Adaptive Markov Chain Monte Carlo for Auxiliary Variable Method and Its Application to Parallel Tempering

Takamitsu Araki Kazushi Ikeda

Graduate School of Information Science, Nara Institute of Science and Technology
E-mail: takamitsu-a@is.naist.jp, kazushi@is.naist.jp

Abstract— Auxiliary variable methods such as the Parallel Tempering and the cluster Monte Carlo methods generate samples that follow a target distribution by using proposal and auxiliary distributions. In sampling from complex distributions, these algorithms are highly more efficient than the standard Markov chain Monte Carlo methods. However, their performance strongly depends on their parameters and determining the parameters is critical. In this paper, we proposed an algorithm for adapting the parameters during drawing samples and proved the convergence theorem of the adaptive algorithm. We applied our algorithm to the Parallel Tempering. That is, we developed adaptive Parallel Tempering that tunes the parameters on the fly. We confirmed the effectiveness of our algorithm through the validation of the adaptive Parallel Tempering, comparing samples from the target distribution by the adaptive Parallel Tempering and samples by conventional algorithms.

Keywords— Adaptive Markov Chain Monte Carlo, Auxiliary Variable Method, Parallel Tempering, Convergence

1 Introduction

Markov chain Monte Carlo (MCMC) methods have been an important algorithm in various scientific fields (Liu 2001, Robert and Casella 2004). MCMC methods can generate samples that follow a target distribution by using a simple proposal distribution. However, in sampling from complex distribution such as multi-modal, the standard MCMC methods produce samples theoretically converge the target distribution but practically do not. The produced samples can be trapped in a local mode for a extremely long period.

To cope with this localization problem, the parallel tempering (PT) a.k.a. exchange Monte Carlo method was proposed (Geyer 1991; Hukushima and Nemoto 1996). The PT algorithm introduces auxiliary distributions with a parameter called the temperature, generates multiple MCMC samples from target and auxiliary distributions in parallel, and exchanges the positions of two samples. An auxiliary distribution is tempered when the temperature is high and one with a low temperature is similar to the target distribution. This “tempering” implementation and the exchange process help samples escape from a local mode.

Auxiliary distributions are also used in other several algorithms. One example is the Gibbs variable selection in Bayesian variable selection, where auxiliary distributions approximate the marginal posterior distribution of coefficient parameters for sampling from the joint posterior (Dellaportas et al. 2002). Another is the cluster Monte Carlo methods, efficiently produce samples by block-wise updates based on auxiliary distributions (Swendsen and Wang 1987; Higdon 1998). These algorithms are referred to as auxiliary variable methods (AVMs) in this paper.

The performance of an AVM depends on both the proposal distribution and the auxiliary distributions. Hence the parameters of the distributions have to be chosen so that the Markov chains of the AVM mix as faster as possible. They have been tuned by rough methods or trial-and-error in pilot runs so far because their relationship to the mixing speed has not been clear.

For standard MCMC methods such as Metropolis-Hastings algorithm (Hastings 1970), Gilks et al. (1998) and Haario et al. (2001) proposed adaptive MCMC algorithms that tuned the parameters of a proposal distribution by using past samples during runs. Haario et al. (2001) also proved the convergence theorem of their algorithms, which was developed later (Andrieu and Moulines 2006; Roberts and Rosenthal 2007).

In this paper, we proposed adaptive AVMs by extending the above adaptive algorithms to general AVMs, where the algorithms adapt the parameters of the proposal and the auxiliary distributions of AVMs during runs. We also proved the convergence theorem of our algorithm in a similar way to Roberts and Rosenthal (2007).

We showed the effectiveness of adaptive AVMs by applying it to the PT algorithm and validating it numerically. That is, we developed an adaptive PT algorithm that tune the proposal parameters, which contain the number of temperatures, and temperatures while it runs, and showed the effectiveness of the algorithm via numerical experiments. We also proved the convergence of the adaptive PT algorithm by using the theorem of the general adaptive AVMs.

The rest of the paper is organized as follows. the conventional PT algorithm is briefly shown in Section 2 and an adaptive PT algorithm is proposed in Section 3. Experimental properties of our algorithm are examined in Section 4. In section 5, the proposed algorithm is extended to general AVMs and the convergence theorems of our algorithms are proved in Section 6. Finally, we give a conclusion in Section 7.

2 Parallel Tempering Algorithm

The PT algorithm is a typical algorithm that uses auxiliary distributions, $\pi_{t_l}(dx_l)$, $l = 2, ..., L$, where $1 = t_1 > t_2 > \cdots > t_L > 0$. The density of the lth auxiliary distribution is parametrized by the inverse temperature t_l as $\pi_{t_l}(x) \propto \pi(x)^{t_l}$ or $\pi_{t_l}(x) \propto \pi(x)^{t_l} p(x)^{1-t_l}$, where $\pi(x)$ is the density of the target distribution and $p(x)$ is the density of a simple distribution that mix fast by using a conventional MCMC method. In other words, the inverse temperature t_l tempers the multi-modality of the target distribution $\pi(dx)$ so that the auxiliary densities, $\pi_{t_l}(x_l)$, gradually connect the target density $\pi(x)$ to a simple density $p(x)$ or the uniform distribution.

The PT algorithm executes either of the parallel step and the exchange step at time n, with probability α_r and $1-\alpha_r$, respectively. The parallel step generates the L samples, $x_l^{(n)}$, $l = 1, \ldots, L$, according to $\pi_{t_l}(dx_l)$ for each by using a standard MCMC method. Note that we employed the Metropolis algorithm with the proposal variance γ_l in this paper. The exchange step randomly chooses a sample $x_l^{(n)}$ from the $L-1$ samples, $x_l^{(n)}$, $l = 1, \ldots, L-1$, and exchange $x_l^{(n)}$ for $x_{l+1}^{(n)}$ with probability

$$\min\left(1, \frac{\pi_{t_l}(x_{l+1}^{(n)})\pi_{t_{l+1}}(x_l^{(n)})}{\pi_{t_l}(x_l^{(n)})\pi_{t_{l+1}}(x_{l+1}^{(n)})}\right). \qquad (1)$$

The performance of the PT algorithm strongly depends on the inverse temperatures, more specifically, their intervals and their number. The interval of two adjacent inverse temperatures determines both the similarity of the two distributions and the acceptance probability of an exchange as seen in Eq. (1). The acceptance ratio for the exchanges, which is referred to as exchange ratio in this paper, should not take extreme value. For example, Liu (2001) said a preferable value is a half at any interval. To avoid extreme values and lead to homogeneous exchange ratios, Hukushima (1999) updated temperatures using a recursive formula through preliminary runs and Goswami and Liu (2007) tuned the intervals by iteratively estimating the expected exchange probability through preliminary runs.

Jasra (2007) treated the intervals as a sequence and experimentally compared three inverse-temperature sequences, equal space, logarithmic decay and power decay. The results showed the last was best. Nagata and Watanabe (2008) proved in the low temperature limit when the sequence of inverse temperatures is a geometric progression the exchange ratios are homogeneous. However, the above methods only discussed the intervals and did not take into account the proposal distributions, on which the mixing of samples and the estimation of exchange ratio also depend. In our setting, for example, the Metropolis algorithm has a parameter to be determined, that is, the proposal variance γ_l. It is necessary to re-set the proposal variance when the inverse temperatures are changed a lot, because the appropriate proposal variances obviously depend on the shape of auxiliary distributions.

The more auxiliary distributions the PT algorithm has, the faster the samples are mixed because flatter auxiliary distributions are available but the more computational complexity is required. To solve the trade-off and determine an appropriate number of distributions, Goswami and Liu (2007) proposed to select the maximum temperature using statistical tests. The tests should be done in an off-line manner, that is, they need preliminary experiments in advance.

3 Adaptive PT Algorithm

We propose an adaptive PT algorithm that adapts the inverse temperatures, the parameters of proposal distribution, and the minimum inverse temperature while the algorithm is running. The three adaptation algorithms are described below.

The exchange ratio should take a moderate value. To converge the exchange ratio for x_{l-1} and x_l to a specific value, $\alpha \in (0, 1)$, typically a half, the log inverse temperature, $\zeta_l = \log(t_l)$, is updated as

$$\zeta_l^{(n+1)} \leftarrow \zeta_l^{(n)} - a_n^l (ER_{l-1,l}^{(n)} - \alpha), \qquad (2)$$

where $ER_{l-1,l}^{(n)}$ is a variable that takes one if the exchange occurs between the samples, $x_{l-1}^{(n)}$ and $x_l^{(n)}$, at time n, and zero otherwise. The learning coefficient, a_n^l, is a decreasing random variable with n that satisfies $\lim_{n\to\infty} a_n^l = 0$ *almost sure.*

The proposal distribution of the Metropolis algorithm for a target and auxiliary distribution should have an appropriate variance, which is an average of the variances of all modes. To control the proposal distribution of the Metropolis algorithm for the distribution $\pi_{t_l}(dx_l)$ on $\mathbb{R}^p$, the the variance $\gamma_l = (\gamma_{l1}, ..., \gamma_{lp}) \in \mathbb{R}^p$ and the auxiliary mean parameter $\mu_l = (\mu_{l1}, ..., \mu_{lp}) \in \mathbb{R}^p$ are updated as

$$\begin{aligned} \mu_{lj}^{(n+1)} &\leftarrow \mu_{lj}^{(n)} + b_n(x_{lj}^{(n+1)} - \mu_{lj}^{(n)}), \\ \gamma_{lj}^{(n+1)} &\leftarrow \gamma_{lj}^{(n)} + b_n\left((x_{lj}^{(n+1)} - \mu_{lj}^{(n+1)})^2 - \gamma_{lj}^{(n)}\right), \end{aligned} \qquad (3)$$

where $x_{lj}^{(n+1)}$ is the jth element of $x_l^{(n+1)} \in \mathbb{R}^p$. When $x_l^{(n)}$ is updated to $x_l^{(n+1)}$ by exchanging to $x_{l-1}^{(n)}$ or $x_{l+1}^{(n)}$, the $\mu_l^{(n)}$ jumps to the exchanged value, that is, $\mu_l^{(n+1)} \leftarrow x_l^{(n+1)}$. The learning coefficient, b_n, is a decreasing function of n that satisfies $\lim_{n\to\infty} b_n = 0$.

The auxiliary distribution with the minimum inverse temperature should be so flat that Metropolis samples can frequently move from one mode to another while the total number of auxiliary distributions should be as small as possible. To determine an appropriate value for the minimum inverse temperature, the auxiliary distributions $\pi_{t_l}(dx_l)$ with $l > l^*$ are removed where l^* is the smallest number that satisfies

$$\prod_{j=1}^{p} \gamma_{lj}^{(n)} \geq \prod_{j=1}^{p} V^{(n)}(x_{lj}), \qquad (4)$$

where $V^{(n)}(x_{lj})$ is the sample variance of x_{lj} at time n. This check is done at time $n = m, 2m, \ldots$, where m is a large number (e.g. 10^4). To improve the reliability, when inequality (4) holds a few times d (e.g. 3) in succession the auxiliary distribution is determined to be enough flat.

Inequality (4) shows the relationship between the sample variance and the proposal variance. Due to

Eq. (3), the latter converges to the variance of local region and hence it is smaller than the sample variance if Metropolis samples are localized in a mode. Otherwise, the auxiliary distribution is flat enough.

A pseudo code of the adaptive PT algorithm is given in the following.

```
Algorithm  Adaptive PT algorithm
  Initialize $x_l^{(0)}$, $\zeta_l^{(0)}$, $\gamma_l^{(0)}$, $\mu_l^{(0)}$ and $c_l = 0$ ,$l = 1, ..., L$. ($\zeta_1^{(0)}$=0, constant).
  for $n = 0$ to $N - 1$ do
    $u \sim U[0, 1]$ (where $U[0, 1]$ is a uniform distribution of the interval (0,1)).
    if $u \leq \alpha_r$ then
      for $l = 1$ to $L$ do
        (parallel sampling step)
        Generate $x_l^{(n+1)}$ via Metropolis Algorithm for $\pi_{t_l^{(n)}}(dx_l)$, which has the proposal variances $\gamma_l^{(n)}$.
        (proposal parameter learning step)
        Update ($\gamma_l^{(n)}$, $\mu_l^{(n)}$) to ($\gamma_l^{(n+1)}$, $\mu_l^{(n+1)}$) by the Eq. (3).
      end for
    else
      (exchange step)
      Randomly Choose a neighboring pair, $x_l^{(n)}$ and $x_{l+1}^{(n)}$, and exchange them with the probability Eq. (1).
      (inverse temperature learning step)
      Update $\zeta_{l+1}^{(n)}$ to $\zeta_{l+1}^{(n+1)}$ by Eq. (2).
      if the exchange is accepted,  then
        $\mu_k^{(n+1)} \leftarrow x_k^{(n+1)}$, for $k = l, l + 1$.
      end if
    end if
    ( minimum inverse temperature decision step)
    if ($n$ mod $m$) == 0 then
      for $l = 1$ to $L$ do
        If Eq. (4) hold, then $c_l \leftarrow c_l + 1$.
      end for
      $L \leftarrow \min\{l | c_l \geq d, l = 1, \ldots, L\}$, if $\{\} \neq \emptyset$.
    end if
  end for
```

The adaptive PT algorithm converges. The proof will be given later as a special case of adaptive MCMC algorithms for general auxiliary variable methods.

4 Experiments

To confirm the effectiveness of our algorithm, the following two computer experiments were carried out:

1. A mixture of four normal distributions.

2. The posterior of a mixture of six normal distributions.

In each of the experiments, the burn-in period was a half of the total number of iterations and sample sets, which are used in an estimation and a scatter plot, are chosen from every 50 samples in post burn-in. The proposal distribution of the Metropolis algorithm was an independent normal distribution. Other parameters were $\alpha = 0.5$, $\alpha_r = 0.5$, $a_n^l = 1/(1 + n/(20 + 10l)) * \log(\exp(-\zeta_l^{(n)}) + 1)$, $b_n = 1/(5 + 0.1n)$, $m = 10^4$ and $d = 3$. $L = 25$ and the intervals of inverse temperatures are equal, that is $t^{(0)} = (1, 24/25, 23/25, \ldots, 1/25)$, at the initial condition. Note that these values are invariant for the each above distribution, i.e., a tuning of these values was not necessary in these experiments.

4.1 A mixture of four normal distributions

To see and visualize the properties of our adaptive PT algorithm, we chose a mixture of four normal distributions in two dimensional space as the target distribution (Fig. 1(a)):

$$g(x) = \sum_{i=1}^{4} \frac{1}{4} \frac{1}{2\pi^2 det(\Sigma_i)^{1/2}} \exp\left(-\frac{1}{2}(x - \mu_i)^T \Sigma_i^{-1}(x - \mu_i)\right),$$

where the parameters of the normal distributions are

$$\begin{aligned} \mu_1 &= (0, 44), & \mu_2 &= (44, 0), \\ \mu_3 &= (0, -44), & \mu_4 &= (0, -44), \\ \Sigma_1 &= \text{diag}(1, 7^2), & \Sigma_2 &= \text{diag}(7^2, 1), \\ \Sigma_3 &= \text{diag}(1, 7^2), & \Sigma_4 &= \text{diag}(7^2, 1). \end{aligned}$$

These normal distributions have quite different variances (1 and 7^2), where the proposal variance learning is difficult.

The adaptive PT algorithm ran for 3×10^5 iterations, where the auxiliary distributions are $\pi_{t_l}(x) \propto g(x)^{t_l}$ and the initial proposal variances are $\gamma_{lj}^{(0)} = 3 \times 10^2$.

As a result, our algorithm mixed well and obtained samples from all possible modes (Fig. 1(b)). In fact, the number of inverse temperatures was reduced to five after 3×10^4 iterations but the auxiliary distribution $\pi_{\hat{t}_5}(dx)$ is flat enough (Fig. 2), where $\hat{t}_5$ is the t_5 obtained by the adaptive PT algorithm.

The larger the variances of the auxiliary distribution became, the larger the proposal variances became. In fact, the sums of proposal variances are $(\hat{\gamma}_{1,1} + \hat{\gamma}_{1,2}, \ldots, \hat{\gamma}_{5,1} + \hat{\gamma}_{5,2}) = (32.26, 41.86, 245.8, 1124, 8704)$.

The estimated exchange ratios converged to $(0.501, 0.507, 0.499, 0.498)$, all of which are almost $\alpha = 0.5$. Then, the inverse temperatures were $(\hat{t}_2, \ldots, \hat{t}_5) = (0.328, 0.108, 0.0307, 0.00937)$.

$t_2^{(n)}$ and $\gamma_2^{(n)}$ converge quickly from even the extreme starting points (Fig. 3). The others also converge as fast as them.

4.2 The posterior of a mixture of six normal distributions

In the second experiment, we estimated the average of component specific means of given the data using Bayesian estimation as is seen in Jasra et al. (2007).

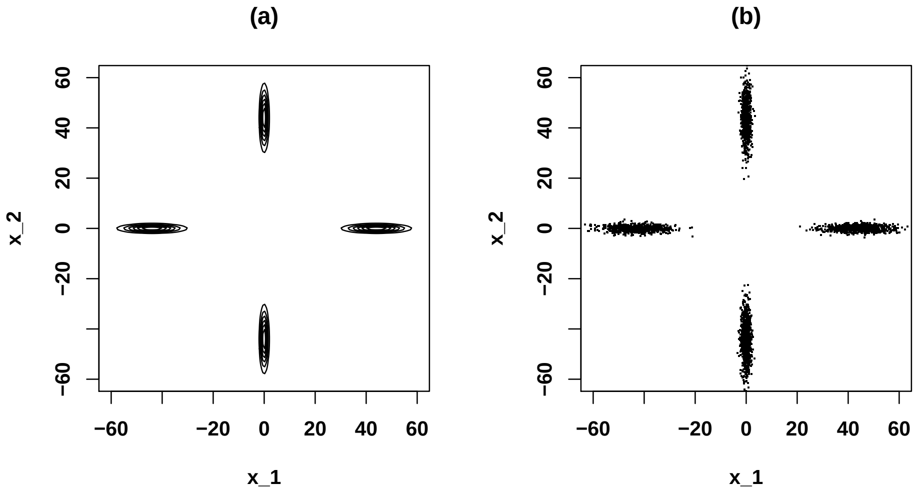

Figure 1: A mixture of four normal distributions. a) The target distribution. b) Samples by the Adaptive PT algorithm.

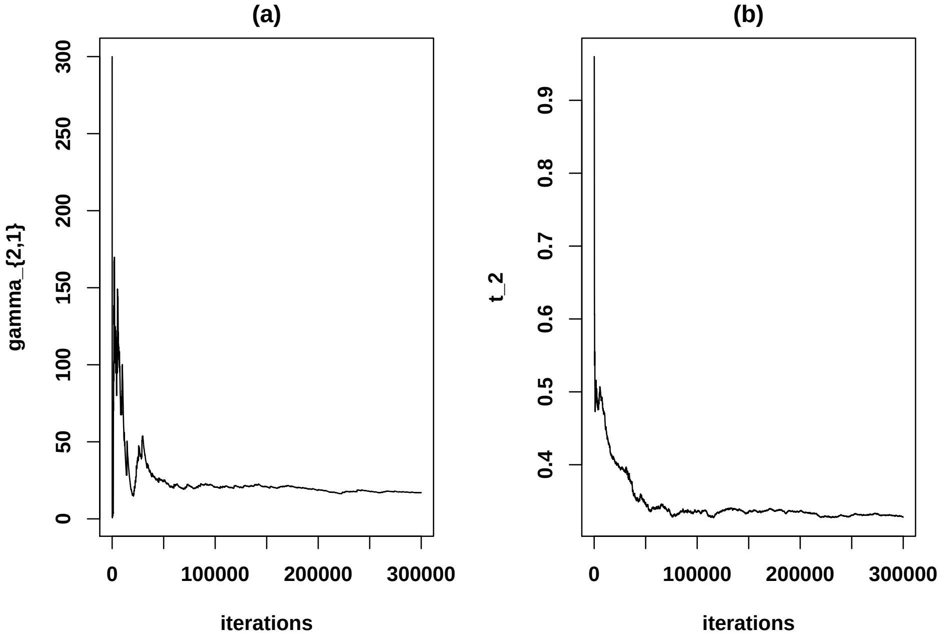

Figure 3: trace plot (a):proposal variance $\gamma_{2,1}^{(n)}$, (b):inverse temperature $t_2^{(n)}$.

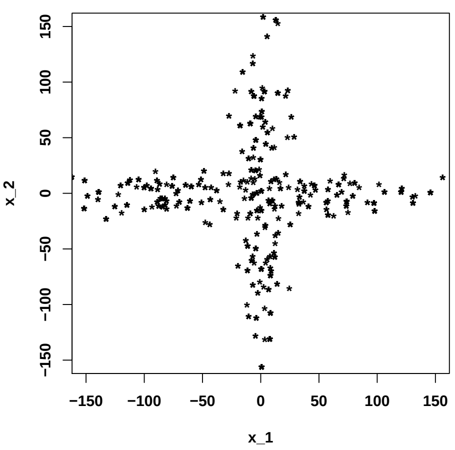

Figure 2: Samples by Metropolis algorithm with $\hat{\gamma}_5$ from the auxiliary distribution $\pi_{\hat{t}_5}(dx)$. They cover all the modes in Fig. 1.

The statistical model was a mixture of normal distributions, that is,

$$f(y|\mu, w, \sigma^2) = \sum_{m=1}^{M} \frac{w_m}{\sqrt{2\pi}\sigma_m} \exp\left(-\frac{1}{2\sigma_m^2}(y-\mu_m)^2\right), \qquad (5)$$

where $w_M = 1 - \sum_{m=1}^{M-1} w_m$. The priors are a normal-inverse Gamma-Dirichlet prior as follows.

$$\begin{aligned} \mu_m &\sim N(\xi, \kappa^2), \quad m = 1, \ldots, M, \\ \sigma_m^2 &\sim IG(\alpha_g, \beta_g), \quad m = 1, \ldots, M, \\ w_m &\sim \mathcal{D}(\varrho), \quad m = 1, \ldots, M-1, \end{aligned}$$

where $\mathcal{D}(\varrho)$ is the symmetric Dirichlet distribution with parameter ϱ. In the following, the hyperparameters were $\alpha_g = 12$, $\beta_g = 10$ and $\varrho = 1$. The parameters ξ and κ^2 were determined by the median and four times the variance of the given data.

The data of size 150, $y_{1:150}$, were independently identically distributed according to a mixture model of the form (5) with parameters, $M = 6$, $w_1 = \cdots = w_6 = 1/6$, $(\mu_1, \ldots, \mu_6) = (-8, -3, 1, 4, 8, 13)$, $\sigma_1^2 = \sigma_6^2 = 1.5^2$ and $\sigma_2^2 = \cdots = \sigma_5^2 = 0.5^2$. In this case, the posterior $\pi(\mu, w, \sigma^2|y_{1:150})$ was a 17 dimensional distribution and had $6! = 720$ symmetric modes due to the invariance against permutation of the labels of the parameters.

The auxiliary distributions were set to

$$\pi_{t_l}(\mu, w, \sigma^2|y_{1:150}) \propto \left(\prod_{i=1}^{150} f(y_i|\mu, w, \sigma^2)\right)^{t_l} p(\mu, w, \sigma^2),$$

where $p(\mu, w, \sigma^2)$ was the prior.

Our algorithm was compared to the conventional PT algorithm with the fixed parameters. The parameter values of the conventional algorithm were shifted from the values obtained by the adaptive PT algorithm as follows.

(a) $\zeta_l \leftarrow \hat{\zeta}_l \cdot \varphi_\zeta$, for $l = 1, \ldots, L$, $(0.5 \le \varphi_\zeta \le 3)$.
$L \leftarrow \hat{L}$, $\gamma_l \leftarrow \hat{\gamma}_l$.

(b) $\gamma_l \leftarrow \hat{\gamma}_l \cdot \varphi_\gamma^2$, for $l = 1, \ldots, L$, $(0.1 \le \varphi_\gamma \le 3)$.
$L \leftarrow \hat{L}$, $\zeta_l \leftarrow \hat{\zeta}_l$.

(c) $L \leftarrow \hat{L} + \varphi_L$, $(-5 \le \varphi_L \le 5)$.
$\gamma_l \leftarrow \hat{\gamma}_l$, $\zeta_l \leftarrow \hat{\zeta}_l$.
(If $\varphi_L > 0$, ζ_l and γ_l were learned, for $l = \hat{L} + 1, ..., \hat{L} + \varphi_L$, to maintain the fairness.)

We ran the adaptive PT algorithm and the conventional PT algorithms for 10^6 iterations respectively. The initial condition were $w_{l,m}^{(0)} = 1/6$, $\sigma_{l,m}^{2(0)} \sim IG(\alpha_g, \beta_g)$, $\mu_{l,m}^{(0)} \sim U[\min(y_{1:150}), \max(y_{1:150})]$ for each run. The initial values of $\gamma_1^{(0)}, \ldots, \gamma_L^{(0)}$ were the sorted L random numbers from $U[0.0001, 800]$. The variables of posterior were divided into four blocks, the numbers of which are (5,4,4,4). Each Metropolis algorithm updated for the every block.

We evaluated the estimators of μ_m, $m = 1, \ldots, 6$. The accuracy of the estimation was evaluated by the root mean square error (RMSE). To evaluate the total error of the six estimators, RMSE takes the root average of the errors of the six estimators, that is,

$$\mathrm{RMSE}(i) = \left(\frac{1}{6}\sum_{m=1}^{6}(\bar{\mu}_m(i) - 2.5)^2\right)^{1/2},$$

where $\bar{\mu}_m(i)$ is the estimator, which is the average of samples $\mu_m^{(n)}$, in the ith trial, and 2.5 is the true value.

As a result, our algorithm can obtain appropriate parameters and achieve very low RMSEs. Fig. 4(a) and (b) show the RMSEs of the adaptive PT algorithm and the conventional PT algorithms in 50 runs were less than those of the conventional PT algorithms with shifted parameters. On the other hand, even if the number of temperatures increases the RMSEs don't increase (Fig. 4(c)), but the computational costs of the algorithms increase. In fact, the shifted inverse temperatures could not control the exchange ratios well (Fig. 5).

5 Generalization to Auxiliary Variable Methods

The idea of the adaptive PT algorithm is applicable to general AVMs. AVMs are mathematically formulated as below.

Let $\pi(dx)$ be a distribution on a state space $\mathcal{X}$ with σ-algebra $F_{\mathcal{X}}$ and $\pi_\lambda(dy|x)$ be a conditional distribution on a state space $\mathcal{Y}$ with σ-algebra $F_{\mathcal{Y}}$ given $F_{\mathcal{X}}$, where $\lambda \in \Lambda$ is a parameter vector. Then, the marginal distribution on $\mathcal{X}$ of the joint distribution $\pi_\lambda(dx, dy) = \pi_\lambda(dy|x)\pi(dx)$ is $\pi(dx)$ irrespective of $\pi_\lambda(dy|x)$.

In case of MCMC methods with auxiliary variables, $\pi(dx)$ corresponds to the target distribution and $\pi_\lambda(dy|x)$ to the auxiliary distributions. We term an MCMC method that draw samples (x', y') from $\pi_\lambda(dx, dy)$ to obtain x' an auxiliary variable method.

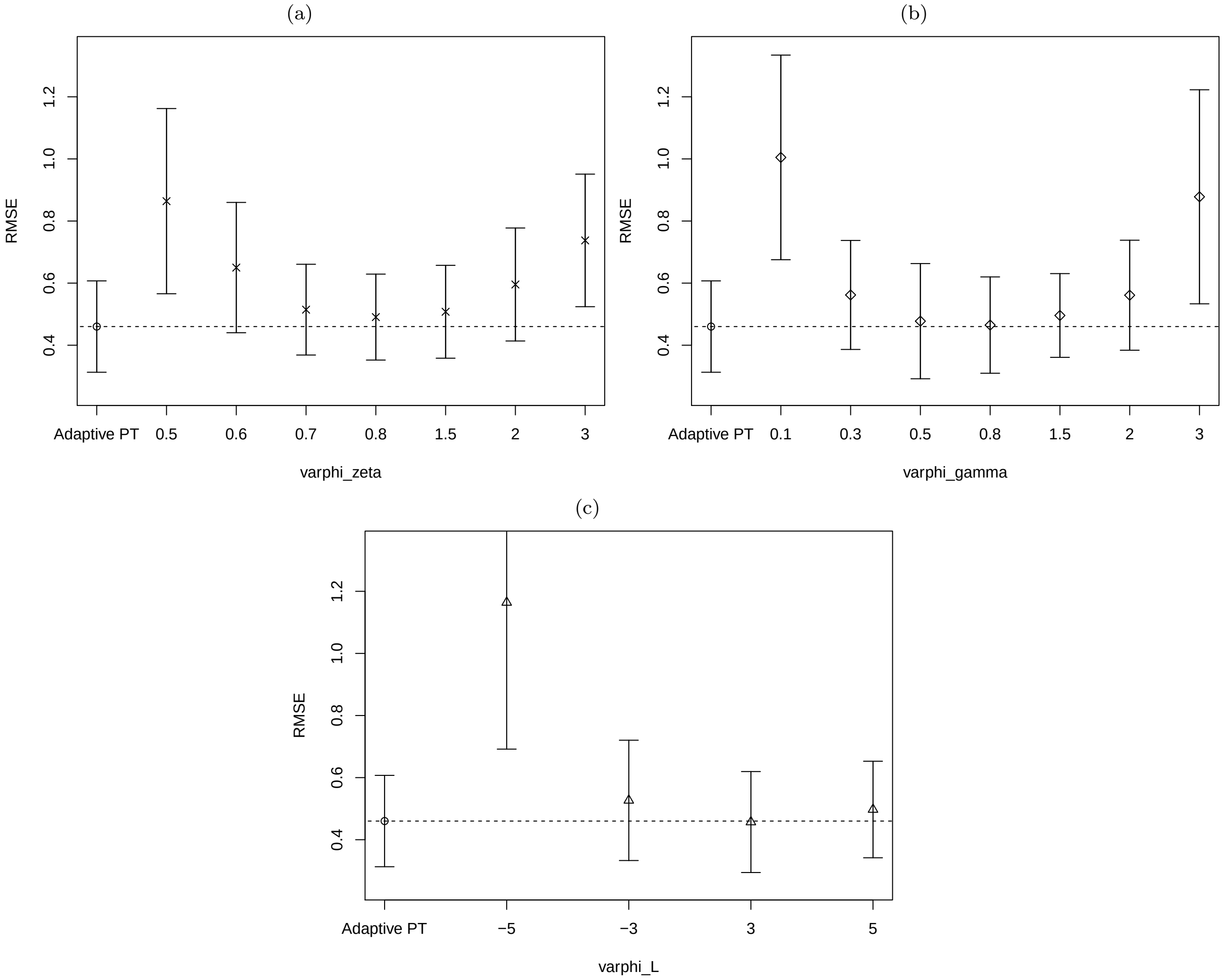

Figure 4: RMSEs in 50 runs. Each plot displays an average and a standard error of RMSEs for each algorithm in a mark and a radius of the error bar, respectively. The adaptive PT algorithm : (○). The conventional PT algorithms with the shifted parameters are separately plotted in these figures. The inverse temperatures : × in (a), The proposal variances : ⋄ in (b), The number of inverse temperatures : △ in (c).

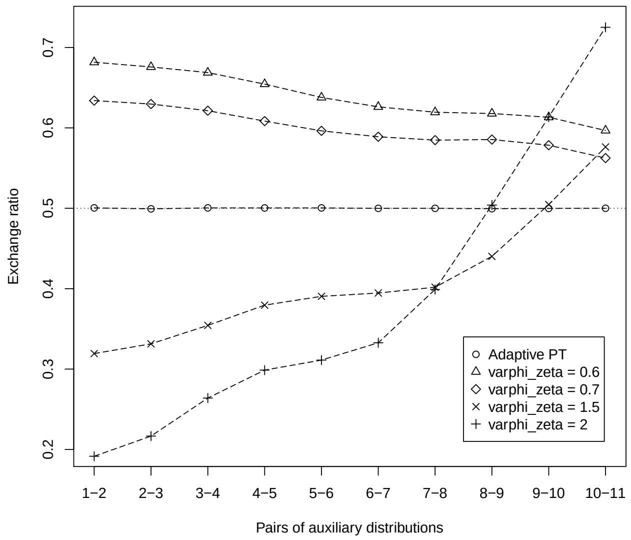

Figure 5: The averages of the estimated exchange ratios in 50 runs. The ratios of the adaptive PT algorithm were almost the predefined value while the conventional PT algorithm with shifted inverse temperatures could not be tuned constant and a moderate value (0.5), a dotted line.

In the PT algorithms, for example, the auxiliary distributions are $\pi_\lambda(dy|x) = \prod_{l=2}^{L} \pi_{t_l}(dx_l)$, $\lambda = (t_2, \ldots, t_L)$ and the auxiliary variables $y = (x_2, ..., x_L)$.

In order to introduce adaptation, we need to consider time-varying parameters. Let $\{P_\theta((x,y),(dx,dy))\}_{\theta\in\Theta}$ be a family of Markov transition kernels on $\mathcal{X} \times \mathcal{Y}$ with stationary distribution $\pi_\lambda(dx,dy)$, that is,

$$\begin{aligned}(\pi_\lambda P_\theta)(dx,dy) &= \iint_{x',y'} \pi_\lambda(dx',dy') P_\theta((x',y'),(dx,dy)) \\ &= \pi_\lambda(dx,dy),\end{aligned}$$

where $\lambda \subseteq \theta$. Then, the adaptive MCMC for AVM updates the parameters θ during generating chains $(x^{(n)}, y^{(n)})$ by P_θ as the following pseudo code.

```
Algorithm  Adaptive MCMC for AVM
  Initialize $(x^{(0)}, y^{(0)}), \theta^{(0)}$.
  for $n = 0$ to $N-1$ do
    [1] $(x^{(n+1)}, y^{(n+1)}) \sim P_{\theta^{(n)}}((x^{(n)}, y^{(n)}), (dx, dy))$
    [2] Update $\theta^{(n)}$ to $\theta^{(n+1)}$ by using the result of
    step 1. such as $(x^{(n+1)}, y^{(n+1)})$.
  end for
```

In the adaptive PT algorithms, for example, the time-varying parameter vector is $\theta = (\gamma_1, \ldots, \gamma_L; t_2, \ldots, t_L)$.

6 Convergence Theorem

Atchade (2011) and Fort et al. (2011) proved convergence theorems of adaptive MCMC algorithms that learn the parameters of the target distribution. The conditions for convergence in their theorems are, however, technical and strict. For example, the stationary distribution must converge. These conditions will considerably restrict the available learning algorithms.

In this section, we show some convergence theorems that our algorithm in the previous section converges under weaker conditions. Here, convergence means that an algorithm is ergodic, that is,

$$\lim_{n\to\infty} \|A^{(n)}((x,y,\theta),dx) - \pi(dx)\| = 0, \qquad \forall (x,y) \in \mathcal{X} \times \mathcal{Y}, \theta \in \Theta,$$

where $\|\mu(dx) - \nu(dx)\| = \sup_{A\in\mathcal{F}_\mathcal{X}} |\mu(A) - \nu(A)|$ and

$$\begin{aligned}&A^{(n)}((x,y,\theta),B_\mathcal{X}) \\ &= P\left[x^{(n)} \in B_\mathcal{X} | x^{(0)} = x, y^{(0)} = y, \theta^{(0)} = \theta\right], \\ &\qquad B_\mathcal{X} \in F_\mathcal{X}.\end{aligned}$$

Theorem 1. *The adaptive MCMC for AVM is ergodic if the following conditions hold:*

(a) Simultaneous uniform ergodicity

$$\begin{aligned}&\forall \varepsilon > 0,\ \exists N \in \mathbb{N}\ s.t. \\ &\quad \|P_\theta^N((x,y),dx) - \pi(dx)\| \le \varepsilon, \\ &\qquad \forall (x,y) \in \mathcal{X} \times \mathcal{Y}, \theta \in \Theta.\end{aligned} \tag{6}$$

(b) Diminishing adaptation

$$\begin{aligned}&\lim_{n\to\infty} \sup_{(x,y)\in\mathcal{X}\times\mathcal{Y}} \\ &\|P_{\theta^{(n+1)}}((x,y),(dx,dy)) - P_{\theta^{(n)}}((x,y),(dx,dy))\| \\ &= 0 \textit{ in probability.}\end{aligned} \tag{7}$$

Proof. See Appendix A. □

The above conditions do not require that the auxiliary parameter $\lambda^{(n)}$ and the stationary distribution $\pi_{\lambda^{(n)}}$ converge. The condition (a) can be replaced with more concrete condition that checks only properties of the Markov transition kernel as follows.

(a') (Simultaneously strongly aperiodically geometrical ergodicity) There exists $C \in \mathcal{F}_{\mathcal{X}\times\mathcal{Y}}$, $V : \mathcal{X} \times \mathcal{Y} \to [1,\infty)$, $\delta > 0$, $\tau < 1$, and $b < \infty$, such that $\sup_C V < \infty$ and the following conditions hold for all $\theta \in \Theta$.

(i) (Strongly aperiodic minorisation condition) There exist a probability measure $\nu_\theta(dx,dy)$ on C such that $P_\theta((x,y),(dx',dy')) \ge \delta\nu_\theta(dx',dy')$ for all $x, y \in C$.

(ii) (Geometric drift condition)

$$(P_\theta V)(x,y) \le \tau V(x,y) + b\mathbf{1}_{\{C\}}(x,y), \qquad for\ all\ x,y \in \mathcal{X} \times \mathcal{Y},$$

where $(P_\theta V)(x,y) \equiv \iint P_\theta((x,y),(dx',dy')) V(x',y') dx'dy'$, and $\mathbf{1}_{\{\cdot\}}(x)$ is the indicator function.

Theorem 2. *The adaptive MCMC for AVM is ergodic if the condition (b) in Theorem 1, the condition (a') and $E[V(x^{(0)}, y^{(0)})] < \infty$ hold.*

Proof. Straightforward from Proposition 3 and the proof of Theorem 3 in Roberts and Rosenthal (2007), and Theorem 1. □

Theorem 3 (Weak law of large numbers)**.** *Suppose an adaptive MCMC for AVM satisfies the conditions (a) and (b) and let $g : \mathcal{X} \to \mathbb{R}$ be a bounded measurable function. Then,*

$$\frac{1}{n}\sum_{i=1}^{n} g(x^{(i)}) \to \int g(x)\pi(dx) \quad \textit{in probability}$$

as $n \to \infty$ for any initial values $(x, y) \in \mathcal{X} \times \mathcal{Y}$ and $\theta \in \Theta$.

Proof. Straightforward from the coupling argument (Roberts and Rosenthal 2007). □

The convergence of the adaptive PT algorithm is proved by applying Theorem 2 as below.

Theorem 4. *The adaptive PT algorithm is ergodic if the following conditions hold:*

(s1) *The support S of the target distribution $\pi(dx)$ is compact and the density $\pi(x)$ is continuous and positive on S.*

(s2) *The family of proposal densities $\{q_\gamma\}_{\gamma\in\Gamma^p}$ is continuous and positive on $S^2\times\Gamma^p$, where $\Gamma = [c, C]$.*

Proof. See Appendix B. □

It will be possible to remove the assumption that S is compact by extending Theorem 6 of Bai et al. (2011).

7 Conclusions

This paper proposed adaptive MCMC algorithms to learn parameters of proposal distributions and auxiliary distributions simultaneously, and proved convergence theorems that give weak sufficient conditions for convergence.

We applied this framework to the Parallel Tempering algorithm and showed that the adaptive PT algorithm can adapts its parameters on the fly so that samples mix rapidly by experiments with a mixture model. We also presented that the performance of the PT algorithm depends on its parameters and the adaptive PT algorithm finds good parameters through experiments for Bayesian estimation.

Although we discussed the PT algorithm in a real space so far, we consider the idea of adaptation is applicable to those in a discrete space. We also consider our adaptive framework is applicable to other auxiliary variable methods such as the Gibbs variable selection, the partial decoupling method (one of the cluster Monte Carlo methods) and so on. We will extend our theory to the new field in the future.

Appendix

A Proof of Theorem 1

Let $\epsilon > 0$, and choose $N \in \mathbb{N}$ as in condition (a). From condition (b) and the coupling argument in the proof of Theorem 1 of Roberts and Rosenthal (2007), the following result holds.

There exists $n^* \in \mathbb{N}$ large enough so that for $K > n^* + N$, there exists a second chain $\{x'^{(n)}, y'^{(n)}\}_{n=K-N}^{K}$, such that $(x'^{(K-N)}, y'^{(K-N)}) = (x^{(K-N)}, y^{(K-N)})$ and $(x'^{(n+1)}, y'^{(n+1)}) \sim P_{\theta(K-N)}((x'^{(n)}, y'^{(n)}), dx, dy)$ for $n = K-N, ..., K-1$, and $P(x^{(K)} \neq x'^{(K)}) \leq 2\epsilon$.

Then it follows that

$$||P(x^{(K)} \in dx) - P(x'^{(K)} \in dx)|| \leq 2\epsilon, \tag{8}$$

where $P(x^{(K)} \in dx)$ denotes the distribution of $x^{(K)}$. (Because of $||P(y \in dx) - P(z \in dx)|| \leq P(y \neq z)$.)

On the other hand, from the condition (a), for all $A_{\mathcal{X}} \in F_{\mathcal{X}}$, we have

$$\begin{aligned}\epsilon &\geq \left|E[P^N_{\theta(K-N)}((x^{(K-N)}, y^{(K-N)}), A_{\mathcal{X}}) - \pi(A_{\mathcal{X}})]\right| \\ &= \left|P(x'^{(K)} \in A_{\mathcal{X}}) - \pi(A_{\mathcal{X}})\right|.\end{aligned}$$

That is,

$$||P(x'^{(K)} \in dx) - \pi(dx)|| \leq \epsilon. \tag{9}$$

From inequality (8) and (9), we have

$$||P(x^{(K)} \in dx) - \pi(dx)|| \leq 3\epsilon. \tag{10}$$

Since $K \geq n^*+N$ is arbitrary, the algorithm is ergodic.

B Proof of Theorem 4

We prove the sufficient conditions of convergence in Theorem 2 are satisfied. Firstly, we prove the condition (a') holds.

Let Borel σ-algebra on $\mathbb{R}^p$ be $\mathcal{B}(\mathbb{R}^p)$. For $\boldsymbol{x} \in S^L$, $\boldsymbol{\gamma} \in \Gamma^{pL}$, $\boldsymbol{t} \in \mathcal{T}^L$ and $\boldsymbol{B} = B_1 \times B_2 \times \cdots \times B_L$, $B_l \in \mathcal{B}(S)$, the transition kernel of the PT algorithm is

$$K_{\boldsymbol{\gamma},\boldsymbol{t}}(\boldsymbol{x}, \boldsymbol{B}) = \alpha_r \prod_{l=1}^{L} P_{\gamma_l,t_l}(x_l, B_l) + (1-\alpha_r)\sum_{l=2}^{L} \varsigma_l k_{l,l-1}(\boldsymbol{x}, \boldsymbol{B}), \tag{11}$$

where $0 \leq \varsigma_l \leq 1$, $\sum_{l=2}^{L} \varsigma_l = 1$, $P_{\gamma_l,t_l}(x_l, dx_l)$ and $k_{l,l-1}(\boldsymbol{x}, d\boldsymbol{x'})$ are the Metropolis transition kernel for $\pi_{t_l}(dx_l)$ and the transition kernel of an exchange process of x_l and x_{l-1}, respectively.

By condition (s1), we have $d \equiv \sup_{x\in S, t\in\mathcal{T}} \pi_t(x) < \infty$. By the compactness of S and condition (s2), we have also $\delta \equiv \inf_{x,x'\in S,\gamma\in\Gamma^p} q_\gamma(x, x') > 0$.

For $x \in S$ and $t \in \mathcal{T}$, denote $R_{x,t} = \left\{y \in S | \frac{\pi_t(y)}{\pi_t(x)} \leq 1\right\}$. For $x_l \in S$, $B_l \in \mathcal{B}(S)$, $t_l \in \mathcal{T}$

and $\gamma_l \in \Gamma$, we have

$$\begin{aligned}
&P_{\gamma_l,t_l}(x_l, B_l) \\
&= \int_{B_l} q_{\gamma_l}(x_l, x_l') \min\left(1, \frac{\pi_{t_l}(x_l')}{\pi_{t_l}(x_l)}\right) dx_l' \\
&+ \mathbf{1}_{\{B_l\}}(x_l) \int_S q_{\gamma_l}(x_l, \tilde{x}_l) \left\{1 - \min\left(1, \frac{\pi_{t_l}(\tilde{x}_l)}{\pi_{t_l}(x_l)}\right)\right\} d\tilde{x}_l \\
&= \int_{B_l \cap R_{x_l,t_l}} q_{\gamma_l}(x_l, x_l') \frac{\pi_{t_l}(x_l')}{\pi_{t_l}(x_l)} dx_l' \\
&+ \int_{B_l \cap R^c_{x_l,t_l}} q_{\gamma_l}(x_l, x_l') dx_l' \\
&\geq \frac{\delta}{d} \int_{B_l \cap R_{x_l,t_l}} \pi_{t_l}(x_l') dx_l' + \frac{\delta}{d} \int_{B_l \cap R^c_{x_l,t_l}} \pi_{t_l}(x_l') dx_l' \\
&= \frac{\delta}{d} \pi_{t_l}(B_l).
\end{aligned}$$

From Eq. (11), this inequality leads to

$$\begin{aligned}
K_{\boldsymbol{\gamma},\boldsymbol{t}}(\boldsymbol{x}, \boldsymbol{B}) &\geq \alpha_r \prod_{l=1}^{L} P_{\gamma_l,t_l}(x_l, B_l) \\
&\geq \alpha_r \prod_{l=1}^{L} \frac{\delta}{d} \pi_{t_l}(B_l) \\
&= \alpha_r \frac{\delta^L}{d^L} \pi_{\boldsymbol{t}}(\boldsymbol{B}), \qquad (12)
\end{aligned}$$

where $\pi_{\boldsymbol{t}}(\boldsymbol{B}) = \prod_{l=1}^{L} \pi_{t_l}(B_l)$ is a probability measure on S^L. Since the inequality (12) holds for all $\boldsymbol{B} \in \mathcal{B}(S^L)$, the condition (a')(i) follows.

Let $0 < \tau < 1$, $V(\boldsymbol{x}) = 1$ if $\boldsymbol{x} \in S^L$, otherwise $V(\boldsymbol{x}) \equiv 1/\tau$, and $b = 1 - \tau$. Then we have

$$(K_{\boldsymbol{\gamma},\boldsymbol{t}} V)(\boldsymbol{x}) \leq \tau V(\boldsymbol{x}) + b \mathbf{1}_{\{S^L\}}(\boldsymbol{x}), \quad \forall \boldsymbol{x} \in \mathbb{R}^{pL}. \qquad (13)$$

This inequality implies that the condition (a')(ii) is satisfied. Also we have $E[V(x^{(0)}, y^{(0)})] \leq 1/\tau < \infty$.

From Eq. (2) and (3), it follows that $t_l^{(n+1)} - t_l^{(n)} \to 0$ $a.s.$ and $\gamma_l^{(n+1)} - \gamma_l^{(n)} \to 0$ as $n \to \infty$. The minimum inverse temperature decision process changes the value of ς_l only finite times. Thus, the condition (b) in Theorem 1 holds.

The proof is complete.

References

[1] Andrieu, C., Moulines, E.: On the ergodicity properties of some adaptive mcmc algorithms. Annals of Applied Probability **16**(3), 1462–1505 (2006)

[2] Atchade, Y.: A computational framework for empirical bayes inference. Stat. Comput. **21**(4), 463–473 (2011)

[3] Bai, Y., Roberts, G., Rosenthal, J.: On the containment condition for adaptive markov chain monte carlo algorithms. Advances and Applications in Statistics **21**(1), 1–54 (2011)

[4] Dellaportas, P., Forster, J.J., Ntzoufras, I.: On bayesian model and variable selection using mcmc. Stat. Comput. **12**(1), 27–36 (2002)

[5] Fort, G., Moulines, E., Priouret, P.: Convergence of adaptive and interacting markov chain monte carlo algorithms. Ann. Stat. **39**(6), 3262–3289 (2011)

[6] Geyer, C.: Markov chain monte carlo maximum likelihood. Proc. 23rd Symp. Interface Comput. Sci. Statist pp. 156–216 (1991)

[7] Gilks, W., Roberts, G., Sahu, S.: Adaptive markov chain monte carlo through regeneration. Journal of the American Statistical Association **93**(443), 1045–1054 (1998)

[8] Goswami, G., Liu, J.: On learning strategies for evolutionary monte carlo. Stat. Comput. **17**(1), 23–38 (2007)

[9] Haario, H., Saksman, E., Tamminen, J.: An adaptive metropolis algorithm. Bernoulli **7**(2), 223–242 (2001)

[10] Hastings, W.: Monte carlo sampling methods using markov chains and their applications. Biometrika **57**(1), 97–109 (1970)

[11] Higdon, D.M.: Auxiliary variable methods for markov chain monte carlo with applications. J. Am. Stat. Assoc. **93**(442), 585–595 (1998)

[12] Hukushima, K., Nemoto, K.: Exchange monte carlo method and application to spin glass simulations. Journal of the Physical Society of Japan **65**(6), 1604–1608 (1996)

[13] Jasra, A., Stephens, D., Holmes, C.: On population-based simulation for statics inference. Stat. Comput. **17**(3), 263–279 (2007)

[14] Liu, J.: Monte Carlo Strategies in Scientific Computing. Springer, New York (2001)

[15] Nagata, K., Watanabe, S.: Asymptotic behavior of exchange ratio in exchange monte carlo method. Neural Networks **21**(7), 980–988 (2008)

[16] Robert, C., Casella, G.: Monte Carlo Statistical Methods. Springer (2004)

[17] Roberts, G., Rosenthal, J.: Coupling and ergodicity of adaptive markov chain monte carlo algorithms. J. Appl. Probab. **44**(2), 458–475 (2007)

[18] Swendsen, R., Wang, J.: Nonuniversal critical dynamics in monte carlo simulations. Physical Review Letters **58**, 86–88 (1987)