Stat Comput (2006) 16: 193–202
DOI 10.1007/s11222-006-6966-6

Markov Chain Monte Carlo in small worlds

Yongtao Guan · Roland Fleißner · Paul Joyce · Stephen M. Krone

Received: November 2004 / Accepted: December 2005

Abstract As the number of applications for Markov Chain Monte Carlo (MCMC) grows, the power of these methods as well as their shortcomings become more apparent. While MCMC yields an almost automatic way to sample a space according to some distribution, its implementations often fall short of this task as they may lead to chains which converge too slowly or get trapped within one mode of a multi-modal space. Moreover, it may be difficult to determine if a chain is only sampling a certain area of the space or if it has indeed reached stationarity.

In this paper, we show how a simple modification of the proposal mechanism results in faster convergence of the chain and helps to circumvent the problems described above. This mechanism, which is based on an idea from the field of "small-world" networks, amounts to adding occasional "wild" proposals to any local proposal scheme. We demonstrate through both theory and extensive simulations, that these new proposal distributions can greatly outperform the traditional local proposals when it comes to exploring complex heterogenous spaces and multi-modal distributions. Our method can easily be applied to most, if not all, problems involving MCMC and unlike many other remedies which improve the performance of MCMC it preserves the simplicity of the underlying algorithm.

Keywords Markov Chain Monte Carlo · Metropolis-Hastings algorithm · Proposal distributions · Small-world networks · Importance sampling

Y. Guan · R. Fleißner · S. M. Krone
Department of Mathematics, P.O. Box 441103, University of Idaho, Moscow, ID 83844-1103
e-mail: yguan@uidaho.edu
e-mail: fleissne@uidaho.edu
e-mail: krone@uidaho.edu

P. Joyce (✉)
Department of Mathematics, P.O. Box 441103, University of Idaho, Moscow, ID 83844-1103
Department of Statistics, P.O. Box 441104, University of Idaho, Moscow, ID 83844-1104
e-mail: joyce@uidaho.edu

1. Introduction

Markov Chain Monte Carlo (Gamerman, 1997) is a sampling scheme for surveying a space S with a prescribed probability measure π. It has particular importance in Bayesian analysis, where $x \in S$ represents a vector of parameters and $\pi(x)$ is the posterior distribution of the parameters conditional on the data. MCMC can as well be used to solve the so-called missing data problem in frequentist statistics. Here, $x \in S$ represents the value of a latent or unobserved random variable, and $\pi(x)$ is its distribution conditional on the data. In either case, MCMC serves as a tool for numerical computation of complex integrals and is often found to be the only workable approach for problems involving a large space with a complex structure where traditional numerical methods are not possible.

As its name implies, MCMC does not attempt to draw elements of S independently of each other, but instead relies on a Markov chain which moves through S. Probably the most widely used version of MCMC is the Metropolis-Hastings algorithm (Hastings, 1970) which works in the following way: Suppose the chain is at a point $x \in S$, the algorithm then proposes a move to $y \in S$ following some proposal distribution $q(x, y)$. The move from x to y is either accepted or rejected and the acceptance probability is given by

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

Therefore, the constructed Markov chain moves from state x to state y with probability $T(x, y) := q(x, y)a(x, y)$. It is easy to check that the detailed balance equation $\pi(x)T(x, y) = \pi(y)T(y, x)$ holds. Hence, setting up the acceptance probability in the above way results in an ergodic reversible Markov chain (Meyn and Tweedie, 1996) with stationary distribution π provided that $q(x, y)$ is ergodic. This means that if we are able to run this chain long enough the frequency with which we observe $x \in S$ in our sample will converge to $\pi(x)$. However, if we do not pick the proposal distribution $q(x, y)$ wisely the Markov chain might reject most of the proposals and thus may not be active enough to reach the stationary distribution in a reasonable number of steps. Furthermore, if π has heavy tails or multiple modes it may become difficult for the chain to explore all the important regions of S.

To ensure activeness of the chain many implementations of the Metropolis-Hastings algorithm use local proposals. Assuming that S is a metric space with metric d, typically taken to be the Euclidean distance, we call a proposal distribution $q(x, y)$ a local proposal if $q(x, y)$ decreases rapidly with increasing distance between x and y, as is the case if $q(x, \cdot)$ is a normal distribution or if it has a compact support, i.e., $q(x, y) = 0$ if $d(x, y) > r$ for some finite r. If π is a smooth function then picking y in the neighbourhood of x will lead to a high acceptance rate $a(x, y)$. However, the resulting small step size means that it will take a large number of steps to move a substantial distance from the starting point. Therefore, it has been argued Jarner and Roberts (2001) that, at least if π is heavy tailed, the proposal distributions should have heavy tails, too.

In the next two sections, we will demonstrate how very simple heavy-tailed proposal distributions—namely mixtures of local proposals and random draws—outperform pure local proposal schemes, especially if π is multi-modal. We will first work out some mathematics to illustrate why these proposal distributions should perform better than traditional MCMC (section 2). Then we will present some simulations to support our argument (section 3).

Throughout this paper, we will assume that S, the state space of the Markov chain, is a space with some metric $d(\cdot, \cdot)$ and that it is equipped with the Borel σ-field $\mathcal{B}(S)$ and two measures: a canonical measure μ and a probability measure π, the stationary distribution of the Markov chain on S. Let $|B| := \mu(B)$ for any $B \in \mathcal{B}(S)$. We assume $|S| < \infty$. In the continuous case, S typically is a compact subset of $\mathbb{R}^n$ typically with Lebesgue measure as its canonical measure, while in the discrete case, S is a finite set with μ being counting measure. Let $N_x = \{y : d(x, y) < r\}$ denote the local neighborhood of x. We further assume that the standard local proposal $q(x, y)$ is the uniform distribution over N_x, i.e., $q(x, y) = 1/|N_x|$ if $y \in N_x$, 0 otherwise. In order to keep the notation simple the rest of this paper will only describe the discrete case. Yet, as any implementation of a continuous problem would necessitate the discretization of S, this does not affect the applicability of our method.

2. Metropolis-Hastings with small-world proposals

2.1. Small-world proposal distributions

Our choice for a modified proposal distribution is motivated by the so-called small-world networks (Watts and Strogatz, 1998). These graphs are characterized by a much shorter average path length than regular lattices in spite of retaining considerable regularity. They can be constructed by randomly rewiring a small fraction of the edges of a regular network. It turns out that replacing a relatively small number of edges with long-range connections results in a drastic decrease in the average path length.

In the context of the Metropolis-Hastings algorithm we may impose such a "small-world effect" by altering the proposal distribution as follows. With a large probability $1 - p$, a move is proposed according to the local proposal distribution. However, with some small probability p, we propose, for example through a random draw from S, a move that is typically far away from the current state. These "wild" proposals play the role of the long-range connections in the small-world networks. Precisely, if $q(x, y)$ is our local proposal distribution and $|S|$ represents the canonical measure of S then let

$$p(x, y) := (1 - p)q(x, y) + p/|S| \tag{2}$$

be the new proposal distribution. We call such a mixture of local proposals and random draws a *small-world proposal distribution*. In the following we refer to a Markov chain which uses small-world proposals as a small-world chain or SWC and we call a chain which only relies on local proposals a local-proposal chain or LPC. Below, we show how the probability of making a large jump in a SWC depends on π (section 2.2), then calculate the cost of a SWC in terms of the chain's average acceptance rate (section 2.3), and provide a rule for how to choose p (section 2.4).

2.2. The probability of large jumps

Suppose A and B are two disjoint subsets of the state space S. Let us also assume that none of the points in B lies in the neighbourhood of a point in A. Then the probability that a

SWC which is currently wandering in A jumps to a site in B in a single step is given by

$$\lambda = \sum_{x\in A}\sum_{y\in B}\frac{\pi(x)}{\pi(A)}\min\left(1,\frac{\pi(y)}{\pi(x)}\right)\frac{p}{|S|} \tag{3}$$

where $\pi(A)=\sum_{x\in A}\pi(x)$. To gain some insight into equation (3), consider three special cases:

1. If A corresponds to a flat region and B to a hill, then $\pi(y)>\pi(x)$, for $y\in B, x\in A$. From (3), we get $\lambda = p\frac{|B|}{|S|}$, which is proportional to the relative size of B.
2. If A is a hill and B is a flat region, then $\pi(y)<\pi(x)$, for $y\in B, x\in A$. From (3), we get $\lambda = p\frac{1}{|S|}\frac{1}{\pi(A)/|A|}\pi(B)$. Notice that $\pi(A)/|A|$ is the average probability, or average height, of A. The higher A, the more difficult it is for a chain to jump out. λ is also proportional to the total measure of the flat region B.
3. If both, A and B, are hills. Using the fact that

$$\min(a,b)=\frac{1}{2}\left((a+b)-|a-b|\right), \tag{4}$$

we get

$$\lambda = \frac{1}{2}p\frac{|B|}{|S|}\left(1+\frac{\pi(B)/|B|}{\pi(A)/|A|} - \frac{1}{\pi(A)|B|}\sum_{A}\sum_{B}|\pi(x)-\pi(y)|\right). \tag{5}$$

The last term in the parentheses is proportional to the average cross-variation between A and B. The second term in the parentheses is the ratio of the average height of hill B and the average height of hill A. $\frac{|B|}{|S|}$ is the relative size of set B. So if the cross-variation is small and if the ratio of the average heights is about 1, then we get $\lambda\approx p|B|/|S|$.

If we ignore all the paths that lead from A to B without a direct jump from the first subset to the latter, we may use λ as an indicator of how fast the SWC travels between distinct regions of S. In case (1) as well as in case (3), the speed is proportional to the relative size of the second hill. Note also that in case (3) the size of the valley between two hills does not influence the speed with which the chain jumps between the two hills. Notice also that the mean time to move from a flat spot to a hill depends only on the size of the space and not on its dimension. From case (2), we can see that the probability of jumping off a hill to a flat region is proportional to the total probability measure of the flat region. In the case that this total probability measure is small, most proposals of jumping off hills will be rejected. Hence, the Markov chain stays for a long time in important regions.

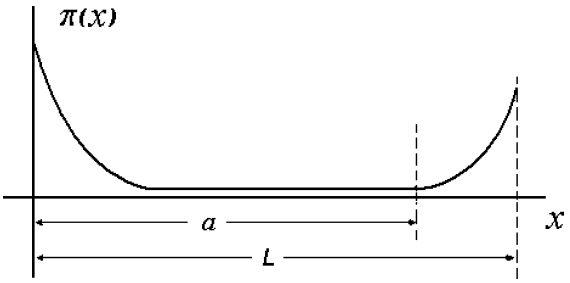

Fig. 1 $\pi(x)$ as used in example 1

That the situation is quite different for local proposals can be seen in the following simple example of a multi-modal space where the mean time to move from the 'major hill' to the 'minor hill' can be calculated explicitly. The point of the exercise is to demonstrate that even for simple spaces the mean time to move from one hill to the next will be an exponential function of the distance between the hills for a local proposal distribution. Whereas, a SWC will move from one peak to the other in a time that is linear in the distance between the two peaks. Here our neighborhood size is one unit in length. A larger neighborhood will only change the scale factor in the exponential distribution.

Example 1. Let $x\in S=[0,L)\cap\mathbb{Z}$ and suppose that $\pi(x)\propto \alpha e^{-\alpha x}$ for $0\le x\le a$ and $\pi(x)\propto e^{(x-L)}$ for $a<x<L$ (Figure 1). Consider a very simple proposal distribution where a move one unit to the right or left is made with equal probability. That is $p(k,k+1)=p(k,k-1)=0.5$. Note that $\frac{\pi(k+1)}{\pi(k)}=e^{-\alpha}$ for $k+1<a$. Now let T_k be the time it takes to move one step down the left hill from state k to state $k+1$. To calculate $E(T_k)$ we develop a recursion equation by conditioning on all the possible one step moves that can be made while in state k. In order to move one step to the right down the hill, we must propose a move to the right and then accept that move. If instead we propose a move to the right, but reject that move then the process starts anew. We may also propose a move to the left and climb back up the hill. Conditioning on these three possibilities leads to the following recursion for $E(T_k)$:

$$E(T_k)=\frac{1}{2}e^{-\alpha}+\frac{1}{2}\left(1-e^{-\alpha}\right)\left[E(T_k)+1\right] + \frac{1}{2}\left[E(T_{k-1})+E(T_k)+1\right]. \tag{6}$$

This implies

$$E(T_k)=e^{\alpha}E(T_{k-1})+2e^{\alpha}. \tag{7}$$

We are ultimately interested in the time it takes to move all the way down the hill to the reach the second mode. Denote this time by S_a where

$$S_a = \sum_{k=0}^{a-1} T_k.$$

Since $E(T_0) = e^\alpha$ it follows from (7) that

$$E(S_a) = e^\alpha E(S_{a-1}) + e^\alpha(2a-1).$$

Since this is a simple linear recursion we can solve it explicitly and get

$$E(S_a) = e^{\alpha a} + [e^\alpha(2a-1)]\left[\frac{e^{\alpha a}-1}{e^\alpha - 1}\right]. \tag{8}$$

The exponential function in this example descends much more slowly than the normal distribution which is often used to test MCMC algorithms. Still, equation (8) shows that the mean time to move between the two hills is an exponential function of their distance. If we replaced the exponential with the normal distribution the situation would be even worse giving $E(S_a) \approx e^{\alpha a^2}$. A SWC, on the other hand, would move between this example's two hills in a time whose mean is a linear function of their distance, as can be seen by replacing $|B|$ in the discussion of equation (3) with $L-a$ and $|S|$ with L.

2.3. Average acceptance rate

By (1) and (2), the average acceptance rate for a small-world chain at equilibrium can be expressed as:

$$a = \sum_{x\in S}\sum_{y\in S} \pi(x)\left((1-p)q(x,y) + \frac{p}{|S|}\right)\min\left(1, \frac{\pi(y)}{\pi(x)}\right). \tag{9}$$

Under the assumption that $q(x,y) = 1/|N_x|$ for $y \in N_x$, we get:

$$\sum_{x\in S}\sum_{y\in S} q(x,y)\min(\pi(x),\pi(y)) = 1 - \frac{1}{2|N_x|}\sum_{x\in S}\sum_{y\in N_x} |\,\pi(x)-\pi(y)\,| \tag{10}$$

and

$$\frac{p}{|S|}\sum_{x\in S}\sum_{y\in S}\min(\pi(x),\pi(y)) = p\left(1-\frac{1}{2|S|}\right)\sum_{x\in S}\sum_{y\in S}|\,\pi(x)-\pi(y)\,|. \tag{11}$$

Denote

$$V_l = \frac{1}{2|N_x|}\sum_{x\in S}\sum_{y\in N_x}|\pi(x)-\pi(y)|$$

$$V_g = \frac{1}{2|S|}\sum_{x\in S}\sum_{y\in S}|\pi(x)-\pi(y)|.$$

It is easy to see that V_l is the average local variation of π whereas V_g is its average global variation. Substituting them back into (9), we get the average acceptance rate as

$$a = 1 - V_l + p(V_l - V_g) \tag{12}$$

Notice that $1-V_l$ is the average acceptance rate for the local proposal. In the case where the probability distribution is not too noisy, we have $V_l < V_g$. Therefore, the average acceptance rate of the SWC decreases compared to that of the LPC and this is what we have to pay when using small-world proposals. However, this is a small price, since the dominant term $1-V_l$ is at least one order of magnitude larger than $p(V_l - V_g)$ and usually the two terms in (12) differ by two to three orders of magnitude. In the case where the probability distribution is very noisy, one may get $V_l > V_g$. In that case the best choice would be to increase p all the way up to 1 and end up with random sampling.

Thus, allowing for wild proposals usually decreases the average acceptance rate. One should, however, keep in mind that a high acceptance rate does not mean fast convergence (Roberts et al., 1997). In all of the examples described below, the LPC had a reasonably good acceptance rate. This only means that the local proposal was doing a good job sampling one of the regions, but because the local proposal did not reach the other regions the LPC did not converge. In fact, we could improve the acceptance rate by making the neighborhood size smaller for the local proposal distribution, but this would only serve to sample one of the regions more thoroughly.

2.4. Simple strategies for choosing p

In this section we assume that the space S is partitioned into two disjoint subsets A and B. B will represent the "important" region, which can be thought of as a collection of hills. A will be the "unimportant" region which can be thought of as the flat region of the space S. We will assume that $\pi(y) > \pi(x)$ for all $y \in B$ and $x \in A$. If we again ignore any paths from A to B which do not involve large jumps, we can use the result from section 2.2, that the probability of moving from a flat spot to a hill is $\lambda = \frac{|B|}{|S|}p$. Hence, the

mean time it takes to move from the unimportant region A to the important region B is approximately

$$\frac{1}{\lambda} = \frac{|S|}{|B|p}.$$

However, the SWC also bears a cost due to its lower acceptance rate. Here, we describe how to choose p so as to minimize the effect of this trade off. We view the time spent in the flat region and the proposed moves from the hill back to the flat region as wasted steps in the chain; all other steps in the chain are used to explore the important part of the space. Suppose that a SWC is run for M steps and let $h(p)$ be the mean number of wasted steps. Then

$$h(p) = \frac{|S|}{|B|p} + Mp\left(1 - \frac{|B|}{|S|}\right). \tag{13}$$

Denote by $r = |B|/|S|$ the fraction of the space that contains the hills. We now solve for p that minimizes $h(p)$. Indeed, since

$$h'(p) = -\frac{1}{rp^2} + M(1-r),$$

setting $h'(p) = 0$ and solving gives

$$p = \frac{1}{\sqrt{Mr(1-r)}} \approx \frac{1}{\sqrt{Mr}}.$$

For example, if you run the SWC for 1 million steps and the important region containing the hills represents one percent of the total then p should be set to 0.01. Normally, we will not know r in advance, but have to make a guess. However, if for example one's guess for r is two orders of magnitude too high, then, due to the square root, p will only be too low by one order of magnitude. Note also that in the simulations described in the next section we chose p in the range of 10^{-4} to 10^{-1} and no matter which value we picked the SWC always performed better than the LPC.

3. Simulations

3.1. A two dimensional distribution with four main modes

In the first simulation we use local-proposal and small-world chains to explore the probability distribution shown in Figure 2. The underlying space S is the grid $\{1, \ldots, 5000\} \times \{1, \ldots, 5000\}$.

For both types of chains, the local proposal was simply proposing one of the 8 surrounding neighbours with equal probability. For the SWC we occasionaly proposed a random point of the grid. Figure 3 shows the results for three runs of the LPC and three runs of the SWC with varying p.

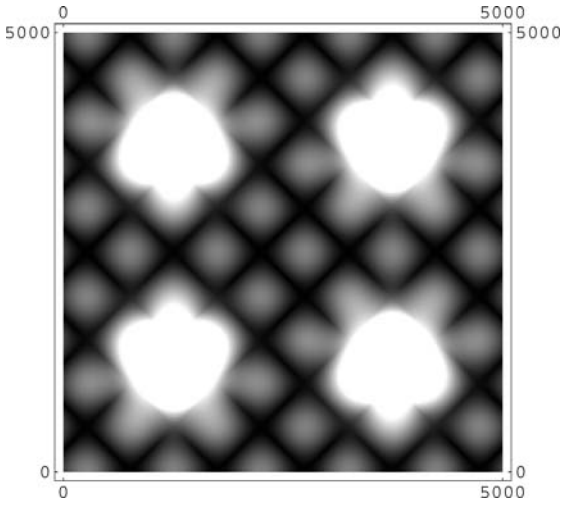

Fig. 2 The probability distribution used in the first simulation; the lightness of a pixel is proportional to its point mass

We can see that for this simulated distribution with four main modes, the LPC gets trapped within one peak, while the SWC surveys the whole distribution appropriately, the highest of the three values of p giving the best fit to the original distribution.

3.2. A high-dimensional example

Here, we simulated a 2-modal mixed normal distribution over the grid $\{0, 1, \ldots, 999\}^{10}$. Thus, the state space had 10^{30} points, which should not be manageable for any numerical method. Figure 4 illustrates the situation with the corresponding 2-dimensional case.

Again, we sampled the space with both LPC and SWC. For the LPC, the proposals were randomly chosen from the $3^{10} - 1$ neighbours of the current state, while the SWC proposed a random state within the whole space with $p = 0.01$. To compare the results between LPC and SWC, we simply counted the sample points in the two peak regions: one being the set $A = (0, 200)^{10}$ and the other one being the set $B = (800, 1000)^{10}$. We ran 10 independent LPC and SWC. Each run had 10^8 steps, yet only every tenth step was recorded. Table 1 shows the results.

The left column is the result for the SWC. We can see that in each run the chain explores both peaks. The right column on the other hand is the result for the LPC. We can see that in each run the chain either gets trapped in set A or in set B.

3.3. SWC in infinite spaces

In order to demonstrate that the applicability of small-world proposals is not limited to the discrete case and finite spaces, we sampled a mixture of two normal distributions

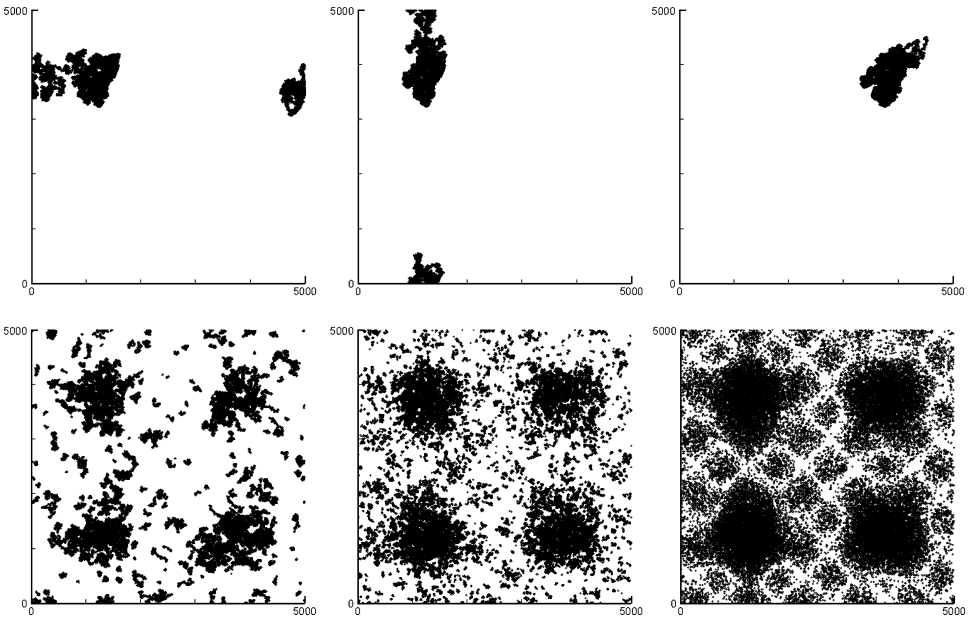

Fig. 3 The results of the first simulation. The top row shows three runs of the LPC with 10^6 steps; the bottom row shows three runs of the SWC with 10^6 steps with different values of p (the leftmost being 0.0001, the middle one 0.001 and the rightmost 0.01)

on $\mathbb{R}^4$, namely the density $\frac{1}{2}f(x_1)f(x_2)f(x_3)f(x_4) + \frac{1}{2}g(x_1)g(x_2)g(x_3)g(x_4)$ where $f(x_i) \sim N(-10, 4)$ and $g(x_i) \sim N(10, 4)$. As in the previous simulation, we ran 10 independent LPC and SWC, each for 10^8 steps, and every tenth step was recorded. The local proposal incremented each component of the current point with a number drawn from $N(0, 0.5)$, while the wild proposals of the SWC used increments drawn from a Cauchy distribution whose full width at half maximum was 20. The frequency p of these wild proposals was set to 0.1. Table 2 summarizes the results of this simulation. Just like in the discrete case, each of the LPC got stuck at one of the peaks whereas the SWC always managed to explore both of them.

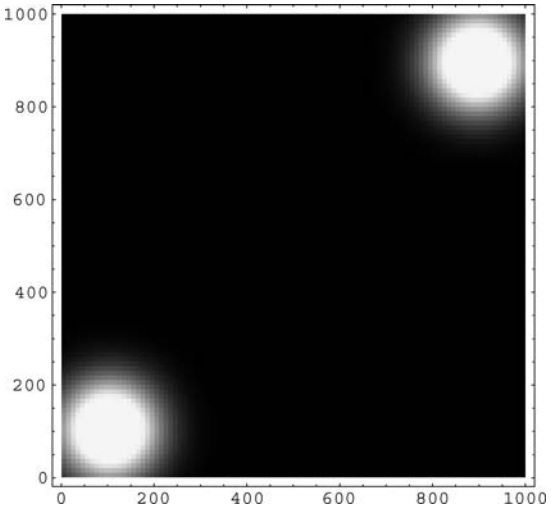

Fig. 4 In the second simulation our probability distribution was the 10-dimensional equivalent of this figure

3.4. An example of a distribution with traps

The probability distribution used in our fourth simulation is shown in Figure 5. This distribution has one main hill in the center and four heaps at each corner surrounded by an almost null recurrent region. As the chance for an LPC to find the heaps is extremely small, we used only SWC in this simulation. We ran 10 independent chains for 10^7 steps, with $p = 0.1$ and recorded every tenth sample point.

The results of this simulation are collected in Table 3. All five hills were found in each run.

3.5. SWC in a heterogeneous space

To see how well an SWC performs in very heterogeneous spaces we took a picture which is often used in image

Table 1 The results of the second simulation showing the number of times the SWC (left column) and the LPC (right column) visited region A and B of the state space. The rows correspond to ten independent runs

SWC		LPC	
Visits to set A	Visits to set B	Visits to set A	Visits to set B
2118493	2079152	0	4068403
2820473	1509184	0	4056310
2728981	1536989	0	4047301
3741123	689508	4497207	0
3374184	934000	0	4069861
3047729	1314131	4452966	0
2462776	1697934	0	4027776
2562090	1696232	0	4096586
2501613	1719777	4532164	0
3285900	1035378	0	4103339

Table 2 The results of the third simulation showing the number of times the SWC (left column) and the LPC (right column) visited regions A and B of the state space, where A and B are hyper-balls with radius 3 centered at (−10,−10,−10,−10) and (10,10,10,10) respectively. The rows correspond to ten independent runs

SWC		LPC	
Visits to A	Visits to B	Visits to A	Visits to B
2484103	1937147	4419221	0
2239322	2179364	0	4430577
2240084	2197521	4417074	0
2179757	2231809	0	4432532
2147068	2268633	0	4426766
2272973	2151260	4417081	0
2228271	2199696	0	4414427
2296054	2123406	0	4421527
2191123	2222683	4430064	0
2297794	2107059	4425759	0

processing, converted it to gray scale and increased the contrast by taking the fourth power of each gray level. Then we ran both LPC and SWC to sample the picture. Again, p was set equal to 0.1 and every tenth sample point was recorded. The results as well as the original image are shown in Figure 6.

Although the difference between the LPC and SWC samples are subtle, one can see, that the image reconstructed with the LPC lacks some details like, for example, the highlight on the hair. One should also note that the SWC image is already very detailed after only 50000 steps.

3.6. SWC and importance sampling

Our final example illustrates how small-world proposals can be used in the context of importance sampling (Hastings, 1970). Suppose we want to estimate the expectation $E_\pi(f(X))$ of a random variable $f(X)$ with respect to a probability measure π. Although Monte Carlo integration (Rice, 1994), i.e., drawing a sample from π and averaging the obtained values of $f(X)$, is one way to achieve this goal, the resulting estimate might have a high variance since $f(x)$ might have maxima where $\pi(x)$ is small. Therefore, it is advisable (Press et al., 1992) to sample not from π but from a new distribution $\pi' \propto f\pi$ which gives an appropriate weight both to the function f and to the original distribution π. Assuming, without loss of generality, that $f > 0$, one can easily check that $E_\pi(f(X)) = 1/E_{\pi'}(1/f(X))$. Hence we can use the following estimator

$$\widehat{E_\pi(f(X))} = \frac{N}{\sum_{t=1}^{N} \frac{1}{f(X(t))}}, \tag{14}$$

where the $X(t)$ are taken from π'. Notice, however, that this importance sampling will in many cases require a chain to sample from a multi-modal space. Thus, its wide use has been restricted due to the limitations of local-proposal chains. Small-world chains, on the other hand, should be well suited for this problem.

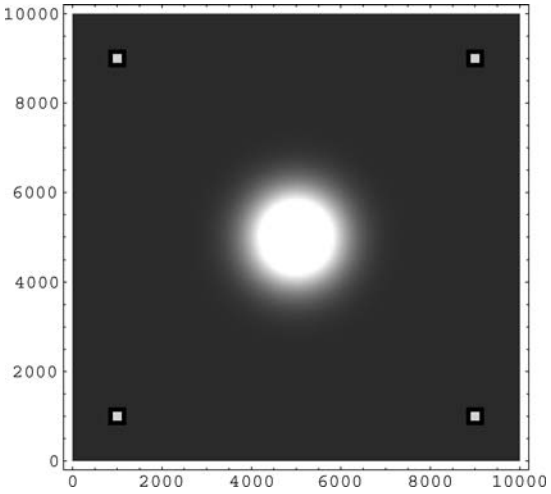

Fig. 5 The two-dimensional distribution used in the fourth simulation. The whole space has size 10000 × 10000

Table 3 The results of the fourth simulation showing the number of times the SWC visited the four heaps (columns A-D) and the central hill ("Center") of the probability space shown in Figure 5. The rows correspond to ten independent runs

A	B	C	D	Center
62194	60062	59019	64989	504718
52897	55617	66427	58123	478288
57943	68062	57264	68815	508021
63832	55247	57274	62762	497264
53808	63851	60561	50230	467570
54432	61555	64119	64871	455125
58708	72168	63017	66713	491471
64462	60580	61251	60949	453775
70234	62663	67625	55654	489553
55683	60116	66087	53879	460234

In our final simulation we therefore compared the performance of local-proposal chains sampling directly from π and of small-world chains which sampled from $(f(x)+1)\pi(x)$ (see Figure 9) for the function $f(x)$ plotted in Figure 7 and the $\pi(x)$ shown in Figure 8. The reason for using $f+1$ instead of f is the high variance which might result from small values of f. This of course has to be taken into account when applying equation (14). Each chain was run for 10^5 steps and every 100th step was recorded. The parameter p of the SWC was set to 0.1. We also ran simulations in which an LPC tried to sample from $(f(x)+1)\pi(x)$, yet they always got stuck at one of the modes (data not shown).

Figures 10 and 11 show the estimated expectations for 100 independent runs of the LPC and the SWC respectively. While there is a considerable variation among the values obtained with the LPC, the estimates produced by the SWC

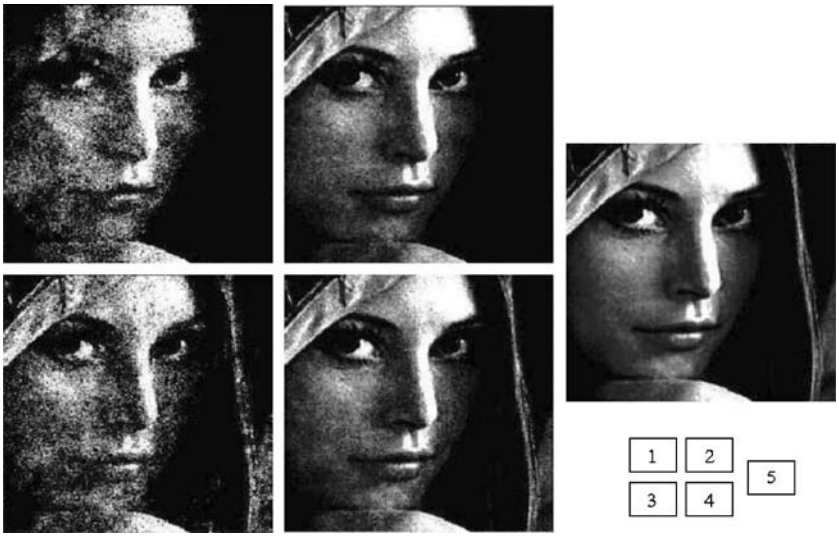

Fig. 6 Top row: the results for the LPC. Bottom row: the results for the SWC. The original picture is displayed on the right-hand side. Figures 6.1 and 6.3 correspond to 50000 steps and Figures 6.2 and 6.4 to 10^7 steps, respectively

are all very close to the true value which is approximately 0.780.

4. Discussion

The purpose of this paper was to present a general yet simple idea for a proposal distribution that leads to a better convergence of MCMC. In all the examples in this paper the small-world chains performed dramatically better than the chains which only relied on local proposals. While the local-proposal chains got stuck at one mode of the distribution (see sections 3.1, 3.2 and 3.3) or missed important details (see section 3.5), the small-world chains were even able to explore the extremely heterogeneous space of section 3.4. The small-world chains' ability to sample multi-modal spaces also permitted the application of importance sampling in a case where local-proposal chains failed (see section 3.6).

Through most of this paper, we stuck to the mathematically tractable case of a Markov chain on a finite grid since this is also the case which is the most relevant one when implementing this algorithm. We also focused on the simplifying scenario that local proposals essentially behave like uniform random walks and that we can generate wild proposals by drawing a point from the space at random. Yet, as demonstrated in simulation 3.3, the small-world idea can also be applied when sampling from a continuous density on an unbounded space. There, one just has to use a sufficiently heavy-tailed distribution for the wild proposals like for example a wide Cauchy distribution. Taking a uniform random walk as exemplary local proposal instead of e.g. a normal random walk is also unproblematic since the central limit theorem guarantees that a uniform random walk and a normal random walk will be very similar local proposal distributions. Hence, replacing the uniform local proposals with the more typical normal proposal should not affect the principal results. We are not advocating exchanging every thinkable local proposal for a mixture of two uniform distributions, but rather relying on occasional wild proposals as a way to improve on any local proposal scheme. Therefore,

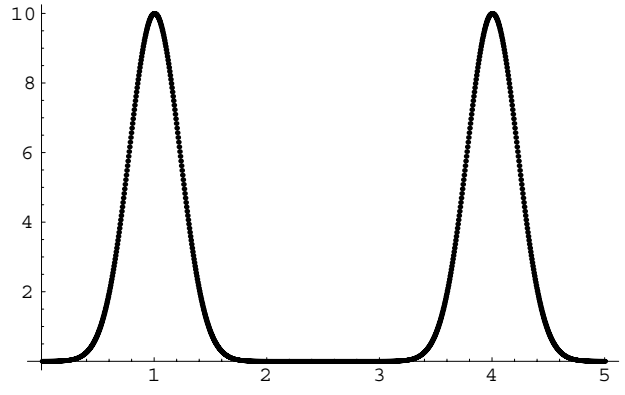

Fig. 7 The function $f(x) = 10(e^{-10(x-1)^2} + e^{-10(x-4)^2})$ to be integrated in section 3.6

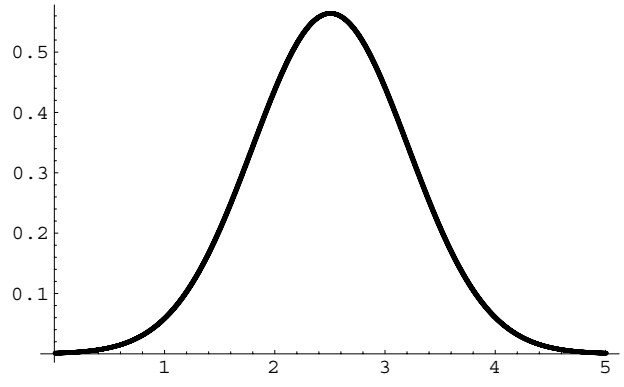

Fig. 8 The density $\pi(x) \propto e^{-(x-2.5)^2}$ from which the LPC in section 3.6 takes its sample

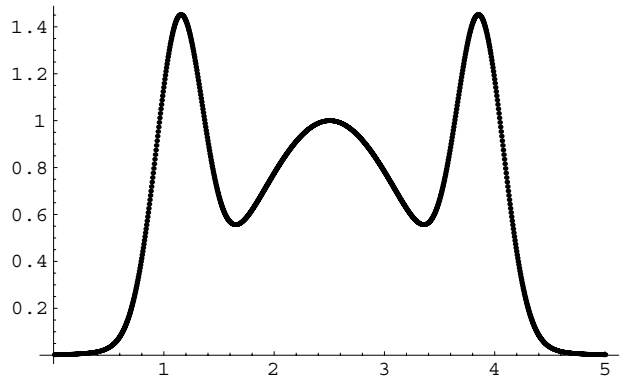

Fig. 9 A plot of $(f(x)+1)\pi(x)$. The SWC in section 3.6 samples from this function

we did not investigate the performance differences which still might exist between different types of local proposals.

Of course, there exists a great variety of other methods which try to ensure convergence of MCMC (cf. Tierney, 1994; Chib and Greenberg, 1995; Gilks et al., 1996). For example, we might know enough about the target distribution $\pi(x)$ so that we can tailor our proposal to this target (Chib et al., 1998). Yet, in many cases that information will not be available.

Another widespread practice is running multiple chains which start from different points of the space. Although this has a superficial resemblance with our idea, exploring the space with multiple chains is in fact equivalent to an incorrect implementation of a single SWC. Suppose for example that an investigator decides to run 100 chains each of length 1 million, starting each chain at a random location. This is the same as running a single chain of length 100 million where every 1 million steps one proposes a wild move and accepts that move with probability 1. So, rather than proposing a wild move with a certain probability, one deterministically decides when to propose a wild move and rather than accepting the move with a certain probability, one always accepts it. These

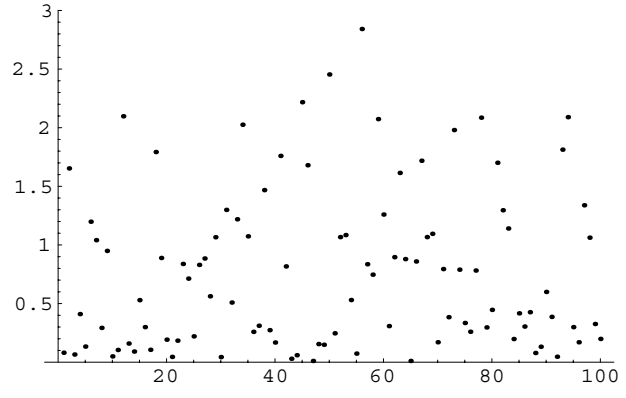

Fig. 10 The estimates $E_\pi(\widehat{f(X)})$ in 100 independent runs of the LPC. The mean of the estimates is 0.770 with a standard deviation of 0.571

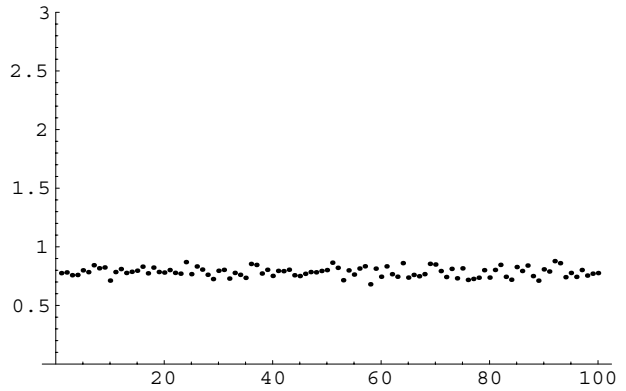

Fig. 11 The estimates $E_\pi(\widehat{f(X)})$ in 100 independent runs of the SWC. The mean of the estimates is 0.785 with a standard deviation of 0.00897

moves violate the assumptions of a homogeneous Markov chain and hence make it unlikely that running multiple chains produces samples from the stationary distribution unless stationarity is reached before the first wild move, in which case the problem is easy and both SWC and multiple chains are unnecessary. As Geyer points out (Geyer, 1992; see also http://www. stat.umn.edu/∼charlie/mcmc/one.html), if the problem is hard, then many short chains are likely to be sampling something closer to the initial distribution from which the starting points were chosen than the stationary distribution $\pi(x)$ of interest. Notice that the SWC accepts wild moves according to the details of the space being explored, as it should, mainly moving between hills or sampling the heavy tails and not returning to the flat regions. Whereas the multiple chain approach, does not make use of any information about the space when it effectively starts the chain over, which will typically be in a flat region. For this reason, a SWC of length N must do better at exploring the space than m multiple chains each of length n with $N = mn$. The only advantage to multiple chains is that they can be run in parallel.

A method which uses multiple chains and indeed samples from the right distribution is the Metropolis-coupled MCMC (Geyer, 1991). There, only one of the chains—the one whose states are recorded—explores π while the others run through flattened versions of that landscape. After every step one checks if it is worthwhile to recombine the chains. If we used this method with only two chains, one which samples from π and one which runs through a completely flattened landscape and if we only attempted to swap chains every $1/p$-th step, this algorithm should behave in the same way as SWC, be it at a higher computational burden.

Another method similar to SWC results from the mixing of transition kernels (e.g. Larget and Simon, 1999). There, instead of having one expression for the Metropolis-Hastings ratio where the proposal probabilities can be written as a

mixture, one has to decide before every step which proposal distribution to use and then only this one goes into the Metropolis-Hastings ratio. This mixing of the transition kernels will indeed give the same result as the mixing of the proposal distributions in the case of symmetric proposals. However, as already mentioned above, the vital point of our approach is not the usage of a mixture as proposal distribution, but the incidental wild jumps. A simple geometric view may give some insight into why the small-world proposals are so effective: If we think of the original space as the inside surface of a large sphere, then a small-world chain's wild proposals tie together distant regions of the sphere. Since these wild links are not static, we get an evolving geometry in which we occasionally stretch two distant points inside the sphere until they touch, then make our jump and let the surface snap back to its spherical shape. While a small-world chain moves through this evolving geometry, it is more likely to jump to another top of a hill than to jump into a valley. Therefore, the hills of the space will behave like neighbors. In that sense, small-world proposals rearrange the space that they are sampling.

It should be noted that this improvement was achieved in an almost automatic way. The only parameter that has to be set in advance is p, the probability of wild proposals and as we have seen in section 2.4 we do not need detailed knowledge of the space in order to produce a reasonable choice of p.

We do not allege that small-world chains always perform better than any other MCMC method, but since using small-world proposals is not more difficult than using local proposals, we are convinced that our method can be a simple and valuable add-on to any of the other methods.

Acknowledgements The authors are members of the University of Idaho Initiative for Bioinformatics and Evolutionary Studies (IBEST). Funding was provided by NSF EPSCoR grant EPS-0132626 (to Paul Joyce, Yongtao Guan, Stephen M. Krone), NSF grant DEB-0089756 (to Paul Joyce), and NIH NCCR grant 1P20PR016448-01 (Roland Fleißner, Paul Joyce, Stephen M. Krone).

References

Chib S. and Greenberg E. 1995, Understanding the Metropolis-Hastings algorithm. The American statistician 49: 327–335.

Chib S., Greenberg E., and Winkelmann R. 1998, Posterior simulation and Bayes factors in panel count data models. Journal of Econometrics 86: 33–54.

Gamerman D. 1997, Markov Chain Monte Carlo. Chapman & Hall, London.

Geyer C. J. 1991, Markov chain Monte Carlo maximum likelihood. In: E. M. Keramides (Ed.): Computing Science and Statistics: Proceedings of the 23rd Symposium on the Interface. Interface Foundation, Fairfax Station, pp. 156–163.

Geyer C. J. 1992, Practical Markov chain Monte Carlo. Statist. Sci. 7: 473–483.

Gilks W. R., Richardson S., and Spiegelhalter D. J. 1996, Markov Chain Monte Carlo in practice. Chapman & Hall, London, 1st edition.

Hastings W. K. 1970, Monte Carlo sampling methods using Markov chains and their applications. Biometrika 57: 97–109.

Jarner S. F. and Roberts G. O. 2001, Convergence of heavy tailed MCMC algorithms (preprint).

Larget B. and Simon D. 1999, Markov chain Monte Carlo algorithms for the Bayesian analysis of phylogenetic trees. Mol. Biol. Evol. 16: 750–759.

Meyn S. P. and Tweedie R. L. 1996, Markov Chains and Stochastic Stability. Springer, New York.

Press W. H., Flannery B. P., Teukolsky S. A., and Vetterling W. T. 1992, Numerical Recipes in C: The Art of Scientific Computing. Cambridge University Press, Cambridge.

Rice J. A. 1994, Mathematical statistics and data analysis. Duxbury Press, Pacific Grove, 2nd edition.

Roberts G. O., Gelman A., and Gilks W. R. 1997, Weak convergence and optimal scaling of random walk Metropolis algorithms. The Annals of Applied Probability 7: 110–120.

Tierney L. 1994, Markov chains for exploring posterior distributions. The Annals of Statistics 22: 1701–1762.

Watts D. J. and Strogatz S. H. 1998, Collective dynamics of 'small-world' networks. Nature 393: 440–442.