Asymptotically Independent Markov Sampling: a new MCMC scheme for Bayesian Inference

James L. Beck and Konstantin M. Zuev[1]

Computing and Mathematical Sciences, Division of Engineering and Applied Science,

California Institute of Technology, USA

Abstract

In Bayesian statistics, many problems can be expressed as the evaluation of the expectation of a quantity of interest with respect to the posterior distribution. Standard Monte Carlo method is often not applicable because the encountered posterior distributions cannot be sampled directly. In this case, the most popular strategies are the importance sampling method, Markov chain Monte Carlo, and annealing. In this paper, we introduce a new scheme for Bayesian inference, called Asymptotically Independent Markov Sampling (AIMS), which is based on the above methods. We derive important ergodic properties of AIMS. In particular, it is shown that, under certain conditions, the AIMS algorithm produces a uniformly ergodic Markov chain. The choice of the free parameters of the algorithm is discussed and recommendations are provided for this choice, both theoretically and heuristically based. The efficiency of AIMS is demonstrated with three numerical examples, which include both multi-modal and higher-dimensional target posterior distributions.

KEY WORDS: Markov chain Monte Carlo, Importance Sampling, Simulated Annealing, Bayesian Inference.

1 Three cornerstones of computational Bayesian inference

In Bayesian statistics, many problems can be expressed as the evaluation of the expectation of a quantity of interest with respect to the posterior distribution. Standard Monte Carlo simulation [MU49], where expectations are estimated by sample averages based on samples drawn independently from the posterior, is often not applicable because the encountered posterior distributions are multi-dimensional non-Gaussian distributions that cannot be explicitly normalized. In this case, the most popular strategies are importance sampling and Markov chain Monte Carlo methods. We briefly review these two methods first because they play an important role in the new MCMC method introduced in this paper.

Importance sampling: This is nearly as old as the Monte Carlo method (see, for instance, [KM53]), and works as follows. Suppose we want to evaluate $\mathbb{E}_\pi[h]$ that is an expectation of a function of interest $h : \Theta \to \mathbb{R}$ under distribution[2] $\pi(\cdot)$ defined on a parameter space $\Theta \subseteq \mathbb{R}^d$,

$$\mathbb{E}_\pi[h] = \int_\Theta h(\theta)\pi(\theta)d\theta. \tag{1}$$

Suppose also that we are not able to sample directly from $\pi(\cdot)$, although we can compute $\pi(\theta)$ for any $\theta \in \Theta$ to within a proportionality constant. Instead, we sample from some

---

[1] Both authors contributed equally to this work. Corresponding author's email: zuev@caltech.edu.

[2] Unless otherwise stated, all probability distributions are assumed to have densities with respect to Lebesgue measure, $\pi(d\theta) = \pi(\theta)d\theta$. For simplicity, the same symbol will be used to denote both the distribution and its density, and we write $\theta \sim \pi(\cdot)$ to denote that θ is distributed according to $\pi(\cdot)$.

other distribution $q(\cdot)$ on Θ which is readily computable for any $\theta \in \Theta$. Let $\theta^{(1)}, \ldots, \theta^{(N)}$ be N i.i.d. samples from $q(\cdot)$, and $w^{(i)} = \pi(\theta^{(i)})/q(\theta^{(i)})$ denote the *importance weight* of the ith sample, then we can estimate $\mathbb{E}_\pi[h]$ by

$$\hat{h}_N = \frac{\sum_{i=1}^N w^{(i)} h(\theta^{(i)})}{\sum_{i=1}^N w^{(i)}}. \tag{2}$$

The estimator $\hat{h}_N$ converges almost surely as $N \to \infty$ to $\mathbb{E}_\pi[h]$ by the Strong Law of Large Numbers for any choice of distribution $q(\cdot)$, provided $\text{supp}(\pi) \subseteq \text{supp}(q)$. Note that the latter condition automatically holds in Bayesian updating using data $\mathcal{D}$ where $q(\theta) = \pi_0(\theta)$ is the prior density and $\pi(\theta) \propto \pi_0(\theta) L(\theta)$ is the posterior $p(\theta|\mathcal{D})$, where L stands for the likelihood function $p(\mathcal{D}|\theta)$.

The estimator $\hat{h}_N$ in (2) generally has a smaller mean square error than a more straightforward unbiased importance sampling estimator:

$$\hat{h}'_N = \frac{1}{N} \sum_{i=1}^N w^{(i)} h(x^{(i)}). \tag{3}$$

This is especially clear when h is nearly a constant: if $h \approx c$, then $\hat{h}_N \approx c$, while $\hat{h}'_N$ has a larger variation. Although $\hat{h}_N$ is biased for any finite N, the bias can be made small by taking sufficiently large N, and the improvement in variance makes it a preferred alternative to $\hat{h}'_N$ [Li01, RC04]. Another major advantage of using $\hat{h}_N$ instead of $\hat{h}'_N$, which is especially important for Bayesian applications, is that in using the former we need to know $\pi(\theta)$ only up to a multiplicative normalizing constant; whereas in the latter, this constant must be known exactly.

The accuracy of $\hat{h}_N$ depends critically on the choice of the *importance sampling distribution* (ISD) $q(\cdot)$, which is also called the *instrumental* or *trial* distribution. If $q(\cdot)$ is chosen carelessly such that the the importance weights $w^{(i)}$ have a large variation, then $\hat{h}_N$ is essentially based only on the few samples $\theta^{(i)}$ with the largest weights, yielding generally a very poor estimate. Hence, for importance sampling to work efficiently, $q(\cdot)$ must be a good approximation of $\pi(\cdot)$ — "the importance sampling density should mimic the posterior density" [Ge89] — so that the variance $\text{var}_q[w]$ is not large. Since usually the prior and posterior are quite different, it is, therefore, highly inefficient to use the prior as the importance sampling distribution. When Θ is high-dimensional, and $\pi(\cdot)$ is complex, finding a good importance sampling distribution can be very challenging, limiting the applicability of the method [AB03].

For the estimator $\hat{h}'_N$ in (3), it is not difficult to show that the optimal importance sampling density, i.e., $q^*(\cdot)$ that minimizes the variance of $\hat{h}'_N$, is $q^*(\theta) \propto |h(\theta)|\pi(\theta)$. This result is sometimes attributed to Rubinstein [Ru81], although it was proved earlier by Kahn and Marshall [KM53]. It is not true, however, that $q^*(\cdot)$ is optimal for the estimator $\hat{h}_N$. Note also that this optimality result is not useful in practice, since when $h(\theta) \geq 0$, the required normalizing constant of $q^*(\cdot)$ is $\int_\Theta h(\theta)\pi(\theta)d\theta$, the integral of interest.

MCMC Sampling: Instead of generating independent samples from an ISD, we could generate dependent samples by simulating a Markov chain whose state distribution converges to the posterior distribution $\pi(\cdot)$ as its stationary distribution. *Markov chain Monte Carlo* sampling (MCMC) originated in statistical physics, and now is widely used in solving statistical problems [Ne93, GRS96, Li01, RC04].

The Metropolis-Hastings algorithm [MR^{2}T^{2}53, Ha70], the most popular MCMC technique, works as follows. Let $q(\cdot|\theta)$ be a distribution on Θ, which may or may not depend on $\theta \in \Theta$. Assume that $q(\cdot|\theta)$ is easy to sample from and it is either computable (up to a multiplicative constant) or symmetric, i.e. $q(\xi|\theta) = q(\theta|\xi)$. The sampling distribution $q(\cdot|\theta)$ is called the *proposal distribution*. Starting from essentially any $\theta^{(1)} \in \text{supp}(\pi)$, the Metropolis-Hastings algorithm proceeds by iterating the following two steps. First, generate a *candidate* state ξ from the proposal density $q(\cdot|\theta^{(n)})$. Second, either accept ξ as the next state of the Markov chain, $\theta^{(n+1)} = \xi$, with probability $\alpha(\xi|\theta^{(n)}) = \min\left\{1, \frac{\pi(\xi)q(\theta^{(n)}|\xi)}{\pi(\theta^{(n)})q(\xi|\theta^{(n)})}\right\}$; or reject ξ and set $\theta^{(n+1)} = \theta^{(n)}$ with the remaining probability $1 - \alpha(\xi|\theta^{(n)})$. It can be shown (see, for example, [RC04]), that under fairly weak conditions, $\pi(\cdot)$ is the stationary distribution of the Markov chain $\theta^{(1)}, \theta^{(2)}, \ldots$ and

$$\lim_{N \to \infty} \frac{1}{N} \sum_{i=1}^{N} h(\theta^{(i)}) = \int_{\Theta} h(\theta)\pi(\theta)d\theta. \tag{4}$$

Since the chain needs some time (so called "burn-in" period) to converge to stationarity, in practice, an initial portion of, say, N_0 states is usually discarded and

$$\tilde{h}_N = \frac{1}{N - N_0} \sum_{i=N_0+1}^{N} h(\theta^{(i)}) \tag{5}$$

is used as an estimator for $\mathbb{E}_\pi[h]$.

The two main special cases of the Metropolis-Hastings algorithm are Independent Metropolis-Hastings (IMH), where the proposal distribution $q(\xi|\theta) = q_g(\xi)$ is independent of θ (so q_g is a *global proposal*), and Random Walk Metropolis-Hastings (RWMH), where the proposal distribution is of the form $q(\xi|\theta) = q_l(\xi - \theta)$, i.e. a candidate state is proposed as $\xi = \theta^{(n)} + \epsilon_n$, where $\epsilon_n \sim q_l(\cdot)$ is a random perturbation (so q_l is a *local proposal*). In both cases, the choice of the proposal distribution strongly affects the efficiency of the algorithms. For IMH to work well, as with importance sampling, the proposal distribution must be a good approximation of the *target distribution* $\pi(\cdot)$, otherwise a large fraction of the candidate samples will be rejected and the Markov chain will be too slow in covering the important regions for $\pi(\cdot)$. When, however, it is possible to find a proposal $q_g(\cdot)$, such that $q_g(\cdot) \approx \pi(\cdot)$, IMH should always be preferred to RWMH because of better efficiency, i.e. better approximations of $\mathbb{E}_\pi[h]$ for a given number of samples N. Unfortunately, such a proposal is difficult to construct in the context of Bayesian inference where the posterior $\pi(\cdot)$ is often complex and high-dimensional. This limits the applicability of IMH.

Since the random walk proposal $q_l(\cdot)$ is local, it is less sensitive to the target distribution. That is why, in practice, RWMH is more robust and used more frequently than IMH. Nonetheless, there are settings where RWMH also does not work well because of the complexity of the posterior distribution. Although (4) is true in theory, a potential problem with RWMH (and, in fact, with any MCMC algorithm) is that the generated samples $\theta^{(1)}, \ldots, \theta^{(N)}$ often consist of highly correlated samples. Therefore, the estimator $\tilde{h}_N$ in (5) obtained from these samples tends to have a large variance for a modest amount of samples. This is especially true when the posterior distribution contains several widely-separated modes: a chain will move between modes only rarely and it will take a long time before it reaches stationarity. If this is the case, an estimate produced

by $\tilde{h}_N$ will be very inaccurate. At first glance, it seems natural to generate several independent Markov chains, starting from different random seeds, and hope that different chains will get trapped by different modes. However, multiple runs will not in general generate a sample in which each mode is correctly represented, since the probability of a chain reaching a mode depends more on the mode's "basin of attraction" than on the probability concentrated in the mode [Ne96].

Annealing: The concept of *annealing* (or *tempering*), which involves moving from an easy-to-sample distribution to the target distribution via a sequence of intermediate distributions, is one of the most effective methods of handling multiple isolated modes. Together with importance sampling and MCMC, annealing constitutes the third cornerstone of computational Bayesian inference.

The idea of using the RWMH algorithm in conjunction with annealing was introduced independently in [KGV83] and [Če85] for solving difficult optimization problems. The resulting algorithm, called *Simulated Annealing*, works as follows. Suppose we want to find the global minimum of a function of interest $h : \Theta \to \mathbb{R}$. This is equivalent to finding the global maximum of $f_T(\theta) = \exp(-h(\theta)/T)$ for any given $T > 0$. By analogy with the Gibbs distribution in statistical mechanics, T is called the *temperature parameter*. Let $T_0 > T_1 > \ldots$ be a sequence of monotonically decreasing temperatures, in which T_0 is large enough so that the probability distribution $\pi_0(\theta) \propto f_{T_0}(\theta)$ is close to uniform, and $\lim_{j \to \infty} T_j = 0$. At each temperature T_j, the Simulated Annealing method generates a Markov chain with $\pi_j(\theta) \propto \exp(-h(\theta)/T_j)$ as its stationary distribution. The final state of the Markov chain at simulation level j is used as the initial state for the chain at level $j + 1$. The key observation is that for any function h such that $\int_\Theta \exp(-h(\theta)/T)d\theta < \infty$ for all $T > 0$, distribution $\pi_j(\cdot)$, as j increases, puts more and more of its probability mass (converging to 1) into a neighborhood of the global minimum of h. Therefore, a sample drawn from $\pi_j(\cdot)$ would almost surely be in a vicinity of the global minimum of h when T_j is close to zero.

The success of Simulated Annealing in finding the global minimum crucially depends on the schedule of temperatures used in the simulation. It was proved in [GG84] that if a logarithmic schedule $T_j = T_0/\log(j + 1)$ is used, then, under certain conditions, there exists a value for T_0 such that use of this schedule guarantees that the global minimum of h will be reached almost surely. In practice, however, such a slow annealing schedule is not computationally efficient. It is more common to use either a geometric schedule, $T_{j+1} = \gamma T_j$ with $0 < \gamma < 1$, or some adaptive schedule, which defines the temperature for the next annealing level based on characteristics of the samples observed at earlier levels. For examples of adaptive annealing schedules, see, for instance, [Ne93].

In Bayesian inference problems, the idea of annealing is typically employed in the following way. First, we construct (in advance or adaptively) a sequence of distributions $\pi_0(\cdot), \ldots, \pi_m(\cdot)$ interpolating between the prior distribution $\pi_0(\cdot)$ and the posterior distribution $\pi(\cdot) \equiv \pi_m(\cdot)$. Next, we generate i.i.d. samples $\theta_0^{(1)}, \ldots, \theta_0^{(N)}$ from the prior, which is assumed to be readily sampled. Then, at each annealing level j, using some MCMC algorithm and samples $\theta_{j-1}^{(1)}, \ldots, \theta_{j-1}^{(N)}$ from the previous level $j - 1$, we generate samples $\theta_j^{(1)}, \ldots, \theta_j^{(N)}$ which are approximately distributed according to $\pi_j(\cdot)$. We proceed sequentially in this way, until the posterior distribution has been sampled. The rationale behind this strategy is that sampling from the multi-modal and, perhaps, high-dimensional posterior in such a way is likely to be more efficient than a straightforward MCMC sampling

of the posterior.

The problem of sampling a complex distribution is encountered in statistical mechanics, computational Bayesian inference, scientific computing, machine learning, and other fields. As a result, many different efficient algorithms have been recently developed, e.g. the method of Simulated Tempering [MP92, GT95], the Tempered Transition method [Ne96], Annealed Importance Sampling [Ne01], the Adaptive Metropolis-Hastings algorithm [BA02], Transitional Markov Chain Monte Carlo method [CC07], to name a few.

In this paper we introduce a new MCMC scheme for Bayesian inference, called *Asymptotically Independent Markov Sampling* (AIMS), which combines the three approaches described above — importance sampling, MCMC, and annealing — in the following way. Importance sampling with $\pi_{j-1}(\cdot)$ as the ISD is used for a construction of an approximation $\hat{\pi}_j^N(\cdot)$ of $\pi_j(\cdot)$, which is based on samples $\theta_{j-1}^{(1)}, \ldots, \theta_{j-1}^{(N)} \sim \pi_{j-1}(\cdot)$. This approximation is then employed as the independent (global) proposal distribution for sampling from $\pi_j(\cdot)$ by the IMH algorithm. Intermediate distributions $\pi_0(\cdot), \ldots, \pi_m(\cdot)$ interpolating between prior and posterior are constructed adaptively, using the essential sample size (ESS) to measure how much $\pi_{j-1}(\cdot)$ differs from $\pi_j(\cdot)$. When the number of samples $N \to \infty$, the approximation $\hat{\pi}_j^N(\cdot)$ converges to $\pi_j(\cdot)$, providing the optimal proposal distribution. In other words, when $N \to \infty$, the corresponding MCMC sampler produces independent samples, hence the name of the algorithm.

Remark 1. The term "Markov sampling" has several different meanings. In this paper it is used as synonymous to "MCMC sampling".

In this introductory section, we have described all the main ingredients that we will need in the subsequent sections. The rest of the paper is organized as follows. In Section 2, the AIMS algorithm is described. The ergodic properties of AIMS are derived in Section 3. The efficiency of AIMS is illustrated in Section 4 with three numerical examples that include both multi-modal and high-dimensional posterior distributions. Concluding remarks are made in Section 5.

2 Asymptotically Independent Markov Sampling

Let $\pi_0(\cdot)$ and $\pi(\cdot)$ be the prior and the posterior distributions defined on a parameter space Θ, respectively, so that, according to Bayes' Theorem, $\pi(\theta) \propto \pi_0(\theta)L(\theta)$, where L denotes the likelihood function for data $\mathcal{D}$. Our ultimate goal is to draw samples that are distributed according to $\pi(\cdot)$.

In Asymptotically Independent Markov Sampling (AIMS), we sequentially generate samples from intermediate distributions $\pi_0(\cdot), \ldots, \pi_m(\cdot)$ interpolating between the prior $\pi_0(\cdot)$ and the posterior $\pi(\cdot) \equiv \pi_m(\cdot)$. The sequence of distributions could be specially constructed for a given problem but the following scheme [Ne01, CC07] generally yields good efficiency:

$$\pi_j(\theta) \propto \pi_0(\theta)L(\theta)^{\beta_j}, \tag{6}$$

where $0 = \beta_0 < \beta_1 < \ldots < \beta_m = 1$. We will refer to j and β_j as the *annealing level* and the *annealing parameter* at level j, respectively. In the next subsection, we assume that β_j is given and therefore the intermediate distribution $\pi_j(\cdot)$ is also known. In Subsection 2.2, we describe how to choose the annealing parameters adaptively.

2.1 AIMS at annealing level j

Our first goal is to describe how AIMS generates sample $\theta_j^{(1)}, \ldots, \theta_j^{(N_j)}$ from $\pi_j(\cdot)$ based on the sample $\theta_{j-1}^{(1)}, \ldots, \theta_{j-1}^{(N_{j-1})} \sim \pi_{j-1}(\cdot)$ obtained at the previous annealing level. We start with an informal motivating discussion that leads to the simulation algorithm. In Section 3, we rigorously prove that the corresponding algorithm indeed generates samples which are asymptotically distributed according to $\pi_j(\cdot)$, as the sample size $N_j \to \infty$. Moreover, the larger N_{j-1}, the less correlated generated samples $\theta_j^{(1)}, \ldots, \theta_j^{(N_j)}$ are — a very desirable, yet rarely affordable, property for any MCMC algorithm.

Let $K_j(\cdot|\cdot)$ be any transition kernel such that $\pi_j(\cdot)$ is a stationary distribution with respect to $K_j(\cdot|\cdot)$. By definition, this means that

$$\pi_j(\theta)d\theta = \int_\Theta K_j(d\theta|\xi)\pi_j(\xi)d\xi \tag{7}$$

Applying importance sampling with the sampling density $\pi_{j-1}(\cdot)$ to integral (7), we have:

$$
\begin{aligned}
\pi_j(\theta)d\theta &= \int_\Theta K_j(d\theta|\xi)\frac{\pi_j(\xi)}{\pi_{j-1}(\xi)}\pi_{j-1}(\xi)d\xi \\
&\approx \sum_{i=1}^{N_{j-1}} K_j(d\theta|\theta_{j-1}^{(i)})\bar{w}_{j-1}^{(i)} \stackrel{\text{def}}{=} \hat{\pi}_j^{N_{j-1}}(d\theta),
\end{aligned}
\tag{8}
$$

where $\hat{\pi}_j^{N_{j-1}}(\cdot)$ will be used as the *global proposal* distribution in the Independent Metropolis-Hastings algorithm, and

$$w_{j-1}^{(i)} = \frac{\pi_j(\theta_{j-1}^{(i)})}{\pi_{j-1}(\theta_{j-1}^{(i)})} \propto L(\theta_{j-1}^{(i)})^{\beta_j - \beta_{j-1}} \quad \text{and} \quad \bar{w}_{j-1}^{(i)} = \frac{w_{j-1}^{(i)}}{\sum_{k=1}^{N_{j-1}} w_{j-1}^{(k)}} \tag{9}$$

are the importance weights and normalized importance weights, respectively. Note that to calculate $\bar{w}_{j-1}^{(i)}$, we do not need to know the normalizing constants of $\pi_{j-1}(\cdot)$ and $\pi_j(\cdot)$. If adjacent intermediate distributions $\pi_{j-1}(\cdot)$ and $\pi_j(\cdot)$ are sufficiently close (in other words, if $\Delta\beta_j = \beta_j - \beta_{j-1}$ is small enough), then the importance weights (9) will not vary wildly, and, therefore, we can expect that, for reasonably large N_{j-1}, approximation (8) is accurate.

Remark 2. In [CB10], the stationary condition (7) was used for an analytical approximation of the target PDF to evaluate the evidence (marginal likelihood) for a model.

Remark 3. Note that for any finite N_{j-1}, distribution $\hat{\pi}_j^{N_{j-1}}(\cdot)$ will usually have both continuous and discrete parts. This follows from the fact that the transition kernel in Markov chain simulation usually has the following form: $K(d\theta|\xi) = k(\theta|\xi)d\theta + r(\xi)\delta_\xi(d\theta)$, where $k(\cdot|\cdot)$ describes the continuous part of the transition kernel, $\delta_\xi(\cdot)$ denotes the Dirac mass at ξ, and $r(\xi) = 1 - \int_\Theta k(\theta|\xi)d\theta$. This is the form, for example, for the Metropolis-Hastings algorithm. Therefore, (8) must be understood as the approximate equality of distributions, not densities. In other words, (8) means that $\mathbb{E}_{\hat{\pi}_j^{N_{j-1}}}[h] \approx \mathbb{E}_{\pi_j}[h]$ and $\mathbb{E}_{\hat{\pi}_j^{N_{j-1}}}[h] \to \mathbb{E}_{\pi_j}[h]$, when $N_{j-1} \to \infty$, for all integrable functions h. See also Example 2.1 below.

From now on, we consider a special case where $K_j(\cdot|\cdot)$ is the random walk Metropolis-Hastings (RWMH) transition kernel. In this case, it can be written as follows:

$$K_j(d\theta|\xi) = q_j(\theta|\xi) \min\left\{1, \frac{\pi_j(\theta)}{\pi_j(\xi)}\right\} d\theta + (1 - a_j(\xi))\delta_\xi(d\theta), \tag{10}$$

where $q_j(\cdot|\xi)$ is a symmetric *local* proposal density, and $a_j(\xi)$ is the probability of having a proper transition ξ to $\Theta \setminus \{\xi\}$:

$$a_j(\xi) = \int_\Theta q_j(\theta|\xi) \min\left\{1, \frac{\pi_j(\theta)}{\pi_j(\xi)}\right\} d\theta \tag{11}$$

Example 2.1. As a simple illustration of (8), consider the case when $\pi_j(\cdot) = \mathcal{N}(\cdot|0,1)$, $\pi_{j-1}(\cdot) = \mathcal{N}(\cdot|0,2)$, and $q_j(\cdot|\xi) = \mathcal{N}(\cdot|\xi, 1/2)$, where $\mathcal{N}(\cdot|\mu, \sigma^2)$ denotes the Gaussian density with mean μ and variance σ^2. The approximation $\hat{\pi}_j^{N_{j-1}}(\cdot)$ based on the samples $\theta_{j-1}^{(1)}, \ldots, \theta_{j-1}^{(N_{j-1})} \sim \mathcal{N}(\cdot|0,2)$ is shown in the top panels of Figure 1, for $N_{j-1} = 5$ and $N_{j-1} = 50$. Suppose that $h_1(\theta) = \theta$ and $h_2(\theta) = \theta^2$ are the functions of interest. Then $\mathbb{E}_{\pi_j}[h_1] = 0$ and $\mathbb{E}_{\pi_j}[h_2] = 1$. The convergence of $h_1^*(N_{j-1}) = \mathbb{E}_{\hat{\pi}_j^{N_{j-1}}}[h_1]$ and $h_2^*(N_{j-1}) = \mathbb{E}_{\hat{\pi}_j^{N_{j-1}}}[h_2]$ is shown in the bottom panel of Figure 1.

For sampling from $\pi_j(\cdot)$, we will use the Independent Metropolis-Hastings algorithm (IMH) with the *global* proposal distribution $\hat{\pi}_j^{N_{j-1}}(\cdot)$. To accomplish this, we have to be able to calculate the ratio $\hat{\pi}_j^{N_{j-1}}(\theta)/\hat{\pi}_j^{N_{j-1}}(\xi)$ for any $\theta, \xi \in \Theta$ as a part of the expression for the acceptance probability $\alpha_j(\xi|\theta) = \min\left\{1, \frac{\pi_j(\xi)\hat{\pi}_j^{N_{j-1}}(\theta)}{\pi_j(\theta)\hat{\pi}_j^{N_{j-1}}(\xi)}\right\}$. However, as it has been already mentioned, the distribution $\hat{\pi}_j^{N_{j-1}}(\cdot)$ does not have a density since it has both continuous and discrete components, and, therefore, the ratio $\hat{\pi}_j^{N_{j-1}}(\theta)/\hat{\pi}_j^{N_{j-1}}(\xi)$ makes no sense. To overcome this "lack-of-continuity problem", taking into account (8) and (10), let us *formally* define the global proposal distribution over Θ as:

$$\hat{\pi}_j^{N_{j-1}}(\theta) \overset{\text{def}}{=} \sum_{i=1}^{N_{j-1}} \bar{w}_{j-1}^{(i)} q_j(\theta|\theta_{j-1}^{(i)}) \min\left\{1, \frac{\pi_j(\theta)}{\pi_j(\theta_{j-1}^{(i)})}\right\}, \tag{12}$$

if $\theta \notin \left\{\theta_{j-1}^{(1)}, \ldots, \theta_{j-1}^{(N_{j-1})}\right\}$, and

$$\hat{\pi}_j^{N_{j-1}}(\theta_{j-1}^{(k)}) \overset{\text{def}}{=} \infty \tag{13}$$

Note that $\hat{\pi}_j^{N_{j-1}}(\cdot)$ is a distribution on Θ, but it does not have a density. However, $\hat{\pi}_j^{N_{j-1}}(\cdot)$ induces another distribution on $\Theta \setminus \left\{\theta_{j-1}^{(1)}, \ldots, \theta_{j-1}^{(N_{j-1})}\right\}$ which does have a density, given by the r.h.s. of (12). This motivates (12).

Now, using (12) and (13), we can calculate the ratio $\hat{\pi}_j^{N_{j-1}}(\theta)/\hat{\pi}_j^{N_{j-1}}(\xi)$ as follows:

I. If $\theta, \xi \notin \left\{\theta_{j-1}^{(1)}, \ldots, \theta_{j-1}^{(N_{j-1})}\right\}$, then

$$\frac{\hat{\pi}_j^{N_{j-1}}(\theta)}{\hat{\pi}_j^{N_{j-1}}(\xi)} = \frac{\sum_{i=1}^{N_{j-1}} \bar{w}_{j-1}^{(i)} q_j(\theta|\theta_{j-1}^{(i)}) \min\left\{1, \frac{\pi_j(\theta)}{\pi_j(\theta_{j-1}^{(i)})}\right\}}{\sum_{i=1}^{N_{j-1}} \bar{w}_{j-1}^{(i)} q_j(\xi|\theta_{j-1}^{(i)}) \min\left\{1, \frac{\pi_j(\xi)}{\pi_j(\theta_{j-1}^{(i)})}\right\}} \tag{14}$$

II. If $\theta \notin \left\{ \theta_{j-1}^{(1)}, \ldots, \theta_{j-1}^{(N_{j-1})} \right\}$ and $\xi = \theta_{j-1}^{(k)}$, then

$$\frac{\hat{\pi}_j^{N_{j-1}}(\theta)}{\hat{\pi}_j^{N_{j-1}}(\xi)} = 0 \quad \text{and} \quad \alpha_j(\xi|\theta) = 0 \tag{15}$$

III. If $\theta = \theta_{j-1}^{(k)}$ and $\xi \notin \left\{ \theta_{j-1}^{(1)}, \ldots, \theta_{j-1}^{(N_{j-1})} \right\}$, then

$$\frac{\hat{\pi}_j^{N_{j-1}}(\theta)}{\hat{\pi}_j^{N_{j-1}}(\xi)} = \infty \quad \text{and} \quad \alpha_j(\xi|\theta) = 1 \tag{16}$$

IV. If $\theta = \theta_{j-1}^{(k)}$ and $\xi = \theta_{j-1}^{(l)}$, then $\frac{\hat{\pi}_j^{N_{j-1}}(\theta)}{\hat{\pi}_j^{N_{j-1}}(\xi)}$ is not defined.

Notice that in the first three cases the ratio $\hat{\pi}_j^{N_{j-1}}(\theta)/\hat{\pi}_j^{N_{j-1}}(\xi)$ is readily computable, while in Case IV, it is not even defined. Therefore, it is very desirable to avoid Case IV. The key observation that allows us to do this is the following: suppose that the initial state $\theta_j^{(1)}$ of the Markov chain that is generated is such that $\theta_j^{(1)} \in \Theta_j^* \stackrel{\text{def}}{=} \Theta \setminus \left\{ \theta_{j-1}^{(1)}, \ldots, \theta_{j-1}^{(N_{j-1})} \right\}$, then $\theta_j^{(i)} \in \Theta_j^*$ for all $i \geq 1$. Indeed, the only way for the chain to enter the set $\left\{ \theta_{j-1}^{(1)}, \ldots, \theta_{j-1}^{(N_{j-1})} \right\}$ is to generate a candidate state $\xi \in \left\{ \theta_{j-1}^{(1)}, \ldots, \theta_{j-1}^{(N_{j-1})} \right\}$; however, according to Case II, such a candidate will always be rejected. Thus, by replacing the state space Θ by Θ_j^* and using (14) and (15) for evaluation of $\hat{\pi}_j^{N_{j-1}}(\theta)/\hat{\pi}_j^{N_{j-1}}(\xi)$, we are able to calculate the acceptance probability $\alpha_j(\xi|\theta) = \min \left\{ 1, \frac{\pi_j(\xi)\hat{\pi}_j^{N_{j-1}}(\theta)}{\pi_j(\theta)\hat{\pi}_j^{N_{j-1}}(\xi)} \right\}$ involved in the IMH algorithm. It is clear that the replacement of Θ by Θ_j^* is harmless for the ergodic properties of the Markov chain when $\Theta \subseteq \mathbb{R}^d$.

Remark 4. One may wonder why not just use the continuous part of $\hat{\pi}_j^{N_{j-1}}(\cdot)$ as the global proposal density within the IMH algorithm. In other words, why not use the density $\hat{\pi}_{j,\text{cont}}^{N_{j-1}}(\cdot)$, which is proportional to the function defined by (12), as the proposal density. Indeed, in this case we would not have any difficulties with calculating the ratio $\hat{\pi}_j^{N_{j-1}}(\theta)/\hat{\pi}_j^{N_{j-1}}(\xi)$. The problem is that it is not clear how to sample from $\hat{\pi}_{j,\text{cont}}^{N_{j-1}}(\cdot)$, while sampling from $\hat{\pi}_j^{N_{j-1}}(d\theta) = \sum_{i=1}^{N_{j-1}} \bar{w}_{j-1}^{(i)} K_j(d\theta|\theta_{j-1}^{(i)})$ is straightforward.

The above discussion leads to the following algorithm for sampling from the distribution $\pi_j(\cdot)$:

AIMS at annealing level j

Input:
> $\theta_{j-1}^{(1)}, \ldots, \theta_{j-1}^{(N_{j-1})} \sim \pi_{j-1}(\cdot)$, samples generated at annealing level $j-1$;
> $\theta_j^{(1)} \in \Theta_j^* = \Theta \setminus \left\{ \theta_{j-1}^{(1)}, \ldots, \theta_{j-1}^{(N_{j-1})} \right\}$, initial state of a Markov chain;
> $q_j(\cdot|\xi)$, symmetric proposal density associated with the RWMH kernel;
> N_j, total number of Markov chain states to be generated.

Algorithm:
    for $i = 1, \ldots, N_j - 1$ do
        1) Generate a global candidate state $\xi_g \sim \hat{\pi}_j^{N_{j-1}}(\cdot)$ as follows:

a. Select k from $\{1, \ldots, N_{j-1}\}$ with probabilities $\bar{w}_{j-1}^{(i)}$ given by (9).

b. Generate a local candidate $\xi_l \sim q_j(\cdot | \theta_{j-1}^{(k)})$.

c. Accept or reject ξ_l by setting

$$
\xi_g = \begin{cases} \xi_l, & \text{with probability } \min\left\{1, \frac{\pi_j(\xi_l)}{\pi_j(\theta_{j-1}^{(k)})}\right\}; \\ \theta_{j-1}^{(k)}, & \text{with the remaining probability.} \end{cases} \tag{17}
$$

2) Update $\theta_j^{(i)} \to \theta_j^{(i+1)}$ by accepting or rejecting ξ_g as follows:

   if $\xi_g = \theta_{j-1}^{(k)}$

       Set $\theta_j^{(i+1)} = \theta_j^{(i)}$

   else

       Set

$$
\theta_j^{(i+1)} = \begin{cases} \xi_g, & \text{with probability } \min\left\{1, \frac{\pi_j(\xi_g)\hat{\pi}_j^{N_{j-1}}(\theta_j^{(i)})}{\pi_j(\theta_j^{(i)})\hat{\pi}_j^{N_{j-1}}(\xi_g)}\right\}; \\ \theta_j^{(i)}, & \text{with the remaining probability.} \end{cases} \tag{18}
$$

       end if

  end for

`Output:`

▶ $\theta_j^{(1)}, \ldots, \theta_j^{(N_j)}$, N_j states of a Markov chain with a stationary distribution $\pi_j(\cdot)$

Schematically, the AIMS algorithm at annealing level j is shown in Figure 2. The proof that $\pi_j(\cdot)$ is indeed a stationary distribution for the Markov chain generated by AIMS is given in Section 3.

Remark 5. As usually for MCMC algorithms, the fact of convergence of a Markov chain to its stationary distribution does not depend on the initial state; however, the speed of convergence does. One reasonable way to chose the initial state $\theta_j^{(1)} \in \Theta_j^*$ in practical applications is the following: generate $\theta_j^{(1)} \sim q_j(\cdot | \theta_{j-1}^{(k^*)})$, where $k^* = \arg\max_k \bar{w}_{j-1}^{(k)}$, i.e. $\theta_{j-1}^{(k^*)}$ has the largest normalized importance weight.

2.2   The full AIMS procedure

At the zero[th] annealing level, $j = 0$, we generate prior samples $\theta_0^{(1)}, \ldots, \theta_0^{(N_0)}$, which usually can be readily drawn directly by a suitable choice of the prior distribution $\pi_0(\cdot)$. Then, using the algorithm described in the previous subsection, we generate samples $\theta_1^{(1)}, \ldots, \theta_1^{(N_1)}$, which are approximately distributed according to intermediate distribution $\pi_1(\theta) \propto \pi_0(\theta) L(\theta)^{\beta_1}$. We proceed like this until the posterior distribution $\pi_m(\theta) \propto \pi_0(\theta) L(\theta)^{\beta_m}$ ($\beta_m = 1$) has been sampled. To make the description of AIMS complete, we have to explain how to choose the annealing parameters β_j, for $j = 2, \ldots, m-1$.

It is clear that the choice of the annealing parameters is very important, since, for instance, it affects the accuracy of the importance sampling approximation (8) and, therefore, the efficiency of the whole AIMS procedure. At the same time, it is difficult to make a rational choice of the β_j-values in advance, since this requires some prior knowledge about the posterior distribution, which is often not available. For this reason, we propose an adaptive way of choosing the annealing scheme.

In importance sampling, a useful measure of degeneracy of the method is the *effective sample size* (ESS) N^{eff} introduced in [KLW94] and [Li96]. The ESS measures how similar the importance sampling distribution $\pi_{j-1}(\cdot)$ is to the target distribution $\pi_j(\cdot)$. Suppose N_{j-1} independent samples $\theta_{j-1}^{(1)}, \ldots, \theta_{j-1}^{(N_{j-1})}$ are generated from $\pi_{j-1}(\cdot)$, then the ESS of these samples is defined as

$$N_{j-1}^{\text{eff}} = \frac{N_{j-1}}{1 + \text{var}_{\pi_{j-1}}[w]} = \frac{N_{j-1}}{\mathbb{E}_{\pi_{j-1}}[w^2]}, \tag{19}$$

where $w(\theta) = \pi_j(\theta)/\pi_{j-1}(\theta)$. The ESS can be interpreted as implying that N_{j-1} weighted samples $(\theta_{j-1}^{(1)}, w_{j-1}^{(1)}), \ldots, (\theta_{j-1}^{(N_{j-1})}, w_{j-1}^{(N_{j-1})})$ are worth $N_{j-1}^{\text{eff}}(\leq N_{j-1})$ i.i.d. samples drawn from the target distribution $\pi_j(\cdot)$. One cannot evaluate the ESS exactly but an estimate $\hat{N}_{j-1}^{\text{eff}}$ of N_{j-1}^{eff} is given by

$$\hat{N}_{j-1}^{\text{eff}}(\bar{w}_{j-1}) = \frac{1}{\sum_{i=1}^{N_{j-1}} (\bar{w}_{j-1}^{(i)})^2}, \tag{20}$$

where $\bar{w}_{j-1} = (\bar{w}_{j-1}^{(1)}, \ldots, \bar{w}_{j-1}^{(N_{j-1})})$ and $\bar{w}_{j-1}^{(i)}$ is the normalized importance weight of $\theta_{j-1}^{(i)}$.

At annealing level j, when β_{j-1} is already known, the problem is to define β_j. Let $\gamma = \hat{N}_{j-1}^{\text{eff}}/N_{j-1} \in (0, 1)$ be a prescribed threshold that characterizes the "quality" of the weighted sample (the larger γ is, the "better" the weighted sample is). Then we obtain the following equation:

$$\sum_{i=1}^{N_{j-1}} (\bar{w}_{j-1}^{(i)})^2 = \frac{1}{\gamma N_{j-1}} \tag{21}$$

Observe that this equation can be expressed as an equation for β_j by using (9):

$$\frac{\sum_{i=1}^{N_{j-1}} L(\theta_{j-1}^{(i)})^{2(\beta_j - \beta_{j-1})}}{\left(\sum_{i=1}^{N_{j-1}} L(\theta_{j-1}^{(i)})^{\beta_j - \beta_{j-1}}\right)^2} = \frac{1}{\gamma N_{j-1}} \tag{22}$$

Solving this equation for β_j gives us the value of the annealing parameter at level j.

Remark 6. Note that when $j \geq 2$, the $\theta_{j-1}^{(1)}, \ldots, \theta_{j-1}^{(N_{j-1})}$ are generated by the Markov chain sampler described in the previous subsection and therefore are not independent. This means that, because of the autocorrelations produced by the Markov chain used, the "true" ESS of this sample is, in fact, smaller than the one given by (19). This is useful to remember when choosing γ. Also, this is another reason to select the prior distribution $\pi_0(\cdot)$ so that samples can be generated independently at the start of each AIMS run.

Combining the AIMS algorithm at a given annealing level with the described adaptive annealing scheme gives rise to the following procedure.

The AIMS procedure

`Input:`
    ▷ γ, threshold for the effective sample size (ESS);
    ▷ $N_0, N_1, \ldots$, where N_j is the total number of Markov chain states to be generated at annealing level j;
    ▷ $q_1(\cdot|\xi), q_2(\cdot|\xi), \ldots$, where $q_j(\cdot|\xi)$ is the symmetric proposal density associated with the RWMH kernel at annealing level j.

```
Algorithm:
```
Set $j = 0$, current annealing level.

Set $\beta_0 = 0$, current annealing parameter.

Sample $\theta_0^{(1)}, \ldots, \theta_0^{(N_0)} \overset{i.i.d}{\sim} \pi_0(\cdot)$.

Calculate $\bar{W}_0^{(i)} = \frac{L(\theta_0^{(i)})^{1-\beta_0}}{\sum_{i=1}^{N_0} L(\theta_0^{(i)})^{1-\beta_0}}$, $i = 1, \ldots, N_0$.

Calculate the ESS $\hat{N}_0^{\text{eff}} = \hat{N}_0^{\text{eff}}(\bar{W}_0)$ using (20), which measures how similar the prior distribution $\pi_0(\cdot)$ is to the target posterior distribution $\pi(\cdot)$.

while $\hat{N}_j^{\text{eff}}/N_j < \gamma$ **do**

Find β_{j+1} from equation (22).

Calculate normalized importance weights $\bar{w}_j^{(i)}$, $i = 1, \ldots, N_j$ using (9).

Generate a Markov chain $\theta_{j+1}^{(1)}, \ldots, \theta_{j+1}^{(N_{j+1})}$ with the stationary distribution $\pi_{j+1}(\cdot)$ using the AIMS algorithm at annealing level $j + 1$.

Calculate $\bar{W}_{j+1}^{(i)} = \frac{L(\theta_{j+1}^{(i)})^{1-\beta_{j+1}}}{\sum_{i=1}^{N_{j+1}} L(\theta_{j+1}^{(i)})^{1-\beta_{j+1}}}$, $i = 1, \ldots, N_{j+1}$.

Calculate the ESS $\hat{N}_{j+1}^{\text{eff}} = \hat{N}_{j+1}^{\text{eff}}(\bar{W}_{j+1})$ using (20), which measures how similar the intermediate distribution $\pi_{j+1}(\cdot)$ is to the posterior $\pi(\cdot)$.

Increment j to $j + 1$.

end while

Set $\beta_{j+1} = 1$, current annealing parameter.

Set $m = j + 1$, the total number of distributions in the annealing scheme.

Set $\bar{w}_{m-1}^{(i)} = \bar{W}_{m-1}^{(i)}$, $i = 1, \ldots, N_{m-1}$.

Generate a Markov chain $\theta_m^{(1)}, \ldots, \theta_m^{(N_m)}$ with the stationary distribution $\pi_m(\cdot) = \pi(\cdot)$ using the AIMS algorithm at annealing level m.

```
Output:
```
▶ $\theta_m^{(1)}, \ldots, \theta_m^{(N_m)} \dot{\sim} \pi(\cdot)$, samples that are approximately distributed according to the posterior distribution.

2.3 Implementation issues

As it follows from the description, the AIMS procedure has the following parameters: γ, the threshold for the effective sample size; N_j, the length of a Markov chain generated at annealing level $j = 1, \ldots, m$; and $q_j(\cdot|\xi)$, the symmetric proposal density associated with the RWMH kernel at level $j = 1, \ldots, m$. Here, we discuss the choice of these parameters and how this choice affects the efficiency of AIMS.

First of all, it is absolutely clear that, as for any Monte Carlo method, the larger the number of generated samples is, the more accurate the corresponding estimates of (1) are. However, we would like to highlight the difference between the roles of N_{j-1} and N_j at annealing level j. While N_j is directly related to the convergence of the chain $\theta_j^{(1)}, \ldots, \theta_j^{(N_j)}$ to its stationary distribution $\pi_j(\cdot)$, N_{j-1} affects this convergence implicitly through the global proposal distribution $\hat{\pi}_j^{N_{j-1}}(\cdot)$: the larger N_{j-1}, the more accurate approximation (8) is, and, therefore, the less correlated $\theta_j^{(1)}, \ldots, \theta_j^{(N_j)}$ are. When $N_{j-1} \to \infty$, samples $\theta_j^{(1)}, \ldots, \theta_j^{(N_j)}$ become independent draws from $\pi_j(\cdot)$, hence the name of the algorithm. Thus, if we increase $N = N_{j-1} = N_j$, the effect is twofold: first, the sample size increases thereby increasing the effective number of independent samples at the jth level (typical for

any Monte Carlo method); second, the samples become less correlated (a useful feature of AIMS), again increasing the effective number of independent samples. As a result of these two effects, increasing N has a strong influence on the effective number of independent posterior samples and so strongly reduces the variance of the estimator for (1).

Suppose now that we are at the last annealing level and generating a Markov chain $\theta_m^{(1)}, \ldots, \theta_m^{(N_m)}$ with the stationary distribution $\pi_m(\cdot) = \pi(\cdot)$. We will refer to this chain as the posterior Markov chain. A critical question faced by users of MCMC methods is how to determine when it is safe to stop sampling from the posterior distribution and use samples $\theta_m^{(1)}, \ldots, \theta_m^{(N_m)}$ for estimation. In other words, how large should N_m be? One possible solution of this "convergence assessment problem" is to use one of the numerous published diagnostic techniques; for example, see [CC96] for a comparative review of MCMC convergence diagnostics. Unfortunately, none of the published diagnostics allows one to say with certainty that a finite sample from an MCMC algorithm is representative of an underlying stationary distribution. A more empirical approach for assessing convergence is to run several posterior Markov chains $\theta_{k,m}^{(1)}, \ldots, \theta_{k,m}^{(N_m)}$, $k = 1, \ldots, K$, in parallel and monitor the corresponding estimators $\hat{h}_1, \ldots, \hat{h}_K$ of $\mathbb{E}_\pi[h]$. A stopping rule for convergence is then

$$\max_{1 \leq i < j \leq K} |\hat{h}_i - \hat{h}_j| < \varepsilon, \tag{23}$$

where ε is a minimum precision requirement. It is important to emphasise, though, that rule (23), although easy-to-understand and easy-to-implement, does not assure convergence of the chains (especially if $\pi(\cdot)$ is multi-modal): "the potential for problems with multiple modes exists whenever there is no theoretical guarantee that the distribution is unimodal" [Ne01].

The threshold γ affects the speed of annealing. If γ is very small, i.e. close to zero, then AIMS will have very few intermediate distributions interpolating between the prior and posterior distributions, and this will lead to inaccurate results for a moderate number of samples. On the other hand, if γ is very large, i.e. close to one, then AIMS will have too many intermediate distributions, which will make the algorithm computationally very expensive.

The proposed method for finding β_j-values is based on the ESS, and β_j is defined from equation (21) (or, equivalently, from (22)). A similar adaptive approach for defining an annealing scheme was proposed in [CC07]. It is based on the coefficient of variation (COV) of the importance weights (9). More precisely, the equation for β_j is given by

$$\frac{\sqrt{\frac{1}{N_{j-1}} \sum_{i=1}^{N_{j-1}} \left(w_{j-1}^{(i)} - \frac{1}{N_{j-1}} \sum_{i=1}^{N_{j-1}} w_{j-1}^{(i)} \right)^2}}{\frac{1}{N_{j-1}} \sum_{i=1}^{N_{j-1}} w_{j-1}^{(i)}} = \delta, \tag{24}$$

where $\delta > 0$ is a prescribed threshold. It is easy to show that the ESS-criterion (21) and the COV-criterion (24) are mathematically equivalent; in fact, $\hat{N}_{j-1}^{\text{eff}} = N_{j-1}/(1 + \delta^2)$. We prefer to use the former criterion since γ has a clear meaning: it is the factor by which the (essential) sample size of the weighted sample is reduced as a penalty for sampling from the importance sampling density instead of the target distribution. It has been found in [CC07] that $\delta = 1$ is usually a reasonable choice of the threshold. This corresponds to $\gamma = 1/2$. Our simulation results (see Section 4) also show that annealing schemes with γ around $1/2$ yield good efficiency.

The choice of the local proposal density $q_j(\cdot|\xi)$ associated with the RWMH kernel determines the ergodic properties of the Markov chain generated by AIMS at level j; it also determines how efficiently the chain explores local neighborhoods of samples $\theta_{j-1}^{(1)}, \ldots, \theta_{j-1}^{(N_{j-1})}$ generated at the previous level. This makes the choice of $q_j(\cdot|\xi)$ very important.

It has been observed by many researchers that the efficiency of Metropolis-Hastings based MCMC methods is not sensitive to the type of the proposal density; however, it strongly depends on its variance (e.g. [GRG96, AB01]). For this reason, we suggest using a Gaussian density as the local proposal:

$$q_j(\theta|\xi) = \mathcal{N}(\theta|\xi, c_j^2 \mathbb{I}), \tag{25}$$

where ξ and $c_j^2 \mathbb{I}$ are the mean and diagonal covariance matrix, respectively. The scaling parameter c_j^2 determines the "spread" of the local proposal distribution. In Section 3, we prove (Theorem 3) that, under certain conditions, the acceptance rate $\bar{\mathcal{A}}_j$ (i.e. the expected probability of having a proper Markov transition $\theta_j^{(i)}$ to $\theta_j^{(i+1)} \neq \theta_j^{(i)}$) satisfies $\bar{\mathcal{A}}_j \geq \frac{1}{M}$, where constant M depends on $q_j(\cdot|\xi)$ and, therefore, on c_j^2. This result can be potentially used for finding an optimal c_j^2 that would minimize M. Alternatively, a more empirical way of choosing the scaling factor consists of adjusting c_j^2 based on the estimated acceptance rate. This works as follows: first, choose an initial value for the scaling factor, $c_{j,0}^2$, and estimate the corresponding acceptance rate $\bar{\mathcal{A}}_j(c_{j,0}^2)$ based on N_j generated Markov states, then modify $c_{j,0}^2$ to obtain an increase in $\bar{\mathcal{A}}_j$. Whether this optimization in c_j^2 is useful depends on whether the accuracy of the estimator that is achieved compensates for the additional computational cost. Finally, note that our simulation results show (see Section 4) that, as j increases, the corresponding optimal scaling factor c_j^2 decreases slightly. This observation coincides with intuition, since when j increases, the intermediate distributions $\pi_j(\cdot)$ become more concentrated.

In the following section we establish the ergodic properties of the Markov chains generated by AIMS.

3 Ergodic properties of AIMS

Since the discussion in Subsection 2.1, which motivated AIMS at annealing level j, involved delta functions and formal equalities (12) and (13), we cannot simply rely on the convergence of the IMH algorithm in verification of AIMS; a rigorous proof is needed. First we prove that the described algorithm indeed generates a Markov chain with a stationary distribution $\pi_j(\cdot)$. We also explain that when the proposal density $q_j(\cdot|\xi)$ is reasonably chosen, $\pi_j(\cdot)$ is the unique (and, therefore, limiting) stationary distribution of the corresponding Markov chain.

Theorem 1. *Let $\theta_j^{(1)}, \theta_j^{(2)}, \ldots$ be the Markov chain on $\Theta_j^* = \Theta \setminus \left\{ \theta_{j-1}^{(1)}, \ldots, \theta_{j-1}^{(N_{j-1})} \right\}$ generated by the AIMS algorithm at annealing level j, then $\pi_j(\cdot)$ is a stationary distribution of the Markov chain.*

Proof. Let $\mathcal{K}_j(\cdot|\cdot)$ denote the transition kernel of the Markov chain generated by AIMS at annealing level j. From the discription of the algorithm it follows that $\mathcal{K}_j(\cdot|\cdot)$ has the following form:

$$\mathcal{K}_j(d\xi|\theta) = \sum_{i=1}^{N_{j-1}} \bar{w}_{j-1}^{(i)} q_j(\xi|\theta_{j-1}^{(i)}) \min\left\{1, \frac{\pi_j(\xi)}{\pi_j(\theta_{j-1}^{(i)})}\right\} \min\left\{1, \frac{\pi_j(\xi)\hat{\pi}_j^{N_{j-1}}(\theta)}{\pi_j(\theta)\hat{\pi}_j^{N_{j-1}}(\xi)}\right\} d\xi \qquad (26)$$
$$+ (1 - \mathcal{A}_j(\theta))\delta_\theta(d\xi),$$

where $\mathcal{A}_j(\theta)$ is the probability of having a proper transition θ to $\Theta_j^* \setminus \{\theta\}$:

$$\mathcal{A}_j(\theta) = \int_{\Theta_j^*} \sum_{i=1}^{N_{j-1}} \bar{w}_{j-1}^{(i)} q_j(\xi|\theta_{j-1}^{(i)}) \min\left\{1, \frac{\pi_j(\xi)}{\pi_j(\theta_{j-1}^{(i)})}\right\} \min\left\{1, \frac{\pi_j(\xi)\hat{\pi}_j^{N_{j-1}}(\theta)}{\pi_j(\theta)\hat{\pi}_j^{N_{j-1}}(\xi)}\right\} d\xi \qquad (27)$$

A sufficient condition for $\pi_j(\cdot)$ to be a stationary distribution is for $\mathcal{K}_j(\cdot|\cdot)$ to satisfy the detailed balance condition:

$$\pi_j(d\theta)\mathcal{K}_j(d\xi|\theta) = \pi_j(d\xi)\mathcal{K}_j(d\theta|\xi) \qquad (28)$$

Without loss of generality, we assume that $\theta \neq \xi$, since otherwise (28) is trivial. In this case $\mathcal{K}_j(d\xi|\theta)$ is given by the first term in (26), since the second term vanishes. Thus, all we need to prove is that function

$$\mathcal{E}(\theta,\xi) \stackrel{\text{def}}{=} \pi_j(\theta) \sum_{i=1}^{N_{j-1}} \bar{w}_{j-1}^{(i)} q_j(\xi|\theta_{j-1}^{(i)}) \min\left\{1, \frac{\pi_j(\xi)}{\pi_j(\theta_{j-1}^{(i)})}\right\} \min\left\{1, \frac{\pi_j(\xi)\hat{\pi}_j^{N_{j-1}}(\theta)}{\pi_j(\theta)\hat{\pi}_j^{N_{j-1}}(\xi)}\right\} \qquad (29)$$

is symmetric with respect to permutation $\theta \leftrightarrow \xi$, for all $\theta, \xi \in \Theta_j^*$. Taking into account (12) and a simple fact that $a\min\{1, b/a\} = b\min\{1, a/b\}$ for all $a, b > 0$, we have:

$$\mathcal{E}(\theta,\xi) = \pi_j(\theta)\hat{\pi}_j^{N_{j-1}}(\xi) \min\left\{1, \frac{\pi_j(\xi)\hat{\pi}_j^{N_{j-1}}(\theta)}{\pi_j(\theta)\hat{\pi}_j^{N_{j-1}}(\xi)}\right\}$$
$$= \pi_j(\xi)\hat{\pi}_j^{N_{j-1}}(\theta) \min\left\{1, \frac{\pi_j(\theta)\hat{\pi}_j^{N_{j-1}}(\xi)}{\pi_j(\xi)\hat{\pi}_j^{N_{j-1}}(\theta)}\right\} = \mathcal{E}(\xi,\theta) \qquad (30)$$

This proves that $\pi_j(\cdot)$ is a stationary distribution of the AIMS Markov chain. $\qquad\square$

A stationary distribution is unique and is the limiting distribution for a Markov chain, if the chain is aperiodic and irreducible (see, for example, [Ti94]). In the case of AIMS, aperiodicity is guaranteed by the fact that the probability of having a repeated sample $\theta_j^{(i+1)} = \theta_j^{(i)}$ is not zero: for example, if the local candidate state ξ_l is rejected in step 1c, then we automatically have $\theta_j^{(i+1)} = \theta_j^{(i)}$. A Markov chain with stationary distribution $\pi(\cdot)$ is irreducible if, for any initial state, it has positive probability of entering any set to which $\pi(\cdot)$ assigns positive probability. It is clear that if the proposal distribution $q_j(\cdot|\xi)$ is "standard" (e.g. Gaussian, uniform, log-normal, etc), then AIMS generates an irreducible Markov chain. In this case, $\pi_j(\cdot)$ is therefore the unique stationary distribution of the AIMS Markov chain, and for every $\theta \in \Theta_j^*$

$$\lim_{n\to\infty} \|\mathcal{K}_j^n(\cdot|\theta) - \pi_j(\cdot)\|_{\text{TV}} = 0, \qquad (31)$$

14

with $\|\cdot\|_{\mathrm{TV}}$ denoting the total variation distance. Recall that the total variation distance between two measures $\mu_1(\cdot)$ and $\mu_2(\cdot)$ on Θ is defined as $\|\mu_1(\cdot) - \mu_1(\cdot)\|_{\mathrm{TV}} = \sup_{A \subset \Theta} |\mu_1(A) - \mu_2(A)|$. In a simulation setup, the most important consequence of convergence property (31) is, of course, that the sample mean converges to the expectation of a measurable function of interest almost surely:

$$\lim_{N_j \to \infty} \frac{1}{N_j} \sum_{i=1}^{N_j} h(\theta_j^{(i)}) = \int_\Theta h(\theta)\pi_j(\theta)d\theta \tag{32}$$

Convergence (31) ensures the proper behavior of the AIMS chain $\theta_j^{(1)}, \theta_j^{(2)}, \ldots$ regardless of the initial state $\theta_j^{(1)}$. A more detailed description of convergence properties involves the study of the speed of convergence of $\mathcal{K}_j^n(\cdot|\theta)$ to $\pi_j(\cdot)$. Evaluation (or estimation) of this speed is very important for any MCMC algorithm, since it relates to a stopping rule for this algorithm: the higher the speed of convergence $\mathcal{K}_j^n(\cdot|\theta) \to \pi_j(\cdot)$, the less samples are need to obtain an accurate estimate in (32). Recall, following [MT09], that a chain $\theta^{(1)}, \theta^{(2)}, \ldots$ is called *uniformly ergodic* if

$$\lim_{n \to \infty} \sup_{\theta \in \Theta} \|\mathcal{K}^n(\cdot|\theta) - \pi(\cdot)\|_{\mathrm{TV}} = 0 \tag{33}$$

The property of uniform ergodicity is stronger than (31), since it guarantees that the speed of convergence is uniform over the whole space. Moreover, a Markov chain is uniformly ergodic if and only if there exist $r > 1$ and $R < \infty$ such that for all $\theta \in \Theta$

$$\|\mathcal{K}^n(\cdot|\theta) - \pi(\cdot)\|_{\mathrm{TV}} \le Rr^{-n}, \tag{34}$$

that is, the convergence in (33) takes place at uniform geometric rate [MT09].

Theorem 2. *If there exists a constant M such that for all $\theta \in \Theta_j^*$*

$$\pi_j(\theta) \le M \hat{\pi}_j^{N_{j-1}}(\theta), \tag{35}$$

then the AIMS algorithm at annealing level j produces a uniformly ergodic chain and

$$\|\mathcal{K}_j^n(\cdot|\theta) - \pi_j(\cdot)\|_{\mathrm{TV}} \le \left(1 - \frac{1}{M}\right)^n \tag{36}$$

Proof. To prove the first part of the theorem we will need the notion of a *small set* [MT09]. A set $A \subset \Theta$ is called a small set if there exists an integer $m > 0$ and a non-trivial measure μ_m on Θ, such that for all $\theta \in A$, $B \subset \Theta$:

$$\mathcal{K}^m(B|\theta) \ge \mu_m(B) \tag{37}$$

In this case we say that A is μ_m-small. It can be shown [MT09] that a Markov chain is uniformly ergodic if and only if its state space is μ_m-small for some m. Thus, to prove the theorem, it is enough to show that Θ_j^* is a small set.

If (35) is satisfied, than the following holds for transition kernel (26) for $\theta \in \Theta_j^*$ and $B \subset \Theta_j^*$:

$$
\begin{aligned}
\mathcal{K}_j(B|\theta) &\geq \int_B \sum_{i=1}^{N_{j-1}} \bar{w}_{j-1}^{(i)} q_j(\xi|\theta_{j-1}^{(i)}) \min\left\{1, \frac{\pi_j(\xi)}{\pi_j(\theta_{j-1}^{(i)})}\right\} \min\left\{1, \frac{\pi_j(\xi)\hat{\pi}_j^{N_{j-1}}(\theta)}{\pi_j(\theta)\hat{\pi}_j^{N_{j-1}}(\xi)}\right\} d\xi \\
&= \int_B \hat{\pi}_j^{N_{j-1}}(\xi) \min\left\{1, \frac{\pi_j(\xi)\hat{\pi}_j^{N_{j-1}}(\theta)}{\pi_j(\theta)\hat{\pi}_j^{N_{j-1}}(\xi)}\right\} d\xi \\
&= \int_B \min\left\{\hat{\pi}_j^{N_{j-1}}(\xi), \pi_j(\xi)\frac{\hat{\pi}_j^{N_{j-1}}(\theta)}{\pi_j(\theta)}\right\} d\xi \\
&\geq \int_B \min\left\{\hat{\pi}_j^{N_{j-1}}(\xi), \frac{\pi_j(\xi)}{M}\right\} d\xi = \frac{1}{M}\int_B \pi_j(\xi)d\xi = \frac{1}{M}\pi_j(B)
\end{aligned}
\tag{38}
$$

The sample space Θ_j^* is therefore $\frac{\pi_j}{M}$-small, and the corresponding Markov chain is uniformly ergodic.

To prove bound (36), first observe, using (38), that

$$
\|\mathcal{K}_j(\cdot|\theta) - \pi_j(\cdot)\|_{\text{TV}} = \sup_A |\mathcal{K}_j(A|\theta) - \pi_j(A)| \leq \sup_A |\pi_j(A) - \frac{1}{M}\pi_j(A)| = 1 - \frac{1}{M}
\tag{39}
$$

For $n > 1$, using the Chapman-Kolmogorov equation $\mathcal{K}^{m+n}(A|\theta) = \int_\Theta \mathcal{K}^m(A|\xi)\mathcal{K}^n(d\xi|\theta)$ and stationarity of $\pi_j(\cdot)$ with respect to $\mathcal{K}_j(\cdot|\cdot)$, we have:

$$
\begin{aligned}
\|\mathcal{K}_j^n(\cdot|\theta) - \pi_j(\cdot)\|_{\text{TV}} &= \sup_A |\mathcal{K}_j^n(A|\theta) - \pi_j(A)| \\
&= \sup_A \left|\int_{\Theta_j^*} \mathcal{K}_j(A|\xi)\mathcal{K}_j^{n-1}(d\xi|\theta) - \int_{\Theta_j^*} \mathcal{K}_j(A|\xi)\pi_j(\xi)d\xi\right| \\
&= \sup_A \left|\int_{\Theta_j^*} \mathcal{K}_j(A|\xi)\left[\mathcal{K}_j^{n-1}(d\xi|\theta) - \pi_j(\xi)d\xi\right]\right| \\
&= \sup_A \left|\int_{\Theta_j^*} [\mathcal{K}_j(A|\xi) - \pi_j(A)]\left[\mathcal{K}_j^{n-1}(d\xi|\theta) - \pi_j(\xi)d\xi\right]\right|,
\end{aligned}
\tag{40}
$$

where the last equality follows from the fact that $\int_{\Theta_j^*} \mathcal{K}_j^{n-1}(d\xi|\theta) = \int_{\Theta_j^*} \pi_j(\xi)d\xi = 1$. Finally, we obtain:

$$
\begin{aligned}
\|\mathcal{K}_j^n(\cdot|\theta) - \pi_j(\cdot)\|_{\text{TV}} &\leq \sup_B \sup_A \left|\int_B [\mathcal{K}_j(A|\xi) - \pi_j(A)]\left[\mathcal{K}_j^{n-1}(d\xi|\theta) - \pi_j(\xi)d\xi\right]\right| \\
&\leq \sup_B \left|\int_B \sup_A |\mathcal{K}_j(A|\xi) - \pi_j(A)|\left[\mathcal{K}_j^{n-1}(d\xi|\theta) - \pi_j(\xi)d\xi\right]\right| \\
&= \|\mathcal{K}_j(\cdot|\theta) - \pi_j(\cdot)\|_{\text{TV}} \cdot \|\mathcal{K}_j^{n-1}(\cdot|\theta) - \pi_j(\cdot)\|_{\text{TV}} \leq \left(1 - \frac{1}{M}\right)^n
\end{aligned}
\tag{41}
$$

$\square$

Remark 7. Note that if there exists a constant M such that (35) holds for all $\theta \in \Theta_j^*$, then $M > 1$ automatically.

16

Corollary 1. *If $\Theta \subset \mathbb{R}^d$ is a compact set and $q_j(\cdot|\xi)$ is a Gaussian distribution centered at ξ, then the AIMS algorithm at annealing level j produces a uniformly ergodic chain and (36) holds with M given by*

$$M = \left(\sum_{i=1}^{N_{j-1}} \bar{w}_{j-1}^{(i)} \frac{\min_{\theta \in \Theta} q_j(\theta|\theta_{j-1}^{(i)})}{\max_{\theta \in \Theta} \pi_j(\theta)} \right)^{-1} \tag{42}$$

Proof. Let us show that in this case condition (35) is always fulfilled. For any $\theta \in \Theta_j^*$ we have:

$$\begin{aligned}
\hat{\pi}_j^{N_{j-1}}(\theta) &= \sum_{i=1}^{N_{j-1}} \bar{w}_{j-1}^{(i)} q_j(\theta|\theta_{j-1}^{(i)}) \min\left\{ 1, \frac{\pi_j(\theta)}{\pi_j(\theta_{j-1}^{(i)})} \right\} \\
&= \sum_{i=1}^{N_{j-1}} \bar{w}_{j-1}^{(i)} q_j(\theta|\theta_{j-1}^{(i)}) \frac{\pi_j(\theta)}{\pi_j(\theta_{j-1}^{(i)})} \min\left\{ 1, \frac{\pi_j(\theta_{j-1}^{(i)})}{\pi_j(\theta)} \right\} \\
&\geq \pi_j(\theta) \sum_{i=1}^{N_{j-1}} \bar{w}_{j-1}^{(i)} \frac{\min_{\theta \in \Theta} q_j(\theta|\theta_{j-1}^{(i)})}{\pi_j(\theta_{j-1}^{(i)})} \min\left\{ 1, \frac{\pi_j(\theta_{j-1}^{(i)})}{\max_{\theta \in \Theta} \pi_j(\theta)} \right\} \\
&= \pi_j(\theta) \sum_{i=1}^{N_{j-1}} \bar{w}_{j-1}^{(i)} \frac{\min_{\theta \in \Theta} q_j(\theta|\theta_{j-1}^{(i)})}{\max_{\theta \in \Theta} \pi_j(\theta)}
\end{aligned} \tag{43}$$

Thus, (35) holds with M given by (42). $\qquad\square$

Remark 8. Note than the assumption of compactness of the sample space Θ is not very restrictive and is typically satisfied in most Bayesian statistics problems. Indeed, to fulfill this condition, it is enough to take a prior distribution $\pi_0(\cdot)$ with compact support. Next, it is clear from the proof, that the conclusion of Corollary 1 holds for different "reasonable" (not only Gaussian) proposal distributions $q_j(\cdot|\xi)$. Therefore, the AIMS algorithm will produce a uniformly ergodic Markov chain in many practical cases.

It has been recognized for a long time that, when using an MCMC algorithm, it is useful to monitor its acceptance rate $\bar{\mathcal{A}}$, i.e. expected probability of having a proper Markov jump $\theta^{(i)}$ to $\theta^{(i+1)} \neq \theta^{(i)}$. While in the case of the RWMH algorithm, the finding of the optimal acceptance rate is a difficult problem: neither high nor low $\bar{\mathcal{A}}$ is good [GRG96]; for IMH the picture is rather simple: the higher $\bar{\mathcal{A}}$, the better [RC04]. Since AIMS is based on the IMH algorithm, their properties are very similar. In particular, one should aim for the highest possible acceptance rate of the global candidate state ξ_g when implementing AIMS.

We finish this section with a result that provides bounds for the acceptance rate of the AIMS algorithms. These bounds can be useful for finding the optimal implementation parameters.

Theorem 3. *Let $\bar{\mathcal{A}}_j$ be the expected probability of having a proper Markov transition associated with the AIMS algorithm at annealing level j. Then*

$$\bar{\mathcal{A}}_j \leq \sum_{i=1}^{N_{j-1}} \bar{w}_{j-1}^{(i)} a_j(\theta_{j-1}^{(i)}), \tag{44}$$

where $a_j(\theta_{j-1}^{(i)})$ is probability (11) associated with having a proper transition under the RWMH transition kernel (10). If (35) holds, then

$$\bar{\mathcal{A}}_j \geq \frac{1}{M} \tag{45}$$

Proof. For every $\theta \in \Theta_j^*$, the probability $\mathcal{A}_j(\theta)$ of transition θ to $\Theta_j^* \setminus \{\theta\}$ is given by (27). For its expected value we have:

$$
\begin{aligned}
\bar{\mathcal{A}}_j &= \int_{\Theta_j^*} \pi_j(\theta) \mathcal{A}_j(\theta) d\theta \\
&= \int_{\Theta_j^*} \int_{\Theta_j^*} \pi_j(\theta) \sum_{i=1}^{N_{j-1}} \bar{w}_{j-1}^{(i)} q_j(\xi|\theta_{j-1}^{(i)}) \min\left\{1, \frac{\pi_j(\xi)}{\pi_j(\theta_{j-1}^{(i)})}\right\} \min\left\{1, \frac{\pi_j(\xi)\hat{\pi}_j^{N_{j-1}}(\theta)}{\pi_j(\theta)\hat{\pi}_j^{N_{j-1}}(\xi)}\right\} d\xi d\theta \\
&\leq \int_{\Theta_j^*} \int_{\Theta_j^*} \pi_j(\theta) \sum_{i=1}^{N_{j-1}} \bar{w}_{j-1}^{(i)} q_j(\xi|\theta_{j-1}^{(i)}) \min\left\{1, \frac{\pi_j(\xi)}{\pi_j(\theta_{j-1}^{(i)})}\right\} d\xi d\theta \\
&= \int_{\Theta_j^*} \pi_j(\theta) \sum_{i=1}^{N_{j-1}} \bar{w}_{j-1}^{(i)} a_j(\theta_{j-1}^{(i)}) d\theta = \sum_{i=1}^{N_{j-1}} \bar{w}_{j-1}^{(i)} a_j(\theta_{j-1}^{(i)})
\end{aligned}
\tag{46}
$$

To prove the lower bound (45), we use (12) in the equation defining $\bar{\mathcal{A}}_j$:

$$
\begin{aligned}
\bar{\mathcal{A}}_j &= \int_{\Theta_j^*} \int_{\Theta_j^*} \pi_j(\theta)\hat{\pi}_j^{N_{j-1}}(\xi) \min\left\{1, \frac{\pi_j(\xi)\hat{\pi}_j^{N_{j-1}}(\theta)}{\pi_j(\theta)\hat{\pi}_j^{N_{j-1}}(\xi)}\right\} d\xi d\theta \\
&= \int_{\Theta_j^*} \int_{\Theta_j^*} \pi_j(\theta)\hat{\pi}_j^{N_{j-1}}(\xi) I\left(\frac{\pi_j(\xi)\hat{\pi}_j^{N_{j-1}}(\theta)}{\pi_j(\theta)\hat{\pi}_j^{N_{j-1}}(\xi)} \geq 1\right) d\xi d\theta \\
&\quad + \int_{\Theta_j^*} \int_{\Theta_j^*} \pi_j(\theta)\hat{\pi}_j^{N_{j-1}}(\xi) I\left(\frac{\pi_j(\theta)\hat{\pi}_j^{N_{j-1}}(\xi)}{\pi_j(\xi)\hat{\pi}_j^{N_{j-1}}(\theta)} \geq 1\right) \frac{\pi_j(\xi)\hat{\pi}_j^{N_{j-1}}(\theta)}{\pi_j(\theta)\hat{\pi}_j^{N_{j-1}}(\xi)} d\xi d\theta \\
&= 2\int_{\Theta_j^*} \int_{\Theta_j^*} \pi_j(\theta)\hat{\pi}_j^{N_{j-1}}(\xi) I\left(\frac{\pi_j(\xi)\hat{\pi}_j^{N_{j-1}}(\theta)}{\pi_j(\theta)\hat{\pi}_j^{N_{j-1}}(\xi)} \geq 1\right) d\xi d\theta \\
&\geq 2\int_{\Theta_j^*} \int_{\Theta_j^*} \pi_j(\theta)\frac{\pi_j(\xi)}{M} I\left(\frac{\pi_j(\xi)}{\hat{\pi}_j^{N_{j-1}}(\xi)} \geq \frac{\pi_j(\theta)}{\hat{\pi}_j^{N_{j-1}}(\theta)}\right) d\xi d\theta \\
&= \frac{2}{M} P\left(\frac{\pi_j(\xi)}{\hat{\pi}_j^{N_{j-1}}(\xi)} \geq \frac{\pi_j(\theta)}{\hat{\pi}_j^{N_{j-1}}(\theta)}\right) = \frac{1}{M},
\end{aligned}
\tag{47}
$$

where the last probability is equal to $1/2$, because θ and ξ are i.i.d. according to $\pi_j(\cdot)$, and hence the result. $\square$

Remark 9. The AIMS algorithm at annealing level j has two accept/reject steps: one is for the local candidate ξ_l (step 1c) and another is for the global candidate ξ_g (step 2). The right-hand side of (44) is nothing else but the local acceptance rate, i.e. expected probability of generating a proper local candidate state $\xi_l \notin \{\theta_{j-1}^{(1)}, \ldots, \theta_{j-1}^{(N_{j-1})}\}$. Basically, (44) says that the global acceptance rate $\bar{\mathcal{A}}_j$ can never exceed the local acceptance rate.

In fact, it can be deduced directly from the description of the algorithm, since if the local candidate ξ_l is rejected, then the global candidate ξ_g is automatically rejected and we have a repeated sample $\theta_j^{(i+1)} = \theta_j^{(i)}$.

4   Illustrative Examples

In this section we illustrate the use of AIMS with three examples: 1) mixture of ten Gaussian distributions in two dimensions (a multi-modal case); 2) sum of two multivariate Gaussian distributions in higher dimensions; and 3) Bayesian updating of a neural network model.

4.1   Multi-modal mixture of Gaussians in 2D

To demonstrate the efficiency of AIMS for sampling from multi-modal distributions, consider simulation from a truncated two-dimensional mixture of M Gaussian densities:

$$\pi(\theta) \propto \pi_0(\theta) \cdot L(\theta) = \mathcal{U}_{[0,a] \times [0,a]}(\theta) \cdot \sum_{i=1}^{M} w_i \mathcal{N}(\theta | \mu_i, \sigma^2 \mathbb{I}_2), \tag{48}$$

where $\mathcal{U}_{[0,a] \times [0,a]}(\cdot)$ denotes the uniform distribution on the square $[0,a] \times [0,a]$. In this example, $a = 10$, $M = 10$, $\sigma = 0.1$, $w_1 = \ldots = w_{10} = 0.1$, and the mean vectors $\mu_1, \ldots, \mu_{10}$ are drawn uniformly from the square $[0,10] \times [0,10]$. Because of our interest in Bayesian updating, we refer to $\pi(\cdot)$ in (48) as a posterior distribution.

Figure 3(a) displays the scatterplot of 10^3 posterior samples obtained from AIMS. Notice there are two clusters of samples that overlap significantly near $\theta = (4, 4)$ that reflect two closely spaced Gaussian densities but the other 8 clusters are widely spaced. The parameters of the algorithm were chosen as follows: sample size $N = 10^3$ per annealing level; the threshold for the ESS $\gamma = 1/2$; the local proposal density $q_j(\cdot | \xi) = \mathcal{N}(\cdot | \xi, c^2 \mathbb{I}_2)$, with $c = 0.2$. The trajectory of the corresponding posterior Markov chain, i.e. the chain generated at the last annealing level with stationary distribution $\pi(\cdot)$, is shown in Figure 3(b). Black crosses $\times$ represent the mean vectors $\mu_1, \ldots, \mu_{10}$. As expected, the chain does not exhibit a local random walk behavior and it moves freely between well-separated modes of the posterior distribution.

The described implementation of AIMS leads to a total number of $m = 6$ intermediate distributions in the annealing scheme. Figure 4 shows how annealing parameter β_j changes as a function of j for 50 independent runs of the algorithm. It is found that in all considered examples, β_j grows exponentially with j.

Let us now compare the performance of AIMS with the Random Walk Metropolis-Hastings algorithm. For a fair comparison, the Metropolis-Hastings algorithm was implemented as follows. First, a sample of $N_0 = 10^3$ points $\theta_0^{(1)}, \ldots, \theta_0^{(N_0)}$ was drawn from the prior distribution $\pi_0(\cdot) = \mathcal{U}_{[0,a] \times [0,a]}(\cdot)$ and the corresponding values of the likelihood function $L(\theta) = \sum_{i=1}^{M} w_i \mathcal{N}(\theta | \mu_i, \sigma^2 \mathbb{I}_2)$ were calculated, $L_i = L(\theta_0^{(i)})$. Then, starting from the point with the largest likelihood, $\theta^{(1)} = \theta_0^{(k)}$, $k = \arg \max L_i$, a Markov chain $\theta^{(1)}, \ldots, \theta^{(N)}$, with stationary distribution $\pi(\cdot)$ was generated using the Metropolis-Hastings algorithm. The proposal distribution used was $q(\cdot | \xi) = N(\cdot | \xi, c^2 \mathbb{I}_2)$ with $c = 0.2$, and the length of the chain was $N = 5 \cdot 10^3$. Thus, the total number of samples used in both AIMS and RWMH was $N_t = 6 \cdot 10^3$. The scatterplot of posterior samples obtained from RWMH

and the trajectory of the corresponding Markov chain are show in Figures 3(c) and 3(d), respectively. While the AIMS algorithm successfully sampled all 10 modes with the approximately correct proportion of total samples, RWHM completely missed 7 modes.

Suppose that we are interested in estimating the posterior mean vector, $\mu^\pi = (\mu_1^\pi, \mu_2^\pi)$, and the components $(\sigma_1^\pi)^2, (\sigma_2^\pi)^2, \sigma_{12}^\pi$ of the posterior covariance matrix Σ^π. Their true values are given in Table 1 along with the AIMS estimates in terms of their means and coefficients of variation averaged over 50 independent simulations, all based on 10^3 posterior samples.

Figure 5 displays the mean square error (MSE) of the AIMS estimator for the posterior mean and covariance matrix for different values of the scaling factor c. The MSE was estimated based on 50 independent runs of the algorithm. An interesting observation is that the MSE as a function of c is nearly flat around the optimal, $c_{\text{opt}} \approx 0.15$, i.e. the one that minimizes the MSE.

4.2 Mixture of two higher-dimensional Gaussians

To demonstrate the efficiency of AIMS for higher dimensionality, consider simulation from a truncated sum of two multivariate Gaussian densities:

$$\pi^d(\theta) \propto \pi_0^d(\theta) \cdot L^d(\theta) = \mathcal{U}_{[-a,a]^d}(\theta) \cdot \left(\mathcal{N}(\theta|\mu_1, \sigma^2 \mathbb{I}_d) + \mathcal{N}(\theta|\mu_2, \sigma^2 \mathbb{I}_d) \right), \qquad (49)$$

where $a = 2$, $\mu_1 = (0.5, \ldots, 0.5)$, $\mu_2 = (-0.5, \ldots, -0.5)$, and $\sigma = 0.5$. Thus, $\pi^d(\cdot)$ is a bimodal distribution on a d-dimensional cube $[-a, a]^d$. Suppose that a quantity of interest is the function $h : [-a, a]^d \to [-a, a]$ that gives the largest component of $\theta = (\theta_1, \ldots, \theta_d) \in [-a, a]^d$:

$$h(\theta) = \max\{\theta_1, \ldots, \theta_d\} \qquad (50)$$

and we want to estimate its expectation with respect to $\pi^d(\cdot)$ using posterior samples $\theta^{(1)}, \ldots, \theta^{(N)} \sim \pi^d(\cdot)$ as follows:

$$\bar{h} = \mathbb{E}_{\pi^d}[h] \approx \hat{h}_N = \frac{1}{N} \sum_{i=1}^{N} h(\theta^{(i)}) \qquad (51)$$

This example is taken from [CC07], where the Transitional Markov chain Monte Carlo method (TMCMC) for sampling from posterior densities was introduced.

Here, we consider five cases: $d = 2, 4, 6, 10$, and 20. The performance of TMCMC was examined for only the first three cases in [CC07]. The last two cases are higher dimensional, and, therefore, more challenging.

The details of implementation and simulation results from 50 independent runs are summarized in Table 2. First of all, observe that AIMS outperforms TMCMC, when $d = 2, 4, 6$. Both methods are capable of generating samples from both modes of the posterior; however, the probabilities of the modes (each is 1/2 in this example) are found more accurately by AIMS.

Remark 10. In addition to the first three cases, five other scenarios with different probabilities of modes and different values of σ were examined in [CC07]. It is found that AIMS outperforms TMCMC in all these cases too.

Results presented in Table 2 help to shed some light on the properties of the optimal scaling parameter c_{opt} for the proposal density $q_j(\cdot|\xi) = \mathcal{N}(\cdot|\xi, c^2 \mathbb{I}_d)$. It appears c_{opt} depends not only on the dimension d, which is expected, but also on N, the number of samples used per each annealing level. The latter dependence is explained by the fact that the global proposal distribution $\hat{\pi}_j^N(\cdot)$ for the AIMS Markov chain depends both on N and c: $\hat{\pi}_j^N(\cdot)$ is a weighted sum of N RWMH transition kernels with Gaussian proposal distributions, whose spread is controlled by c. When N is fixed, c_{opt} is a monotonically increasing function of d, since in higher dimensions, for optimal local exploration of the neighborhoods of $\theta_{j-1}^{(1)}, \ldots, \theta_{j-1}^{(N)}$, we have to be able to make larger local jumps from $\theta_{j-1}^{(k)}$ to ξ_l. When d is fixed, c_{opt} is a monotonically decreasing function of N, since the more samples $\theta_{j-1}^{(1)}, \ldots, \theta_{j-1}^{(N)}$ that have been generated at the previous level, the more we can focus on local exploration of their neighborhoods without worrying too much about regions that lie far away. If we think of the support of $q_j(\cdot|\theta_{j-1}^{(k)}) = \mathcal{N}(\cdot|\theta_{j-1}^{(k)}, c^2 \mathbb{I}_d)$ as lying mostly in a d-dimensional ball of radius c centered at $\theta_{j-1}^{(k)}$, then we can explain the dependence of c_{opt} on N as follows: the more d-dimensional balls of radius c we have, the smaller c we can use for covering the sample space.

It is interesting to look at how the local and global acceptance rates (see Remark 9) depend on the scaling parameter c. Figures 6, 7, and 8 display these acceptance rates along with the coefficient of variation δ of the AIMS estimator for the first three cases: $d = 2, 4$ and 6, based on 50 independent runs. As expected, the global acceptance rate is always smaller than the local acceptance rate, and the minimum value of δ corresponds to the maximum value of the global acceptance rate. Observe also that the peak of the global acceptance rate slides to the left, when j increases. This suggests that it is more efficient to use smaller values of c at higher annealing levels. Indeed, it is natural to expect that $c_j^{\text{opt}} > c_{j+1}^{\text{opt}}$, since the intermediate distribution $\pi_{j+1}(\cdot)$ is more concentrated than $\pi_j(\cdot)$.

Finally, we draw attention to Case 4 in Table 2 where $d = 10$ with $N = 10^3$ and $N = 2 \cdot 10^3$ samples per annealing level. Usually for Monte Carlo based methods, the coefficient of variation δ of the estimator is proportional to $1/\sqrt{N_t}$, where N_t is the total number of samples. Thus, the doubling of sample size will result in the reduction of δ by the factor of $1/\sqrt{2} \approx 0.71$. For AIMS, however, the decrease of δ is more significant: from $\delta = 26.7\%$ to $\delta = 12.2\%$, i.e. approximately by the factor of 0.46. This is because, as explained in Subsection 2.3, the increase of N affects not only the total sample size, but also improves the global proposal distribution $\hat{\pi}_j^N(\cdot)$. This improvement of $\hat{\pi}_j^N(\cdot)$ results in the generation of less correlated samples at each annealing level and, therefore, leads to an additional reduction of the coefficient of variation δ.

4.3 Bayesian updating of a neural network

To illustrate the use of AIMS for Bayesian updating, consider its application to a feed-forward neural network model, one of the most popular and most widely used models for function approximation. The goal is to approximate a (potentially highly nonlinear) function $f : X \to \mathbb{R}$, where $X \subset \mathbb{R}^p$ is a compact set, based on a finite number of measurements $y_i = f(x_i)$, $i = 1, \ldots, n$, by using a finite sum of the form

$$\hat{f}(x, \theta) = \sum_{j=1}^{M} \alpha_j \Psi(\langle x, \beta_j \rangle + \gamma_j) \tag{52}$$

where θ denotes the model parameters $\alpha_j, \gamma_j \in \mathbb{R}$ and $\beta_j \in \mathbb{R}^p$, $\langle \cdot, \cdot \rangle$ is the standard scalar product in $\mathbb{R}^p$, and Ψ is a sigmoidal function, the typical choice being either the logistic function or the tanh function that is used in this example:

$$\Psi(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}. \tag{53}$$

Model (52) is called a feed-forward neural network (FFNN) with activation function (53), p input units, one hidden layer with M hidden units, and one output unit. The parameters β_j and α_j are called the connection weights from the input units to the hidden unit j and the connection weights from the hidden unit j to the output unit, respectively. The term γ_j is a designated bias of the hidden unit j and it can be viewed as a connection weight from an additional constant unit input. Schematically, the FFNN model is shown in Figure 9.

The rationale behind the FFNN approximation method follows from the universal approximation property of FFNN models [Cy89, HSW89]; that is, a FFNN with sufficient number of hidden units and properly adjusted connection weights can approximate most functions arbitrarily well. More precisely, finite sums (52) over all positive integers M are dense in the set of real continuous functions on the p-dimensional unit cube.

Let $\mathcal{A}$ denote the FFNN architecture, i.e. the input-output model (52) together with information about the type of activation function Ψ, number of input units p, and number of hidden units M. In this example, we use $p = 1$, $M = 2$, and Ψ is given by (53), so the model parameters $\theta = (\alpha_1, \alpha_2, \beta_1, \beta_2, \gamma_1, \gamma_2) \in \Theta = \mathbb{R}^6$.

Deterministic model $\mathcal{A}$ of function f given by $\hat{f}(x, \theta)$ in (52) can be used to construct a Bayesian (stochastic) model $\mathcal{M}$ of function f by stochastic embedding (see the details in [Be08, Be10]). Recall, that by definition, a Bayesian model $\mathcal{M}$ consists of two components:

1. An input-output probability model $y \sim p(y|x, \theta, \mathcal{M})$, which is obtained by introducing the prediction-error

$$\varepsilon = y - \hat{f}(x, \theta), \tag{54}$$

which is the difference between the true output $y = f(x)$ and the deterministic model output $\hat{f}(x, \theta)$. A probability model for ε is introduced by using the Principle of Maximum Entropy [Ja57, Ja03], which states that the probability model should be selected to produce the most uncertainty subject to constraints that we wish to impose (the selection of any other probability model would lead to an unjustified reduction in the prediction uncertainty). In this example, we impose the following constraints: $\mathbb{E}[\varepsilon] = 0$ and $\text{var}[\varepsilon] = \sigma^2$ with ε unbounded. The maximum entropy PDF for the prediction-error is then $\varepsilon \sim \mathcal{N}(0, \sigma^2)$. This leads to the following input-output probability model:

$$p(y|x, \theta, \mathcal{M}) = \mathcal{N}\left(y \mid \hat{f}(x, \theta), \sigma^2\right) \tag{55}$$

Here, the prediction-error variance σ^2 is included in the set of model parameters where, for convenience, we define $\theta_7 = \log \sigma^{-2}$, so the parameter space is now $\Theta = \mathbb{R}^7$.

2. A prior PDF $\pi_0(\theta|\mathcal{M})$ over the parameter space which is chosen to quantify the initial relative plausibility of each value of θ in Θ. In this example, the prior distributions are assumed to be:

$$\alpha_j \sim \mathcal{N}(0, \sigma_\alpha^2), \quad \beta_j \sim \mathcal{N}(0, \sigma_\beta^2), \quad \gamma_j \sim \mathcal{N}(0, \sigma_\gamma^2), \quad \theta_7 = \log \sigma^{-2} \sim \mathcal{N}(0, \sigma_{\theta_7}^2), \tag{56}$$

with $\sigma_\alpha = \sigma_\beta = \sigma_\gamma = \sigma_{\theta_7} = 5$. Thus, the prior PDF in our case is

$$\pi_0(\theta|\mathcal{M}) = \mathcal{N}(\theta_7|0, \sigma_{\theta_7}^2) \prod_{j=1}^{M} \mathcal{N}(\alpha_j|0, \sigma_\alpha^2) \mathcal{N}(\beta_j|0, \sigma_\beta^2) \mathcal{N}(\gamma_j|0, \sigma_\gamma^2). \tag{57}$$

Let $\mathcal{D}$ denote the training data, $\mathcal{D} = \{(x_1, y_1), \ldots, (x_n, y_n)\}$, treated as independent samples, then the likelihood function which expresses the probability of getting data $\mathcal{D}$ based on the probability model (55) is given by

$$L(\theta) = p(\mathcal{D}|\theta, \mathcal{M}) = \prod_{i=1}^{n} p(y_i|x_i, \theta, \mathcal{M}) \tag{58}$$

In this example, data are synthetically generated from (55) with $\alpha_1 = 5$, $\alpha_2 = -5$, $\beta_1 = -1$, $\beta_2 = -3$, $\gamma_1 = 5$, $\gamma_2 = 2$, $\sigma = 0.1$, and the input $x_i = i/10$, for $i = 1, \ldots, n = 100$.

Finally, using Bayes' theorem, we can write the posterior PDF $\pi(\theta|\mathcal{D}, \mathcal{M})$ for the uncertain model parameters:

$$\pi(\theta|\mathcal{D}, \mathcal{M}) \propto \pi_0(\theta|\mathcal{M}) \cdot L(\theta)$$
$$= \mathcal{N}(\theta_7|0, \sigma_{\theta_7}^2) \prod_{j=1}^{M} \mathcal{N}(\alpha_j|0, \sigma_\alpha^2) \mathcal{N}(\beta_j|0, \sigma_\beta^2) \mathcal{N}(\gamma_j|0, \sigma_\gamma^2) \cdot \prod_{i=1}^{n} p(y_i|x_i, \theta, \mathcal{M}) \tag{59}$$

Under the Bayesian framework, the mean prediction of $y = f(x)$ from observable x can be obtained by integrating out the nuisance parameters:

$$\mathbb{E}_\pi[y|x, \mathcal{D}, \mathcal{M}] = \int_\Theta \hat{f}(x, \theta) \pi(\theta|\mathcal{D}, \mathcal{M}) d\theta \tag{60}$$

To demonstrate the efficiency of AIMS for the mean prediction problem, we use it to sample from the posterior PDF (59) and use Monte Carlo simulation in (60). The parameters of the AIMS algorithm are chosen as follows: sample size $N = 3 \times 10^3$ per annealing level; the threshold for the ESS $\gamma = 1/2$; the proposal density $q_j(\cdot|\xi) = \mathcal{N}(\cdot|\xi, c^2 \mathbb{I}_7)$, with $c = 0.5$. This implementation of AIMS leads to a total number of $m = 10$ intermediate distributions in the annealing scheme. The obtained posterior samples $\theta_m^{(1)}, \ldots, \theta_m^{(1)}$ are then used to approximate the integral on the right-hand side of (60):

$$\int_\Theta \hat{f}(x, \theta) \pi(\theta|\mathcal{D}, \mathcal{M}) d\theta \approx \frac{1}{N} \sum_{i=1}^{N} \hat{f}(x, \theta_m^{(i)}) \stackrel{\text{def}}{=} \bar{\hat{f}}_m(x) \tag{61}$$

The true function $y = f(x)$ as well as its AIMS approximation $\bar{\hat{f}}_m(x)$ are shown in Figure 10. A few "intermediate approximations" $\bar{\hat{f}}_j(x)$, which are based on $\theta_j^{(1)}, \ldots, \theta_j^{(1)} \sim \pi_j$, are plotted to show how $\bar{\hat{f}}_j(x)$ approaches $f(x)$ when $j \to m$. To visualize the uncertainty for the AIMS approximation, we plot its 5th and 95th percentiles in Figure 11.

5 Concluding Remarks

In this paper, a new scheme for sampling from posterior distributions, called Asymptotically Independent Markov Sampling (AIMS), is introduced. The algorithm is based on

three well-established and widely-used stochastic simulation methods: importance sampling, MCMC, and simulated annealing. The key idea behind AIMS is to use N samples drawn from $\pi_{j-1}(\cdot)$ as an importance sampling density to construct an approximation $\hat{\pi}_j^N(\cdot)$ of $\pi_j(\cdot)$, where $\pi_0(\cdot), \ldots, \pi_m(\cdot)$ is a sequence of intermediate distributions interpolating between the prior $\pi_0(\cdot)$ and posterior $\pi(\cdot) = \pi_m(\cdot)$. This approximation is then employed as the independent proposal distribution for sampling from $\pi_j(\cdot)$ by the independent Metropolis-Hastings algorithm. When $N \to \infty$, the AIMS sampler generates independent draws from the target distribution, hence the name of the algorithm.

Important ergodic properties of AIMS are derived. In particular, it is shown that, under certain conditions (that are often fulfilled in practice), the AIMS algorithm produces a uniformly ergodic Markov chain. The choice of the free parameters of the algorithm is discussed and recommendations are provide for their values, both theoretically and heuristically based. The efficiency of AIMS is demonstrated with three examples, which include both multi-modal and higher-dimensional target posterior distributions.

Acknowledgements

References

[AB01] Au S. K. and Beck J. L. (2001) "Estimation of small failure probabilities in high dimensions by subset simulation", *Probabilistic Engineering Mechanics*, vol. 16, No. 4, pp. 263–277.

[AB03] Au S. K. and Beck J. L. (2003) "Importance sampling in high dimensions", *Structural Safety*, vol. 25, No. 2, pp 139-163.

[Be08] Beck J. L. (2008) "Probability Logic, Information Quantification and Robust Predictive System Analysis", Technical Report EERL 2008-05, Earthquake Engineering Research Laboratory, California Institute of Technology, Pasadena, California.

[Be10] Beck J. L. (2010) "Bayesian system identification based on probability logic", *Structural Control and Health Monitoring*, vol. 17, pp. 825-847.

[BA02] Beck J. L. and Au S. K. (2002) "Bayesian updating of structural models and reliability using Markov chain Monte Carlo simulation", *Journal of Engineering Mechanics*, vol. 128, No. 4, pp. 380–391.

[CB10] Cheung S. H. and Beck J. L. (2010) "Calculation of posterior probabilities for Bayesian model class assessment and averaging from posterior samples based on dynamic system data", *Journal of Computer-aided Civil and Infrastructure Engineering*, vol. 25, No. 5, pp. 304-321.

[Če85] Černý V. (1985) "A thermodynamical approach to the travelling salesman problem: an efficient simulation algorithm" *Journal of Optimization Theory and Applications*, vol. 45, No. 1, pp. 41-51.

[CC07] Ching J. and Chen Y-C (2007) "Transitional Markov chain Monte Carlo method for Bayesian model updating, model class selection, and model averaging", *Journal of Engineering Mechanics*, vol. 133, No. 7, pp. 816-832.

[CC96] Cowles M. K. and Carlin B. P. (1996) "Markov chain Monte Carlo convergence diagnostics: a comparative review", *Journal of the American Statistical Association*, Vol. 91, No. 434, pp. 883-904.

[Cy89] Cybenko G. (1989) "Approximations by superpositions of a sigmoidal functions", *Mathematics of Control, Signals and Systems*, vol. 2, pp. 303-314.

[Ge89] Geweke J. (1989) "Bayesian inference in econometric models using Monte Carlo integration", *Econometrica*, vol. 57, No. 6, pp. 1317-1339.

[GG84] Geman S. and Geman D. (1984) "Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, No. 6, pp. 721-741.

[GRG96] Gelman A., Roberts G.O., and Gilks W.R. (1996) "Efficient Metropolis Jumping Rules", *Bayesian Statistics*, vol. 5, pp. 599-607.

[GRS96] Gilks W. R., Richardson S., and Spiegelhalter, D. J. (1996) *Markov Chain Monte Carlo in Practice*, Chapman and Hall, London.

[GT95] Geyer C. J. and Thompson E. A. (1995) "Annealing Markov chain Monte Carlo with applications to ancestral inference", *Journal of the American Statistical Association*, vol. 90, No. 431, pp. 909-920.

[Ha70] Hastings W. K. (1970) "Monte Carlo sampling methods using Markov chains and their applications", *Biometrika*, vol. 57, No. 1, pp. 97–109.

[HSW89] Hornik K., Stinchcombe M., and White H. (1989) "Multilayer feedforward networks are universal approximators", *Neural Networks*, vol. 2, pp. 359-366.

[Ja57] Jaynes E. T. (1957) "Information theory and statistical mechanics", *Physical Review*, vol. 106, pp. 620-630.

[Ja03] Jaynes E. T. (2003) *Probabiity Theory: The Logic of Science*, (Ed. G.L. Bretthorst), Cambridge University Press.

[KGV83] Kirkpatrick S., Gelatt C. D., and Vecchi M. P. (1983) "Optimization by simulated annealing", *Science*, vol. 220, No. 4598, pp. 671–680.

[KLW94] Kong A., Liu J. S., and Wong W. H. (1994) "Sequential imputations and Bayesian missing data problems", *Journal of the American Statistical Association*, vol. 89, No. 425, pp. 278–288.

[KM53] Kahn H. and Marshall A. W. (1953) "Methods of reducing sample size in Monte Carlo computations", *Journal of the Operations Research Society of America*, vol. 1, No. 5, pp. 263-278.

[Li96] Liu J. S. (1996) "Metropolized independent sampling with comparison to rejection sampling and importance sampling" *Statistics and Computing*, vol. 6, No. 2, pp. 113–119.

[Li01] Liu J. S. (2001) *Monte Carlo Strategies in Scientific Computing*, Springer Series in Statistics.

[MP92] Marinari E. and Parisi G. (1992) "Simulated tempering: A new Monte Carlo scheme", Europhysics Letters, vol. 19, No. 6, pp. 451-458.

[MR^{2}T^{2}53] Metropolis N., Rosenbluth A. W., Rosenbluth M. N., Teller A. H., and Teller E. (1953), "Equation of state calculations by fast computing machines", *J. Chemical Physics*, vol. 21, No. 6, pp. 1087–1092.

[MT09] Meyn S. and Tweedie R. L. (2009) *Markov chains and Stochastic Stability*, Cambridge University Press.

[MU49] Metropolis, N. and Ulam, S. (1949) "The Monte Carlo method", *Journal of the American Statistical Association*, vol. 44, pp. 335-341.

[Ne93] Neal R. M. (1993) "Probabilistic Inference Using Markov Chain Monte Carlo Methods", Technical Report CRG-TR-93-1, Dept. of Computer Science, University of Toronto.

[Ne96] Neal R. M. (1996) "Sampling from multimodal distributions using tempered transitions", *Statistics and Computing*, vol. 6, pp. 353-366.

[Ne01] Neal R. M. (2001) "Annealed importance sampling", *Statistics and Computing*, vol. 11, pp. 125-139.

[RC04] Robert C. P. and Casella G. (2004) *Monte Carlo Statistical Methods*, 2nd ed. Springer Texts in Statistics.

[Ru81] Rubinstein R. (1981) *Simulation and the Monte Carlo Method*, John Wiley, New York.

[Ti94] Tierney L. (1994) "Markov chains for exploring posterior distributions", *The Annals of Statistics*, vol. 22, No. 4, pp. 1701-1762.

Parameter	μ_1^π	μ_2^π	$(\sigma_1^\pi)^2$	$(\sigma_2^\pi)^2$	σ_{12}^π
True value	5.23	5.75	4.51	3.37	-1.30
AIMS mean	5.20	5.73	4.56	3.32	-1.25
AIMS cov	2.4%	2.0%	8.2%	8.2%	27.7%

Table 1: True values of the posterior parameters and the AIMS estimates in terms of their means and coefficients of variation averaged over 50 simulations [Example 4.1].

Case	d	$\bar{h}$	TMCMC: $\hat{h}_N, (\delta)$	AIMS: $\hat{h}_N, (\delta)$	N	γ	c_{opt}	$\bar{m}$
1	2	0.29	0.28 (12.3%)	0.29 (8.8%)	10^3	1/2	0.2	3
2	4	0.51	0.54 (10.0%)	0.51 (6.9%)	10^3	1/2	0.4	4
3	6	0.64	0.65 (15.7%)	0.64 (10.4%)	10^3	1/2	0.6	4.95
4	10	0.76	—	0.76 (26.7%)	10^3	1/2	0.7	5.84
	10	0.76	—	0.76 (12.2%)	$2 \cdot 10^3$	1/2	0.6	5.98
5	20	0.92	—	0.95 (42.1%)	$4 \cdot 10^3$	1/2	0.5	5.58

Table 2: Summary of the simulation results: d is the dimension of the sample space; $\bar{h}$ and $\hat{h}_N$ are the exact value of $\mathbb{E}_{\pi^d}[h]$ and its estimated value, respectively; δ in parentheses is the corresponding coefficient of variation; N, γ, c_{opt}, and $\bar{m}$ are the number of samples used per annealing level, the threshold for the ESS, the (nearly) optimal value of the scaling parameter, and the average number of distributions in the annealing scheme, respectively. The AIMS results are based on 50 independent runs. The TMCMC results are taken from [CC07] and are based on 50 independent runs [Example 4.2].
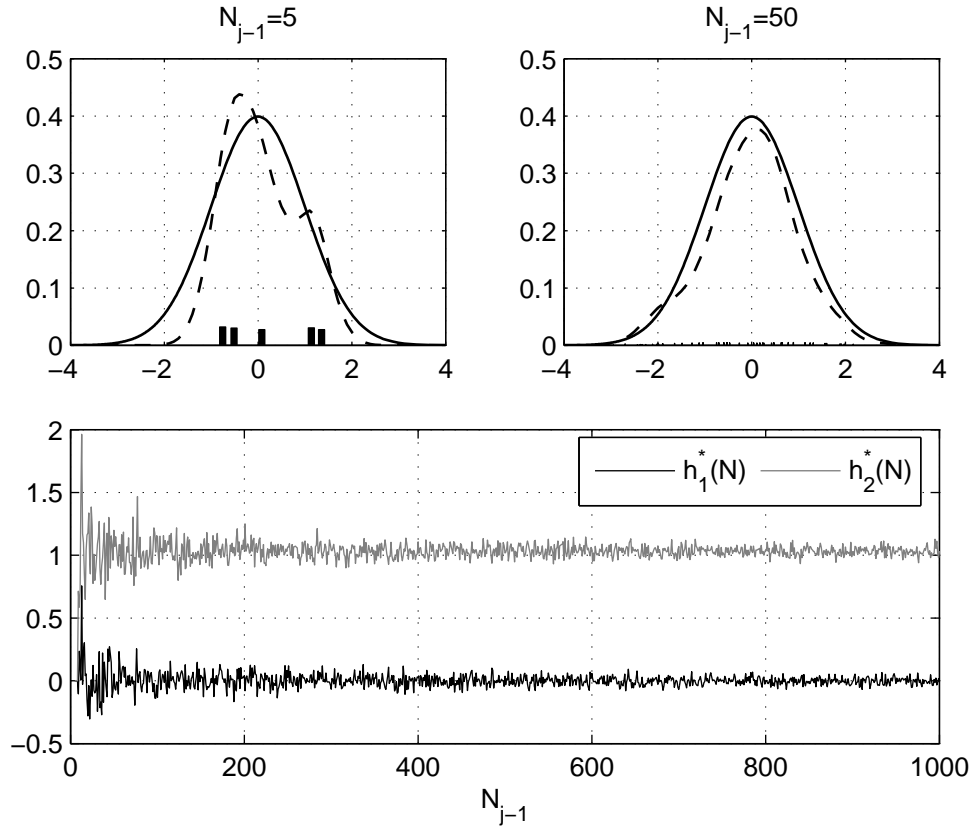
Figure 1: The top panels show the distribution $\pi_j(\cdot)$ (solid lines) and its approximation $\hat{\pi}_j^{N_{j-1}}(\cdot)$, for $N_{j-1} = 5$ (left) and $N_{j-1} = 50$ (right). Dashed lines and bars correspond to the continuous and discrete parts of $\hat{\pi}_j^{N_{j-1}}(\cdot)$, respectively. The bottom panel shows the convergence of $h_1^*(N_{j-1}) = \mathbb{E}_{\hat{\pi}_j^{N_{j-1}}}[h_1]$ and $h_2^*(N_{j-1}) = \mathbb{E}_{\hat{\pi}_j^{N_{j-1}}}[h_2]$ to the true values, 0 and 1, respectively [Example 2.1].
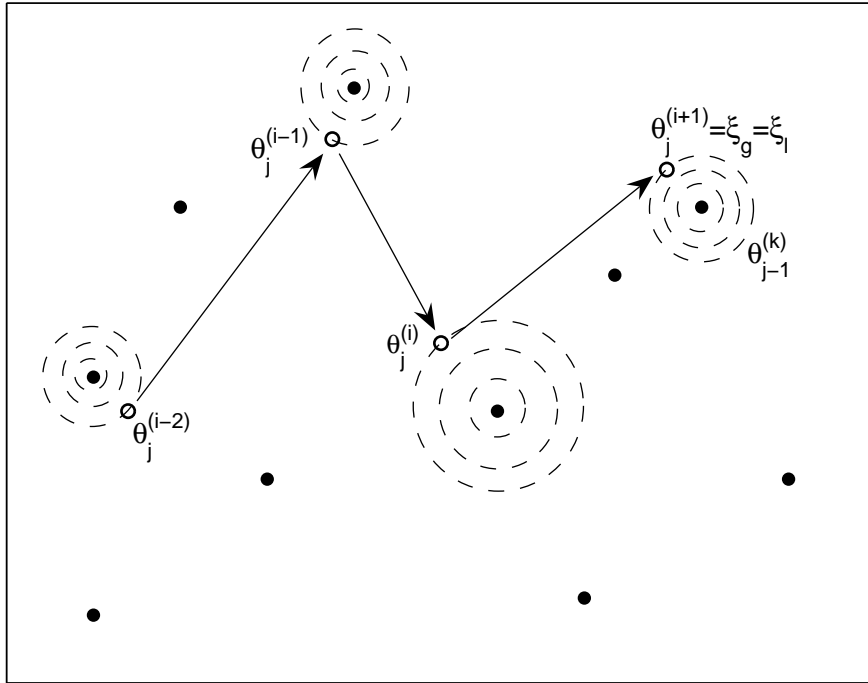
Figure 2: AIMS at annealing level j: disks $\bullet$ and circles $\circ$ represent $\theta_{j-1}^{(1)}, \ldots, \theta_{j-1}^{(N_{j-1})}$ and $\theta_j^{(1)}, \ldots, \theta_j^{(N_j)}$, respectively; concentric circles show the correspondence between $\theta_{j-1}^{(k)}$ that has been chosen in step 1a and the corresponding local candidate $\xi_l \sim q(\cdot|\theta_{j-1}^{(k)})$ that has been generated in step 1b. In this schematic picture, all shown candidate states are accepted as new states of the Markov chain.
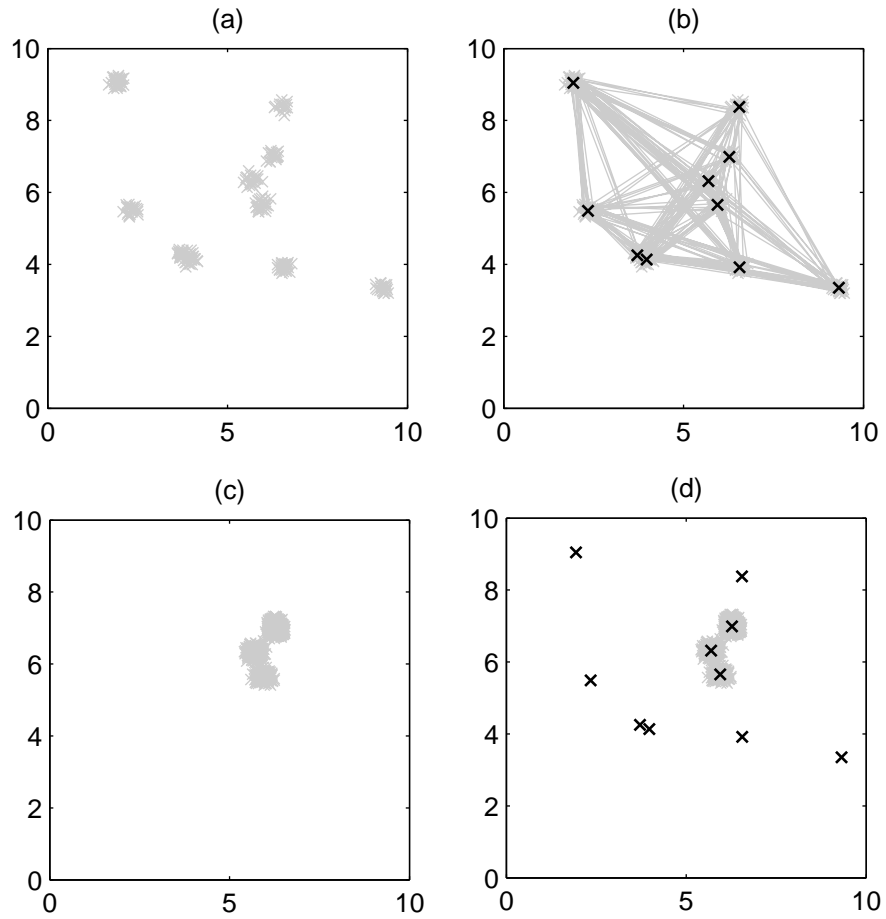
Figure 3: (a) Scatterplots of 10^3 posterior samples; (b) the trajectories of the corresponding posterior Markov chain obtained from AIMS; and (c), (d) corresponding plots from RWMH. Black crosses $\times$ represent the modes $\mu_1, \ldots, \mu_{10}$ of $\pi(\cdot)$ [Example 4.1].
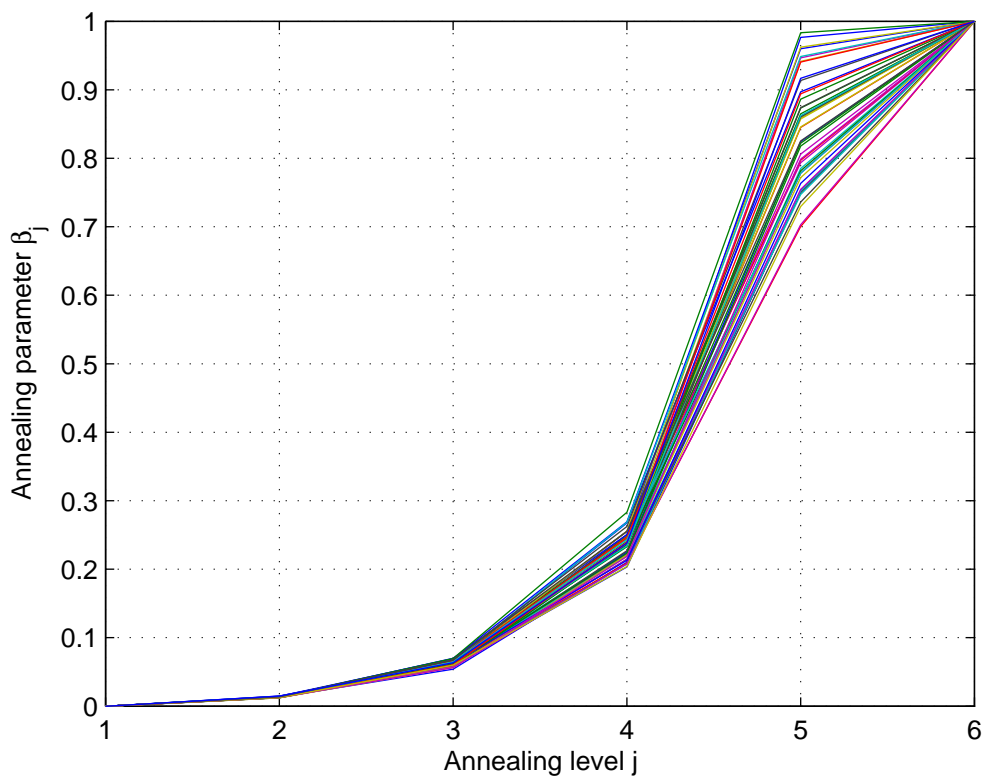
Figure 4: Annealing parameter β_j as a function of annealing level j for 50 independent runs of AIMS [Example 4.1].
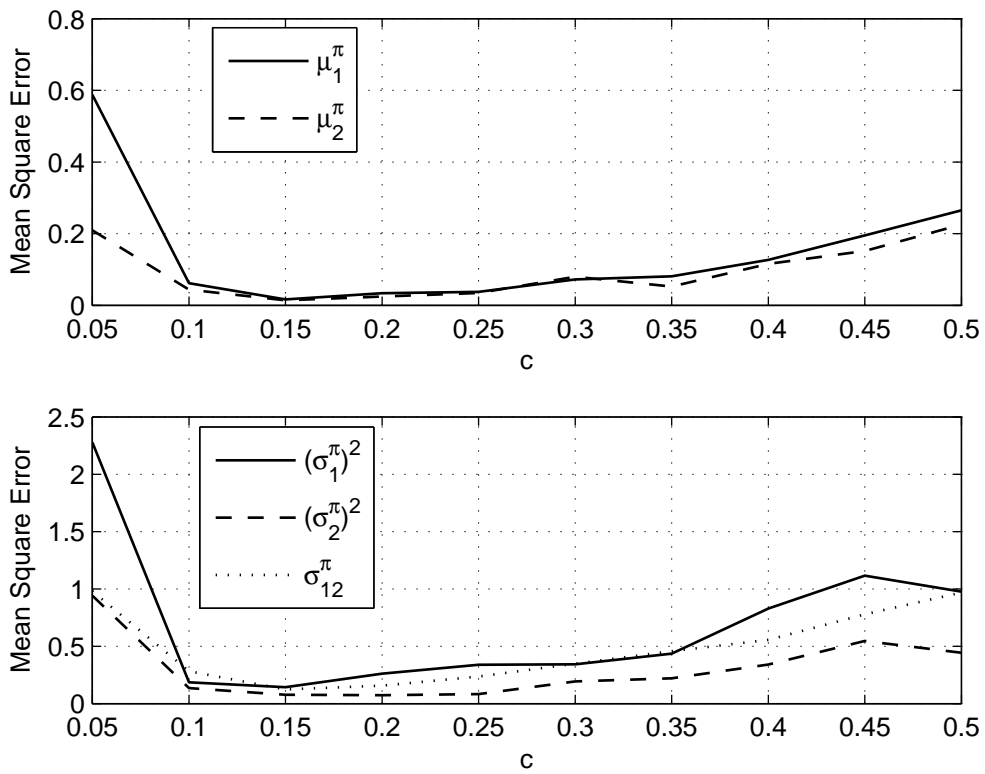
Figure 5: Mean square error of the AIMS estimator for the mean and covariance matrix as a function of the scaling factor c showing the optimal value is $c_{\text{opt}} \approx 0.15$ [Example 4.1].
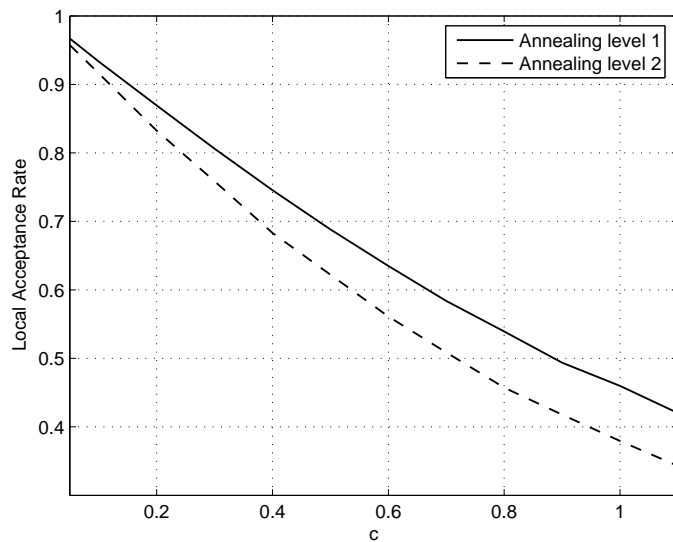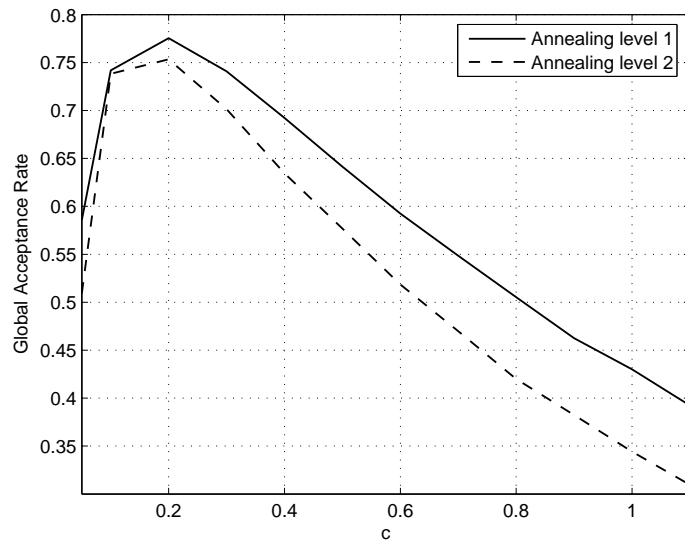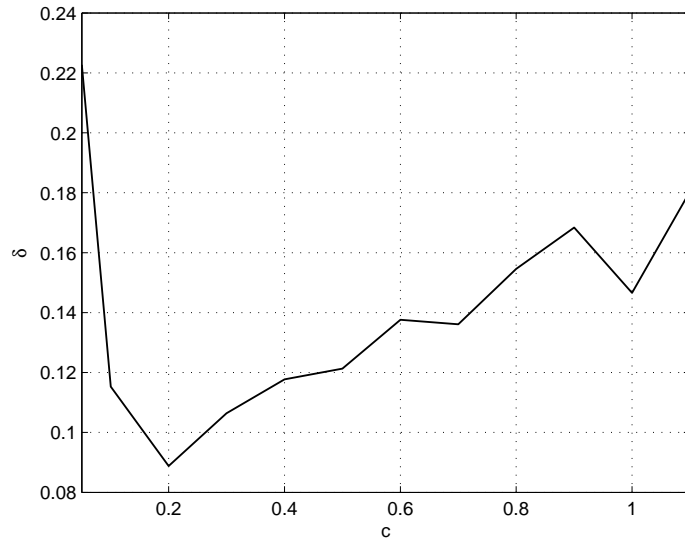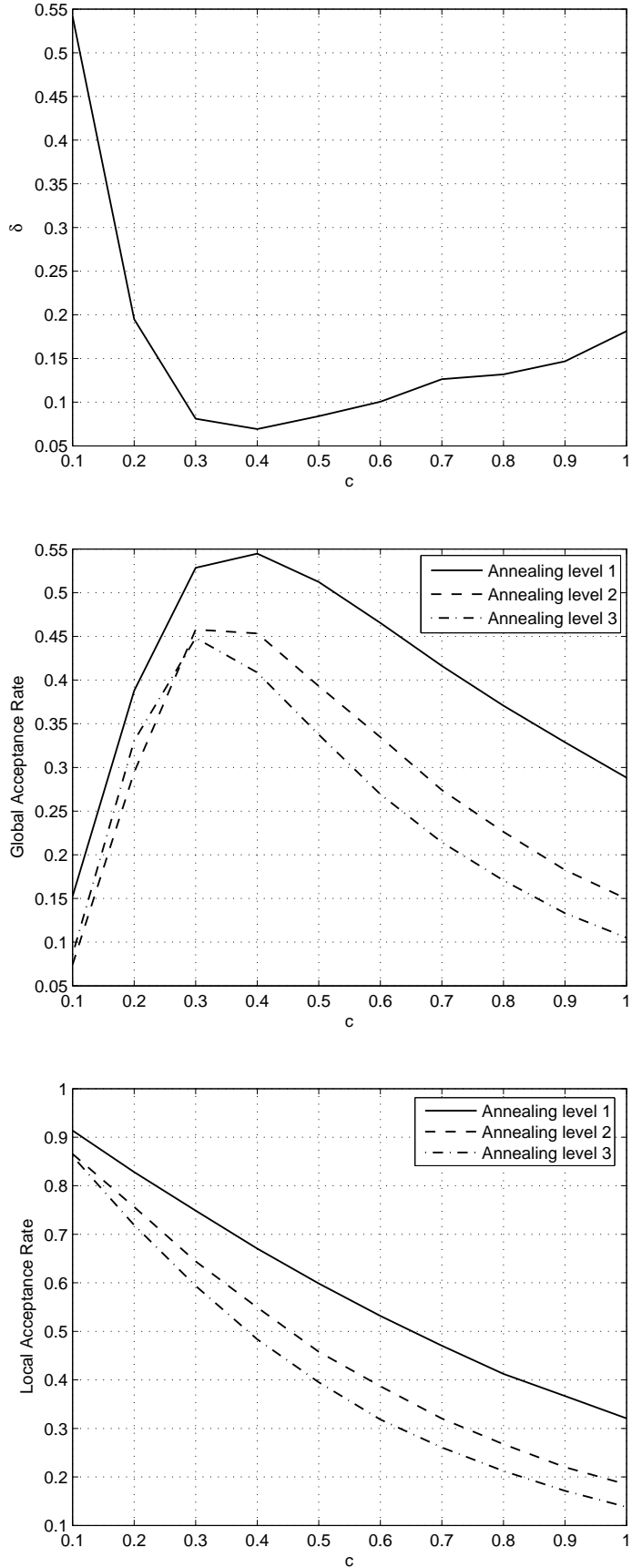
Figure 6: Coefficient of variation δ of the AIMS estimate (top panel), global acceptance rate (middle panel), and local acceptance rate (bottom panel) as functions of c for Case 1 ($d = 2$) [Example 4.2].
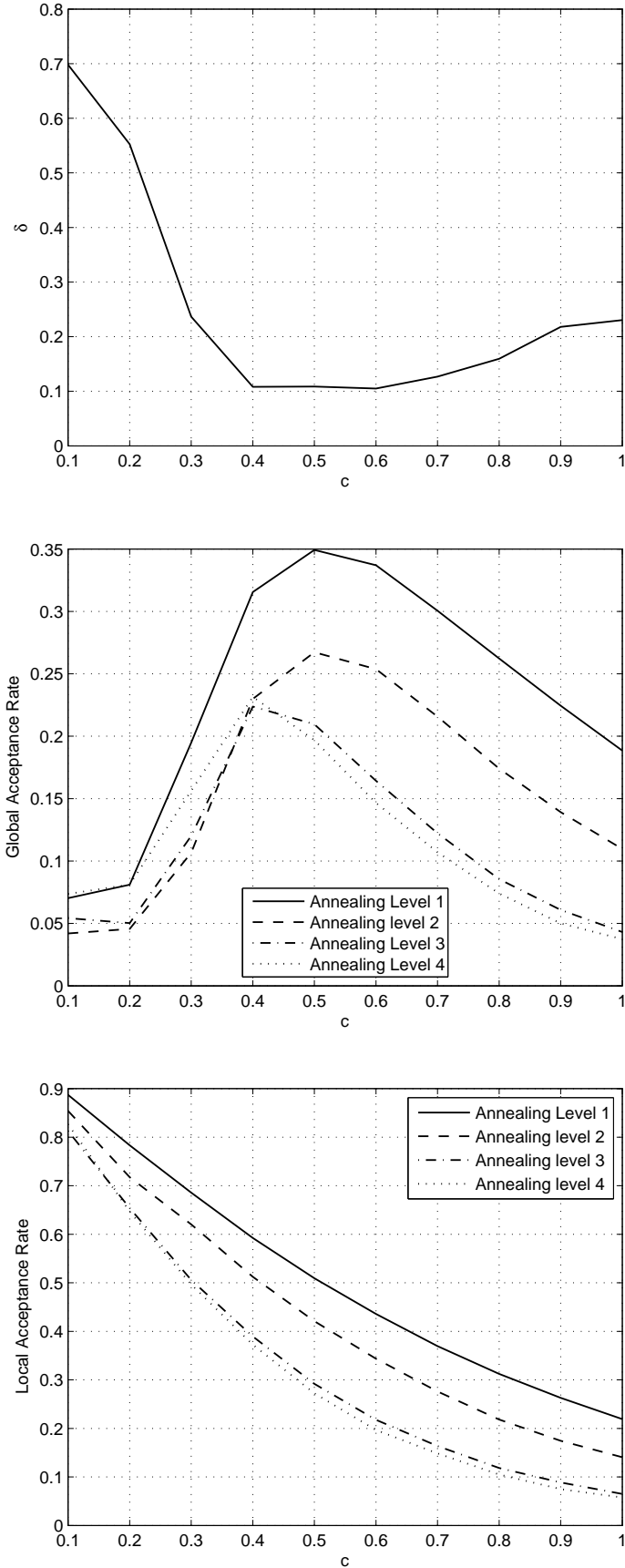
Figure 7: Coefficient of variation δ of the AIMS estimate (top panel), global acceptance rate (middle panel), and local acceptance rate (bottom panel) as functions of c for Case 2 ($d = 4$) [Example 4.2].
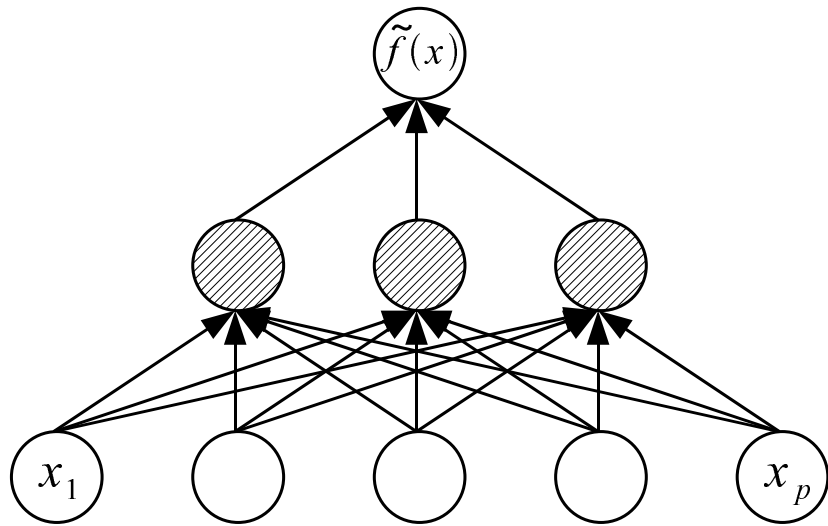
Figure 8: Coefficient of variation δ of the AIMS estimate (top panel), global acceptance rate (middle panel), and local acceptance rate (bottom panel) as functions of c for Case 3 ($d = 6$) [Example 4.2].

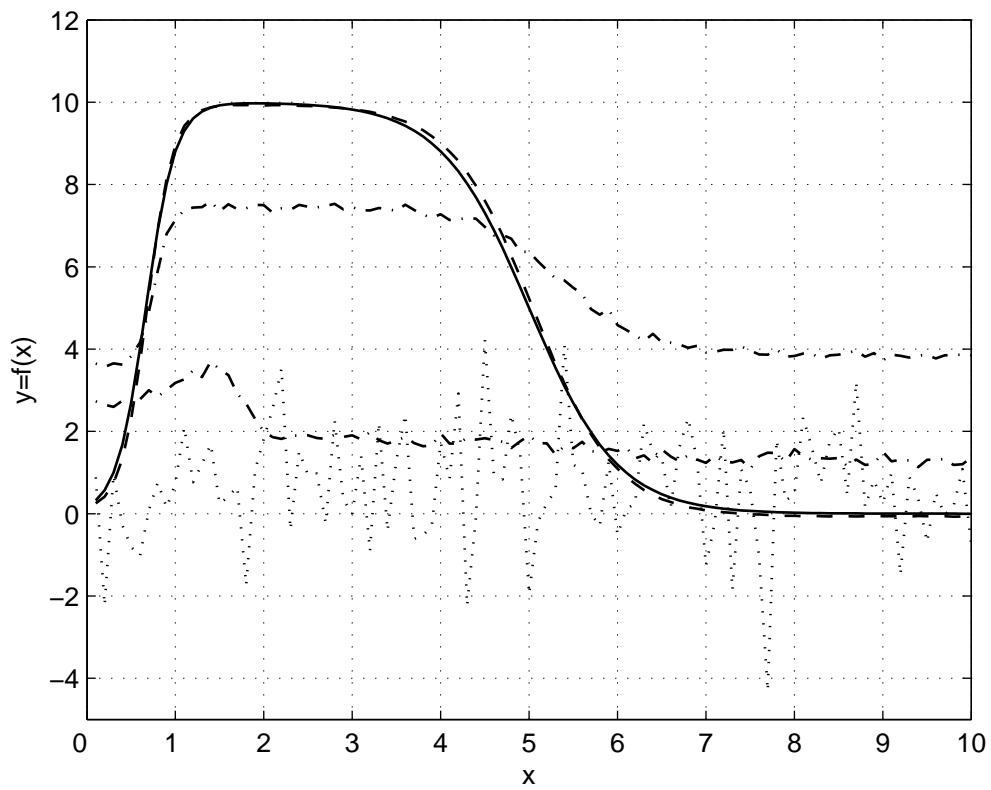Figure 9: The feed-forward neural network model with one hidden layer (shown by hatching) [Example 4.3].

Figure 10: The true function $f(x)$ (solid curve), its posterior approximation $\bar{\hat{f}}_{10}(x)$ (dashed curve) which is constructed using AIMS, and "intermediate annealing approximations": $\hat{\bar{f}}_0(x)$ (dotted curve) which is based on prior samples, $\hat{\bar{f}}_2(x)$ and $\bar{\hat{f}}_3(x)$ (dashed-dotted curves) [Example 4.3].

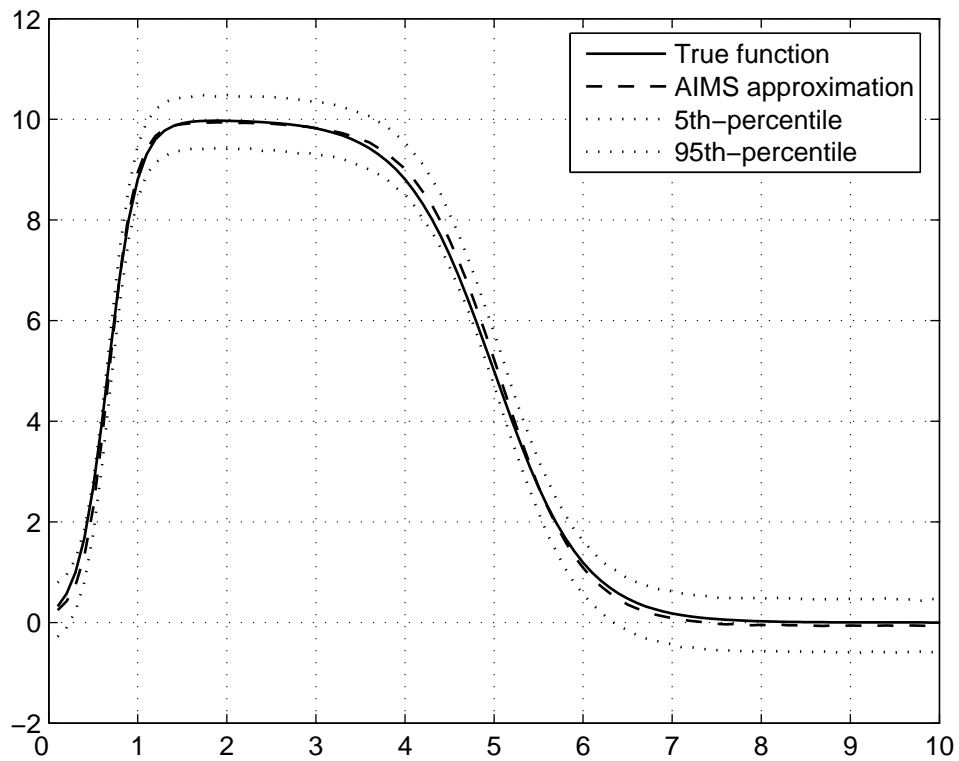Figure 11: The true function $f(x)$ (solid curve), its AIMS approximation $\bar{\hat{f}}_{10}(x)$ (dashed curve), and 5th and 95th percentiles of $\hat{\bar{f}}_{10}(x)$ (dotted curves) [Example 4.3].