Markov Chains Monte Carlo Algorithms

Kai Yang*
University of Toronto

STA496H Readings in Statistics:

Markov Chains Monte Carlo Algorithms

Fall 2010

Supervisor: Prof. Jeffrey S. Rosenthal

December 23, 2010

*Email: kai.b.yang@utoronto.ca

Contents

In this paper we study various aspects of general state space Markov chains and MCMC algorithms. We will start by an introduction to MCMC in chapter 1 followed by topics in optimal scalings and adaptive MCMC in chapter 2 and 3. Then we study extensively on an application of adaptive MCMC algorithm in chapter 4. In the last part of the paper, chapter 5, we give a proof of law of large numbers for Markov chains.

1 Introduction

Markov Chain Monte Carlo (MCMC) Algorithm is a way to approximately sample from certain distributions with high dimensionality. The idea is the following: If we run a Markov chain with certain conditions satisfied (e.g. ergodicity) for long enough period of time, the random variable is going to approximately have the stationary distribution. We can study the parameter of interest of some distributions by direct sampling, but sometimes this may be impossible notably due to high dimensionality. So we can construct a Markov chain with this particular distribution as precisely the invariant distribution and sample by simply running the chain for a long time.

1.1 Motivations

Bayesian inference is based on posterior distributions. It is often the case that posterior quantities of interests can only be approximated by simulations. Sampling from the posterior distribution is one of the major challenges of Bayesian statisticians. MCMC is a way and sometimes the only way to sample from due to the high dimensionality of the posterior distributions. Hence MCMC is widely applied in Bayesian statistics to help to calculate various quantity of interests.

1.2 Preliminaries

In this paper we study the following type of Markov chains: Let $E \subset \mathbb{R}^k$ where $\mathbb{R}^k$ is the Euclidean space and E is called the state space. Let $X_0, X_1, ...$ be a series of E-valued random variables satisfying the Markov condition. That is, conditionally on $X_0 = x_0, X_1 = x_1, ...X_n = x_n$, for states $x_0, x_1, ...x_n \in E$, X_{n+1} could either stay as the same state as $X_n = x_n$ with probability $0 \leq r(x_n) < 1$ only depending on x_n or move to a new state y according to a probability density function $p(x_n, y)$ which depends only on state x_n. In this case, we say a jump has occurred. By law of total probability, at time $n + 1$ a jump either

occurs or not. We have

$$\int p(x_n, y)dy = 1 - r(x_n)$$

Transition kernel does not change with respect to time, so we call this Markov chain $\{X_n\}$ a *time-homogeneous* Markov chain. We denote the probability of the chain being in set A after one step from state x as $P(x, A) = P(X_{n+1} \in A | X_n = x)$ for any n. Let $\delta_x(A)$ be 1 if $x \in A$ and 0 if $x \notin A$. Using the notations that we have already established,

$$\mathbf{P}(x, A) = \int_A p(x, y)dy + \delta_x(A)r(x)$$

Suppose now X_n has a probability distribution function $f_n(x)$, we can derive the probability density function for X_{n+1},

$$\begin{aligned}
\mathbf{P}(X_{n+1} \in A) &= \int_E f_n(x)P(x, A)dx \\
&= \int_E f_n(x)\big[\int_A p(x, y)dy + \delta_x(A)r(x)\big]dx \\
&= \int_E \int_A f_n(x)p(x, y)dydx + \int_E \delta_x(A)r(x)f_n(x)dx \\
&= \int_A \int_E f_n(x)p(x, y)dxdy + \int_A r(y)f_n(y)dy \\
&= \int_A \big[\int_E f_n(x)p(x, y)dx + r(y)f_n(y)\big]dy
\end{aligned}$$

Hence we can see the p.d.f of X_{n+1} is

$$f_{n+1}(y) = \int_E f_n(x)p(x, y)dx + f_n(y)r(y) \tag{1}$$

The transition density functions $p(x, y)$ and the staying probabilities $r(x)$ acts like an operator on the space of all distributions on the state space. We call the operator a Markov operator denoted as P. In the new notation, (1) could be expressed as:

$$f_n(x)P = f_{n+1}(x)$$

1.3 Invariant Distribution and Reversibility

A distribution π on the state space of a Markov chain is called invariant if

$$\pi P = \pi$$

or

$$\int_E \pi(x)p(x,y)dx + \pi(y)r(y) = \pi(y)\ \forall y \in E$$

A distribution λ on the state space of a Markov chain is called reversible if

$$\lambda(x)p(x,y) = \lambda(y)p(y,x)\ \forall x, y \in E$$

Theorem 1. *If a distribution π of a Markov chain is reversible, it is invariant.*

Proof.

$$\int_E \pi(x)p(x,y)dx + \pi(y)r(y) = \int_E \pi(y)p(y,x)dx + \pi(y)r(y) = \pi(y)(1-r(y)) + \pi(y)r(y) = \pi(y)$$

□

1.4 The Metropolis-Hastings Algorithm

The idea of MCMC is simple, i.e. to construct a Markov chain with the desired distribution being invariant and run the chain for long enough period of time waiting for convergence and hence we get a sample for the distribution. Such a construction might not seem to be easy at first thought. Metropolis-Hastings algorithm is one of the most classical and important MCMC algorithms.

Suppose our target distribution has p.d.f π. We define a proposed Markov chain on the state space with transition density function $q(x,y)$ at current state x. We call $q(x,y)$ the proposal distribution at x. The Metropolis algorithm is the following: When at state $X_n = x_n$, first a proposed value y_n of Y_{n+1} is generated according to the proposal distribution $p(x_n, y)$. Then y_n is accepted with probability $\alpha(x,y)$ where

$$\alpha(x,y) = \begin{cases} \min\{\frac{\pi(y)}{\pi(x)}\frac{q(y,x)}{q(x,y)}, 1\} & \text{if } \pi(x)q(x,y) \neq 0, \\ 1 & \text{if } \pi(x)q(x,y) = 0. \end{cases}$$

If the proposal is accepted, $X_{n+1} = Y_{n+1}$, otherwise $X_{n+1} = X_n$. The number of accepted

steps divided by the total number of steps is called the mean acceptance rate. In the case of rejection, the chain will stay put. The probability density function for a jump from state x is $p(x, y) = q(x, y)\alpha(x, y)$, and the probability for staying at state x is

$$r(x) = 1 - \int_E p(x, y)dy = 1 - \int_E q(x, y)\alpha(x, y)dy$$

Note that if the proposal distributions are symmetric, i.e. $q(x, y) = q(y, x)$ for all $x, y \in E$, the acceptance probability reduces to

$$\alpha(x, y) = \begin{cases} \min\{\frac{\pi(y)}{\pi(x)}, 1\} & \text{if } \pi(x)q(x, y) \neq 0, \\ 1 & \text{if } \pi(x)q(x, y) = 0. \end{cases}$$

Symmetric Metropolis chains are easy to deal with because the acceptance probability is relatively easy. We will focus on *symmetric random walk Metropolis Algorithm* in this paper which is an example of a symmetric Metropolis algorithm.

Theorem 2. *The Metropolis Algorithm defined as above produces a Markov chain (X_n) that has exactly the target distribution π being invariant.*

Proof. If (X_n) is reversible to distribution π, so that π is invariant for (X_n). Need to show that $\pi(x)p(x, y) = \pi(y)p(y, x)$. The reversibility equation is trivially true when $x = y$, so we consider the case when $x \neq y$,

$$\pi(x)p(x, y) = \pi(x)q(x, y)\alpha(x, y) = \pi(x)q(x, y)\min\{\frac{\pi(y)}{\pi(x)}\frac{q(y, x)}{q(x, y)}, 1\}$$

$$\pi(y)p(y, x) = \pi(y)q(y, x)\alpha(y, x) = \pi(y)q(y, x)\min\{\frac{\pi(x)}{\pi(y)}\frac{q(x, y)}{q(y, x)}, 1\}$$

When $\frac{\pi(y)}{\pi(x)}\frac{q(y,x)}{q(x,y)} \leq 1$,

$$\pi(x)p(x, y) = \pi(x)q(x, y)\frac{\pi(y)}{\pi(x)}\frac{q(y, x)}{q(x, y)} = \pi(y)q(y, x) * 1 = \pi(y)p(y, x)$$

When $\frac{\pi(y)}{\pi(x)}\frac{q(y,x)}{q(x,y)} > 1$,

$$\pi(x)p(x, y) = \pi(x)q(x, y) * 1 = \pi(y)q(y, x)\frac{\pi(x)}{\pi(y)}\frac{q(x, y)}{q(y, x)} = \pi(y)p(y, x)$$

Therefore, (X_n) is reversible to distribution π and π is invariant. □

1.5 The Symmetric Random Walk Metropolis Algorithm(RWM) and Optimal Scalings

The choice of proposal distribution is crucial to rapid convergence for a Markov chain. Intuitively the best choice of the proposal distribution is the target distribution. However the reason that we are doing MCMC is that it is extremely difficult to sample for the target distribution so that we come across such an intricate way i.e. MCMC to sample from it. Therefore we start with an easy proposal distribution which is usually symmetric n-dimensional normal distribution with mean $\mathbf{0}$ and covariance matrix $\sigma^2 I_d$. In other words, our proposed value $Y_{n+1} = X_n + Z_{n+1}$ where Z_n are i.i.d samples from $N(\mathbf{0}, \sigma^2 I_d)$.

Now the question is that what σ value should we choose so that we can optimize this MCMC algorithm.(i.e.Optimal Scaling) If σ is too small then although most of the moves will be accepted, we are moving with very small step size so it takes longer to possible reach all the states in the state space. And the X_n's that are close in time would very likely to take similar states which increases the correlations between the samples. This is not what we want because we want a Markov chain that gives approximately independent samples from the distribution in one run instead of running the Markov chain over and over again. Also if σ is too large then most of the proposed moves will be rejected then again it will take longer for the Markov chain to converge to stationarity and the correlation between samples increases because of rejections.

1.6 Two Fundamental Theorems of MCMC Algorithms

Under certain conditions one can prove two important fundamental theorems of Markov chains: law of large number and central limit theorem (see Nummelin 2002, Geyer 1992.) They are stated as follows: Let $X_0, X_1, ...$ be a Markov chain with stationary distribution π. Suppose random variable X has distribution π, and g is some functional of X. Let

$$\mu = E(g(X)) \text{ and } \mu_n = \frac{1}{n}\sum_{i=0}^{n} g(X_i)$$

Law of large number tells us:

$$\mu_n \xrightarrow{a.s.} \mu \text{ as n} \to \infty$$

This gives us the theoretic support that the idea to use time-averages of a Markov chain

with certain distribution of interest as invariant distribution to approximate the parameter of interest of this particular distribution works as long as we run the Markov chain for long enough period of time.

Central limit theorem tells us:

$$\sqrt{n}(\mu_n - \mu) \xrightarrow{\mathcal{D}} N(0, \sigma^2) \text{ as n} \to \infty$$

The central limit theorem gives us some idea of how much error there is between the time-averages and the true parameter of interest. (i.e. the size of the Monte Carlo error) It is very difficult to calculate the value of the variance σ^2 theoretically, but its value could still be approximated from the Markov chain itself. To do this we need some time-series methods.

1.7 Definitions of Better MCMC Algorithms

In this section we make clear what we mean by one MCMC algorithm is better than the other. As we all understand, for the same target distribution, we can construct different Metropolis Algorithms to sample from it just by changing the proposal distributions. As long as Ergodicity is satisfied and we are willing to wait for long enough period of time we will always get a sample for the target distribution. However we want the quickest way to get an approximately i.i.d. sample from the target distribution by one run of the Markov chain instead of millions of long runs of Markov chains in practice. So there are several criteria to examine whether an MCMC algorithm is good or not:

(1)Faster convergence to stationarity. Suppose P_1 and P_2 are the transition kernels of two Markov chains, P_1 converges to stationarity faster if the n-step total variation distance of P_1 is always smaller than that of P_2 no matter what the initial state is for any number of steps. This condition could be expressed as:

$$\sup_{A \subset E} |P_1^n(x, A) - \pi(A)| \leq \sup_{B \subset E} |P_2^n(x, B) - \pi(B)| \text{ for all } x \in E \text{ and n = 1,2,...}$$

Where $P_{1(2)}^n(x, A)$ is if we start at state x the probability that we in set A after n-steps, i.e. $P(X_n \in A | X_0 = x)$. $\pi(A) = P(X \in A)$ is the probability that $X \in A$ where the random variable X has the target distribution π.

(2)Smaller variance of functionals of chains. By law of large number, we understand that time averages of a functional of a Markov chain is going to converge to the functional of a random variable with the invariant distribution. Moreover, we want to have a Markov chain such that the time-averages of the functional has a smaller variance. So our approximation

to the real parameter of interest would be more precise. Let $X_0, X_1, ...$ and $Y_0, Y_1, ...$ be two Markov chains with the same invariant distributions. We say $X_0, X_1, ...$ is a better Markov chain if

$$Var(\frac{1}{n}\sum_{i=1}^{n} g(X_i)) \leq Var(\frac{1}{n}\sum_{i=1}^{n} g(Y_i))$$

This definition might depend on functional g chosen, n and initial state. The way to estimate this variance directly involves running the chain for multiple times. An alternative way to estimate this quantity is called the integrated autocorrelation time τ_g. If the Markov chain X_n is in stationarity with stationary distribution π, then for large n,

$$Var(\frac{1}{n}\sum_{i=1}^{n} g(X_i)) \approx \frac{1}{n} Var(g(X))\tau_g \text{ where X has stationary distribution } \pi$$

$$\text{and } \tau_g = \sum_{k=-\infty}^{+\infty} Corr(g(X_0), g(X_k)) = 1 + 2\sum_{k=1}^{+\infty} Corr(g(X_0), g(X_k))$$

The Markov chain $\{X_n\}$ is said to have smaller asymptotic variance than $\{Y_n\}$ if τ_g for $\{X_n\}$ than that for $\{Y_n\}$. The integrated autocorrelation time could be estimated by the following method: to estimate $Corr(g(X_0), g(X_3))$ we can use the sample correlation of $(g(x_0), g(x_3)), (g(x_1), g(x_4)), (g(x_2), g(x_5))$, etc. This calculation is still difficult because it involves many many steps when the chain has been run for a fairly long enough period of time.

(3)Larger mean square difference between X_n and X_{n-1}. In order to get approximately i.i.d. samples we want the states to change drastically as the chain runs rather than getting stuck at some point or moving slowly and getting a lot of similar states from the state space. We say Markov chain $\{X_n\}$ mixes faster than $\{Y_n\}$ if $E[(X_n - X_{n-1})^2]$ is larger. This mean square error is estimated by $\frac{1}{n-B+1}\sum_{i=B}^{n}(X_i - X_{i-1})^2$ to allow a burn-in B for the chain to approximately reach stationarity. Whether Markov chain mixes well could be also observed from the trace plot. When the step sizes are small or when there are a lot of rejections, $E[(X_n - X_{n-1})^2]$ will tend to be smaller.

2 Optimal Scalings of RWM

We restrict ourselves to the RWM(random walk Metropolis) algorithms in this section and a few theorems are proven for optimal scalings for RWM under certain conditions. We assume the chain that we consider here are ergodic (i.e. no matter where we start we will end up

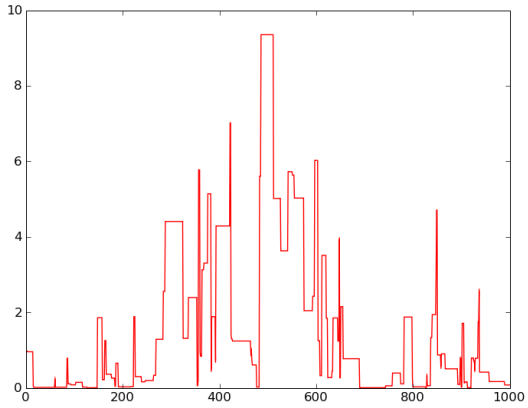

Figure 1: σ too large, too many rejections, bad mixing as states always stay put.

with a sample from the target distribution if we run the chain long enough of time). Our goal is to construct a Markov chain(i.e. to find the optimal scaling σ) that converges to stationarity as quickly as possible. Also we want the correlation between X_i's decreases to 0 in as few iterations as possible.

2.1 Trace Plots

Trace plots are plots of values of X_n versus values of n. It describes the process of Markov chain pictorially. The trace plot for a good MCMC algorithm should be vibrating quickly so that for the next state we have chances to get to anywhere in the state space, and this reduces the correlation of the chain. Also the trace plot should not have too many horizontal lines which stands for rejections. If our proposed values are always rejected it will take much longer for the Markov chain to reach stationarity and the correlation of the chain will be greater because close states are more likely to be the same. We can see from the trace plot whether close samples takes similar states and the number of rejections. See Figure 1, 2 and 3 for examples of trace plots.

2.2 Theorems about Optimal Scalings

Theorem 3 (Roberts, Gelman, and Gilks (1997))**.**
For special target densities on $\mathbb{R}^n$ *of the form*

$$\pi(x_1, x_2, ..., x_d) = f(x_1)f(x_2)...f(x_d)$$

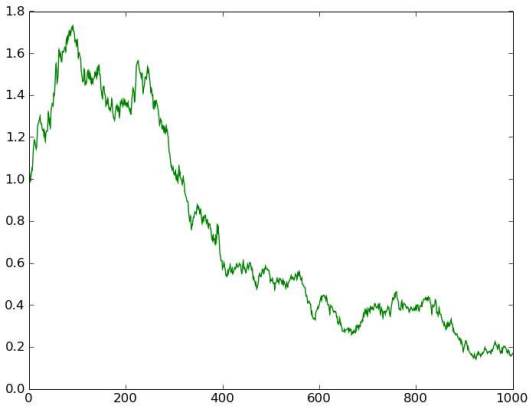

Figure 2: σ too small, step size too small, bad mixing as states always stay close.

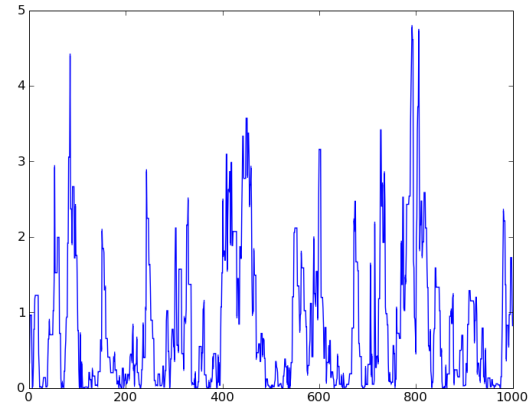

Figure 3: σ medium, decent step size, reasonable amount of rejections, good mixing.

for some one dimensional smooth density f. Assuming proposal increment distribution is $N(0, \sigma^2 I_d)$, the optimal acceptance rate is precisely 0.234 *as* $d \to \infty$.

This assumption is highly unreasonable in the MCMC context because the target density in this case would be of a random vector consists of i.i.d. components. Sampling from this kind of distributions is relatively easy by sampling independently from the one-dimensional density f. The proof is rather complicated we omit it here.

For finite dimension situation, as long as the dimension is greater or equal to 5 the limiting results for optimal scaling still seems to be a good approximation but in 1 dimension case, numerical studies on normal distributions show that the optimal acceptance rate is approximately 0.44.

Theorem 4 (Roberts, Rosenthal (2001))**.**
For the inhomogeneous target densities on $\mathbb{R}^n$ of the form

$$\pi(x_1, x_2, ..., x_d) = \prod_{i=1}^{d} C_i f(C_i x_i)$$

where C_i are i.i.d. from some fixed distribution. Assuming proposal increment distribution is $N(0, \sigma^2 I_d)$, the optimal acceptance rate is precisely 0.234 *as* $d \to \infty$.

Theorem 3 corresponds to the case when C_i are constants. In this case the speed of the algorithm is slowed down by an inhomogeneity factor of $b = E(C_i^2)/(E(C_i))^2$.

Corollary 5.
If the target distribution is $N(0, \Sigma)$ for some d-dimensional covariance matrix Σ, and the increment distribution is $N(0, \Sigma_p)$, it is the best if Σ_p is proportional to Σ. i.e. $\Sigma = k\Sigma_p$ for some $k > 0$. The optimal value of k is $(2.38)^2/d$.

2.3 A Numerical Example

In this section, let us consider a simple MCMC algorithm to sample from the $d = 10$ dimensional normal distribution with covariance Σ. Let M be the 10×10 matrix having diagonal entries 1, and off-diagonal elements given by the product of the number of row and number of column divided by 100. i.e. if m_{ij} is the $ij - th$ entry of M, then $m_{ij} = \frac{ij}{100}$ when $i \neq j$ and $m_{ii} = 1$ for $i = 1, 2, ...10$. Let $\Sigma^{-1} = M^2$, and let the target density π be that of $N(\mathbf{0}, \Sigma)$. We compute numerically the top-left entry of Σ is approximately equal to 1.0305. We know that once the Markov chain reaches stationarity the first component of the random vector is going to have mean 0 and variance 1.0305. We would expect the second moment

of this random variable to be 1.0305, and hence the mean value of the square of the first component should be close to 1.0305 when the chain reaches stationarity. The probability density function of this target multivariate normal distribution is

$$f(\mathbf{x}) = \frac{1}{(2\pi)^5|\Sigma|^{1/2}} e^{-\frac{1}{2}\mathbf{x}^T\Sigma^{-1}\mathbf{x}}$$

We use RWM to sample from this target distribution with initial state $X_0 = (1, 0, ...0)$, and with increment distribution given by $N(\mathbf{0}, \sigma^2 I_d)$ for various choices of σ. For each choice of σ we run the algorithm for 100,000 iterations and average all the values of the square of the first coordinate to estimate its stationary mean. We repeat this 10 times(10 independent runs) for each σ, to compute a 95% confidence interval for the stationary mean and a root mean squared error(RMSE) for each choice of σ. The results are the following (different from Rosenthal, 2008):

σ	mean accurate rate	95% confidence interval	RMSE
0.1	0.984012	0.997846 ± 0.375312	0.605530
0.6	0.484989	1.043671 ± 0.015606	0.025179
0.7	0.363837	1.024226 ± 0.015426	0.024890
0.8	0.262166	1.031090 ± 0.011843	0.019109
3	0	1	0

When σ takes value of 3, all the proposal values are rejected. Notice that there is a weakness in this simulation that the first component of the initial state is too close to stationarity which makes RWM algorithms with small scaling look good. Actually if we start at initial state $X_0 = (5, 0, ...0)$, the consequence of optimal scaling is more obvious. The results are the following:

σ	mean accurate rate	95% confidence interval	RMSE
0.1	0.982364	3.235990 ± 0.560614	0.904500
0.6	0.484908	1.042814 ± 0.016851	0.027188
0.7	0.364171	1.033113 ± 0.013935	0.022483
0.8	0.262154	1.032918 ± 0.016444	0.026532
3	$1.599 * 10^{-5}$	14.2527 ±4.709836	7.598883

We can see when σ is small although the acceptance rate is high, it is very hard for the chain to explore the whole state space and for the chain to sample from the right distribution

because the step size is too small. If we start at state $(5, 0, ..., 0)$ we end up with a sample mean of the second moment of 3.235990 which is far away from the true mean 1.0305. We can see if σ is too large most of the moves are rejected, the chain does not converge to stationarity quickly and the sampling distribution is still pretty far away from the target distribution.

3 Adaptive MCMC Algorithms

The idea of adaptive MCMC algorithms is to have a collection of Markov chain transitional kernels, denoted as $\{P_\gamma\}$, each having the same stationary distribution π. Let Γ_n be the transitional kernel chosen at step n for $n = 1, 2, ...$ Such an MCMC algorithm is called an adaptive MCMC algorithm.

3.1 Examples of Adaptive MCMC Algorithms

Example 1. As we know that under some conditions, the optimal scaling for RWM algorithm gives acceptance rate 0.234. It is possible to find optimal scaling by trial and error, i.e. manually add if the acceptance rate is too high and subtract the scaling factor if the acceptance rate is too low until the acceptance rate is about 0.234. We can find optimal scaling by running a adaptive MCMC algorithm which will find the scaling to result in acceptance rate of 0.234 automatically.

Example 2. Adaptive MCMC algorithm is generally more useful in finding optimal covariance matrix for the increment distribution. We stated in the previous chapter that it is optimal at least in the normal target distribution case that the covariance matrix of the proposal distribution is proportional to the covariance matrix of the target distribution. We can run an adaptive MCMC algorithm to constantly calculate the empirical covariance matrix so that we can have the covariance matrix of the increment distribution closer and closer to a multiple of the covariance matrix of the target distribution.

3.2 Ergodicity of Adaptive MCMC Algorithms

It seems nice that we can sort of combine different MCMC algorithms that have the same target distribution and run a computer program to make decision of which algorithm is the best with some pre-imposed criteria (acceptance rate or covariance matrix of the proposal distribution). A big question is that whether this mixture of algorithms is still going to be

a valid MCMC algorithm that converges to the desired target distribution. The answer to this question is No as illustrated by Rosenthal in 2004.

A simple counter example is the following set of discrete state space MCMC algorithms: Let the state space $\chi = \{1, 2, 3, 4\}$ and the target distribution be $\pi(1) = \pi(3) = \pi(4) = 0.333$ and $\pi(2) = 0.001$. We consider the following two RWMs. For the first one, the proposal distribution $Y^1_{n+1} \sim Uniform\{X_n - 1, X_n + 1\}$ and for the second one, the proposal distribution $Y^2_{n+1} \sim Unifrom\{X_n - 2, X_n + 2\}$. They are both valid MCMC algorithms converging to the same target distribution. However, if our adaptation is the following: if our n^{th} proposal is rejected the next proposal comes from Y^1_{n+1} otherwise the proposal comes from Y^2_{n+1}. Once we run this adaptive algorithm we will see the chain is always trapped at state 1 and limiting distribution is weighted heavily towards 1 compared to 3 and 4. The reason for this to happen is when the chain is at state 3 or 4, it does not get trapped. For state 3, the chain could either move to 1 with probability 1/4 with second algorithm or move to 4 with probability 1/2. For state 4 it could move to state 3 with probability 1/2 once a proposal is rejected. However for state 1, once the first proposal does not move to state 3 the chain gets trapped at state 1 with only escape probability 0.5*0.01/0.333.

Hence we understand that adaptive MCMC algorithms do not necessarily guarantee ergodicity. We need to find out some sufficient conditions to ensure ergodicity.

Theorem 6 (Roberts and Rosenthal 2005)**.**
Let $\chi \subset \mathbb{R}^n$ be the state space with target distribution $\pi(x)$. Let Γ_n be the transitional kernel used at step n with stationary distribution π for all n.
If the adaptive algorithm also satisfies the Diminishing Adaptation condition,

$$\lim_{n\to\infty} \sup_{x\in\chi} \|P_{\Gamma_{n+1}}(x, \cdot) - P_{\Gamma_n}(x, \cdot)\| = 0 \text{ in probability}$$

and the Containment condition,

$$\forall \epsilon > 0, \{M_\epsilon(X_n, \Gamma_n)\}_{n=0}^{\infty} \text{ is bounded in probability}$$

where $M_\epsilon(x, \gamma) = \inf\{n \geq 1 : \|P^n_\gamma(x, \cdot) - \pi(\cdot)\| \leq \epsilon\}$

Then the Markov chain with this adaptive algorithm is ergodic with stationary distribution π. We have,

$$\lim_{n\to\infty} \sup_{A\subset\chi} \|P(X_n \in A) - \pi(A)\| = 0 \text{ asymptotic convergence}$$

and law of large number, i.e. for any bounded real-valued function g we have,

$$\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} g(X_i) = \pi(g)$$

4 An N-Point Process

In this section, let us consider an N-point process on $\mathbb{R}^2$. We would like to restrict ourself to the 3-point process and the higher dimensional cases are not much more complicated than that.

4.1 Introduction

Suppose we have N points in $\mathbb{R}^2$ denoted as $(x_1, y_1), (x_2, y_2), ..., (x_N, y_N)$ and we are given the joint density function of these N points on $\mathbb{R}^2$. It is defined as following: A,B and C are constants,

$$H(x_1, y_1, ..., x_N, y_N) = A \sum_{i<j} \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} + B \sum_{i<j} \frac{1}{\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}} + C \sum_i x_i$$

and the target distribution of these N points is defined as

$$\pi(x_1, y_1, ..., x_N, y_N) = e^{-H(x_1, y_1, ..., x_N, y_N)}$$

Instead of sampling directly from such a high dimensional distribution which is kind of difficult, the only option left to us is to run an MCMC algorithm to sample from the target distribution and we hope we could get an approximately i.i.d. sample.

4.2 A Three-Point Process in a Restricted Region in $\mathbb{R}^2$

We want to do numerical simulation to study this process hence we restrict ourselves to the region of $[0, 1.5] \times [0, 1]$ and also we create a scaling factor of 0.4 before B to avoid truncation errors. Now the target density becomes:

$$H(\mathbf{x}) = A \sum_{i<j} \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} + 0.4B \sum_{i<j} \frac{1}{\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}} + C \sum_i x_i$$

$$\pi(x_1, y_1, ..., x_3, y_3) = \begin{cases} e^{-H(x_1, y_1, ..., x_3, y_3)} & \text{if } (x_i, y_i) \in [0, 1.5] \times [0, 1] \text{ for } i = 1, 2, 3 \\ 0 & \text{otherwise} \end{cases}$$

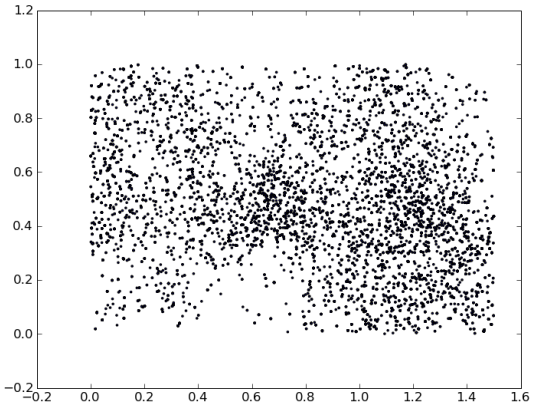

Figure 4: Scatter plot, $A = B = C = 0$, Metropolis within Gibbs algorithm, uniform $[-0.1, 0.1] \times [-0.1, 0.1]$ proposals

First let us understand what these parameters A, B, C are for. If A is large that means points cannot be too far away from each other because when points are far away from each other, $\sum_{i<j} \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$ will be large and the corresponding value of the target density is small. If B is large that means points cannot be too close to each other because in that case $\sum_{i<j} \frac{1}{\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}}$ will be large resulting in small value of the target density. So here comes the problem when both A and B are large, say both are 200, the points can neither get too close to each other nor too far away from each other. Simulation results tell us that the triangle is not moving too much despite the fact that all the congruent triangles in the restricted region have the same probability to be sampled. So the Markov chain is not mixing very well in this case. In the following sections we study this 3-point process case by case.

4.2.1 Case 1. A=B=C=0, Metropolis within Gibbs Algorithm with Uniform Proposal

We run a Metropolis within Gibbs algorithm to see what happens when $A = B = C = 0$. In this case there is no restriction on any of these three points so they should be uniformly distributed in the restricted region. We obtain the scatter plot of y_1 versus x_1 to see the coverage of the state space and the trace plot of x_1 and y_1 to see mixing. (See figure 4 and figure 5) We choose our increment distribution to be uniform over region $[-0.1, 0.1] \times [-0.1, 0.1]$ so that the Markov chain is irreducible and proposal distribution is symmetric.

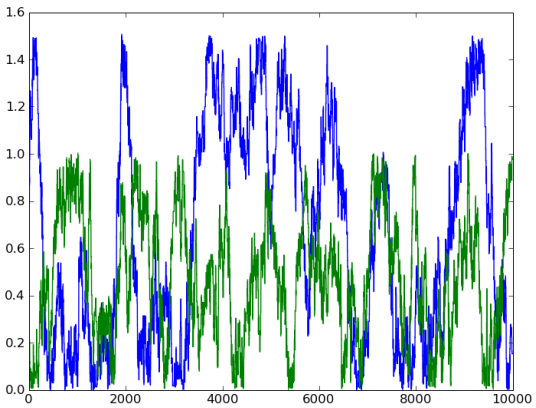

Figure 5: Trace plot, $A = B = C = 0$, Metropolis within Gibbs algorithm, uniform $[-0.1, 0.1] \times [-0.1, 0.1]$ proposals, Blue for x_1, Green for y_1

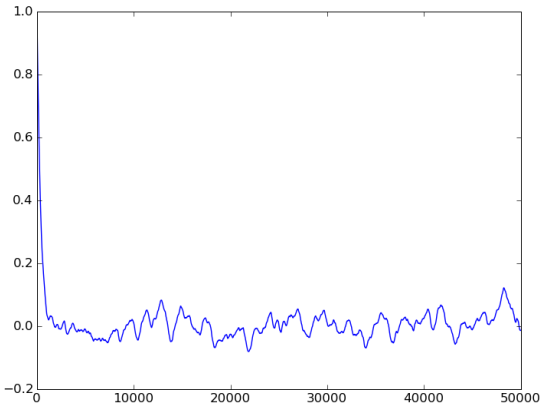

Figure 6: Autocorrelation function with respect to lag in case 1.

As we can see the first point pretty much covers the whole space. We can also observe in the trace plot the Markov chain is mixing pretty well. We can also approximate the variance of g where g is taking the first coordinate x_1, $Var(\frac{1}{n}\sum_{i=1}^{n} g(X_i))$ by running the Markov chain multiple times. In this case $Var(\frac{1}{n}\sum_{i=1}^{n} g(X_i))$ is approximately 0.000467019, calculated by running the Markov chain for 100 times each with 300,000 steps. Take g as the first coordinate of the vector x_1, we can approximate the integrated autocorrelation time $\tau_g = 733.48$, calculated by running the Markov chain for 300,000 steps since we can see the autocorrelation approximately reaches 0 when the lag is as large as about 1795. By convention, the integrated autocorrelation time is approximated only using the terms before the second negative term to avoid random noise error in the calculation.

4.2.2 Case 2. $A = B = 50$ $C = 0$, Metropolis within Gibbs Algorithm with Uniform Proposal

We run a Metropolis within Gibbs algorithm to see what happens when $A = B = 50$ $C = 0$. In this case there is some restrictions on the distances between the three points, so we would expect that the first particle would not be able to freely cover the whole state space. We obtain the scatter plot of y_1 versus x_1 to see the coverage of the state space and the trace plot of x_1 and y_1 to see mixing. We choose our increment distribution to be uniform over region $[-0.1, 0.1] \times [-0.1, 0.1]$ so that the Markov chain is irreducible and proposal distribution is symmetric.

We compute the integrated autocorrelation time in this case. Take g as the first coordinate of the vector x_1, we can approximate the integrated autocorrelation time $\tau_g = 8417.96$, calculated by running the Markov chain for 300,000 steps since we can see the autocorrelation approximately reaches 0 after a lag of length $15,053$. Larger integrated autocorrelation time tells us that correspondingly $Var(\frac{1}{n}\sum_{i=1}^{n} g(X_i))$ is larger and the Markov chain is mixing more poorly than case 1. $Var(\frac{1}{n}\sum_{i=1}^{n} g(X_i)) = 0.0037538$ in this case.

We can look at the scatter plot for the first particle to analyze state space coverage and also we can look at the trace plot of x_1 and y_1 to compare the mixing with case 1. (See figure 7 and figure 8)

4.2.3 Case 3. A=B=200 C=0, Metropolis within Gibbs Algorithm with Uniform Proposal

The problem arises when A and B are large say A=B=200. Since the particles cannot

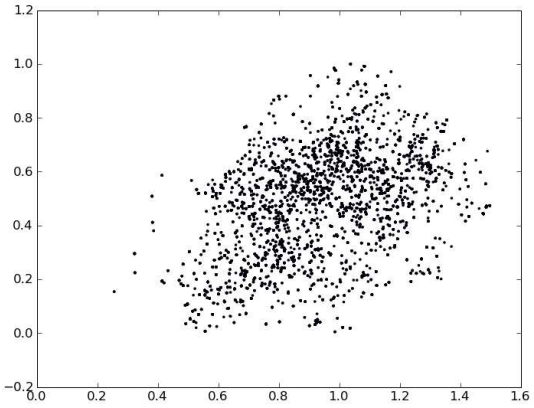

Figure 7: Scatter plot, $A = B = 50$ $C = 0$, Metropolis within Gibbs algorithm, uniform $[-0.1, 0.1] \times [-0.1, 0.1]$ proposals

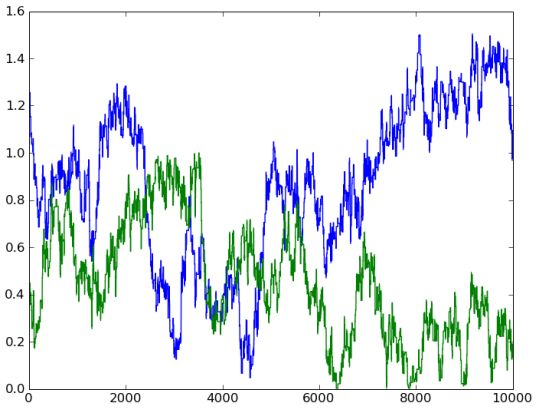

Figure 8: Trace plot, $A = B = 50$ $C = 0$, Metropolis within Gibbs algorithm, uniform $[-0.1, 0.1] \times [-0.1, 0.1]$ proposals, Blue for x_1, Green for y_1

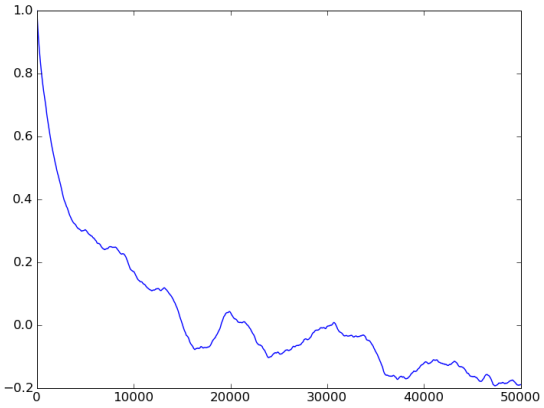

Figure 9: Autocorrelation function with respect to lag in case 2.

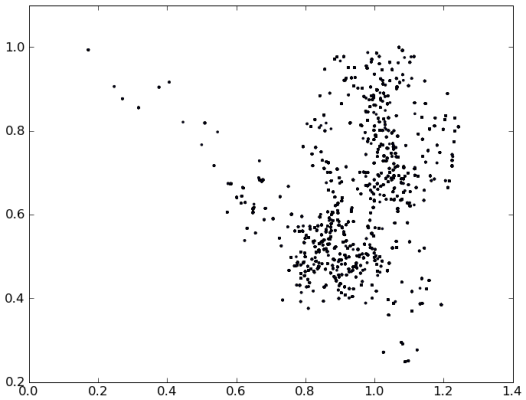

Figure 10: Scatter plot, $A = B = 200$ $C = 0$, Metropolis within Gibbs algorithm, uniform $[-0.1, 0.1] \times [-0.1, 0.1]$ proposals

be too far away from each other and they cannot be too close to each other what happens is that any move that changes the distances between particles will very likely be rejected. Hence the particles are not moving by much resulting in highly similar samples.

In figure 10 we can see the coverage of the state space. The first vertex of the triangle only covers about a half of the state space in the 10,000 runs where but we understand that all congruent triangles have the same probability to be sampled from the target distribution. So the first vertex should not have any preference over some region. Also we can take a look at the trace plot of x_1 and y_1 figure 11. Obviously we can see a lot more rejections than before and it is not mixing so well as the case when A=B=0. The acceptance rate in this case is 0.1788 and $Var(\frac{1}{n}\sum_{i=1}^{n} g(X_i))$ is approximately 0.0041553, calculated by running the Markov chain for 300,000 steps for 100 times.

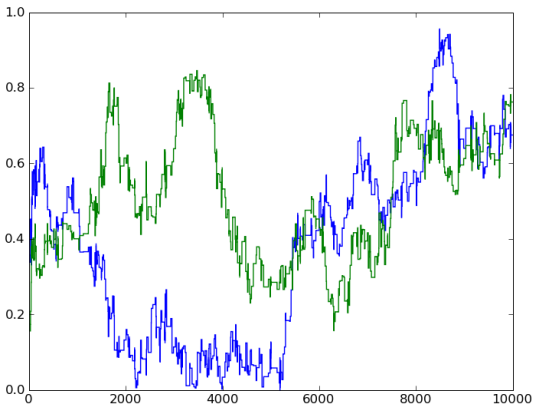

Figure 11: Trace plot, $A = B = 200$ $C = 0$, Metropolis within Gibbs algorithm, uniform $[-0.1, 0.1] \times [-0.1, 0.1]$ proposals, Blue for x_1, Green for y_1

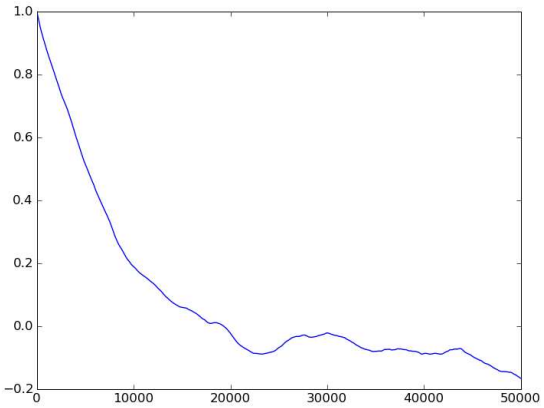

Figure 12: Autocorrelation function with respect to lag in case 3.

We compute the integrated autocorrelation time in this case. Take g as the first coordinate of the vector x_1, we can approximate the integrated autocorrelation time $\tau_g = 12192.66$, calculated by running the Markov chain for 300,000 steps. The number of lags for the autocorrelation time to be approximately 0 is 19263. Large integrated autocorrelation time indicates poor mixing of the Markov chain. Hence in the following sections, we explore various adaptive MCMC algorithms to improve the mixing in the case when A=B=200.

4.2.4 Case 4. A=B=200 C=0, Metropolis within Gibbs Adaptive Algorithm for Arbitrary Target Acceptance Rate with Uniform Proposal

We notice that the acceptance rate is 0.1788 which is not the optimal value 0.234 or 0.44. So we run an Adaptive MCMC algorithm to expand the proposal when acceptance rate is too high and shrink the proposal when the acceptance rate is too low. We will break the entire run into 50 steps per batch and shrink/expand the size the possible proposal region by $min\{0.01, 1/n\}$ after each batch to satisfy the diminishing adaptation condition (Containment condition is satisfied because we are working with this bounded region)and try to obtain the optimal acceptance rate 0.234 or 0.44.

Figure 15 and 16 are the scatter plot and trace plot for target acceptance rate 0.234 and figure 17 and 18 are for target acceptance rate 0.44. We compute the autocorrelation time for the adaptive chain with target acceptance rate 0.234, $\tau_{g,0.234} = 12808.084$ with the required lag of autocorrelation time to be 34798 and autocorrelation time for the adaptive chain with target acceptance rate 0.44, $\tau_{g,0.44} = 12488.05$ with the required lag of autocorrelation time to be 15877, calculated by running the chain for 300,000 steps respectively. Also, $Var_{0.44}(\frac{1}{n}\sum_{i=1}^{n} g(X_i))$ is approximately 0.00713251 and $Var_{0.234}(\frac{1}{n}\sum_{i=1}^{n} g(X_i))$ is approximately 0.007095017.

4.2.5 Case 5. A=B=200 C=0, Full Metropolis Adaptive Algorithm for Arbitrary Target Acceptance Rate with Uniform Proposals and Normal Proposals

One question is if the problem of the constraints on three particles is caused by the Metropolis within Gibbs algorithm. Since the other two particles do not move, this restricts the movement of the 3rd particle. So to answer this question we run the full Metropolis algorithm (propose 3 points together each time) using uniform proposal and normal distribution proposal. It turns out that with about the same scaling factor the acceptance rate of the full Metropolis algorithm is even lower than the Metropolis within Gibbs algorithm.(It

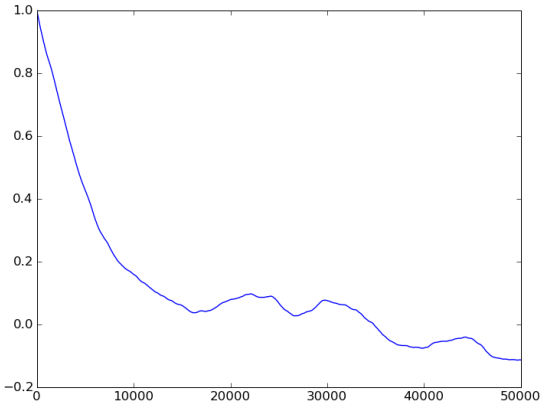

Figure 13: Autocorrelation function with respect to lag in case 4(0.234).

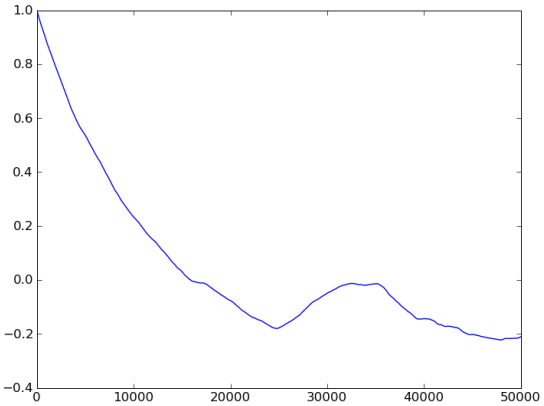

Figure 14: Autocorrelation function with respect to lag in case 4(0.44).

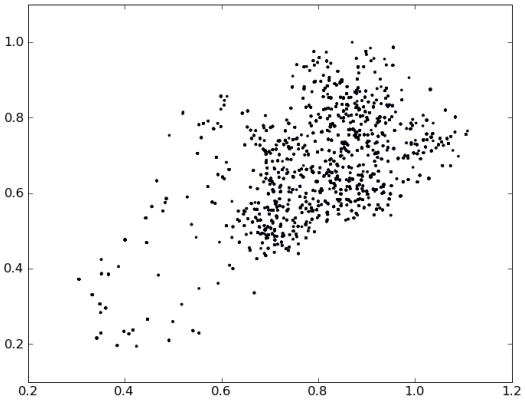

Figure 15: Scatter plot, $A = B = 200$ $C = 0$, Metropolis within Gibbs algorithm, adaptive proposals with target acceptance rate 0.234

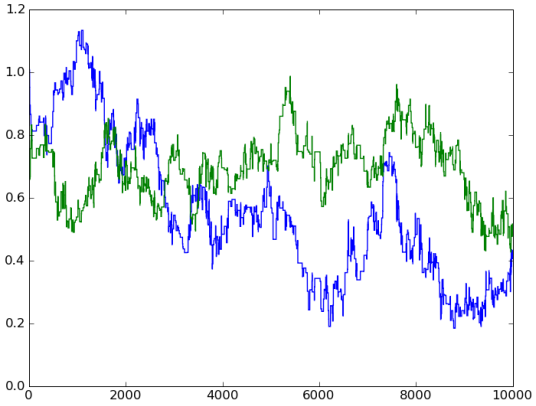

Figure 16: Trace plot, $A = B = 200$ $C = 0$, Metropolis within Gibbs algorithm, adaptive proposals with target acceptance rate 0.234, Blue for x_1, Green for y_1

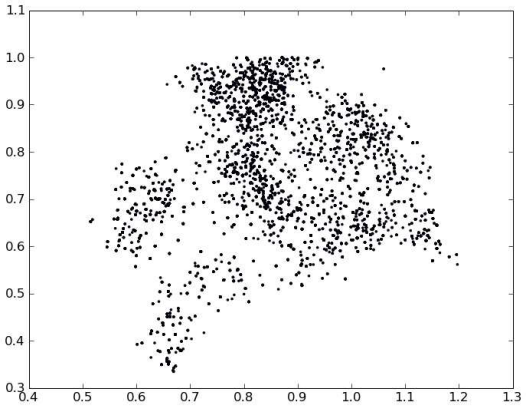

Figure 17: Scatter plot, $A = B = 200$ $C = 0$, Metropolis within Gibbs algorithm, adaptive proposals with target acceptance rate 0.44

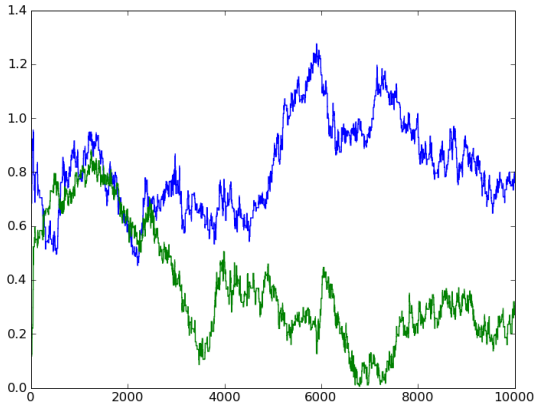

Figure 18: Trace plot, $A = B = 200$ $C = 0$, Metropolis within Gibbs algorithm, adaptive proposals with target acceptance rate 0.44, Blue for x_1, Green for y_1

is even harder for proposals to be accepted when 3 particles are moving at the same time.) See figure 19 and 20 for the trace plots.

We can also compute the integrated autocorrelation time for these two adaptive algorithms. We have $\tau_{g,uniform} = 18193.93169$. The number of lags for the autocorrelation time to be approximately 0 is 20016 in the uniform case. $\tau_{g,normal} = 26878.986499$, both with target acceptance rate of 0.234. The number of lags for the autocorrelation time to be approximately 0 is 48018 in the normal case. Hence integrated autocorrelation time gives us some evidence that the full metropolis algorithm is not working more efficiently as the Metropolis within Gibbs algorithm.

4.2.6 Case 6. A=B=200 C=0, Metropolis within Gibbs Algorithm with Translation for Arbitrary Target Acceptance Rate with Uniform Proposals

We understand that for the target distribution, all congruent triangles have equal probability to appear because the density only depends on the length of the sides of the triangle. However because of the constraint on distances between particles either rejection rate is high or step size becomes small for adaptive algorithms. We wish we can combine this algorithm with some translation proposal so that samples will not always come from the same region of the state space. We combine algorithm used in case 4 with a probability implementing a translation algorithm. i.e. 3 points move in the same direction so the proposal is always accepted in this case. This movement in this case is chosen to be uniform[-0.4, 0.4] distribution in both x and y direction.

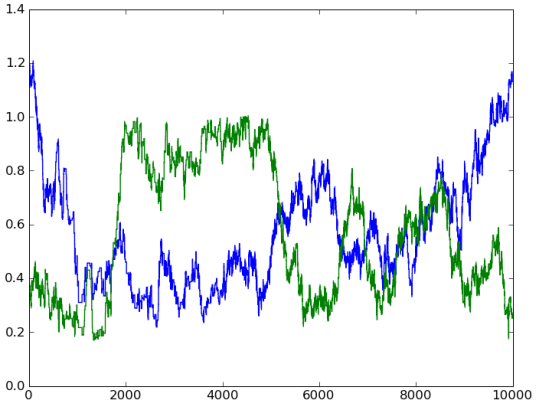

Figure 19: Trace plot, $A = B = 200$ $C = 0$, Full Metropolis algorithm with adaptive normal proposals with target acceptance rate 0.234, Blue for x_1, Green for y_1

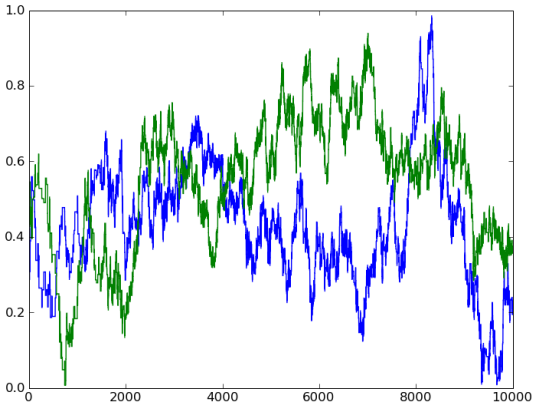

Figure 20: Trace plot, $A = B = 200$ $C = 0$, Full Metropolis algorithm with adaptive uniform proposals with target acceptance rate 0.234, Blue for x_1, Green for y_1

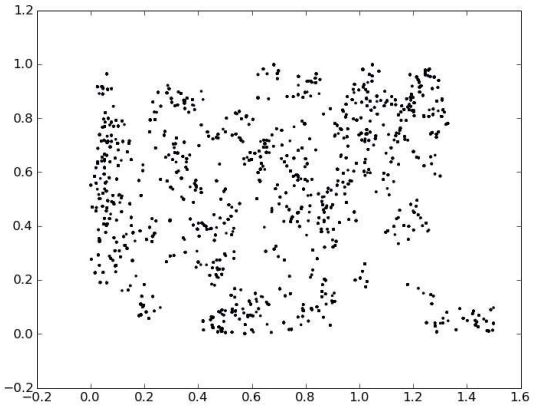

Figure 21: Scatter plot, $A = B = 200$ $C = 0$, Metropolis within Gibbs algorithm, adaptive proposals with target acceptance rate 0.234 combined with translation

The result for this modification is pretty stunning. See figure 21 and 22 for scatter plot and trace plot. We can see from the figure that it is almost very close to case 1.

Of course we compute the integrated autocorrelation time for this algorithm, taking g as the first coordinate of the vector x_1, we can approximate the integrated autocorrelation time $\tau_g = 889.6057958$, calculated by running the Markov chain for 300,000 steps. We also calculate the $Var_{0.234}(\frac{1}{n}\sum_{i=1}^{n} g(X_i)) = 0.0003924$ by running the chain for 300,000 steps for 100 times. The required lag for autocorrelation to be approximately 0 is 1668. The integrated autocorrelation time implies this algorithm mixes very well.

4.3 Summary

Now we have studies 6 cases of the 3-point process and we summarize the results in the following table:

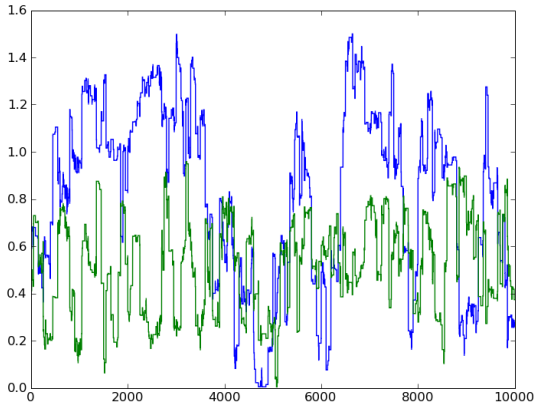

Figure 22: Trace plot, $A = B = 200$ $C = 0$, Metropolis within Gibbs algorithm, adaptive proposals with target acceptance rate 0.234 combined with translation, Blue for x_1, Green for y_1

Case	$Var(\frac{\sum_{i=1}^{n} g(X_i)}{n})$	Required Lag	τ_g	Estimated $Var(g(X))$	$\frac{nVar(\frac{\sum_{i=1}^{n} g(X_i)}{n})}{\tau_g}$
1	0.0004670	1795	733.48	0.1881343	0.19100725
2	0.0037538	15053	8417.96	0.1563531	0.13377825
3	0.0041553	19263	12192.66	0.1354317	0.10224102
4(0.234)	0.0070950	34798	12808.08	0.1324289	0.16618450
4(0.44)	0.0071325	15877	12488.05	0.1582583	0.17134404
5(uniform)	0.0068531	20016	18193.93	0.1316732	0.11300219
5(normal)	0.0110523	48018	26878.98	0.1541377	0.12335636
6	0.0003924	1668	889.60	0.1349172	0.11096076

The cases that we have studied here are:

Case 1 - $A = B = C = 0$, Metropolis within Gibbs Algorithm with Uniform Proposal

Case 2 - $A = B = 50$ $C = 0$, Metropolis within Gibbs Algorithm with Uniform Proposal

Case 3 - $A = B = 200$ $C = 0$, Metropolis within Gibbs Algorithm with Uniform Proposal

Case 4(0.234) - $A = B = 200$ $C = 0$, Metropolis within Gibbs Adaptive Algorithm for Target Acceptance Rate 0.234 with Uniform Proposal

Case 4(0.44) - $A = B = 200$ $C = 0$, Metropolis within Gibbs Adaptive Algorithm for Target Acceptance Rate 0.44 with Uniform Proposal

Case 5(uniform) - $A = B = 200$ $C = 0$, Full Metropolis Adaptive Algorithm for Target Acceptance Rate 0.234 with Uniform Proposals

Case 5(normal) - $A = B = 200$ $C = 0$, Full Metropolis Adaptive Algorithm for Target Acceptance Rate 0.234 with Normal Proposals

Case 6 - $A = B = 200$ $C = 0$, Metropolis within Gibbs Algorithm with Translation for Target Acceptance Rate 0.234 with Uniform Proposals

5 A Proof of Law of Large Numbers for MCMC

In section 1.6, we introduced the two most important theorems in MCMC theory - Law of Large Numbers and Central Limit Theorem. In this section we will give a formal mathematically rigorous proof of the theorem LLN. We follow the methods discussed in [1](Nummelin, 2002).

5.1 Definition of a Small Set

To formulate the irreducibility assumption we need the definition of a small set:

Definition 7. A subset $I \subset E$ with positive Lebesgue measure is called *small* if there exists a subset $J \subset E$ with positive Lebesgue measure such that $p(x, y) \geq \beta$ for some constant $\beta > 0$, for all $x \in I$ and $y \in J$.

5.2 Two Hypotheses to Prove LLN

To prove LLN for Markov Chain we need to assume the following two hypotheses:

Hypothesis 1 There exists a small set I such that for each initial state $x \in E$,

$$P^{n(x)}(x, I) = P(X_{n(x)} \in I | X_0 = x) > 0$$

for some integer $n(x)$ which depends on x. That means no matter where we start the chain from there is always some positive probability to be in the small set in some steps.

Hypothesis 2 The Markov chain has an invariant distribution π and $\pi(x) > 0$ for all $x \in E$.

5.3 Formal Formulation of LLN

Suppose random variable X has distribution π, and f is some functional of X. Let $\pi(f) =$

$\int_E \pi(x)f(x)dx$ be finite. We call f to be $\pi-$integrable. Now we formulate LLN as follows:

Theorem 8 (Law of Large Numbers for MCMC)**.** *Under the hypothesis 1 and 2, let $X_0, X_1, ...$ be a Markov chain with invariant distribution π. We have*

$$\lim_{n\to\infty} \frac{f(X_0) + f(X_1) + \ldots + f(X_{n-1})}{n} = \pi(f)$$

with probability 1 for all initial states $X_0 = x$ and all $\pi-$integrable functions f.

5.4 The Proof of LLN for Markov Chains

We are going to give a rigorous mathematical proof of LLN under the two hypotheses posed in 5.2. The approach we use is called regeneration - breaking the Markov chain into a sum of two Markov chains. One of the Markov chain in the sum regenerates itself so we can break the chain into random tours which are i.i.d. and finally we appeal to the usual LLN to prove the theorem. We break the proof into several sections.

5.4.1 Regeneration and Various Definitions

In this section we talk about regeneration of a Markov Chain to accomplish the goal of decomposing the path of a Markov chain into i.i.d. blocks. We denote the uniform distribution on J to be $\nu(y) = \frac{1}{|J|}$ for $y \in J$ and 0 otherwise. $|J|$ denotes the Lebesgue measure of J. Also we define the function $s(x) = \beta|J|$ for $x \in I$ and 0 otherwise.

Now we rewrite the small set condition as:

$$p(x, y) \geq s(x)\nu(y) \text{ for all x,y} \in E$$

We now break the Markov chain into two Markov chains:

$$P(x, A) = s(x) \int_A \nu(y)dy + (1 - s(x))R(x, A) \text{ for all x} \in E \text{ and } A \subset E$$

Also we denote the sub-Markov operator $(1 - s(x))R(x, A)$ as $Q(x, A)$ which is equal to $P(x, A) - s(x)\int_A \nu(y)dy$. It turns out to be useful notation later on.

We can verify that $R(x, A)$ is a Markov chain: When $x \notin I$, $s(x) = 0$ we have $R(x, A) = P(x, A)$ hence $R(x, E) = P(x, E) = 1$. When $x \in I$, $s(x) = \beta|J|$.

$$1 = P(x, E) = \beta|J|\frac{1}{|J|}|J \cap E| + (1 - \beta|J|)R(x, E)$$

Hence we have $(1 - \beta|J|)R(x, E) = (1 - \beta|J|)$ and $R(x, E) = 1$.

We define a bivariate Markov Chain (X_n, Y_n) as follows. Let $X_0, X_1, ...$ be a sequence of $E-valued$ random variables and let $Y_0, Y_1, ...$ be a sequence of binary $\{0,1\}$-valued random variables. Transition probabilities of the bivariate MC is defined by:

$$P(X_n \in A; Y_n = 1 | X_{n-1} = x; Y_{n-1}) = s(x) \int_A \nu(y)dy$$

$$P(X_n \in A; Y_n = 0 | X_{n-1} = x; Y_{n-1}) = (1 - s(x))R(x, A)$$

for all $n \geq 1, A \subset E$, independently of the random variable Y_{n-1}.

Remark. The marginal probability law of X_n is still that of the original MC since we have

$P(X_n \in A; Y_n = 1 | X_{n-1} = x; Y_{n-1}) + P(X_n \in A; Y_n = 0 | X_{n-1} = x; Y_{n-1})$
$= s(x) \int_A \nu(y)dy + (1 - s(x))R(x, A) = P(x, A)$

Now we define the following probability rules for Y_n:

$$P(Y_n = 1 | X_{n-1} = x) = s(x)$$

$$P(Y_n = 0 | X_{n-1} = x) = 1 - s(x)$$

Such definition for Y_n is crucial(it is the key for regeneration method) since under such construction, the bivariate MC *regenerates* itself when $Y_n = 1$:

$$P(X_n \in A | X_{n-1} = x; Y_n = 1) = \frac{P(X_n \in A; Y_n = 1 | X_{n-1} = x)}{P(Y_n = 1 | X_{n-1} = x)} = \frac{s(x) \int_A \nu(y)dy}{s(x)} = \int_A \nu(y)dy \tag{2}$$

More generally, by the Markov property once $Y_n = 1$ the Chain probabilistically regenerates itself, we have,

$P(X_n \in A_0, X_{n+1} \in A_1 ..., Y_{n+1} = y_1, Y_{n+2} = y_2 ... | X_0, X_1, ... X_{n-1}, Y_1, ... Y_{n-1}, Y_n = 1)$
$= P_\nu(X_0 \in A_0, X_1 \in A_1, ... Y_1 = y_1, Y_2 = y_2 ...)$ where subscript ν is the distribution of X_0.
for all $A_0, A_1, ... \subset E$ and $y_1, y_2 ... \in \{0,1\}$.

Also, when $Y_n = 0$, we have the following inequality:

$$P(X_n \in A_0 | X_0, X_1, ... X_{n-1} = x, Y_1, ... Y_{n-1}, Y_n = 0) = \frac{(1 - s(x))R(x, A_0)}{1 - s(x)} \leq \frac{P(x, A_0)}{1 - \beta|J|}$$

for all $n \geq 0$, $x \in E$.

Definition 9. We define the first regeneration epoch as $T_1 = \min\{n \geq 1 : Y_n = 1\}$. More generally, We denote the successive regeneration epoches as $T_1, T_2, ...$ where $1 \leq T_1 \leq T_2 \leq T_3$.... And $T_i = \min\{n > T_{i-1} : Y_n = 1\}$ for $i = 2, 3, ...$

By the Markov property,

$P(X_{T_i} \in A_0, X_{T_i+1} \in A_1..X_{T_i+m-1} \in A_{m-1}; T_{i+1} - T_i = m | X_0, X_1, ...X_{T_i-1}, Y_1, ...Y_{T_i-1}, T_i = n) = P_\nu(X_0 \in A_0, X_1 \in A_1, ...X_{m-1} \in A_{m-1}, T_1 = m)$ where subscript ν is the distribution of X_0, for all $m, n \geq 1$ and $A_0, A_1, ...A_{m-1} \subset E$.

Definition 10. We define the random blocks(tours) of the Markov chain with respect to regeneration times:

$\xi_0 = (X_0, X_1, ...X_{T_1-1})$
$\xi_1 = (X_{T_1}, X_{T_1+1}, ...X_{T_2-1})$
$\xi_2 = (X_{T_2}, X_{T_2+1}, ...X_{T_3-1})$
...
Or in general, $\xi_i = (X_{T_i}, X_{T_i+1}, ...X_{T_{i+1}-1})$ for i = 1,2,3,...

Definition 11. We define the random sum of the Markov chain with respect to random blocks:

$\zeta_0(f) = \sum_{m=0}^{T_1-1} f(X_m)$
$\zeta_1(f) = \sum_{m=T_1}^{T_2-1} f(X_m)$
$\zeta_2(f) = \sum_{m=T_2}^{T_3-1} f(X_m)$
...
Or in general, $\zeta_i(f) = \sum_{m=T_i}^{T_{i+1}-1} f(X_m)$ for i = 1,2,3,...

5.4.2 Independence of Regeneration Times and Random Blocks

Having defined all the important concepts, we show the independence of regeneration times and random blocks.

Lemma 12. *The differences between regeneration epochs are i.i.d. distributed. In mathematical language, $T_{i+1} - T_i$ are i.i.d. for all $i = 1, 2, 3...$*

Proof. The proof is really is just one line. We show that the probability of $T_{i+1} - T_i$ equal to m does not depend on previous regeneration times and has the same probability for all i. We have,

$$P(T_{i+1} - T_i = m | X_0, X_1, ...X_{n_2-1}, T_1, ...T_{i-1} = n_1, T_i = n_2) = P_\nu(T_1 = m) \tag{3}$$

Hence the conditional probability $P(T_{i+1}-T_i = m|T_{i-1}, T_i)$ does not depend on T_{i-1}, T_i hence these $T_{i+1}-T_i$ are independent for all $i = 1, 2, 3....$ And $P(T_{i+1}-T_i = m|T_{i-1}, T_i) = P_\nu(T_1 = m)$ which is the same for all i. Hence we know these difference are identically distributed as well. Equation (3) is true because when we look at X_{T_i} for any i, it always has distribution ν which does not depend on the past by our construction and Markov property. The next regeneration epoch just come as if the chain starts fresh off X_0 where X_0 has distribution ν. □

Lemma 12 turns out to be crucial in the proof of LLN. The following corollary describes the structure of the Markov chain in terms of regeneration times.

Corollary 13. *The random blocks are i.i.d and so that the random sums are i.i.d. Mathematically, ξ_i's are i.i.d random vectors. Hence $\zeta_i(f)$'s are i.i.d. random variables because they are functions of i.i.d. random vectors.*

Proof. Consider $\xi_{i+1} = (X_{T_{i+1}}, X_{T_{i+1}+1}, ...X_{T_{i+2}-1})$, because of the Markov property the only component that might depend on ξ_i is $X_{T_{i+1}}$. However we showed in equation (2) that $X_{T_{i+1}}$ just has distribution ν which does not dependent on $X_{T_{i+1}-1}$ and hence is independent of ξ_i because of the Markov property. $\zeta_i(f)$'s are functions of independent random vectors ξ_i's so they are independent random variables. Since each block starts off from distribution ν, follows the same Markov transition kernel with the same terminating rule (stop when next regeneration happens), each block are also identically distributed. And it follows that each random sum are identically distributed as well since they are functions of the random blocks. □

5.4.3 Properties of the First Regeneration Time Starting from Initial Distribution ν

Now we have understood that once the regeneration happens, the chain probabilistically regenerates from distribution ν. So first we study the properties of regeneration times when we start from distribution ν and later we will consider the case when we start from arbitrary initial state in the state space.

Definition 14. We define here *potential function* with respect to a p.d.f. on the state space E and a sub-Markov operator Q. The potential function $\mu(x)$ is the following:

$$\mu(x) = \sum_{n=0}^{\infty} \nu Q^n(x)$$

where $\nu Q^n(x)$ is the distribution of the random variable under the transformation of sub-Markov operator Q for n steps starting from initial distribution ν.

Lemma 15. *If the chain starts with invariant distribution π as initial distribution we have*

$$1 = P_\pi(T_1 = \infty) + \int_E \mu(x)dx \int_E \pi(x)s(x)dx$$

Proof. Let $n \geq 1$ be a fixed integer. We may decompose the event $\{T_1 \leq n\}$ according to lag between n and the last regeneration time.

$$\{T_1 \leq n\} = \cup_{k=0}^{n-1}\{L_n = k\}$$

where $L_n = \min\{0 \leq k \leq n-1 : Y_{n-k} = 1\}$

Now for any $n \geq 1$ and any $A \subset E$, suppose we start the chain with stationary distribution π.

$$\begin{aligned}
\mathbf{P}_\pi(X_n \in A) &= P_\pi(X_n \in A, T_1 > n) + P_\pi(X_n \in A, T_1 \leq n) \\
&= P_\pi(X_n \in A, T_1 > n) + \sum_{k=0}^{n-1} P_\pi(X_n \in A, L_n = k) \quad [1] \\
&= P_\pi(X_n \in A, T_1 > n) + \sum_{k=0}^{n-1} P_\pi(Y_{n-k} = 1, Y_{n-k+1} = 0, ..., Y_n = 0, X_n \in A) \quad [2] \\
&= P_\pi(X_n \in A, T_1 > n) + \sum_{k=0}^{n-1} P_\pi(Y_{n-k} = 1) P_\nu(Y_1 = 0, ..., Y_k = 0, X_k \in A) \quad [3] \\
&= P_\pi(X_n \in A, T_1 > n) + \sum_{k=0}^{n-1} \int_E \pi(x)s(x)dx \int_A \nu Q^k(x)dx \quad [4] \\
&= P_\pi(X_n \in A, T_1 > n) + \int_E \pi(x)s(x)dx \int_A \sum_{k=0}^{n-1} \nu Q^k(x)dx \quad [5]
\end{aligned}$$

Some Remarks:

[1] As we decompose the event $\{T_1 \leq n\}$ into disjoint events.

[2] As regeneration happened k steps ago and never happened anymore afterwards.

[3] By Markov property and time homogeneity as we know the chain regenerates at time n-k.

[4] Since π is invariant and also because $P_\nu(Y_1 = 0, ..., Y_k = 0, X_k \in A)$ is the probability starting from ν, not regenerating for k steps, and X_k is in A.

[5] Switch the order of summation and integration.

Let $n \to \infty$, we have

$$\int_A \pi(x)dx = P_\pi(X_n \in A, T_1 = \infty) + \int_E \pi(x)s(x)dx \int_A \mu(x)dx$$

Now we take A as E so we get the following result:

$$\mathbf{1} = P_\pi(T_1 = \infty) + \int_E \pi(x)s(x)dx \int_E \mu(x)dx$$

□

Corollary 16.

$$\int_E \mu(x)dx < \infty$$

Proof. The proof is easily followed from the previous lemma.

$$\int_E \pi(x)s(x)dx = \beta|J| \int_I \pi(x)dx$$

Since $\beta > 0$ and J has positive Lebesgue measure, we have $\beta|J| > 0$. Also since we have $\pi(x) > 0$ everywhere on E and I has positive Lebesgue measure, we have $\int_I \pi(x)dx > 0$. Hence we proved that $\int_E \pi(x)s(x)dx > 0$. Since $P_\pi(T_1 = \infty) \in [0, 1]$, if $\int_E \mu(x)dx = \infty$ we will not have $1 = P_\pi(T_1 = \infty) + \int_E \pi(x)s(x)dx \int_E \mu(x)dx$ as $\int_E \pi(x)s(x)dx > 0$. Hence we proved that $\int_E \mu(x)dx < \infty$. □

Theorem 17. *Let $E_\nu(\zeta_0(f))$ be the expectation of the first random sum, with initial distribution ν. We have $E_\nu(\zeta_0(f)) = \int_E \mu(x)f(x)dx$. Let $E_\nu(T_1)$ denote the expected value of the first regeneration time starting from distribution ν. As a consequence, we have $E_\nu(T_1) = \int_E \mu(x)dx < \infty$.*

Proof. If we start from initial distribution ν, we want to understand the expectation of the first random sum,

$$E_\nu(\zeta_0(f)) = E_\nu(\sum_{n=0}^{T_1-1} f(X_n)) = E_\nu(\sum_{n=0}^{\infty} f(X_n)I(T_1 > n)) = \sum_{n=0}^{\infty} E_\nu(f(X_n)I(T_1 > n))$$

where I is the indicator function.

$$\sum_{n=0}^{\infty} E_\nu(f(X_n)I(T_1 > n)) = \sum_{n=0}^{\infty} \int_E \nu Q^n(x)f(x)dx = \int_E \sum_{n=0}^{\infty} \nu Q^n(x)f(x)dx = \int_E \mu(x)f(x)dx$$

If we take function f as the constant function 1, we have

$$E_\nu(\sum_{n=0}^{T_1-1} 1) = E_\nu(T_1) = \int_E \mu(x)dx$$

And from the previous corollary we have $E_\nu(T_1) = \int_E \mu(x)dx < \infty$ □

Lemma 18.

$$P_\nu(T_1 < \infty) = \int_E \mu(x)s(x)dx = 1$$

Proof. It is quite obvious why $P_\nu(T_1 < \infty) = 1$ since we have $E_\nu(T_1) = \int_E \mu(x)dx < \infty$. If $P_\nu(T_1 = \infty) > 0$, we have $E_\nu(T_1) = \infty$ which is a contradiction. Hence $P_\nu(T_1 < \infty) = 1$. We are left with the proof that $P_\nu(T_1 < \infty) = \int_E \mu(x)s(x)dx$. Since we have

$$P_\nu(T_1 = n) = \int_E \nu Q^{n-1}(x)P(Y_n = 1|X_{n-1} = x, Y_{n-1} = 0)dx = \int_E \nu Q^{n-1}(x)s(x)dx$$

Now,

$$\begin{aligned}
\mathbf{P}_\nu(T_1 < \infty) &= \sum_{n=1}^{\infty} P_\nu(T_1 = n) \\
&= \sum_{n=1}^{\infty} \int_E \nu Q^{n-1}(x)s(x)dx \\
&= \int_E \sum_{n=1}^{\infty} \nu Q^{n-1}(x)s(x)dx \\
&= \int_E \mu(x)s(x)dx
\end{aligned}$$

Hence we have $P_\nu(T_1 < \infty) = \int_E \mu(x)s(x)dx = 1$. □

5.4.4 The Unique Invariant Distribution

Under the hypotheses 1 and 2. It turns out that there exists only one unique invariant distribution for the Markov chain. And we will prove this claim in this section. First let us start with two lemmas.

Lemma 19. *If $f(x) > 0$ for all $x \in E$ and $\lambda(x)$ is a p.d.f. on E, we have $\int_E f(x)\lambda(x)dx > 0$.*

Proof. Let $A_m = \{x : f(x) \geq \frac{1}{m}\}$,

$$\cup_{m=1}^{\infty} A_m = E$$

Also A_m is an increasing sequence of events so $\lim_{m\to\infty} A_m = E$. We have,

$$\lim_{m\to\infty} \int_{A_m} \lambda(x)dx = \int_E \lambda(x)dx = 1$$

Hence there exists m_0 such that

$$\int_{A_{m_0}} \lambda(x)dx > 0$$

Finally we have,

$$\int_E f(x)\lambda(x)dx \geq \int_{A_{m_0}} f(x)\lambda(x)dx \geq \int_{A_{m_0}} \frac{1}{m_0}\lambda(x)dx > 0$$

□

Lemma 20. *Let $\lambda(x)$ be any invariant distribution of the Markov chain which satisfies the two hypotheses. We have $\int_E \lambda(x)s(x)dx > 0$.*

Proof. By hypothesis 1 we have the following,

$$\sum_{n=1}^{\infty} 2^{-n} P^n(x, I) > 0 \text{ for all } x \in E$$

Where $P^n(x, I)$ is the probability of going from state x to set I in n steps. By the previous lemma, $\beta > 0$ and $|J| > 0$, we have

$$\beta|J| \int_E \lambda(x) \sum_{n=1}^{\infty} 2^{-n} P^n(x, I)dx > 0$$

Now since λ is invariant under P,

$$\begin{aligned}
\beta|J| \int_E \lambda(x) \sum_{n=1}^{\infty} 2^{-n} P^n(x, I)dx &= \beta|J| \sum_{n=1}^{\infty} 2^{-n} \int_E \lambda(x) P^n(x, I)dx \\
&= \beta|J| \sum_{n=1}^{\infty} 2^{-n} \int_I (\lambda P^n)(x)dx \\
&= \beta|J| \sum_{n=1}^{\infty} 2^{-n} \int_I \lambda(x)dx \\
&= \beta|J| \int_I \lambda(x)dx \\
&= \int_E \lambda(x)s(x)dx
\end{aligned}$$

Hence we proved that $\int_E \lambda(x)s(x)dx > 0$.

□

Theorem 21. *The p.d.f.* $\frac{\mu(x)}{\int_E \mu(x)dx}$ *is the unique invariant distribution of the chain,* π*. We denote constant* $\int_E \mu(x)dx$ *as* M *here. Hence we have* $M^{-1}\mu(x)$ *being the unique invariant distribution of the chain, where* $M^{-1} = \int_E \pi(x)s(x)dx$*.*

Proof. Consider the potential function $\mu(x)$ which is defined on E. It is a non-negative function on E because it is a sum of p.d.f.'s. So after the normalization it becomes a p.d.f on E. However we shall prove that is it just π. (i.e. the unique invariant distribution on the state space)

Step 1. $M^{-1}\mu$ is invariant. Since Q is linear operator, we can interchange the order of operations. Write $M^{-1}\mu$ as

$$\begin{aligned} M^{-1}\mu(y) &= M^{-1}\sum_{n=0}^{\infty}\nu Q^n(y) \\ &= M^{-1}[\nu(y) + \sum_{n=1}^{\infty}\nu Q^n(y)] \\ &= M^{-1}[\nu(y) + \sum_{n=0}^{\infty}(\nu Q^n)Q(y)] \\ &= M^{-1}[\nu(y) + \mu Q(y)] \end{aligned}$$

By the previous lemma since $\int_E \mu(x)s(x)dx = 1$,

$$M^{-1}\mu(y) = M^{-1}[\nu(y) + \mu Q(y)] = M^{-1}[(\int_E \mu(x)s(x)dx)\nu(y) + \mu Q(y)] = M^{-1}\mu P(y)$$

by the definition of the transition kernel P. Thus $M^{-1}\mu$ is invariant under P.

Step 2. The invariant distribution of this Markov chain is unique. In the proof of lemma 15, we have the following result that holds for any subset $A \subset E$ and any invariant distribution π for the Markov chain,

$$\int_A \pi(x)dx = P_\pi(X_n \in A, T_1 = \infty) + \int_E \pi(x)s(x)dx \int_A \mu(x)dx$$

Thus,

$$\int_A \pi(x)dx - \int_E \pi(x)s(x)dx \int_A \mu(x)dx \geq 0$$

for all $A \subset E$. Hence,

$$\lambda(x) = \pi(x) - \big[\int_E \pi(x)s(x)dx\big]\mu(x) \geq 0$$

non-negative almost everywhere.

Now the claim is that $\int_E \lambda(y)dy = 0$. We prove this claim by contradiction. First observe that $\lambda(y) = \pi(y) - \mu(y)$. Since P is a linear operator and we have both π and μ being invariant, we have $\pi P = \pi$ and $\mu P = \mu$. This implies that $(\pi - \mu)P = \pi - \mu$ by linearity. So if $\int_E \lambda(y)dy > 0$, then $\frac{\lambda(y)}{\int_E \lambda(y)dy}$ is an invariant distribution of the Markov chain. Hence we have $\int_E \lambda(x)s(x)dx > 0$ by lemma 20. However, since $\lambda(x) = \pi(x) - \big[\int_E \pi(x)s(x)dx\big]\mu(x)$, if we just compute this integral $\int_E \lambda(x)s(x)dx$ directly:

$$\begin{aligned}
\int_E \lambda(y)s(y)dy &= \int_E \pi(y)s(y)dy - \big[\int_E \pi(x)s(x)dx\big]\int_E \mu(y)s(y)dy \\
&= \int_E \pi(y)s(y)dy - \int_E \pi(x)s(x)dx \\
&= 0
\end{aligned}$$

Since we have proved that $\int_E \mu(y)s(y)dy = 1$ in lemma 18.

Hence we have a contradiction here that $\int_E \lambda(x)s(x)dx = 0$. So we have $\int_E \lambda(y)dy = 0$, which means $\pi(x)$ is equal to $\big[\int_E \pi(x)s(x)dx\big]\mu(x)$ almost everywhere. Hence the unique invariant distribution of the Markov chain is $\big[\int_E \pi(x)s(x)dx\big]\mu(x)$. We also have $\int_E \pi(x)s(x)dx = M^{-1}$ because it is the normalizing constant of the p.d.f. μ. □

5.4.5 Recurrence

In this section we prove the last tool we use to prove LLN. i.e. The regeneration of the Markov chain is recurrent, which means regeneration occurs infinitely often with probability 1. Let $N(n)$ be the random number of regenerations up to time n.

Theorem 22.

$$P(T_i < \infty | X_0 = x) = 1$$

for all $x \in E$ and $i = 1, 2, 3, ...$ And hence we have $N(n) \to \infty$ as $n \to \infty$ wp 1.

Proof. Lemma 18 tells us if we start with initial distribution ν, we will have a finite regeneration time with probability 1. (i.e. $P_\nu(T_1 < \infty) = 1$) Once we regenerate we understand that we will start fresh off from initial distribution ν and hence the regeneration time is finite with probability 1. So by induction the theorem follows naturally if we can prove the following:

Lemma 23.

$$P(T_1 < \infty | X_0 = x) = 1$$

for all $x \in E$.

Step 1. We prove that the first regeneration time T_1 is finite from almost every initial state $x \in E$. Since we proved before that

$$\mathbf{1} = P_\pi(T_1 = \infty) + \int_E \pi(x)s(x)dx \int_E \mu(x)dx$$

and we also have $\int_E \pi(x)s(x)dx = M^{-1}$ and $\int_E \mu(x)dx = M$, we have that $P_\pi(T_1 < \infty) = 1$. By law of total probability, we have

$$\int_E P(T_1 < \infty | X_0 = x)\pi(x)dx = 1$$

Hence we have T_1 is finite starting from almost every state in E. i.e.

$$P(T_1 < \infty | X_0 = x) = 1$$

for a.e. $x \in E$. Since we have $\pi(x)$ as a p.d.f. on E and $P(T_1 < \infty | X_0 = x) \leq 1$.

Step 2. We prove the first regeneration time is finite with probability 1 starting from every state $x \in E$. We define function $h_\infty(x) = P(T_1 = \infty | X_0 = x)$, for $x \in E$. We have

$$h_\infty(x) = P(T_1 = \infty | X_0 = x) = \lim_{n\to\infty} P(T_1 > n | X_0 = x) = \lim_{n\to\infty} Q^n(x, E)$$

By Monotone Convergence Theorem,

$$\begin{aligned}
\int_E Q(x,y)h_\infty(y)dy &= \int_E Q(x,y) \lim_{n\to\infty} Q^n(y,E)dy \\
&= \lim_{n\to\infty} \int_E Q(x,y)Q^n(y,E)dy \\
&= \lim_{n\to\infty} Q^{n+1}(x,E) \\
&= h_\infty(x)
\end{aligned}$$

Since we have $P_\nu(T_1 < \infty) = 1$ (i.e. if starting from ν we have a finite regeneration time with probability 1), we have

$$\int_E \nu(y)h_\infty(y)dy = 0$$

Then it follows from the definition of the transition kernel P that,

$$\int_E P(x,y)h_\infty(y)dy = \int_E (s(x)\nu(y) + Q(x,y))h_\infty(y)dy = \int_E Q(x,y)h_\infty(y)dy = h_\infty(x)$$

Now we complete the proof by contradiction, Suppose there exists some $x_0 \in E$ such that $h_\infty(x_0) > 0$, it follows that

$$h_\infty(x_0) = \int_E P(x_0,y)h_\infty(y)dy > 0$$

However this implies that

$$\int_E h_\infty(y)dy > 0$$

but we already showed in step 1 that starting from almost every state in E will have finite regeneration time with probability 1. So the integral

$$\int_E h_\infty(y)dy = 0$$

which is a contradiction. Hence we proved $P(T_1 < \infty | X_0 = x) = 1$ for all $x \in E$. □

5.4.6 The Final Step

In this section we appeal to the usual Strong Law of Large Number in probability theory to complete the proof. Recall in the last section we defined $N(n)$ to be the random number of regenerations up to time n and we proved that $N(n) \to \infty$ as $n \to \infty$ by proving regeneration is recurrent starting from any initial state in the state space.

Let x be any state in the state space E and $X_0 = x$. Since T_1 is finite with probability 1 as we proved in the last section,

$$\zeta_0(f) = \sum_{i=0}^{T_1-1} f(X_i)$$

is finite with probability 1.

Also since $T_{N(n)} \leq n < T_{N(n)+1}$, we define the tail of the sum $\zeta'_{N(n)}(f)$ to be:

$$\zeta'_{N(n)}(f) = \begin{cases} \sum_{i=T_{N(n)}}^{n-1} f(X_i) & \text{if } T_{N(n)} \leq n-1, \\ 0 & \text{if } T_{N(n)} = n. \end{cases}$$

Hence we have,

$$\frac{\sum_{i=0}^{n-1} f(X_i)}{n} = \frac{\sum_{i=0}^{T_1-1} f(X_i)}{n} + \frac{\sum_{i=T_1}^{T_{N(n)}-1} f(X_i)}{n} + \frac{\zeta'_{N(n)}(f)}{n}$$
$$= \frac{\zeta_0(f)}{n} + \frac{\sum_{i=1}^{N(n)-1} \zeta_i(f)}{n} + \frac{\zeta'_{N(n)}(f)}{n}$$

Part I.

Since

$$\zeta_0(f) = \sum_{i=0}^{T_1-1} f(X_i)$$

is finite with probability 1.

$$\lim_{n\to\infty} \frac{\zeta_0(f)}{n} = 0 \text{ with probability 1}$$

Part II.

By triangle inequality,

$$\frac{|\zeta'_{N(n)}(f)|}{n} \leq \frac{\sum_{i=T_{N(n)}}^{T_{N(n)+1}-1} |f(X_i)|}{n} \tag{4}$$

By regeneration theorem, $\sum_{i=T_{N(n)}}^{T_{N(n)+1}-1} |f(X_i)|$ has the same probability distribution as $\sum_{i=0}^{T_1-1} |f(X_i)|$ where X_0 has p.d.f. ν and T_1 is the first regeneration epoch. Since T_1 is finite with probability 1, $\sum_{i=0}^{T_1-1} |f(X_i)|$ is finite with probability 1.

Hence $\sum_{i=T_{N(n)}}^{T_{N(n)+1}-1} |f(X_i)|$ is finite with probability 1 and

$$\lim_{n\to\infty} \frac{\sum_{i=T_{N(n)}}^{T_{N(n)+1}-1} |f(X_i)|}{n} = 0 \text{ with probability 1}$$

Now by inequality (4),

$$\lim_{n\to\infty} \frac{\zeta'_{N(n)}(f)}{n} = 0 \text{ with probability 1}$$

Part III.

By lemma 12 and lemma 23, T_1 is finite with probability 1 and $T_{i+1} - T_i$'s are i.i.d. with $i \geq 1$. Hence we have

$$\lim_{n\to\infty} \frac{T_n}{n} = \lim_{n\to\infty} \frac{T_1 + (T_2 - T_1) + (T_3 - T_2) + \ldots + (T_n - T_{n-1})}{n}$$
$$= \lim_{n\to\infty} \frac{T_1}{n} + \lim_{n\to\infty} \frac{(T_2 - T_1) + (T_3 - T_2) + \ldots + (T_n - T_{n-1})}{n-1} \lim_{n\to\infty} \frac{n-1}{n}$$

$$= E_\nu(T_1) = \int_E \mu(x)dx = M \text{ with probability 1}$$

Since we have $T_{N(n)} \leq n < T_{N(n)+1}$,

$$\lim_{n\to\infty} \frac{T_{N(n)}}{N(n)} \leq \lim_{n\to\infty} \frac{n}{N(n)} \leq \lim_{n\to\infty} \frac{T_{N(n)+1}}{N(n)+1} \lim_{n\to\infty} \frac{N(n)+1}{N(n)}$$

By theorem 22, $N(n) \to \infty$ as $n \to \infty$ with probability 1,

$$M \leq \lim_{n\to\infty} \frac{n}{N(n)} \leq M \text{ with probability 1}$$

Therefore we have the result that,

$$\lim_{n\to\infty} \frac{N(n)}{n} = M^{-1} \text{ with probability 1}$$

Finally we can prove the following,

$$\begin{aligned}
\lim_{n\to\infty} \frac{\sum_{i=1}^{N(n)-1} \zeta_i(f)}{n} &= \lim_{n\to\infty} \frac{\sum_{i=1}^{N(n)-1} \zeta_i(f)}{N(n)-1} \lim_{n\to\infty} \frac{N(n)-1}{n} \\
&= E(\zeta_1(f))M^{-1} \\
&= M^{-1} \int_E \mu(x)f(x)dx = \pi(f) \text{ with probability 1}
\end{aligned}$$

Since $M^{-1}\mu$ is the unique invariant distribution π.

By part I, II, and III, we complete the proof of LLN:

$$\begin{aligned}
\lim_{n\to\infty} \frac{\sum_{i=0}^{n-1} f(X_i)}{n} &= \lim_{n\to\infty} \frac{\sum_{i=0}^{T_1-1} f(X_i)}{n} + \lim_{n\to\infty} \frac{\sum_{i=T_1}^{T_{N(n)}-1} f(X_i)}{n} + \lim_{n\to\infty} \frac{\zeta'_{N(n)}(f)}{n} \\
&= \lim_{n\to\infty} \frac{\zeta_0(f)}{n} + \lim_{n\to\infty} \frac{\sum_{i=1}^{N(n)-1} \zeta_i(f)}{n} + \lim_{n\to\infty} \frac{\zeta'_{N(n)}(f)}{n} \\
&= 0 + M^{-1} \int_E \mu(x)f(x)dx + 0 \\
&= \pi(f) \text{ with probability 1}
\end{aligned}$$

Acknowledgments I would like to thank my supervisor Jeffrey Rosenthal for his guidance throughout the semester. He makes probability easy and fun for me and makes it possible for me to understand these topics. It is to him that I dedicate this paper. Also I would like to thank my friends Asad, Chaveen, Ahim for so much support in programming and interesting discussions.

References

[1] Numelin, E. (2002) MC for MCMC'ists. *International Statistical Review* Vol.70 No.2 215–240.

[2] Rosenthal, J. (2008) Optimal Proposal Distributions and Adapative MCMC. *Chapter for MCMC handbook.*

[3] Bednorz,W.,Latuszynski,K.,Latala,R. (2008) A Regeneration Proof of the Central Limit Theorem for Uniformly Ergodic Markov Chains. *Elect. Comm. in Probab.* 13 85–98.

[4] Roberts,G.O.,Rosenthal,J. (2004) General State Space Markov Chains and MCMC Algorithms. *Probability Surveys.* 1 20–71.

[5] Gelfand,A.,Smith,A. (1990) Sampling-Based Approaches to Calculating Marginal Densities. *Journal of the American Statistical Association.* Vol 85 No.410 398–409.

[6] Geyer,C. (1992) Practical Markov Chain Monte Carlo. *Statistical Science.* Vol 7 No.4 473–483.

[7] Roberts,G.O.,Rosenthal,J. (2001) Optimal scaling for various Metropolis-Hastings algorithms. *Statistical Science.* Vol 16 No.4 351–367.

[8] Tierney,L. (1994) Markov Chain for Exploring Posterior Distributions. *The Annals of Statistics.* Vol 22 No.4 1701–1762.

[9] Durrett, R. *Probability: Theory and Examples,* third edition. Duxbury Press, 2005.

[10] Rosenthal,J. *First Look at Rigorous Probability Theory,* second edition. World Scientific Publishing Company, 2006.

[11] Chen,M.-H., Shao,Q.-M., Ibrahim,J. *Monte Carlo Methods in Bayesian Computation,* New York : Springer, 2000.