that are designed to improve the ordinary MCMC methods within the limitations of the Markovian framework, in contrast to, e.g., the so-called *adaption sampling techniques*. Clearly, the prevailing "memory property" of the methods will then be fixed at "minimal memory" (due to the Markov structure), and so the best we can do is to increase the mixing of the chain. In other words, the focus will be on the task of *making suitable transitions in the momentary mode*. In particular, we will focus on the strategy known as multiple proposal MCMC.

5 Multiple proposals

In this section we will introduce a type of MCMC methods that are somewhat different from the ordinary methods described in Section 3. Instead of proposing a single transition for acceptance or rejection at each step, the so-called *multiple proposal methods* generates $m \geq 1$ proposals that are considered for transition.

The main idea of using more than one proposal at each step is that the proposal strategy then may be bolder, in the sense that the typical area of impact of the proposal density may be increased. As accounted for in Sections 3.1 and 3.3, the efficiency of the MCMC algorithms is intrinsically governed by the interaction between the acceptance rate and the proposal mechanism, with the dominating relationship that smaller steps are more likely to be accepted. In the multiple proposal methods, at most one of the proposals will be accepted at each step, and so it will not be critical if most of the proposals are highly unrealistic transition alternatives, as long as at least one of them has a fair chance of being accepted. Naturally, there will be a trade-off between the momentary range and the transition rate in the multiple proposal case as well, but the relationship may then be less restrictive, so that larger steps may be accepted more often.

Of course, there is no such thing as a free lunch. Evaluating several proposals will necessarily be computationally more expensive than evaluating just one. The question of actual gain may therefore be posed: is there any reason that using m proposals at each step will be more efficient than m sequential steps with a single proposal? Fortunately, the possible benefits of multiple proposals will be relevant.

The most apparent advantage is that the m proposals may be correlated, and so the neighborhood of the present state of the chain may be examined more systematically than with m individual proposals. Such a facility may be important if, for instance, the target distribution is multi-modal or has statistical "bottlenecks" of some sort. It is also possible that when several of the proposals have a fair chance of being accepted, they may more easily be weighted according to some suitable criteria.

Another advantage is that some of the multiple proposal methods are much better adjusted for parallel programming than the ordinary methods. In addition, some of the methods may be used to strengthen various other simulation schemes (e.g., random-ray, hit-and-run, random-grid Metropolis), as suggested in Liu et al. (2000); Qin and Liu (2001). Finally, it is actually possible to utilize the information about the target distribution that is obtained from the evaluation of all the proposals, not only the part corresponding to the accepted states, as described in Tjelmeland (2004).

In the subsequent sections, we will examine the multiple proposal methods of Liu et al. (2000) and Tjelmeland (2004). A method similar to that of Liu et al. (2000) is proposed in Qin and Liu (2001), but for reasons of moderation, we will not consider it here. However, we mention that the modified method may provide computational simplifications, as it allows old proposals to be reused.

5.1 The Kingpin-method

In this section we will present the multiple proposal method of Liu et al. (2000). We will consider proposal sets $\mathcal{Y} = (y_1, ..., y_m)$ with an associated density function $q(x \rightarrow \mathcal{Y})$ such that the proposals are exchangeable, i.e., for each permutation $\tilde{\mathcal{Y}} = (\tilde{y}_1, ..., \tilde{y}_m)$ of $\mathcal{Y}$, we

have that

$$q(x \rightarrow \mathcal{Y}) = q(x \rightarrow \tilde{\mathcal{Y}}).$$

In particular, we then have that the marginal density function of each proposal y_j, $j \in \{1, \ldots, m\}$ is identical, and we will denote it by $q_0(x \rightarrow y_j)$. We will also impose the modest requirement that $q_0(x \rightarrow y) > 0$ if and only if $q_0(y \rightarrow x) > 0$. By defining the transition weight function $w(x, y) \equiv w(x \rightarrow y)$ as

$$w(x \rightarrow y) = \pi(x) q_0(x \rightarrow y) \lambda(x, y), \tag{16}$$

where $\lambda(x, y)$ is any non-negative, symmetric function, we can specify the method of Liu et al. (2000) by the following algorithm.

Algorithm 3 Kingpin-method

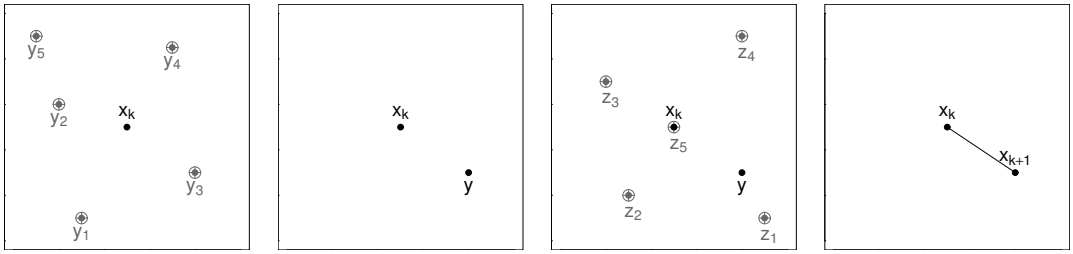
Initialize X_0 with some $x_0 \in \mathcal{S}$
for $i = 1$ to n **do**
  $x = X_{i-1}$
  Generate proposal set $\mathcal{Y} = (y_1, ..., y_m)$ from $q(x \rightarrow \mathcal{Y})$
  Compute the reversed transition weights $w(y_j \rightarrow x), \quad j = 1, \ldots, m$
  Select y among $\{y_1, \ldots, y_m\}$ with probability proportional to $w(y_j \rightarrow x)$
  Generate set of dummy proposals $\mathcal{Z} = (z_1, ..., z_m)$ from $q(y \rightarrow \mathcal{Z})$ with $z_m = x$
  Compute acceptance rate $\alpha(x \rightarrow y) = \min \left\{1, \frac{\sum_{j=1}^{m} w(y_j \rightarrow x)}{\sum_{j=1}^{m} w(z_j \rightarrow y)}\right\}$
  Generate u from Unif $([0, 1))$
  if $u < \alpha(x \rightarrow y)$ **then**
    $X_i = y$
  else
    $X_i = x$
  end if
end for

We will refer to the method described in Algorithm 3 as the *Kingpin-method*, since among all the proposals, only the most dominant state ("the Kingpin") is actually considered for transition. The different steps of the method is illustrated in Figure 8.

All together, the steps leading up to the selection of the final proposal can be seen as simply a special type of proposal mechanism. In that sense, the Kingpin-method is quite similar to the standard Metropolis-Hastings algorithm, except that the acceptance rate is of a slightly different type. More precisely, the acceptance rate depends on both the final proposal and on the other proposals. However, the intention with the particular acceptance/rejection design is that the ratio of the total "incoming flow" to x from a set of neighboring states $\mathcal{Y}$ and the total "incoming flow " to y from a corresponding set of neighbors $\mathcal{Z}$ should be balanced. The current state x will always will be included in $\mathcal{Z}$ and y will necessarily be included in $\mathcal{Y}$, and so the Kingpin-method with $\lambda(x, y) = 1$ can easily be seen as a natural extension of the Metropolis-Hastings algorithm, as presented in Section 3.2, since for $m = 1$, the two methods will be identical.

5.2 The Theater-method

In this section we will present the multiple proposal method of Tjelmeland (2004). The method can be specified by the following algorithm. We will supply sufficient conditions



(a) Step 1: Get proposals set. **(b)** Step 2: Select "Kingpin". **(c)** Step 3: Get dummy set. **(d)** Step 4: Accept or reject transition.

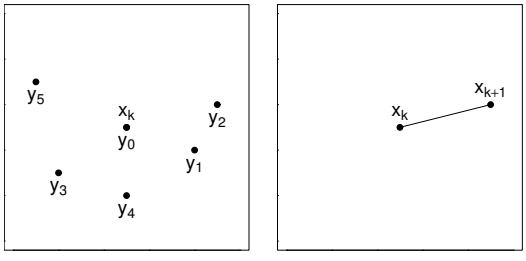
Figure 8: Kingpin-method. Step 1: Generate set of proposals with current state x_k as point of origin. Step 2: Select final proposal from the proposal set. Step 3: Generate set of dummy-proposals with the final proposal as point of origin. Dummy-proposals are needed for decision making in the last step. Step 4: Reject or accept the final proposal.

for the induced chain to satisfy the detailed balance condition in Section 5.4. The method can be specified by the following algorithm.

Algorithm 4 Theater-method

Initialize X_0 with some $x_0 \in \mathcal{S}$
for $i = 1$ to n **do**
  $x = X_{i-1}$
  Generate proposal set $\mathcal{Y} = (y_1, ..., y_m)$ from $q(x \to \mathcal{Y})$
  Compute transition probabilities $p\left(x \to \mathcal{Y}\right) = \left(p_0\left(x \to \mathcal{Y}\right), p_1\left(x \to \mathcal{Y}\right), \ldots, p_m\left(x \to \mathcal{Y}\right)\right)$
  Select y from $(x, y_1, \ldots, y_m)$ with probabilities given by $p\left(x \to \mathcal{Y}\right)$
  $X_i = y$
end for

We will refer to the method described in Algorithm 4 as the *Theater-method*, since all the proposals are considered as candidates for the "next-state part", in similarity with the actors in a theater audition. The general method leaves quite a bit of freedom in the assignment of weights to the candidates, but of course, the most suited should be given some priority. On the other hand, to avoid unwholesome cliques, it may perhaps be necessary to support the more unusual of the lot as well. The different steps of the method is illustrated in Figure 9.



(a) Step 1: Get proposals set. **(b)** Step 2: Select transition.

Figure 9: Theater-method. Step 1: Generate proposal set $(y_1, \ldots, y_m)$ at x_k. Step 2: Select the next state from the candidates $(x_k, y_1, \ldots, y_m)$.

5.3 A unitary view

In this section we will show how the Kingpin-method can be seen to be a particular instance of a generalized Theater-method. We will proceed by reformulating the Theater-method as given by Algorithm 5.

Algorithm 5 Theater-method - reformulated

```
Initialize X_0 with some x_0 ∈ S
for i = 1 to n do
    x = X_{i-1}
    Generate proposal set Y = (y_1, ..., y_m) from q(x→Y)
    Compute transition probabilities p(x→Y) = (p_0(x→Y), p_1(x→Y), ..., p_m(x→Y))
    Select y from (y_1, ..., y_m) with probability proportional to p_j(x→Y)
    Generate u from Unif([0,1))
    if u < (1 − p_0(x→Y)) then
        X_i = y
    else
        X_i = x
    end if
end for
```

It should be fairly obvious that the new formulation is equivalent with the original, as formalized by Proposition 5.3.1

Proposition 5.3.1. *The Markov chain induced by Algorithm 5 (the reformulated Theater-method) is indistinguishable from the chain induced by Algorithm 4 (the original formulation of the Theater-method).*

Proof. Consider a random variable A_1 on the set $\mathcal{Y} = \{y_0, y_1, \ldots, y_m\}$ with an associated probability distribution given by $p_0, p_1, \ldots, p_m$, i.e.,

$$\Pr(A_1 = y_i) = p_i, \quad i \in \{0, 1, \ldots, m\}, \qquad \sum_{i=0}^{m} p_i = 1.$$

Correspondingly, let B be a random variable on the truncated set $\tilde{\mathcal{Y}} = \{y_1, \ldots, y_m\}$ with probability distribution given by $\tilde{p}_1, \ldots, \tilde{p}_m$, where

$$\tilde{p}_i = \Pr(B = y_i) = \frac{p_i}{\sum_{j=1}^{m} p_j}, \quad i \in \{1, \ldots, m\},$$

and let the random variable $A_2|B = b$ with $b \in \tilde{\mathcal{Y}}$ be defined on the set $\mathcal{Y}$ with distribution

$$\Pr(A_2 = y_0|B = b) = p_0, \qquad \Pr(A_2 = b|B = b) = 1 - p_0.$$

Clearly, the marginal distribution of A_2 is then

$$\begin{aligned}\Pr(A_2 = y_i) &= \sum_{j=1}^{m} \Pr(A_2 = y_i|B = y_j) \Pr(B = y_j) \\ &= \begin{cases} p_0 \sum_{j=1}^{m} \Pr(B = y_j) & = p_0, & y_i = y_0, \\ (1 - p_0) \Pr(B = y_i) & = (1 - p_0) \frac{p_i}{\sum_{j=1}^{m} p_i} = p_i, & y_i \in \{y_1, \ldots, y_m\}, \end{cases}\end{aligned}$$

so A_1 and A_2 has the same distribution. Thus it is irrelevant if we make the draw in one or two steps, that is, if we use a realization of A_1 or a realization of A_2. □

The only principal difference between the Kingpin-method and the reformulated Theater-method is that the acceptance rate in the former is random, with the randomness introduced by the set of dummy-proposals, while in the latter, it is deterministically dependent on x and $\mathcal{Y}$, as given by $1 - p_0(x \to \mathcal{Y})$. Consequently, it should be possible, but not necessarily desirable, to formulate a generalized Theater-method such that the Kingpin-method will correspond to a particular instance of that class of methods.

5.4 Method validation

In this section, we will examine some sufficient conditions for the Markov chains associated with the Kingpin-method and Theater-method to have π as their stationary distributions. In fact, the chains satisfy the detailed balance condition for a large class of proposal densities, which we will show by using the results given in the next section. In addition to stationarity, it will be necessary to verify aperiodicity and irreducibility, but this will usually be quite simple for the particular applications.

5.4.1 Preliminary results

When examining the properties of the transition kernels associated with the multiple proposal methods, the following simple results will be convenient.

Proposition 5.4.1. *Let (X_k) be a Markov chain with transition kernel on the form*

$$\kappa(x \to y) = \sum_{j=1}^{m} t_j(x \to y) + r(x)\delta_x(y).$$

If each route of actual transition satisfies the detailed balance condition individually, i.e.,

$$\pi(x)t_j(x \to y) = \pi(y)t_j(y \to x), \qquad j \in \{1, \ldots, m\},$$

then (X_k) satisfies the detailed balance condition.

Proof. The overall satisfaction of the detailed balance follows directly from the assumptions and the simple property of the Dirac delta function that $f(y)\delta_x(y) = f(x)\delta_y(x)$, since then

$$\begin{aligned}
\pi(x)\kappa(x \to y) &= \pi(x)\sum_{j=1}^{m} t_j(x \to y) + \pi(x)r(x)\delta_x(y) \\
&= \sum_{j=1}^{m} \pi(x)t_j(x \to y) + \pi(x)r(x)\delta_x(y) \\
&= \sum_{j=1}^{m} \pi(y)t_j(y \to x) + \pi(y)r(y)\delta_y(x) \\
&= \pi(y)\sum_{j=1}^{m} t_j(y \to x) + \pi(y)r(y)\delta_y(x) = \pi(y)\kappa(y \to x).
\end{aligned}$$

□

Proposition 5.4.2. *If a function $f(x \to y)$ can be written on the form*

$$f(x \to y) = \int \dots \int h\left(x, y, g_1, \dots, g_r\right) \mathrm{d}g_1 \cdots \mathrm{d}g_r$$

for some continuous function $h\left(x, y, g_1, \dots, g_r\right)$ with the property that

$$h\left(x, y, g_1, \dots, g_r\right) = h\left(y, x, \tilde{g}_1, \dots, \tilde{g}_r\right),$$

for some permutation $(\tilde{g}_1, \dots, \tilde{g}_r)$ of $(g_1, \dots, g_r)$, then $f(x \to y)$ is symmetric in x and y.

Proof. The proof is immediate. By changing the order of integration (Fubini's theorem), it follows from the conditions that

$$\begin{aligned} f(x \to y) &= \int \dots \int h\left(x, y, g_1, \dots, g_r\right) \mathrm{d}g_1 \cdots \mathrm{d}g_r \\ &= \int \dots \int h\left(y, x, \tilde{g}_1, \dots, \tilde{g}_r\right) \mathrm{d}g_1 \cdots \mathrm{d}g_r \\ &= \int \dots \int h\left(y, x, \tilde{g}_1, \dots, \tilde{g}_r\right) \mathrm{d}\tilde{g}_1 \cdots \mathrm{d}\tilde{g}_r = f(y \to x) \end{aligned}$$

□

5.4.2 Assessing the Kingpin-method

Consider a Markov chain (X_k) constructed by Algorithm 3. Given that $X_k = x$, there are at most $m+1$ events that will concur with $X_{k+1} \in \mathcal{A}$. Either some $y \in \mathcal{A}$ is proposed as one of the m members of the proposal set when that particular member is selected and accepted, or, if $x \in \mathcal{A}$, the final transition is rejected. Denote by $R(x)$ the event of rejecting any transition proposed from x, and let $T_j(x \to \mathcal{A})$ denote the event that, when in x, a state $y \in \mathcal{A}$ is proposed as the jth member of the proposal set, and that the jth member of the proposal set is selected and accepted. In particular, Figure 8 illustrates an event T_3 for a Kingpin-chain with $m = 5$ when $y_3 \in \mathcal{A}$. Obviously, the events $\{T_1, \dots, T_m, R\}$ are mutually exclusive, and so

$$\Pr\left(X_{k+1} \in \mathcal{A} | X_k = x\right) = \sum_{j=1}^{m} \Pr\left(T_j(x \to \mathcal{A})\right) + \Pr\left(R(x)\right).$$

To express the transition kernel of the Kingpin-method in density form, the following short hand notation will be useful. Let

$$\begin{aligned} \mathcal{Y}_{-j} &\equiv (y_1, \dots, y_{j-1}, y_{j+1}, \dots, y_m), & \mathrm{d}\mathcal{Y}_{-j} &\equiv \mathrm{d}y_1 \dots \mathrm{d}y_{j-1}\mathrm{d}y_{j+1} \dots \mathrm{d}y_m, \\ \mathcal{Z}_{-j} &\equiv (z_1, \dots, z_{j-1}, z_{j+1}, \dots, z_m), & \mathrm{d}\mathcal{Z}_{-j} &\equiv \mathrm{d}z_1 \dots \mathrm{d}z_{j-1}\mathrm{d}z_{j+1} \dots \mathrm{d}z_m, \\ \mathcal{Y}_{y_j:=x} &\equiv (y_1, \dots, y_{j-1}, x, y_{j+1}, \dots, y_m), & \mathrm{d}\mathcal{Y} &\equiv \mathrm{d}y_1 \dots \mathrm{d}y_m, \\ \mathcal{Z}_{z_j:=x} &\equiv (z_1, \dots, z_{j-1}, x, z_{j+1}, \dots, z_m), & \mathrm{d}\mathcal{Z} &\equiv \mathrm{d}z_1 \dots \mathrm{d}z_m, \end{aligned}$$

and let

$$P_{KP}\left(x \to \mathcal{Y}_{y_j:=y}\right) \equiv \frac{w(y \to x)}{w(y \to x) + \sum_{i=1, i \neq j}^{m} w(y_i \to x)}, \tag{17}$$

$$P_A\left(x \to \mathcal{Y}_{y_j:=y}, \mathcal{Z}_{z_m:=x}\right) \equiv \min\left\{1, \frac{w(y \to x) + \sum_{i=1, i \neq j}^{m-1} w(y_j \to x)}{w(x \to y) + \sum_{i=1}^{m-1} w(z_j \to y)}\right\}. \tag{18}$$

In the above notation, $P_{KP}\left(x \rightarrow \mathcal{Y}_{y_j:=y}\right)$ denotes the probability of selecting $y_j = y$ as the final proposal ("Kingpin") conditionally on the location x and the other members of the proposal set, $\mathcal{Y}_{-j}$, as specified in the algorithm. Further on, $P_A\left(x \rightarrow \mathcal{Y}_{y_j:=y}, \mathcal{Z}_{z_m:=x}\right)$ denotes the probability of accepting a final transition $x \rightarrow y$ when y is the selected jth proposal, conditionally on the other members of the proposal set, $\mathcal{Y}_{-j}$, and the set of remaining dummy-proposals, $\mathcal{Z}_{-m}$. From the definitions (17) and (18), let the transition probability associated with the proposal $x \rightarrow y_j$ for $y_j = y$ be given as

$$P_{KPA}\left(x \rightarrow \mathcal{Y}_{y_j:=y}, \mathcal{Z}_{z_m:=x}\right) \equiv P_{KP}\left(x \rightarrow \mathcal{Y}_{y_j:=y}\right) P_A\left(x \rightarrow \mathcal{Y}_{y_j:=y}, \mathcal{Z}_{z_m:=x}\right). \tag{19}$$

Consequently, the resultant $P_{KPA}\left(x \rightarrow \mathcal{Y}_{y_j:=y}, \mathcal{Z}_{z_m:=x}\right)$ will then give the probability of selecting and accepting the proposal $y_j = y$ at x, conditionally on $\mathcal{Y}_{-j}$ and $\mathcal{Z}_{-m}$. Similarly, let $P_{KPR}\left(x \rightarrow \mathcal{Y}, \mathcal{Z}_{z_m:=x}\right)$ denote the probability of rejecting all the proposals, conditionally on $\mathcal{Y}$ and $\mathcal{Z}_{-m}$, that is, let

$$P_{KPR}\left(x \rightarrow \mathcal{Y}, \mathcal{Z}_{z_m:=x}\right) \equiv P_{KP}\left(x \rightarrow \mathcal{Y}_{y_j:=y_j}\right)\left[1 - P_A\left(x \rightarrow \mathcal{Y}_{y_j:=y_j}, \mathcal{Z}_{z_m:=x}\right)\right].$$

By the law of total probability, we can then express the transition kernel of Algorithm 3 in density form as

$$\kappa(x \rightarrow y) = \sum_{j=1}^{m} t_j(x \rightarrow y) + r(x)\delta_x(y) \tag{20}$$

where $t_j(x \rightarrow y)$ is the probability density corresponding to the event $T_j(x \rightarrow y)$, and $r(x)$ is probability density corresponding to $R(x)$, i.e.,

$$\begin{aligned}
&t_j(x \rightarrow y) \\
&\quad = \int \cdots \int P_{KPA}\left(x \rightarrow \mathcal{Y}_{y_j:=y}, \mathcal{Z}_{z_m:=x}\right) q(x \rightarrow \mathcal{Y}_{y_j:=y}) \frac{q(y \rightarrow \mathcal{Z}_{z_m:=x})}{\int \cdots \int q(y \rightarrow \mathcal{Z}_{z_m:=x}) \mathrm{d}\mathcal{Y}_{-j}} \mathrm{d}\mathcal{Y}_{-j}\, \mathrm{d}\mathcal{Z}_{-m}, \\
&\quad = \int \cdots \int P_{KPA}\left(x \rightarrow \mathcal{Y}_{y_j:=y}, \mathcal{Z}_{z_m:=x}\right) q(x \rightarrow \mathcal{Y}_{y_j:=y}) \frac{q(y \rightarrow \mathcal{Z}_{z_m:=x})}{q_0(y \rightarrow x)} \mathrm{d}\mathcal{Y}_{-j}\, \mathrm{d}\mathcal{Z}_{-m}, \qquad (21) \\
&r(x) = \int \cdots \int P_{KPR}\left(x \rightarrow \mathcal{Y}, \mathcal{Z}_{z_m:=x}\right) q(x \rightarrow \mathcal{Y}) q(y \rightarrow \mathcal{Z}_{z_m:=x}) \mathrm{d}\mathcal{Y}\, \mathrm{d}\mathcal{Z}_{-m}.
\end{aligned}$$

In the derivation of (21), we have used the fact that

$$\int \cdots \int q(y \rightarrow \mathcal{Z}_{z_m:=x}) \mathrm{d}\mathcal{Y}_{-j} = q_0(y \rightarrow x)$$

by assumption, and Bayes' theorem for probability densities, that states, when using generic notation, that $f(x|y) = f(x, y) / \int f(x, y) \mathrm{d}x$.

The weight function $w(x \rightarrow y)$ specified in (16) is designed to ensure that the flow of the chain is strongly balanced. While the proposals are generated from the proposal distribution in accordance with the proposal density function, the final proposal is selected with somewhat of the opposite intention; the proposal with the highest weight on the *reverse transition* is most likely to be selected. The result is then that the overall dynamics of the Kingpin-method will be balanced for large classes of q and λ, and the induced Markov chain will be reversible and have π as it stationary distribution. These considerations are formalized in the following result of Liu et al. (2000).

Proposition 5.4.3. *The transition rule given by Algorithm 3 (the Kingpin-method) satisfies the detailed balance condition.*

Proof. Let the auxiliary function $a\left(x\rightarrow y, \mathcal{Y}_{y_j:=y}, \mathcal{Z}_{z_m:=z}\right)$ be defined from (16) and (19) as

$$a\left(x\rightarrow y, \mathcal{Y}_{y_j:=y}, \mathcal{Z}_{z_m:=x}\right) \equiv \frac{P_{KPA}\left(x\rightarrow \mathcal{Y}_{y_j:=y}, \mathcal{Z}_{z_m:=x}\right)}{w(y\rightarrow x)}.$$

It then follows from the definitions (17) and (18) that

$$\begin{aligned} & a\left(x\rightarrow y, \mathcal{Y}_{y_j:=y}, \mathcal{Z}_{z_m:=x}\right) \\ & \quad = \frac{1}{w(y\rightarrow x)+\sum_{i=1,i\neq j}^{m} w(y_i\rightarrow x)} \min\left\{1, \frac{w(y\rightarrow x)+\sum_{i=1,i\neq j}^{m} w(y_i\rightarrow x)}{w(x\rightarrow y)+\sum_{i=1,i\neq j}^{m} w(z_i\rightarrow y)}\right\} \\ & \quad = \min\left\{\frac{1}{w(y\rightarrow x)+\sum_{i=1,i\neq j}^{m} w(y_i\rightarrow x)}, \frac{1}{w(x\rightarrow y)+\sum_{i=1}^{m-1} w(z_i\rightarrow y)}\right\}. \end{aligned} \tag{22}$$

From (22) and the trivial fact that $\min\{a,b\} = \min\{b,a\}$ for any numbers $a, b \in \mathbb{R}$, it is clear that

$$a\left(x\rightarrow y, \mathcal{Y}_{y_j:=y}, \mathcal{Z}_{z_m:=x}\right) = a\left(x\rightarrow y, \mathcal{Z}_{z_m:=x}, \mathcal{Y}_{y_j:=y}\right),$$

and so the function $h\left(x\rightarrow y, \mathcal{Y}_{y_j:=y}, \mathcal{Z}_{z_m:=x}\right)$ defined by

$$h\left(x\rightarrow y, \mathcal{Y}_{y_j:=y}, \mathcal{Z}_{z_m:=x}\right) = a\left(x\rightarrow y, \mathcal{Y}_{y_j:=y}, \mathcal{Z}_{z_m:=x}\right) q(x\rightarrow \mathcal{Y}_{y_j:=y}) q(y\rightarrow \mathcal{Z}_{z_m:=x})$$

obviously has the property that

$$h\left(x\rightarrow y, \mathcal{Y}_{y_j:=y}, \mathcal{Z}_{z_m:=x}\right) = h\left(y\rightarrow x, \mathcal{Z}_{z_m:=x}, \mathcal{Y}_{y_j:=y}\right).$$

From Proposition 5.4.2, it then follows that the function $f(x\rightarrow y)$ defined by

$$f(x\rightarrow y) = \int\!\!\ldots\!\!\int h\left(x\rightarrow y, \mathcal{Y}_{y_j:=y}, \mathcal{Z}_{z_m:=x}\right) \mathrm{d}\mathcal{Y}_{-j}\, \mathrm{d}\mathcal{Z}_{-m}$$

is a symmetric function, i.e.,

$$f(x\rightarrow y) = f(y\rightarrow x).$$

The density function associated with the individual routs of actual transition, as given by (21), can be written as

$$\begin{aligned} t_j(x\rightarrow y) &= \int\!\!\ldots\!\!\int P_{KPA}\left(x\rightarrow \mathcal{Y}_{y_j:=y}, \mathcal{Z}_{z_m:=x}\right) q(x\rightarrow \mathcal{Y}_{y_j:=y}) \frac{q(y\rightarrow \mathcal{Z}_{z_m:=x})}{q_0(y\rightarrow x)} \mathrm{d}\mathcal{Y}_{-j}\, \mathrm{d}\mathcal{Z}_{-m} \\ &= \frac{w(y\rightarrow x)}{q_0(y\rightarrow x)} \int\!\!\ldots\!\!\int h\left(x\rightarrow y, \mathcal{Y}_{y_j:=y}, \mathcal{Z}_{z_m:=x}\right) \mathrm{d}\mathcal{Y}_{-j}\, \mathrm{d}\mathcal{Z}_{-m} \\ &= \frac{w(y\rightarrow x)}{q_0(y\rightarrow x)} f(x\rightarrow y) \\ &= \pi(y)\lambda(y,x) f(x\rightarrow y), \end{aligned} \tag{23}$$

where the last equality follows from the definition (16) of $w(x \rightarrow y)$. Since both $\lambda(y, x)$ and $f(x \rightarrow y)$ is symmetric, it follows from (23) that

$$\begin{aligned} \pi(x)t_j(x \rightarrow y) &= \pi(x)\pi(y)\lambda(y, x)f(x \rightarrow y) \\ &= \pi(y)\pi(x)\lambda(x, y)f(y \rightarrow x) = \pi(y)t_j(y \rightarrow x), \end{aligned} \tag{24}$$

and then the overall detailed balance follows readily from (24), due to Proposition 5.4.1 and the form (20) of the transition kernel.

□

5.4.3 Assessing the Theater-method

To express the transition kernel of the Theater-method, it will be convenient to adopt some of the short hand notation introduced in the previous section for the Theater-method. In particular, by letting

$$\begin{aligned} \mathcal{Y}_{-j} &\equiv (y_1, \ldots, y_{j-1}, y_{j+1}, \ldots, y_m), & \mathrm{d}\mathcal{Y}_{-j} &\equiv \mathrm{d}y_1 \ldots \mathrm{d}y_{j-1}\mathrm{d}y_{j+1} \ldots \mathrm{d}y_m, \\ \mathcal{Y}_{y_j:=x} &\equiv (y_1, \ldots, y_{j-1}, x, y_{j+1}, \ldots, y_m), & \mathrm{d}\mathcal{Y} &\equiv \mathrm{d}y_1 \ldots \mathrm{d}y_m, \end{aligned}$$

the transition kernel density of Algorithm 4 can be expressed as

$$\kappa(x \rightarrow y) = \sum_{j=1}^{m} t_j(x \rightarrow y) + r(x)\delta_x(y), \tag{25}$$

in density form, where

$$\begin{aligned} t_j(x \rightarrow y) &= \int \ldots \int p_j\left(x \rightarrow \mathcal{Y}_{y_j:=y}\right) q(x \rightarrow \mathcal{Y}_{y_j:=y})\mathrm{d}\mathcal{Y}_{-j}, \\ r(x) &= \int \ldots \int p_0\left(x \rightarrow \mathcal{Y}\right) q(x \rightarrow \mathcal{Y})\mathrm{d}\mathcal{Y}. \end{aligned}$$

Due to the particular form of the transition weights $w(x \rightarrow y)$ and the use of dummy-proposals, the Kingpin-method satisfies the detailed balance condition for practically any proposal distribution as long as the proposals are exchangeable. For the Theater-method, however, the form of the transition probabilities is generally unspecified, and the method do not make use of any reference states. Consequently, there will be somewhat more restrictions on the agreement of the proposal and decision mechanisms. A sufficient condition for the Theater-method to satisfy the detailed balance is specified in the following proposition.

Proposition 5.4.4. *The transition rule given by Algorithm 4 (the Theater-method) satisfies the detailed balance condition if*

$$\pi(x)p_j\left(x \rightarrow \mathcal{Y}_{y_j:=y}\right) q(x \rightarrow \mathcal{Y}_{y_j:=y}) = \pi(y)p_j\left(y \rightarrow \mathcal{Y}_{y_j:=x}\right) q(y \rightarrow \mathcal{Y}_{y_j:=x}).$$

for each $j \in \{1, \ldots, m\}$.

Proof. The proof is immediate, since from the assumption,

$$\begin{aligned}
\pi(x)t_j(x\rightarrow y) &= \pi(x)\int\ldots\int p_j\left(x\rightarrow \mathcal{Y}_{y_j:=y}\right) q(x\rightarrow \mathcal{Y}_{y_j:=y})\mathrm{d}\mathcal{Y}_{-j} \\
&= \int\ldots\int \pi(x)p_j\left(x\rightarrow \mathcal{Y}_{y_j:=y}\right) q(x\rightarrow \mathcal{Y}_{y_j:=y})\mathrm{d}\mathcal{Y}_{-j} \\
&= \int\ldots\int \pi(y)p_j\left(y\rightarrow \mathcal{Y}_{y_j:=x}\right) q(y\rightarrow \mathcal{Y}_{y_j:=x})\mathrm{d}\mathcal{Y}_{-j} \\
&= \pi(y)\int\ldots\int p_j\left(y\rightarrow \mathcal{Y}_{y_j:=x}\right) q(y\rightarrow \mathcal{Y}_{y_j:=x})\mathrm{d}\mathcal{Y}_{-j} \\
&= \pi(y)t_j(y\rightarrow x), \qquad (26)
\end{aligned}$$

and the overall satisfaction of the detailed balance then follows from Proposition 5.4.1 and the form (25) of the transition kernel. □

We will now present two of the transition alternatives suggested in Tjelmeland (2004).

Transition alternative 1 ($\mathcal{T}1$) Let the transition probabilities be proportional to

$$\begin{aligned}
\hat{p}_0\left(x\rightarrow\mathcal{Y}\right) &= \pi(x)q(x\rightarrow\mathcal{Y}) \\
\hat{p}_j\left(x\rightarrow\mathcal{Y}\right) &= \pi(y_j)q(y_j\rightarrow\mathcal{Y}_{y_j:=x}), \quad j\in\{1,\ldots,m\},
\end{aligned}$$

that is, take

$$p_j\left(x\rightarrow\mathcal{Y}\right) = \frac{\hat{p}_j\left(x\rightarrow\mathcal{Y}\right)}{\sum_{i=0}^{m}\hat{p}_i\left(x\rightarrow\mathcal{Y}\right)}, \quad j\in\{0,1,\ldots,m\}. \qquad (27)$$

For $m=1$, the Theater-method with transition probabilities given by $\mathcal{T}1$ will correspond to the so-called Barker algorithm (Barker, 1965), which then is similar to the Metropolis-Hastings algorithm except that the acceptance probability

$$\min\left\{1, \frac{\pi(y)q(y\rightarrow x)}{\pi(x)q(x\rightarrow y)}\right\} \quad \text{is replaced by} \quad \frac{\pi(y)q(y\rightarrow x)}{\pi(y)q(y\rightarrow x)+\pi(x)q(x\rightarrow y)}.$$

More importantly, the above transition rule will satisfy the detailed balance condition for arbitrary $m\geq 1$.

Proposition 5.4.5. *The transition rule given by Algorithm 4 (the Theater-method) with transition probabilities given by $\mathcal{T}1$ (Transition alternative 1) satisfies the detailed balance condition.*

Proof. Define $c=\sum_{j=0}^{m}\hat{p}_j\left(x\rightarrow\mathcal{Y}\right)$. For each $j\in\{1,\ldots,m\}$, we have from (27) that

$$\begin{aligned}
\pi(x)p_j\left(x\rightarrow\mathcal{Y}_{y_j:=y}\right)q(x\rightarrow\mathcal{Y}_{y_j:=y}) &= \pi(x)\pi(y)q(y\rightarrow\mathcal{Y}_{y_j:=x})q(x\rightarrow\mathcal{Y}_{y_j:=y})/c, \\
&= \pi(y)p_j\left(y\rightarrow\mathcal{Y}_{y_j:=x}\right)q(y\rightarrow\mathcal{Y}_{y_j:=x}),
\end{aligned}$$

so the condition of Proposition 5.4.4 is clearly satisfied. □

For computational convenience, we note that the following Metropolis-type simplification, similar to (10) for the standard Metropolis-Hastings algorithm, applies for the transition probabilities of $\mathcal{T}1$.

Proposition 5.4.6. *It the proposal density $q(x \to \mathcal{Y})$ is symmetric, so that in particular*

$$q(x \to \mathcal{Y}) = q(y_j \to \mathcal{Y}_{y_j:=x}), \quad j \in \{1, \ldots, m\},$$

then the transition probabilities of $\mathcal{T}1$ is proportional to the target density at each proposal. That is, $p_0(x \to \mathcal{Y}) = \pi(x)/c$ and $p_j(x \to \mathcal{Y}) = \pi(y_j)/c$ for $j \in \{1, \ldots, m\}$, with the normalizing constant given by $c = \pi(x) + \sum_{i=1}^m \pi(y_i)$.

Proof. The proof is immediate, since

$$p_0(x \to \mathcal{Y}) = \frac{\pi(x)q(x \to \mathcal{Y})}{\pi(x)q(x \to \mathcal{Y}) + \sum_{i=1}^m \pi(y_i)q(y_i \to \mathcal{Y}_{y_i:=x})} = \frac{\pi(x)}{\pi(x) + \sum_{i=1}^m \pi(y_i)},$$

and for $j \in \{1, \ldots, m\}$,

$$p_j(x \to \mathcal{Y}) = \frac{\pi(y_j)q(y_j \to \mathcal{Y}_{y_j:=x})}{\pi(x)q(x \to \mathcal{Y}) + \sum_{i=1}^m \pi(y_i)q(y_i \to \mathcal{Y}_{y_i:=x})} = \frac{\pi(y_j)}{\pi(x) + \sum_{i=1}^m \pi(y_i)}.$$

□

In order to specify the second transition alternative, which will be an enhancement of the first, we will have to define a so-called *peskunization* procedure for transition matrices. The peskunization procedure is inspired by the rather well-known *Peskun's Theorem* (Peskun, 1973), that states, informally speaking, that a modification of a reversible Markov chain on a discrete sample space such that the probability of state changes is increased can never be unfavorable with respect to asymptotic variance. For a more thorough treatment of Peskun's theorem, we refer to Neal (2004).

Definition 5.4.1 (Peskunized transition matrix)**.** *Let $Q = [Q_{i,j}]_{i,j=0}^m$ be a $(m+1) \times (m+1)$ transition matrix, i.e., all the elements are non-negative and the sum of the elements on each row is 1. Let the $(m+1) \times (m+1)$ matrix $\widetilde{Q}$ be defined from Q via the following process.*

1: *Set $t = 0$ and let $Q^0 = Q$*
2: *Set $\mathcal{A}^t = \{i \mid Q_{i,i}^t > 0, i = 0, 1, \cdots, m\}$*
3: *If $|\mathcal{A}^t| \leq 1$, set $\widetilde{Q} = Q^t$ and stop the process*
4: *Set*

$$u = \min_{i \in \mathcal{A}^t} \left(\frac{1 - \sum_{j \notin \mathcal{A}^t} Q_{i,j}^t}{\sum_{j \in \mathcal{A}^t \setminus \{i\}} Q_{i,j}^t} \right)$$

5: *Construct Q^{t+1} from Q^t by setting (in the subsequent order)*

$$\begin{aligned} Q_{i,j}^{t+1} &= Q_{i,j}^t && \text{if } i \notin \mathcal{A}^t \text{ or } j \notin \mathcal{A}^t, \\ Q_{i,j}^{t+1} &= u Q_{i,j}^t && \text{if } i, j \in \mathcal{A}^t, \\ Q_{i,i}^{t+1} &= 1 - \sum_{j=1, j \neq i}^m Q_{i,j}^{t+1} && \text{for } j \in \mathcal{A}^t. \end{aligned}$$

6: *Repeat from step 3 with $t = t + 1$*

The process will be completed in maximally m iterations, and we will refer to the resulting transition matrix $\widetilde{Q}$ as the peskunization of Q.

Following Tjelmeland (2004), we can demonstrate the conveniently preservative nature of the peskunization procedure with respect to a stationary distribution by the following proposition.

Proposition 5.4.7. *Let $Q = [Q_{i,j}]_{i,j=0}^{m}$ be a $(m+1) \times (m+1)$ transition matrix and let $\Psi = (\psi_0, \ldots, \psi_m)$ be a vector of length (m+1) such that $\psi_i \geq 0$ and $\sum_{i=0}^{m} \psi_i = 1$. If Q satisfies the detailed condition for Ψ, i.e.,*

$$\psi_i Q_{i,j} = \psi_j Q_{j,i}, \tag{28}$$

for all $i, j \in \{0, 1, \ldots, m\}$, then $\widetilde{Q}$, the peskunization of Q, will satisfy the detailed condition for Ψ, i.e.,

$$\psi_i \widetilde{Q}_{i,j} = \psi_j \widetilde{Q}_{j,i}, \tag{29}$$

for all $i, j \in \{0, 1, \ldots, m\}$.

Proof. For some $u \in \mathbb{R}$ and $\mathcal{A}^t \subseteq \{0, 1, \ldots, m\}$, let the the $(m+1) \times (m+1)$ matrix Q^{t+1} be defined from Q^t by setting (in the subsequent order)

$$Q_{i,j}^{t+1} = Q_{i,j}^t \qquad \text{if } i \notin \mathcal{A}^t \text{ or } j \notin \mathcal{A}^t, \tag{30}$$

$$Q_{i,j}^{t+1} = u Q_{i,j}^t \qquad \text{if } i, j \in \mathcal{A}^t, \tag{31}$$

$$Q_{i,i}^{t+1} = 1 - \sum_{j=1, j \neq i}^{m} Q_{i,j}^{t+1} \qquad \text{for } j \in \mathcal{A}^t. \tag{32}$$

If, for all $i, j \in \{0, 1, \ldots, m\}$, we have that $\psi_i Q_{i,j}^t = \psi_j Q_{j,i}^t$, then it is clear from (30)-(32) that $\psi_i Q_{i,j}^{t+1} = \psi_j Q_{j,i}^{t+1}$ for all $i, j \in \{0, 1, \ldots, m\}$, since

$$\begin{aligned}
&\psi_i Q_{i,j}^{t+1} = \psi_i Q_{i,j}^t = \psi_j Q_{j,i}^t = \psi_j Q_{j,i}^{t+1} && \text{if } i \notin \mathcal{A}^t \text{ or } j \notin \mathcal{A}^t, \\
&\psi_i Q_{i,j}^{t+1} = \psi_i u Q_{i,j}^t = u \psi_i Q_{i,j}^t = u \psi_j Q_{j,i}^t = \psi_j u Q_{j,i}^t = \psi_j Q_{j,i}^{t+1} && \text{if } i, j \in \mathcal{A}^t, \\
&\psi_i Q_{i,i}^{t+1} = \psi_j Q_{j,j}^{t+1} && \text{for } i = j.
\end{aligned}$$

Now the detailed balance condition for Ψ holds for $Q^0 = Q$ by assumption, as given by (28), and since $\widetilde{Q} = Q^k$ for some $k \in \{0, 1, \ldots, m\}$, the condition (29) holds by induction. □

From the peskunization procedure, the second transition alternative, $\mathcal{T}2$, can be specified from the first, $\mathcal{T}1$, in the following way.

Transition alternative 2 ($\mathcal{T}2$) Let $\Psi(x \to \mathcal{Y}) = (\psi_0(x \to \mathcal{Y}), \ldots, \psi_m(x \to \mathcal{Y}))$ be the transition probabilities corresponding to $\mathcal{T}1$, as given by (27), and let $Q(x \to \mathcal{Y})$ be the $(m+1) \times (m+1)$ matrix with each row equal to $\Psi(x \to \mathcal{Y})$, i.e., $Q_{i,j}(x \to \mathcal{Y}) = \psi_j(x \to \mathcal{Y})$ for all $i, j \in \{0, 1, \ldots, m\}$. The choice of transition probabilities that define $\mathcal{T}2$ will then be

$$p_j(x \to \mathcal{Y}) = \tilde{p}_{0,j}(x \to \mathcal{Y}), \quad j \in \{0, 1, \ldots, m\}, \tag{33}$$

where $\tilde{p}_{i,j}(x \to \mathcal{Y})$ is the elements of the peskunization $\widetilde{Q}(x \to \mathcal{Y})$ of $Q(x \to \mathcal{Y})$.

As pointed out by Tjelmeland (2004), the Theater-method with $\mathcal{T}2$ will correspond to a standard Metropolis-Hastings algorithm when $m = 1$. The detailed balance condition will continue to hold for $\mathcal{T}2$.

Proposition 5.4.8. *The transition rule given by Algorithm 4 (the Theater-method) with transition probabilities given by $\mathcal{T}2$ (Transition alternative 2) satisfies the detailed balance condition.*

Proof. Clearly, $Q(x \to \mathcal{Y})$ satisfy the detailed balance condition for $\Psi(x \to \mathcal{Y})$, since for all $i, j \in \{0, 1, \ldots, m\}$,

$$\psi_j(x \to \mathcal{Y}) Q_{j,i}(x \to \mathcal{Y}) = \psi_j(x \to \mathcal{Y}) \psi_i(x \to \mathcal{Y}) = \psi_i(x \to \mathcal{Y}) Q_{i,j}(x \to \mathcal{Y}),$$

and so by Proposition 5.4.7, $\widetilde{Q}(x \to \mathcal{Y})$ also satisfies the detailed balance condition for $\Psi(x \to \mathcal{Y})$. Consequently, for each $j \in \{1, \ldots, m\}$, it then follows from (27) and (33), that

$$\begin{aligned}
\pi(x) p_j\left(x \to \mathcal{Y}_{y_j:=y}\right) q(x \to \mathcal{Y}_{y_j:=y}) &= \pi(x) \tilde{p}_{0,j}\left(x \to \mathcal{Y}_{y_j:=y}\right) q(x \to \mathcal{Y}_{y_j:=y}) \\
&= \tilde{p}_{0,j}\left(x \to \mathcal{Y}_{y_j:=y}\right) \psi_0\left(x \to \mathcal{Y}_{y_j:=y}\right) \\
&= \tilde{p}_{j,0}\left(x \to \mathcal{Y}_{y_j:=y}\right) \psi_j\left(x \to \mathcal{Y}_{y_j:=y}\right) \\
&= \tilde{p}_{j,0}\left(x \to \mathcal{Y}_{y_j:=y}\right) \pi(y) q(y \to \mathcal{Y}_{y_j:=x}) \\
&= \tilde{p}_{0,j}\left(y \to \mathcal{Y}_{y_j:=x}\right) \pi(y) q(y \to \mathcal{Y}_{y_j:=x}), \\
&= \pi(y) p_j\left(y \to \mathcal{Y}_{y_j:=x}\right) q(y \to \mathcal{Y}_{y_j:=x}) \qquad (34)
\end{aligned}$$

where we have used that $\tilde{p}_{j,0}\left(x \to \mathcal{Y}_{y_j:=y}\right) = \tilde{p}_{0,j}\left(y \to \mathcal{Y}_{y_j:=x}\right)$. From (34), the detailed balance of the chain (with respect to the target distribution) follows from Proposition 5.4.4. □

The transition probabilities do not need to depend on x and $\mathcal{Y}$ only through π and q, which then is the case for $\mathcal{T}1$ and $\mathcal{T}2$. For instance, it is possible to assign more weight to the proposals that result in the most radical state changes. One such alternative is proposed by Tjelmeland (2004), in the form of a linear optimization problem. However, when imposing such regulations on the Markov chain, the detailed balance will easily be lost, and the correct stationary distribution must then be proved more directly.

6 Multiple proposal strategies

For both of the multiple proposal methods described in the previous section, some consequential choices have to be made before any implementation is possible. For the Kingpin-method, a non-negative, symmetric function $\lambda(x, y)$ and a proposal distribution $q(x \to \mathcal{Y})$ for which the proposals are exchangeable need to be selected, and similarly, for the Theater-method, where a set of transition probabilities $p_j\,(y \to \mathcal{Y})$ and a suitable proposal distribution $q(x \to \mathcal{Y})$ are required. In this section, we will examine different types of proposal strategies for the two methods, that is, different choices of $q(x \to \mathcal{Y})$.

6.1 Random exploration

A particularly simple form of proposal strategy would be to just sample each of the proposals independently from some suitable distribution, that is, to take

$$q(x \to \mathcal{Y}) = \prod_{j=1}^{m} q_0(x \to y_j),$$

where $q_0(x \to y_j)$ is the marginal density of y_j at x. Alternatively, as a simple modification of the plain i.i.d.sampling scheme, it is possible to sample the proposals i.i.d.*conditionally* on some other random quantity Φ, i.e.,

$$q(x \to \mathcal{Y}) = \int \prod_{j=1}^{m} q_\phi(x \to y_j) f(x \to \phi) \mathrm{d}\phi,$$

where $f(x \to \phi)$ is the density of ϕ at x and $q_\phi(x \to y_j)$ is the marginal density of y_j at x given ϕ. If $q_\phi(x \to y_j) = f(\phi \to y_j)$, then clearly the joint distribution function $q(x \to \mathcal{Y})$ will be symmetric.

6.2 Systematic exploration

One of the strongest points of the multiple proposal methods is, as already mentioned, that the proposals may be prominently dependent. In this section, we will introduce two different approaches for a systematic exploration of the neighborhood of the current state; QMC point set utilization and maximally spread directions.

6.2.1 QMC point sets

Let the function $g_x\,(u)$ denote the inverse of the cumulative distribution function of the density function $q_0(x \to y)$, i.e.,

$$U \sim \mathrm{Unif}\,([0, 1)) \quad \Rightarrow \quad Y = g_x\,(U) \sim q_0(x \to y).$$

Let $\mathcal{U} = \{u_1, \ldots, u_m\}$ be some (QMC) point set such that $\mathcal{U} \subseteq [0, 1)^d$, and let $\widetilde{\mathcal{U}}\,(\phi) = (\tilde{u}_1(\phi), \ldots, \tilde{u}_m(\phi))$ be the Cranley-Patterson rotation of $\mathcal{U}$ with rotation vector given by ϕ, i.e.,

$$\tilde{u}_i(\phi) = u_i + \phi \mod 1, \qquad i \in \{1, \ldots, m\}.$$

The following QMC schemes are then possible designs for the multiple proposal methods, with the Kingpin edition given as proposed in Craiu and Lemieux (2005).

QMC-Kingpin Let $\mathcal{U} = \{u_1, \ldots, u_m\}$ be a QMC point set.

Generate ϕ from Unif $\left([0,1)^d\right)$
Take proposal set $y_j = g_x\left(\tilde{u}_i(\phi)\right), \quad j = 1, \ldots, m$
Compute transition weights $w(y_j \rightarrow x), \quad j = 1, \ldots, m$
Select y among $\{y_1, \ldots, y_m\}$ with probability proportional to $w(y_j \rightarrow x)$
Find w such that $g_y\left(\tilde{u}_m(w)\right) = x$
Take dummy set $z_j = g_y\left(\tilde{u}_j(w)\right), \quad j = 1, \ldots, m$

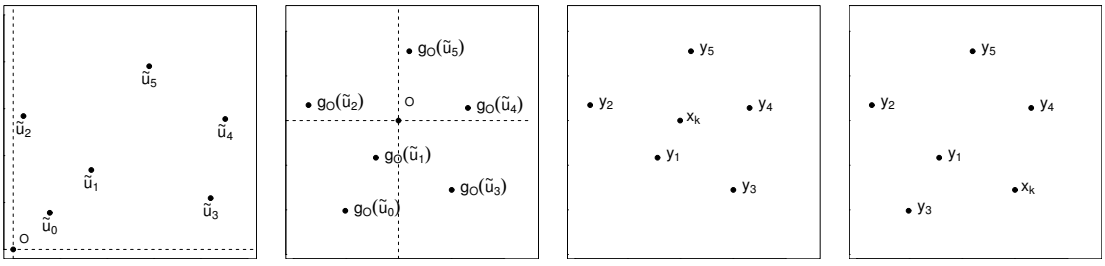
The general idea is as follows. Generate the proposal set $\mathcal{Y}$ from the randomized point set $\widetilde{\mathcal{U}}(\phi)$ and select the final proposal y. Find the rotation vector w that would have produced x as the last proposal from y, and then generate the dummy set $\mathcal{Z}$ from y with the rotation vector w. The last dummy, z_m, will then be equal to x by definition of w. Moreover, the joint distribution of $\mathcal{Z}$ at y will be the same as the joint distribution of $\mathcal{Y}$ at x.

QMC-Theater Let $\mathcal{U} = \{u_0, \ldots, u_m\}$ be a QMC point set, and let $\mathcal{O}$ denote the origin.

Generate ϕ from Unif $([0,1))$
Generate s from Unif $(\{0, 1, \ldots, m\})$
Take $\Delta = x - g_{\mathcal{O}}\left(\tilde{u}_s(\phi)\right)$
Take proposal set $y_j = g_{\mathcal{O}}\left(\tilde{u}_j(\phi)\right) + \Delta, \quad j = 0, 1, \ldots, m$
Swap y_s and y_0, so that $y_0 = x$ and $y_s = g_{\mathcal{O}}\left(\tilde{u}_0(\phi)\right) + \Delta$

The general idea is as follows. Generate the randomized point set $\widetilde{\mathcal{U}}(\phi)$ of size $m+1$ and select one of them at random, as indicated by s. Translate the point set $\widetilde{\mathcal{U}}(\phi)$ so that sth point is relocated to x, and take the proposal set as the remaining m translated points. By employing the random spatial locator s, the joint proposal density $q(x \rightarrow \mathcal{Y})$ will be symmetric.

The QMC point set utilizations are illustrated in Figure 10 for dimension $d = 2$.



(a) Randomized QMC point set. **(b)** Transformed point set. **(c)** Kingpin scheme. **(d)** Theater scheme.

Figure 10: Employment of QMC point sets for the multiple proposal methods with $m = 5$. Subfigures (b)-(d) are plotted on a different scale than subfigure (a). **(a)** : Randomized QMC point set $\widetilde{\mathcal{U}} \subseteq [0,1)^2$, with the 0th point only included in the Theater case. **(b)** : Transformed point set, $\{g_{\mathcal{O}}\left(\tilde{u}_j\right)\}$. **(c)** : Utilization of the transformed point set with the Kingpin-method. For simplicity, we have assumed that $g_x\left(\tilde{u}_j\right) = x + g_{\mathcal{O}}\left(\tilde{u}_j\right)$, which for instance is the case when $q_0(x \rightarrow y)$ is the density function of a Gaussian distribution centered at x. **(d)** : Utilization of the transformed point set with the Theater-method. In particular, the spatial locator s is given as $s = 3$, so that the 0th point and the 3rd point are swapped.

Technically, for the proposals to be exchangeable, it would be necessary to randomly permute the QMC point set at each step. In practice, however, the order in which the *new* proposals are produced will not be important, and so the above procedure may be employed without the minor modification (see Craiu and Lemieux (2005) for an appropriate proof).

For both of the above schemes, the corresponding QMC point set $\mathcal{U}$ is constructed only once, before the actual Markov chain simulation is undertaken. In fact, it will not be necessary to generate the point sets by a QMC construction method. Any point set that is well-distributed in the unit hypercube may successfully be employed. For instance, some kind of simple grid structure will be adequate. However, a regular grid may require an impractically large value m of proposals to excel, and need not be a better choice in practice. The general strategy will be to start off with a point set of an appropriate size that is as well-distributed over the unit hypercube as possible. The point sets of the QMC construction methods will then typically be suboptimal, but they are simple to apply, and may very well be sufficiently well-distributed for practical purposes.

6.2.2 Maximally spread directions

A perhaps more intuitive way of exploring the neighborhood systematically will be by employing a set of maximally spread search directions, similar in spirit to the employment of QMC point sets. We will sketch a construction method for $k \leq d+1$ such points on the unit sphere centered at the origin, $\mathbb{S}^d$. Of course, if $k = 1$ or $k = 2$, the construction task is trivial: just draw a random vector y from $\mathbb{S}^d$ and take the first point as y and the second as $-y$. For $k \geq 3$, the points can be constructed as given by Algorithm 6, where $|\cdot|$ denotes vector length operator, i.e., the Euclidean distance from the origin. The principles of construction is illustrated in Figure 11.

Algorithm 6 Generation of $3 \leq k \leq d+1$ maximally spread points on $\mathbb{S}^d$

Generate $k-1$ (linearly independent) direction vectors $\{\hat{e}_1, \ldots, \hat{e}_{k-1}\}$ from Unif$\left(\mathbb{S}^d\right)$
Construct orthonormal set $\{e_1, \ldots, e_{k-1}\}$ from $\{\hat{e}_1, \ldots, \hat{e}_{k-1}\}$
Take $y_1 = e_1$ and $y_2 = -e_1$
for $i = 3$ to k **do**
  Set new point $y_i = e_{i-1}$
  Determine transformation parameters $\gamma, \rho \in (0,1)$ for the old points $\{y_j\}_{j=1}^{i-1}$ from

$$|\gamma(y_1 - \rho e_{i-1}) - e_{i-1}| = |\gamma(y_1 - \rho e_{i-1}) - \gamma(y_2 - \rho e_{i-1})| \qquad \text{(equidistance)}$$
$$|\gamma(y_j - \rho e_{i-1})| = 1 \qquad \text{(unit sphere)}$$

  for $j = 1$ to $i-1$ **do**
    $y_j = \gamma(y_j - \rho e_{i-1})$
  end for
end for

The appropriateness of the method can be seen by noticing that $k \leq d+1$ points in $\mathbb{R}^d$ always will define a k dimensional hyperplane, and that the maximally spread points on $\mathbb{S}^d$ then necessarily will form an equidistant point set. The latter declaration is analogous with the fact that of all triangles with vertices on a circle, the equilateral triangle will be the one where the sum of the edge lengths is the greatest.

By some rather straightforward calculations, using the orthonormality of the points y_1, y_2 and e_{i-1} at step i, the simultaneous solution of the equidistance condition and the unit sphere condition for the transformation parameters $\gamma, \rho \in (0, 1)$ can be seen to be

$$\gamma = \frac{2\sqrt{\alpha}}{\alpha + 1}, \qquad \rho = \frac{\alpha - 1}{2\sqrt{\alpha}},$$

with $\alpha = |y_1 - y_2|^2 - 1$. However, since the conditions themselves are more intuitive than the solution, we have retained the original equations in the procedure listing. The reason that we have put parentheses around the words *linearly independent* in the first statement of the procedure is that when $k - 1 \leq d$ directions vectors are drawn uniformly from $\mathbb{S}^d$, they will be linearly independent with probability one.

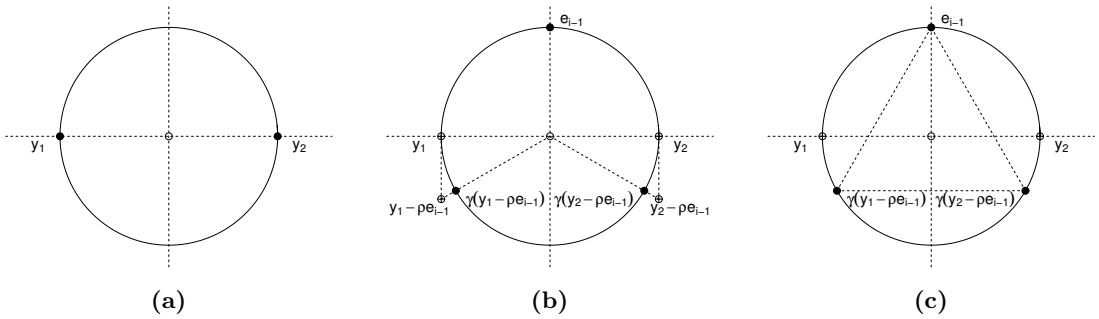


Figure 11: Construction principles of Algorithm 6. **(a)** : The point set generated at the preceding iteration is equidistant and defines a hyperplane. **(b)** A new point is then added, orthogonally on the hyperplane of the already given points. The previously generated points are translated in opposite direction of the point just inserted and scaled back to the unit sphere. **(c)** The extended point set defines a hyperplane in the incremented dimension and the points are equidistant.

By construction, the point set $\{y_1, \ldots, y_k\}$ generated with Algorithm 6 will be maximally spread on the unit sphere centered at the origin. Of course, the center of the points may easily be relocated to any point χ in $\mathbb{R}^d$ by a simple translation, i.e., $\hat{y}_j = y_j + \chi$. Similarly, it is possible to increase or decrease the span by means of standard scaling, either independently, i.e., $\hat{y}_j = c_j y_j$, $c_j \in \mathbb{R}$, or collectively, i.e., $\hat{y}_j = c y_j$, $c \in \mathbb{R}$, with $j = 1, \ldots, k$. Further on, it is clear from Algorithm 6 that the last point generated will be equal to the last direction vector, e_{k-1}, and so if one of the maximally spread points are required to be fixed, say, at the current state, then the last direction vector may be set accordingly.

By employing the points on a sphere surface, that is, points such that the difference from some center point is equal for all the points, the following simple schemes will give two possible utilizations of maximally spread points for the multiple proposal methods.

Max. spread-Kingpin

Generate m maximally spread points $\{\hat{y}_1, \ldots, \hat{y}_m\} \subset \mathbb{S}^d$
Generate some $c \in \mathbb{R}$, and take the proposal set $y_j = x + c\hat{y_j}, \quad j = 1, \ldots, m$
Compute transition weights $w(y_j \rightarrow x), \quad j = 1, \ldots, m$
Select y among $\{y_1, \ldots, y_m\}$ with probability proportional to $w(y_j \rightarrow x)$
Generate m maximally spread points $\{\hat{z}_1, \ldots, \hat{z}_m\} \subset \mathbb{S}^d$ with $\hat{z}_m = (x-y)/\lvert x-y \rvert$
Take dummy set $z_j = y + \lvert x - y \rvert \hat{z}_j, \quad j = 1, \ldots, m$

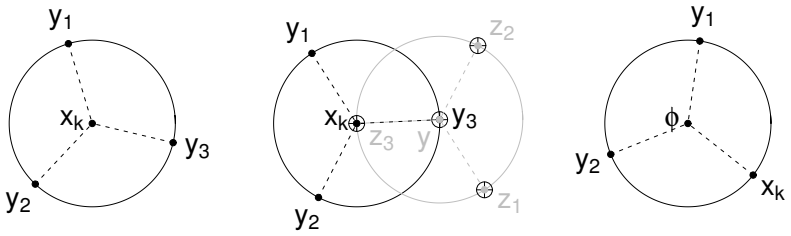
The general idea is as follows. Select a radius c and generate m maximally spread points on the sphere of radius c centered at x. Select one of them as final proposal y and generate m maximally spread points on the corresponding sphere centered at y with one of the points fixed as the current state x. The joint distribution of $\mathcal{Z}$ at y will then be the same as the joint distribution of $\mathcal{Y}$ at x. For computational reasons, it will be convenient to note that since the marginal density of each proposal will be the uniform density on some sphere, the simpler form of transition weights $\hat{w}(y_j \to x) = \pi(y_j)\lambda(y_j, x)$ and $\hat{w}(z_j \to x) = \pi(z_j)\lambda(z_j, x)$ can be employed, since the factor $q_0(y_j \to x) = q_0(z_j \to y)$ will be equal for each $j \in \{1, \ldots, m\}$, and it may therefore be canceled out.

Max. spread-Theater

Generate ϕ from $q_0(x \to \phi) = q_0(\lvert\phi - x\rvert)$
Set $u_{\phi \to x} = (x - \phi)/\lvert x - \phi\rvert$
Generate $m+1$ maximally spread points $\{\hat{y}_1, \ldots, \hat{y}_{m+1}\} \subset \mathbb{S}^d$ with $\hat{y}_{m+1} = u_{\phi \to x}$
Take proposal set $y_j = \phi + \lvert x - \phi\rvert \hat{y}_j, \quad j = 1, \ldots, m$

The general idea is as follows. Select at random a point ϕ in the neighborhood of the current state x. Generate $m + 1$ maximally spread points on the sphere with center at ϕ such that x is one of them. By letting the proposals be centered at ϕ with the corresponding density function $q_0(x \to \phi) = q_0(\lvert\phi - x\rvert)$ for ϕ, the joint proposal density $q(x \to \mathcal{Y})$ will be symmetric.

The maximally spread schemes are illustrated in Figure 12 for dimension $d = 2$, with $m = 3$ proposals for the Kingpin-method and $m = 2$ proposals for the Theater-method.



(a) Kingpin: Step 1 **(b)** Kingpin: Step 2-3 **(c)** Theater

Figure 12: Multiple proposals using maximally spread search directions. **(a)** Kingpin-method, step 1: The proposals are taken from a sphere centered at the current state x. **(b)** Kingpin-method, step 2-3: One of the $m = 3$ proposals is selected as final proposal y, and a set of dummy proposals are generated on a sphere centered at y in such a way that the last dummy coincide the current state x. **(c)** Theater-method: The $m = 2$ proposals are taken from a sphere centered at some point ϕ such that the current state x is equally far from each of the other proposals.

7 Numerical results

In this section, we will examine the performance of the multiple proposal methods numerically on a few test problems. In particular, we will compare the different proposal mechanisms described in the previous sections. For the sake of simplicity, we will restrict our attention to the setting where $\lambda(x, y) = 1$ for the Kingpin-method, and for the Theater-method, we will consider exclusively the two transition alternatives referred to as $\mathcal{T}1$ and $\mathcal{T}2$. Further on, we will let all marginal densities correspond to a Gaussian random walk model, i.e., $q_0(x \to y)$ will be taken according to a $N\left(x, \sigma^2 I\right)$ distribution for some value of the scalar parameter σ, where I denotes the identity matrix. All QMC point sets will be taken from the so-called Faure sequences, as presented by Glasserman (2003). Details on the multiple proposal algorithms that we will try out is given in Table 1.

Name	**Description**
KP-iid	Kingpin-method with proposals sampled i.i.d. from $N\left(x, \sigma^2 I\right)$.
KP-max	Proposals sampled according to the "Max. spread-Kingpin" scheme with $c \sim N\left(0, \sigma^2\right)$.
KP-qmc	Proposals sampled according to the "QMC-Kingpin" scheme with $g_x(u) = x + \sigma I \Phi^{-1}(u)$, where $\Phi^{-1}(u)$ is the inverse of the cumulative distribution function of the standard normal density.
T1-iid	Theater-method with proposals sampled i.i.d. conditionally on ϕ, with $\phi \sim N\left(x, \frac{1}{2}\sigma^2 I\right)$ and $y_j \sim N\left(\phi, \frac{1}{2}\sigma^2 I\right)$ for $j = 1, \ldots, m$. Transition probabilities $p(x \to \mathcal{Y})$ given by $\mathcal{T}1$.
T1-max	Proposals sampled according to the "Max. spread-Theater" scheme with $\phi \sim N\left(x, \sigma^2 I\right)$ and $p(x \to \mathcal{Y})$ given by $\mathcal{T}1$.
T1-qmc	Proposals sampled according to the "QMC-Theater" scheme with $g_{\mathcal{O}}(u) = \sigma I \Phi^{-1}(u)$, where $\Phi^{-1}(u)$ is the inverse of the cumulative distribution function of the standard normal density. Transition probabilities $p(x \to \mathcal{Y})$ given by $\mathcal{T}1$.
T2-iid	Same as *T1-iid*, except that $p(x \to \mathcal{Y})$ is given by $\mathcal{T}2$.
T2-max	Same as *T1-max*, except that $p(x \to \mathcal{Y})$ is given by $\mathcal{T}2$.
T2-qmc	Same as *T1-qmc*, except that $p(x \to \mathcal{Y})$ is given by $\mathcal{T}2$.

Table 1: List of algorithms.

The reason that we take the covariance matrices equal to $\frac{1}{2}\sigma^2 I$ for *T1-iid* is that the marginal distribution of each proposal y_j at x then will be $N\left(x, \sigma^2 I\right)$, due to the dependence structure with the random quantity ϕ. For the inverse of the normal cumulative distribution function, $\Phi^{-1}(u) = \left(\Phi^{-1}(u_{(1)}), \ldots, \Phi^{-1}(u_{(d)})\right)$, no closed-form solution exists, and so we will have to evaluate each component $\Phi^{-1}(u_{(i)})$ numerically. Whenever convenient, we will use the collective term *KP* to denote the algorithms associated with the Kingpin-method, and similarly, the terms *T1* and *T2* will then denote the algorithms associated with the Theater-method for $\mathcal{T}1$ and $\mathcal{T}2$, respectively.

7.1 Statistical efficiency

As our point of departure, we will assume that we are interested in finding the expectation of some random variable $g(X)$ with $X \sim \pi(x)$, i.e.,

$$\mu(g) = \int g(x)\pi(x)\mathrm{d}x,$$

and that we intend to approximate it by simulating a Markov chain (X_k). Following Liu (2001), we will start off by defining a measure on the statistical efficiency of the MCMC sample with respect to $\mu(g)$. Under the assumption that the chain is started from stationarity, such that $X_0 \sim \pi(x)$, we have that

$$\begin{aligned}\operatorname{Var}\hat{\mu}_n(g) = \operatorname{Var}\left(\frac{1}{n}\sum_{i=1}^{n} g(X_i)\right) &= \frac{\sigma^2(g)}{n}\left[1 + 2\sum_{i=1}^{n-1}(1 - \frac{i}{n})\operatorname{Cor}\left\{g(X_1), g(X_i)\right\}\right] \\ &\approx \frac{\sigma^2(g)}{n}\left[1 + 2\sum_{i=1}^{n-1}\operatorname{Cor}\left\{g(X_1), g(X_i)\right\}\right], \end{aligned} \tag{35}$$

where $\sigma^2(g)$ is the variance of $g(X)$. The term $\operatorname{Cor}\left\{g(X_1), g(X_i)\right\}$ denotes the correlation between $g(X_1)$ and $g(X_i)$, and so it represent the autocorrelation function of the time series $g(X_k)$ at lag i. By defining the *integrated autocorrelation time* of $g(X_k)$ as

$$\tau_{\text{int}}(g) = \frac{1}{2} + \sum_{i=1}^{\infty}\operatorname{Cor}\left\{g(X_1), g(X_i)\right\}, \tag{36}$$

we get from (35) that

$$\operatorname{Var}\hat{\mu}_n(g) \approx \frac{\sigma^2(g)}{2n/\tau_{\text{int}}(g)}, \tag{37}$$

which corresponds to an MC estimator based on $2n/\tau_{\text{int}}(g)$ *independent* samples, as described in Section 1.2. Consequently, we may introduce the following notion of statistical efficiency.

Definition 7.1.1 (Effective sample size)**.** *Let (X_k) be a Markov chain. The effective sample size of $(g(X_1), \ldots, g(X_n))$ is given as*

$$\eta_n(g) = \frac{2n}{\tau_{\text{int}}(g)},$$

where $\tau_{\text{int}}(g)$ is the integrated autocorrelation time of $g(X)$ as given by (36).

When considering the mixing properties of the multiple proposal methods on the different test problems, we will use the effective sample size as basis for comparison, with the integrated autocorrelation time estimated from the sample autocorrelation function. More precisely, we will compare the *relative* effective sample size, i.e., $\eta_n(g)/n$. The numbers we report need not be significant to the last digit, but we will take n large enough for the relative performance of the methods to be evident.

In general, the optimal value of the dispersal parameter σ will be different for each of the algorithms listed in Table 1. To make the comparison as just as possible, we will

take the estimate corresponding to the most beneficial σ-value for each algorithm as the relevant measure. However, if an unreasonably large σ-value is found to be optimal, then it may indicate that the boldness of the proposal mechanism is artificially high. Such a potentially unfavorable situation will be concealed by the fact that as long as one realistic transition is proposed at each step, the observed statistical efficiency will be unaffected by the appropriateness of the other proposals. As mentioned in the introduction of Section 5, it is possible to utilize the information obtained from the assessment of all proposed states, including the rejected ones, as suggested by Tjelmeland (2004). If such an approach is to be taken, then clearly, the information provided by each proposal will be of importance, and a moderate σ may be more sensible. We will not take this into account, as it will be a second-line topic.

The computational cost associated with the different methods will vary considerably. For instance, the Kingpin-method will be far more expensive than the Theater-method with respect to the number of necessary evaluations of the target density. Similarly, the transition step of the peskunized version of the Theater-method will be substantially more expensive than for the original. Such considerations will naturally be of great practical importance. However, we will not address these issues in detail, as the computational efficiency will depend heavily on the target distribution, and to some extent also on the environment of implementation. In any case, the computational aspects of the multiple proposal methods are not so much of primary concern for us as the actual effect on the mixing properties. As mentioned in Section 5, a method closely related to the Kingpin-method is given in Qin and Liu (2001), for which the old proposals may be reused for simplified computations.

For $m = 1$, each of the *KP* algorithms will correspond to a standard, single proposal Metropolis-Hastings algorithm (cf. Section 5.1). More precisely, for the specific models listed in Table 1, they will amount to a Metropolis-Hastings algorithm with proposal density $N\left(x, \sigma^2 I\right)$. Consequently, we may use the effective sample size of one of these methods as reference. In particular, we will employ k times the effective sample size of *KP-iid* with $m = 1$ as a benchmark for a multiple proposal method with $m = k \geq 1$. As pointed out in Section 5.4.3, the *T2* algorithms will also correspond to Metropolis-Hastings algorithms, but for *T2-max* and *T2-qmc*, the proposal densities will then be different from $N\left(x, \sigma^2 I\right)$.

For all the considered cases, direct i.i.d. sampling from the target distribution will be possible, and so MCMC simulation will in reality be unnecessary. Of course, the given distributions will still be valid for the purpose of testing MCMC algorithms.

7.2 Example I

The first task we will consider is to sample a bivariate random variable X from a Gaussian mixture distribution, which somewhat informally can be specified as

$$X \sim \frac{1}{2} N\left(\mu_1, \Sigma_1\right) + \frac{1}{2} N\left(\mu_2, \Sigma_2\right), \qquad \Sigma_j = \begin{bmatrix} 1 & \rho_j \\ \rho_j & 1 \end{bmatrix}, j = 1, 2,$$

with $\mu_1 = (-2, -4)$, $\mu_2 = (2, -4)$, $\rho_1 = 0.85$ and $\rho_2 = -0.85$. The target distribution will then be equal to the mixture distribution illustrated in Figure 5. As test function

$g\colon \mathbb{R}^2 \to \mathbb{R}$, we will take the indicator function on the left half plane, i.e.,

$$g(x) = \begin{cases} 1, & x_{(1)} > 0, \\ 0, & x_{(1)} \leq 0, \end{cases}$$

where we employ the coordinate notation $x_{(i)}$ introduced in Section 1.1. The test function will then be 1 where the left component dominates and 0 where the right component dominates. As a time series, $g(X)$ will then express the way that the chain moves between the two modes of the distribution.

Numerical results from the simulation of $g(X)$ on the mixture distribution with $n = 10^6$ and $\sigma^2 \in \{0.1, 0.3, 0.5, 0.7, 1, 1.5, 2, 3, 4, 6, 8, 10, 20, 30, 40, 80\}$ are given in Table 2 and 3. The performance of the algorithms is additionally illustrated in Figure 13.

Alg.$\backslash m$	1	2	3	5	7	10	15	20
KP-iid	0.036	0.066	0.090	0.119	0.148	0.183	0.226	0.255
KP-max	0.036	0.073	0.105					
KP-qmc	0.037	0.073	0.098	0.114	0.147	0.183	0.207	0.225
T1-iid	0.024	0.045	0.063	0.096	0.121	0.153	0.189	0.229
T1-max	0.023	0.048						
T1-qmc	0.013	0.024	0.033	0.080	0.102	0.165	0.230	0.250
T2-iid	0.037	0.067	0.097	0.143	0.177	0.215	0.279	0.325
T2-max	0.038	0.075						
T2-qmc	0.018	0.034	0.044	0.113	0.144	0.239	0.336	0.357

Table 2: Relative effective sample size of $g(X)$ for the multiple proposal algorithms on the Gaussian mixture distribution. Estimated for different number of proposals, m, using samples of size $n = 10^6$.

Alg.$\backslash m$	1	2	3	5	7	10	15	20
KP-iid	6	10	10	10	20	20	20	20
KP-max	6	8	10					
KP-qmc	10	10	10	20	20	20	20	20
T1-iid	10	10	10	10	10	10	20	20
T1-max	2	3						
T1-qmc	1	2	1.5	3	4	6	6	8
T2-iid	8	8	10	10	10	10	20	20
T2-max	2	3						
T2-qmc	1	1.5	1.5	3	4	6	8	10

Table 3: Optimal value of the dispersal parameter σ^2 corresponding to the effective sample size reported in Table 2 .

The empty slots of Table 2 and 3 is due to the fact that we only have specified the maximally spread methods for $k \leq d + 1$ directions. Of course, for the specific example, where $d = 2$, it would be straightforward to generate an arbitrary number of maximally spread direction vectors by means of standard rotation. For consistency, we will restrict ourselves to the outlined procedure.

As can be seen from Table 2 and Figure 13, and as should be clear from intuition, the collective distribution property of the QMC point sets will not be relevant for small values of m. For larger m, the QMC methods seems to be more distinguished. For the largest values of m, the gain of the correlated proposals seems to diminish, as the i.i.d. proposals

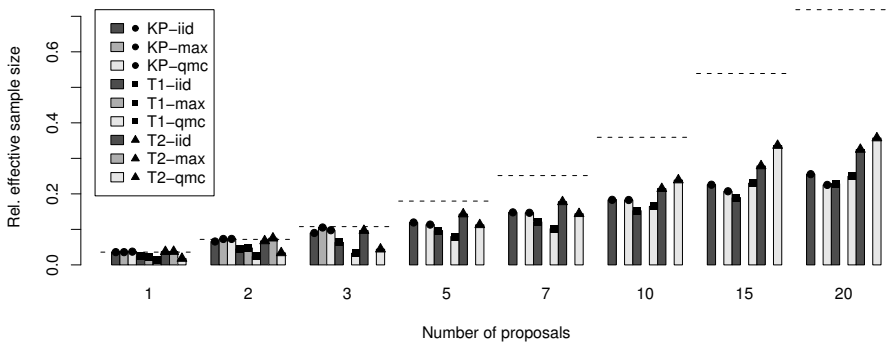


Figure 13: Graphical representation of the simulation results given in Table 2. The proposal type of the different algorithms is indicated by shading, while the transition type is indicated by a dot-like symbol on top of each bar. Reference lines for the single proposal benchmark are indicated by stippled lines.

collectively will function better. The trend of improvement for the QMC schemes is most pronounced for the Theater-method. We will make some supplementary comments on the behavior of the Kingpin QMC scheme in the next section. For both of the distinct methods, the maximally spread algorithms seem to be beneficial for the values of m that are supported.

As expected, the peskunized versions of the Theater-method, *T2*, will be superior to the native version, *T1*. It also seems that each of the *KP* methods in general performs better than the corresponding *T1* method, but that the *T2* method performs even better. For $m = 1$, the *KP* algorithms appear more efficient than the *T1* algorithms, which then agree with the fact that the Barker algorithm at least will be no better than the Metropolis-Hastings algorithm with respect to asymptotic variance.

For $m > 3$, the single proposal equivalent, as indicated by the stippled lines of Figure 13, appears to represent the most efficient sampling strategy, at least with the understanding that n iterations with m proposals amounts computationally to mn single iterations. Of course, such calculations need not hold in practice. Moreover, the multiple proposal methods are well suited for parallel computation, and so they may still be given preference.

From table 3, it is seems that some of the *KP* methods typically have a higher optimal value of σ than the *T1* and *T2* methods, at least for the given problem. This may then be a result of the slight difference between them when it comes to the area of local exploration. In particular, the proposals of the Kingpin methods are generally centered at the current state x, while the Theater methods we are considering will typically explore some random neighborhood of x, as given by the random vectors ϕ for the conditionally i.i.d. and maximally spread cases, and $\Delta(s)$ for the QMC case. The smaller values of σ may imply that the *T1* and *T2* methods are a little less daring than the *KP* methods, and so it is possible that the information contained in the rejected proposals are greater for the former two. As mentioned in Section 7.1, this would be of importance if the extended sample information were to be utilized. We will not make further comment on the issue, since we are not considering such applications.

7.3 Example II

Let X be a d-variate random variable with a normal distribution with expectation 0 and covariance matrix Σ, i.e.,

$$X \sim N(0, \Sigma).$$

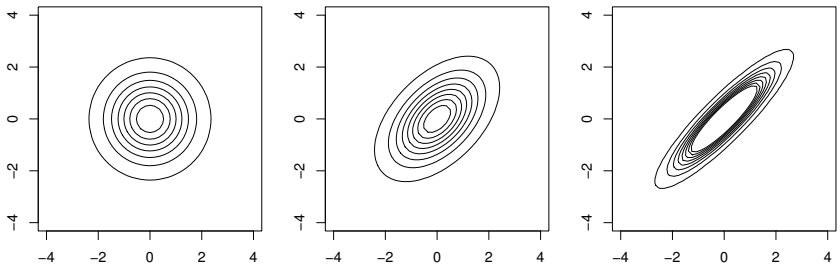
We will take the covariance matrix of X to be on the form

$$\Sigma = \begin{pmatrix} 1 & & \rho & \\ & 1 & & \\ & & \ddots & \\ \rho & & & 1 \end{pmatrix},$$

so that the variance of each component of X is 1 and the covariance between component i and component j is ρ for $i \neq j$. For our second test problem, we shall assume that we want to estimate the mean of the random variable $g(X)$, where $g\colon \mathbb{R}^2 \to \mathbb{R}$ is given by

$$g(x) = x_{(1)}.$$

For increasing value of the correlation parameter, ρ, the probability mass of X will be less uniformly distributed around the mean. In particular, the probability mass will be increasingly concentrated along a line, as illustrated in Figure 14 for $d = 2$. Consequently, we expect that the gain of the algorithms that employ systematic exploration, i.e., the QMC and maximally spread methods, will increase with the value of ρ.



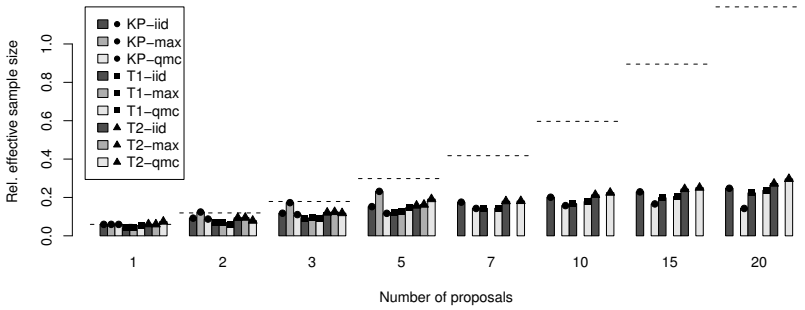
(a) $\rho = 0$ **(b)** $\rho = 0.5$ **(c)** $\rho = 0.9$

Figure 14: Contour plots for the simple Gaussian distribution $N(0, \Sigma)$ in dimension $d = 2$ for correlation parameter $\rho = 0, 0.5, 0.9$. For increasing coordinate correlation, the probability mass will be increasingly concentrated along the line given by the eigenvector of Σ that corresponds to the largest eigenvalue, that is, along the vector $(1, 1)$.

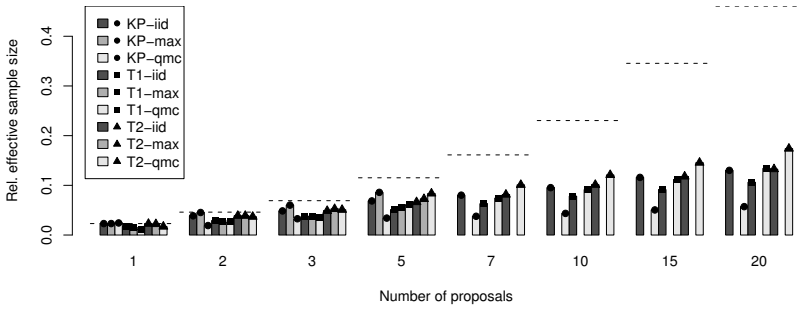
Numerical results from a series of simulations of $g(X)$ with the multiple proposal algorithms for $\rho \in \{0.0, 0.5, 0.9\}$ in dimension $d = 5$ are shown in Figures 15, respectively. For reference, we will include the numerical values of the relative effective sample size and the corresponding σ-value in tabular form in Appendix A-3. We also surveyed higher dimensions, i.e., $d = 10, 15, 20$, and we found that the trend was the same as for $d = 5$.

As shown by Figures 15, the gain of strongly correlated proposals appears to increase with ρ, as anticipated. For $\rho = 0.9$, the Theater QMC schemes can be seen to outperform even the single proposal equivalent. Further on, the *KP-max* algorithm appears to be better than *KP-iid* for the values of m that are supported.

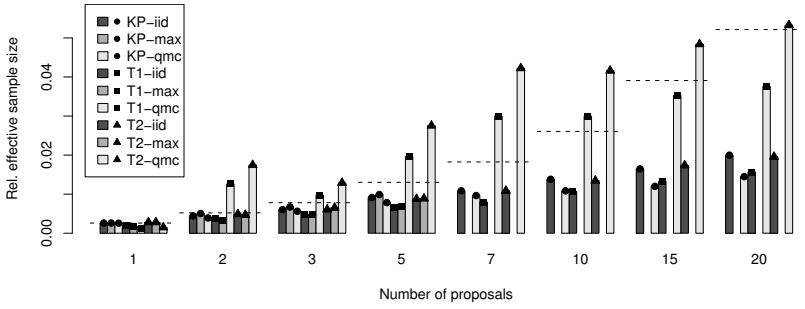
On the other hand, the Kingpin QMC scheme appears to be inferior to the i.i.d. scheme, which then is a rather unexpected situation. We can think of no apparent reason why



(a) $\rho = 0$



(b) $\rho = 0.5$



(c) $\rho = 0.9$

Figure 15: Relative effective sample size of $g(X)$ for the multiple proposal algorithms on the simple Gaussian distribution in dimension $d = 5$. Estimated for different number of proposals, m, and different values of the correlation parameter, ρ, using samples of size $n = 10^6$. The proposal type of the different algorithms is indicated by shading, while the transition type is indicated by a dot-like symbol on top of each bar. Reference lines for the single proposal benchmark are indicated by stippled lines. Numerical values are tabulated in Appendix A-3, Tables A-3.1 - A-3.6. **(a)** : Correlation parameter $\rho = 0$. **(b)** : Correlation parameter $\rho = 0.5$. **(c)** : Correlation parameter $\rho = 0.9$.

this should be the case, and it is attempting to explain is as an implementational mishap, especially since the maximally spread algorithm performs so much better. Despite a series of energetic and persistent investigations, we have not been able to point out any error in the code. We have therefore included the results of the simulations for the *KP-qmc* as recorded, but call attention to the somewhat suspicious behavior. The results of the other algorithms will in any case be unaffected. We should perhaps also mention that even though Craiu and Lemieux (2005) find that the QMC schemes generally perform better than the i.i.d. scheme in their numerical examination of the Kingpin-method, they also report reduced performance in one particular case.

From the above simulation results, it is clear that the multiple proposal methods may be used to construct Markov chains with more rapid mixing than the single proposal edition, especially when the target distribution is uneven or in some sense spatially isolated. Consequently, the employment of multiple proposals may be advisable in certain situations, even if either parallel computation or extended sample information will be utilized.

In a practical application, we expect that it will be favorable to conduct some initial experimentation with different values of m and σ (or any other relevant simulation parameters). Of course, efficient tuning of the parameters will be a full topic on its own, but a gross determination of suitable settings might be available from some relatively short test runs.

8 Summary

The robustness of the MCMC framework is impressive, as it provides the means to sample from practically any distribution. By constructing an irreducible, aperiodic Markov chain with the target distribution as its stationary distribution, random samples can be obtained by straightforward simulations. The statistical efficiency of the MCMC sampler will be governed by the mixing properties of the simulated chain.

The Metropolis-Hastings algorithm has unquestionably been one of the most successful MCMC algorithms, often with the employment of random walk proposals for otherwise intractable target distributions. However, there is an inherent trade-off between the radicalness of the proposal mechanism and the acceptance rate of the proposed transitions for such methods. To remedy the situation, the multiple proposal methods have recently been proposed as an alternative. The multiple proposal methods are designed to induce more rapid mixing by allowing more daring transition proposals. An essential property of the methods is then that highly correlated proposals are supported, providing a more systematic exploration on the momentary level.

In this paper, we have considered two such multiple proposal methods: the method of Tjelmeland (2004) and the method of Liu et al. (2000), which we have referred to as the Theater-method, and the Kingpin-method, respectively. As we have pointed out, there is also a close connection between them. Both of methods are valid under rather general conditions, with significant freedom of choice left for the user, especially with the Theater-method. For the Theater-method, a suitable pair of proposal density and transition rule has to be selected, while the Kingpin-method requires the specification of a proposal density with exchangeable proposals and a symmetric, non-negative function $\lambda(x, y)$. For the latter case, we have considered exclusively $\lambda(x, y) = 1$, and for the former case, we have considered two of the transition alternatives of Tjelmeland (2004). Of course, other choices may prove better. For instance, some kind of distance favorization could be advantageous with the transition rule of the Theater-method.

To motor the chains, we have considered different proposal strategies. Contrasted with the random exploration approach of i.i.d. proposal generation, we have seen how the principles of QMC can be utilized for a more systematic exploration of the local neighborhoods and improved mixing properties. In particular, we have described how QMC point sets and QMC randomization procedures may be employed, as well as the more direct approach of the maximally spread directions. For the presented QMC methods, the point sets do not actually need to be proper QMC point sets, as any well-distributed point set in the unit hypercube will be adequate.

The computational cost associated with the multiple proposal methods will typically increase with the number of proposals. Consequently, the single proposal methods may be more efficient for simple target distributions. Still, if the probability mass of the target density is "easy to miss" with random exploration, the multiple proposals may excel. From the numerical test cases, this could be seen for the larger values of the correlation parameter ρ. In practice, it will probably be a good idea to make some initial short-run experimentation with different values of the simulation parameters. Further on, the multiple proposal methods are well suited for for parallel computation, and the effective increase in the computational cost may therefore be reduced. However, the Kingpin-method will have to be synchronized when the final proposal is selected and at the acceptance/rejection step. Similarly, the Theater-method must be synchronized when

the transition probabilities are to be computed. The ability of parallel computation will be especially significant if the target density is expensive to evaluate.

A convenient property of the multiple proposal methods is that they are relatively simple to implement. In addition, it is possible to employ the extended sample information, and with such an application, the multiple proposal methods will be even more relevant.

Rather than a full assessment of the relative performance of the two multiple proposal methods, the main purpose of this paper has been to examine the potential improvement of the methods by utilizations of QMC ideas. From the numerical experimentation, the efficiency of the Kingpin-method appeared to be lower for the QMC scheme than for the i.i.d. scheme. We have no convincing arguments to account for the situation, and it is tempting to dismiss it as an error of implementation. However, we have not been able to detect any incorrectness. In any case, it is apparent from the remaining results that both of the multiple proposal methods indeed may be strengthened by the systematic exploration supplied by the heavy correlated proposals. For the Kingpin-method, this was evident from the results of the maximally spread scheme. The most pronounced improvement was recorded for the Theater-method with the QMC scheme, where the improvement was quite impressive in some of the cases. Put together, we find that the employment of multiple proposals may be advantageous in its own right, and particularly so when combined with the ideas of QMC.

A Appendix

A-1 Measure theory - a short review

In mathematics, a measure is a function that assigns a number, e.g., a "size", "volume", or "probability", to subsets of a given set. The concept has developed in connection with a desire to carry out integration over arbitrary sets rather than on an interval as traditionally done, and is important in mathematical analysis and probability theory. [Quotation, Wikipedia.]

Definition A-1.1 (Sigma algebra)**.** *Let Ω be an arbitrary set. A* sigma-algebra *(σ-algebra) in Ω is a collection $\mathcal{F}$ of subsets of Ω such that*

(i) $\emptyset \in \mathcal{F}$,

(ii) the union of countably many set in $\mathcal{F}$ is an element of $\mathcal{F}$,

(iii) the complement of any element of $\mathcal{F}$ is an element of $\mathcal{F}$.

Property *(ii)* of the above definition states that a σ-algebra is closed under countable unions, but the definition also implies that it will be closed under countable intersections. In fact, we may say that a σ-algebra is a collection of sets that are closed under countable set operations of with the empty set included. The "countable-closure" property has given rise to the term *sigma*, as capital sigma is the mathematical symbol of summation, which then stands as the prime representative of operators on countable sets. The smallest σ-algebra containing the open sets is called the Borel σ-algebra. Formally, the Borel σ-algebra, $\mathcal{B}$, can be defined as the intersection of all σ-algebras containing the open sets.

Definition A-1.2 (Borel sigma-algebra)**.** *Let Ω be an arbitrary set. The Borel σ-algebra in Ω is given by*

$$\mathcal{B}(\Omega) = \bigcap \{\mathcal{F} | \mathcal{F} \text{ is a } \sigma\text{-algebra in } \Omega \text{ containing the open sets}\}.$$

Definition A-1.3 (Measurable space)**.** *Let Ω be set and let $\mathcal{F}$ be σ-algebra in Ω. The pair $(\Omega, \mathcal{F})$ is called a* measurable space*. The elements of $\mathcal{F}$ are called* measurable sets*.*

Definition A-1.4 (Measure)**.** *Let $(\Omega, \mathcal{F})$ be a measurable space. A measure on $(\Omega, \mathcal{F})$ is a function $\mu \colon \mathcal{F} \to \mathbb{R} \cup \{\infty\}$ such that*

(i) $\mu(\mathcal{A}) \geq 0$ for all $\mathcal{A} \in \mathcal{F}$ with equality if $\mathcal{A} = \emptyset$,

(ii) $\mu\left(\bigcup_{i=0}^{\infty} \mathcal{A}_i\right) = \sum_{i=0}^{\infty} \mu\left(\mathcal{A}_i\right)$ *for any sequence of pairwise disjoint sets $\{\mathcal{A}_i\} \subseteq \mathcal{F}$.*

If $\mu(\Omega) = 1$, then the measure μ is called a probability measure *on $(\Omega, \mathcal{F})$.*

An important example of a measure is the so-called *Lebesgue measure*, which will be presented shortly after the concept of a *measure space*.

Definition A-1.5 (Measure space)**.** *Let μ be a measure on the measurable space $(\Omega, \mathcal{F})$. The triplet $(\Omega, \mathcal{F}, \mu)$ is called a* measure space*. If $\mu(\Omega) = 1$, then $(\Omega, \mathcal{F}, \mu)$ is called a* probability space*, and the measure μ is called a* probability measure*.*

Definition A-1.6 (Lebesgue outer measure on $\mathbb{R}^d$)**.** *Let $\mathcal{I}$ be a rectangle in $\mathbb{R}^d$, i.e., a subset of $\mathbb{R}^d$ which can be defined as the Cartesian product $\prod_{k=1}^{d}(a_k, b_k)$ of open intervals (a_k, b_k) with $a_k \leq b_k$ for each k, and define the volume of $\mathcal{I}$ as $vol(\mathcal{I}) = \prod_{k=1}^{d}(b_k - a_k)$. For any subset $\mathcal{A}$ of $\mathbb{R}^d$, the* Lebesgue outer (exterior) measure *of $\mathcal{A}$ is given by*

$$\mu^*(\mathcal{A}) = \inf\left\{\sum_{k=1}^{\infty} vol(\mathcal{I}_k) : \mathcal{A} \subset \bigcup_{k=1}^{\infty} \mathcal{I}_k, \right\},$$

where each $\mathcal{I}_k$ is a rectangle in $\mathbb{R}^d$.

The outer Lebesgue measure as given by Definition A-1.6 is in fact *not* a measure on $\mathbb{R}^d$, as it fails to completely satisfy condition *(ii)* of Definition A-1.4 on $\mathbb{R}^d$. On a large subset of $\mathbb{R}^d$, however, it is indeed a measure.

Definition A-1.7 (Lebesgue measurable set)**.** *A subset $\mathcal{A}$ of $\mathbb{R}^d$ is said to be* Lebesgue measurable *if for any $\epsilon > 0$, there exists an open set $\mathcal{G}$ such that $\mathcal{A} \subset \mathcal{G}$ and $\mu^*(\mathcal{G} \setminus \mathcal{A}) < \epsilon$, where μ^* denotes the Lebesgue outer measure on $\mathbb{R}^d$. It is common to denote the collection of all Lebesgue measurable sets on $\mathbb{R}^d$ by $\mathcal{L}(\mathbb{R}^d)$.*

Definition A-1.8 (Lebesgue measure on $\mathcal{L}(\mathbb{R}^d)$)**.** *The* Lebesgue measure *of a Lebesgue measurable set $\mathcal{A}$ of $\mathbb{R}^d$ is given as*

$$\mu(\mathcal{A}) = \mu^*(\mathcal{A}),$$

where μ^* denotes the Lebesgue outer measure on $\mathbb{R}^d$.

Consequently, the Lebesgue measure differs only from the Lebesgue outer measure by the the domain of definition. Practically all subsets of $\mathbb{R}^d$ are Lebesgue measurable, as for instance is the case with the elements of the Borel σ-algebra.

A-2 Probability theory - notions of convergence

Definition A-2.1 (Convergence in probability)**.** *A sequence of random variables,* $X_1, X_2, \ldots,$ *converges* in probability *to a random variable* X *if, for every* $\epsilon > 0$*,*

$$\lim_{n\to\infty} \Pr(|X_n - X| \geq \epsilon) = 0 \qquad \textit{or, equivalently,} \qquad \lim_{n\to\infty} \Pr(|X_n - X| < \epsilon) = 1.$$

Definition A-2.2 (Convergence in distribution)**.** *A sequence of random variables,* $X_1, X_2, \ldots,$ *converges* in distribution *to a random variable* X *if*

$$\lim_{n\to\infty} F_{X_n} = F_X(x)$$

at all points x *where* $F_X(x)$ *is continuous.*

Definition A-2.3 (Almost surely convergence)**.** *A sequence of random variables,* $X_1, X_2, \ldots,$ converges almost surely *to a random variable* X *if, for every* $\epsilon > 0$*,*

$$\Pr\left(\lim_{n\to\infty} |X_n - X| < \epsilon\right) = 1.$$

Definition A-2.4 (Norm of total variation)**.** *The norm of total variation between two probability measures* ν_1 *and* ν_2 *on the measure space* $(\Omega, \mathcal{F})$ *is given by*

$$\|\nu_1(\cdot) - \nu_2(\cdot)\|_{TV} = \sup_{\mathcal{A}\in\mathcal{F}} \|\nu_1(\mathcal{A}) - \nu_2(\mathcal{A})\|_{TV}.$$

A-3 Tabulated simulation data

The numerical values illustrated by the figures of Example II will be listed below. For ease, the relevant notation will be recaptured by the following summary.

Tabulated quantities:

- $\eta_n(g)/n$ - relative effective sample size
- σ^2 - proposal dispersal parameter

Simulation parameters:

- d - problem dimension
- ρ - target correlation parameter

The simulation data of Section 7.3 will be tabulated consecutively.

Alg.$\backslash m$	1	2	3	5	7	10	15	20
KP-iid	0.05967	0.09219	0.11782	0.15215	0.17542	0.20112	0.22918	0.24778
KP-max	0.05967	0.12401	0.17280	0.23170				
KP-qmc	0.05949	0.08740	0.11064	0.11717	0.14272	0.15751	0.16633	0.14257
T1-iid	0.04187	0.06860	0.09083	0.12159	0.14345	0.16692	0.19816	0.22396
T1-max	0.04146	0.06876	0.09365	0.12765				
T1-qmc	0.05321	0.05681	0.08870	0.14565	0.14373	0.17887	0.20466	0.23732
T2-iid	0.06027	0.09358	0.12102	0.15744	0.18036	0.21341	0.24382	0.27184
T2-max	0.05972	0.09451	0.12403	0.16244				
T2-qmc	0.07400	0.07884	0.11817	0.19120	0.18139	0.22474	0.25090	0.29659

Table A-3.1: Numerical estimates for $\eta_n(g)/n$, as shown in Figure 15a ($d = 5, \rho = 0.0$).

Alg.$\backslash m$	1	2	3	5	7	10	15	20
KP-iid	1.00	1.50	2.00	2.00	2.50	2.50	3.00	4.00
KP-max	1.00	1.50	1.50	2.00				
KP-qmc	1.00	1.50	2.00	3.50	4.00	4.00	5.00	5.50
T1-iid	1.00	1.50	1.50	2.00	2.00	2.50	2.50	2.50
T1-max	0.30	0.50	0.50	0.70				
T1-qmc	0.50	0.70	0.70	1.00	1.00	1.00	1.00	1.50
T2-iid	1.00	1.50	1.50	2.00	2.00	2.50	3.00	3.00
T2-max	0.30	0.50	0.50	0.70				
T2-qmc	0.50	0.50	0.70	1.00	1.00	1.00	1.50	1.50

Table A-3.2: Optimal σ^2-value corresponding to Figure 15a ($d = 5, \rho = 0.0$), with $\sigma^2 \in \{0.01, 0.05, 0.1, 0.3, 0.5, 0.7, 1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5, 5, 5.5, 6, 6.5\}$.

Alg.\m	1	2	3	5	7	10	15	20
KP-iid	0.02304	0.03863	0.04852	0.06860	0.08021	0.09534	0.11597	0.13011
KP-max	0.02304	0.04534	0.06019	0.08584				
KP-qmc	0.02429	0.01905	0.03273	0.03408	0.03772	0.04353	0.05048	0.05713
T1-iid	0.01710	0.02831	0.03795	0.05062	0.06301	0.07826	0.09139	0.10619
T1-max	0.01540	0.02802	0.03761	0.05521				
T1-qmc	0.01105	0.02642	0.03548	0.06103	0.07429	0.09144	0.11165	0.13406
T2-iid	0.02265	0.03873	0.04859	0.06628	0.08131	0.10106	0.11773	0.13257
T2-max	0.02206	0.03875	0.05244	0.07259				
T2-qmc	0.01695	0.03680	0.05081	0.08361	0.10097	0.12091	0.14562	0.17387

Table A-3.3: Numerical estimates for $\eta_n(g)/n$, as shown in Figure 15b ($d = 5, \rho = 0.5$).

Alg.\m	1	2	3	5	7	10	15	20
KP-iid	1.00	1.50	1.50	2.00	2.00	2.00	2.50	3.00
KP-max	1.00	1.00	1.00	1.50				
KP-qmc	1.00	1.50	1.50	2.00	2.00	2.50	2.00	2.50
T1-iid	1.00	1.00	1.50	1.50	2.00	2.00	2.00	2.50
T1-max	0.30	0.50	0.50	0.50				
T1-qmc	0.30	0.70	0.70	1.00	1.00	1.00	1.50	1.50
T2-iid	1.00	1.00	1.50	2.00	2.00	2.00	2.50	3.00
T2-max	0.30	0.30	0.50	0.70				
T2-qmc	0.30	0.70	0.70	1.00	1.00	1.00	1.50	2.00

Table A-3.4: Optimal σ^2-value corresponding to Figure 15b ($d = 5, \rho = 0.5$), with $\sigma^2 \in \{0.01, 0.05, 0.1, 0.3, 0.5, 0.7, 1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5, 5, 5.5, 6, 6.5\}$.

Alg.\m	1	2	3	5	7	10	15	20
KP-iid	0.00261	0.00444	0.00605	0.00915	0.01087	0.01381	0.01647	0.01999
KP-max	0.00261	0.00504	0.00672	0.00990				
KP-qmc	0.00257	0.00391	0.00562	0.00785	0.00965	0.01085	0.01201	0.01448
T1-iid	0.00189	0.00367	0.00488	0.00664	0.00797	0.01063	0.01327	0.01567
T1-max	0.00181	0.00327	0.00473	0.00681				
T1-qmc	0.00116	0.01261	0.00972	0.01965	0.02996	0.02997	0.03525	0.03757
T2-iid	0.00280	0.00488	0.00608	0.00876	0.01087	0.01346	0.01737	0.01956
T2-max	0.00280	0.00468	0.00653	0.00888				
T2-qmc	0.00149	0.01747	0.01289	0.02755	0.04225	0.04161	0.04839	0.05328

Table A-3.5: Numerical estimates for $\eta_n(g)/n$, as shown in Figure 15c ($d = 5, \rho = 0.9$).

Alg.\m	1	2	3	5	7	10	15	20
KP-iid	0.30	0.30	0.50	0.70	0.50	0.70	0.70	1.00
KP-max	0.30	0.50	0.30	0.50				
KP-qmc	0.30	0.30	0.70	0.50	0.70	0.70	0.70	1.50
T1-iid	0.30	0.30	0.50	0.50	0.50	0.50	0.70	1.00
T1-max	0.05	0.05	0.10	0.10				
T1-qmc	0.05	3.50	5.00	3.50	3.50	3.00	3.50	2.50
T2-iid	0.30	0.30	0.30	0.50	0.70	0.70	1.00	0.70
T2-max	0.10	0.10	0.10	0.30				
T2-qmc	0.05	4.50	3.50	3.50	4.00	3.50	3.00	2.50

Table A-3.6: Optimal σ^2-value corresponding to Figure 15c ($d = 5, \rho = 0.9$), with $\sigma^2 \in \{0.01, 0.05, 0.1, 0.3, 0.5, 0.7, 1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5, 5, 5.5, 6, 6.5\}$.

References

Anderson, E. C. "Monte Carlo Methods and Importance Sampling." Published on http://ib.berkeley.edu/labs/slatkin/eriq/classes/guest_lect/mc_lecture_notes.pdf (1999). Lecture Notes for Stat 578C Statistical Genetics, Berkely University of California.

Barker, A. A. "Monte Carlo calculations of the radial distribution functions for a proton-electron plasma." *Australien Journal of Physics*, 18:119–133 (1965).

Barndorff-Nielsen, O. E., Cox, D. R., and Kluppelberg, C. (eds.). *Complex Stochastic Systems*, chapter 1, 1–62. Chapman and Hall (2001).

Caflisch, R. E., Morokoff, W. J., and Owen, A. B. "Valuation of mortgage backed securities using Brownian bridges to reduce effective dimension." *J. Comp. Finance*, 1(1):27–46 (1997).

Casella, G. and Berger, R. L. *Statistical Inference*. Duxburt, second edition (2002).

Craiu, R. and Lemieux, C. "Acceleration of the Multiple-Try Metropolis using Antithetic and Stratified Sampling." (2005). Available on http://www.math.ucalgary.ca/~lemieux/myftp/PAPERS/mctm.pdf.

Geyer, C. http://www.stat.umn.edu/~charlie/ (2006).

Glasserman, P. *Monte Carlo Methods in Financial Engineering*. Springer (2003).

Hickernell, F. J. "A generalized discrepancy and quadrature error bound." *Math. Comput.*, 67(221):299–322 (1998).

Jiang, R. "Notes on MCMC (I): Introduction." (2003). Available on www-scf.usc.edu/~rongjian/phd/notes_mcmc_1.pdf.

Liu, J. S. *Monte Carlo Strategies in Scientific Computing*. Springer (2001).

Liu, J. S., Liang, F., and Wong, W. H. "The Multiple-Try Method and Local Optimization in Metropolis Sampling." *J. Am. Statist. Ass.*, 95:121–134 (2000).

Metropolis, N. C. "The Beginning of the Monte Carlo Method." *Los Alamos Science Special Issue*, 15 (1987).

Neal, R. M. "A New Proof of Peskun's Theorem Regarding the Asymptotic Variance of MCMC Estimators." Technical report, University of Toronto (2004). ISBA 2004, Viña Del Mar, Chile, May 2004. Available on http://www.cs.toronto.edu/~radford/ftp/peskun.pdf.

Niederreiter, H. *Random Number Generation and Quasi-Monte Carlo Methods*. Society for Industrial and Applied Mathematics (1992).

Owen, A. B. and Tribble, S. D. "A quasi-Monte Carlo Metropolis algorithm." *PNAS*, 102(25):8844–8849 (2005a).
URL `http://www.pnas.org/cgi/content/abstract/102/25/8844`

—. "Sampling strategies for MCMC." (2005b).
URL http://www-stat.stanford.edu/~owen/reports/

Peskun, P. H. "Optimum Monte-Carlo sampling using Markov chains." *Biometrika*, 607–612 (1973).

Qin, Z. S. and Liu, J. S. "Multipoint metropolis method with application to hybrid Monte Carlo." *J. Comput. Phys.*, 172(2):827–840 (2001).

Robert, C. P. and Casella, G. *Monte Carlo Statistical Methods*. Springer, second edition (2004).

Roberts, G. and Rosenthal, J. "Understanding MCMC." (2003). Slides for week-long U.K. post-graduate course.

Stroock, D. W. *A Concise Introduction to the Theory of Integration*. Birkhauser, third edition (1999).

Tjelmeland, H. "Using all Metropolis-Hastings Proposals to Estimate Mean Values." Statistics No. 4, Department of Mathematical Sciences, Norwegian University of Science and Technology, Trondheim, Norwway (2004). Available on http://www.math.ntnu.no/preprint/statistics/2004/S4-2004.ps.