Variance Reduction via Antithetic Markov Chains

James Neufeld
University of Alberta
jneufeld@ualberta.ca

Michael Bowling
University of Alberta
mbowling@ualberta.ca

Dale Schuurmans
University of Alberta
daes@ualberta.ca

Abstract

We present a Monte Carlo integration method, *antithetic Markov chain sampling* (AMCS), that incorporates local Markov transitions in an underlying importance sampler. Like *sequential Monte Carlo sampling*, the proposed method uses a sequence of Markov transitions to guide the sampling toward influential regions of the integrand (modes). However, AMCS differs in the type of transitions that may be used, the number of Markov chains, and the method of chain termination. In particular, from each point sampled from an initial proposal, AMCS collects a sequence of points by simulating two independent, but *antithetic* Markov chains, which are terminated by a sample-dependent stopping rule. Such an approach provides greater flexibility for targeting influential areas while eliminating the need to fix the length of the Markov chain *a priori*. We show that the resulting estimator is unbiased and can reduce variance on peaked multimodal integrands that challenge current methods.

1 Introduction

We consider Monte Carlo algorithms for approximating integrals of the form

$$\mathcal{I} \doteq \int h(x)\pi(x)dx, \tag{1}$$

where $h : \mathbb{R}^d \to \mathbb{R}$ is a bounded measurable function, π is a probability density, and both h and π are efficiently evaluable at any point x in the domain. We assume the standard measure space $(\mathbb{R}^d, \mathcal{B}, dx)$ where $\mathcal{B}$ is the Borel σ-algebra and dx the Lebesgue measure.

In practical settings, π is often only evaluable up to an unknown constant $\mathcal{Z}_{\hat{\pi}}$, in which case we assume access to an unnormalized function $\hat{\pi}$ such that $\hat{\pi}(x) \doteq \pi(x)\mathcal{Z}_{\hat{\pi}}$, hence $\mathcal{Z}_{\hat{\pi}} = \int \hat{\pi}(x)dx$. In such cases, approximations of the normalizing constant $\mathcal{Z}_{\hat{\pi}}$ are also of interest, either to aid in approximating Eq. (1) or to conduct separate tasks such as Bayesian model comparison [13].

A straightforward approach for approximating $\mathcal{I}$ and $\mathcal{Z}_{\hat{\pi}}$ is *importance sampling*, where an i.i.d. sample $\{X^{(1)}, ..., X^{(N)}\}$ is first simulated from a fixed *proposal distribution*, π_0, then the following estimator computed

$$\mathcal{I}_{IS}^N \doteq N^{-1} \sum_{i=1}^{N} w(X^{(i)})h(X^{(i)}). \tag{2}$$

Here $w(X^{(i)}) \doteq \pi(X^{(i)})/\pi_0(X^{(i)})$ is referred to as the *importance weight*. By ensuring $\text{supp}(\pi) \subseteq \text{supp}(\pi_0)$ and that the variance is bounded, $\mathbb{V}\left(h(X)w(X)\right) < \infty$, the resulting estimate Eq. (2) is *unbiased*, $\mathbb{E}\left[\mathcal{I}_{IS}^N\right] = \mathcal{I}$, consistent, and has a mean square error (MSE) of $\mathbb{V}\left(w(X)h(X)\right)/N$.

Additionally, when only $\hat{\pi}$ is known, one can approximate the normalizing constant $\mathcal{Z}_{\hat{\pi}}$ using the unbiased estimator

$$\mathcal{Z}_{IS}^N \doteq N^{-1} \sum_{i=1}^{N} w(X^{(i)}), \tag{3}$$

where $\hat{\pi}$ is used in place of π in the importance weight. This estimator can also aid in approximating $\mathcal{I}$ via the consistent *weighted importance sampling* estimator[1]

$$\mathcal{I}_{WIS}^N \doteq N^{-1} \sum_{i=1}^{N} h(X^{(i)})w(X^{(i)})/\mathcal{Z}_{IS}^N. \tag{4}$$

The primary limitation of importance sampling is that the proposal density π_0 must be specified *a priori*, yet the quality of the estimator depends critically on how well it matches the integrand. In particular, $\mathbb{V}(\mathcal{I}_{IS}^N)$ is minimized by using $\pi_0(x) \propto |h(x)|\pi(x)$, and $\mathbb{V}(\mathcal{Z}_{IS}^N)$ when $\pi_0(x) \propto \hat{\pi}(x)$. In practice, effective proposal densities are notoriously difficult to construct since the locations of the high-magnitude (important) regions are unknown. In this paper, we develop an approach for overcoming a weak

Appearing in Proceedings of the 18^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2015, San Diego, CA, USA. JMLR: W&CP volume 38.

[1] This estimator can still be used with π, often resulting in lower variance. Although such an estimator is biased, it is consistent with bias decreasing at a rate of $\mathcal{O}(1/N)$ [12].

proposal density in scenarios where the integrand is multimodal and *peaked* so that the majority of the integral is concentrated in modes that cover a small proportion of the domain; e.g., as illustrated in Fig. 1.

One popular approach to tackling these problems is to exploit local structure in the integrand by augmenting the proposal with a series of local transitions, such as Markov chain Monte Carlo (MCMC) moves. For instance, the method of *annealed importance sampling* (AIS) [7], or more generally *sequential Monte Carlo sampling* (SMCS) [3], attempts to direct the sampler toward more influential areas of the integrand (modes) using such a strategy. An important limitation of these methods, however, is that the Markov chains must be defined by the same fixed-length move sequence regardless of the starting point or integrand values. Consequently, when modes are separated by plateaus, simulating local Markov chains often provide no discernible benefit to the sampler. This difficulty can sometimes be mitigated through the use of resampling, which allows computation to be reallocated toward samples with larger importance weights, or through *adaptive* parameter tuning, but such extensions are often unsuitable for parallel processing or for use with limited memory [4, 1].

In this work we present a related but novel approach, *Antithetic Markov Chain Sampling* (AMCS), that augments a fixed proposal density by simulating two (independent) antithetic Markov chains from each proposal point. A key advantage of this approach is the ability to terminate chains using predefined *stopping conditions*; for example, when the integrand values are unchanging (plateau). This allows the sampler to reallocate computation toward more influential regions without requiring a large sample population. We show that returning the average of the integrand evaluated at each point yields an unbiased estimate, often with a significant reduction in variance. The utility of the proposed method is demonstrated on a Bayesian k-means posterior and a robot localization task where highly accurate relational sensors (i.e., LIDAR) are known to create sharply peaked posterior distributions [15].

Notation We use upper case letters to denote random variables and lower case to denote non-random counterparts in the same domain. We will use $\mathbb{E}_\pi[X]$, $\mathbb{V}_\pi(X)$ and $\mathbb{P}_\pi(Q(X))$ to denote the expectation and variance of $X \sim \pi$ and the probability of event $Q(X)$ respectively, omitting the subscript when the distribution is clear from context. Also, $\mathbb{I}\{p\}$ will denote the indicator function that returns 1 if the predicate p is true and 0 otherwise, and $x_{i:j} \doteq (x_i, x_{i+1}, ..., x_j)$ will denote a sequence.

Additionally, throughout this paper we make use of Markov transition kernels; formally, on the measurable space $(\mathcal{X}, \mathcal{B})$ we define a kernel as a mapping $K : \mathcal{X} \times \mathcal{B} \to [0, 1]$. Following standard practice in the Monte Carlo literature, we also let $K(x, x')$ denote the conditional density of the transition $K(x, \cdot)$; that is, $\mathbb{P}(X \in A|x) = \int_A K(x, x')dx'$.

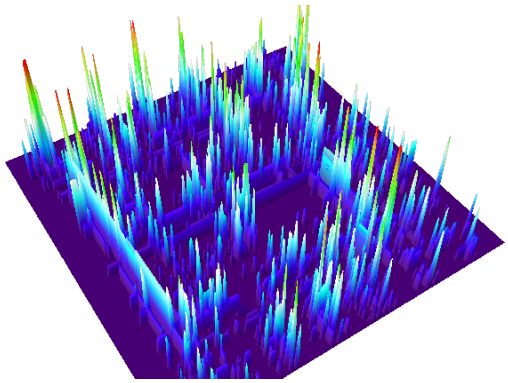

Figure 1: Log-likelihood function of position given sensor readings in a Bayesian robot localization problem (Sec. 4)

2 Background and Related Work

We begin by detailing the popular and closely related sequential Monte Carlo sampling (SMCS) approach [3]. SMCS is an extension of importance sampling that augments the proposal by simulating a Markov chain of fixed length (n) specified by *forward* transition kernels $\{F_1, ..., F_n\}$. To reduce variance, SMCS also exploits a sequence of *backward* kernels $\{B_1, ..., B_n\}$, where it is assumed that all F_i and B_i can be efficiently evaluated pointwise and that all F_i are simulable. Additionally, SMCS employs a sequence of (potentially un-normalized) *intermediate distributions* $\{\pi_1, ..., \pi_n\}$ that (ideally) blend smoothly between the proposal distribution (π_0) and $\pi_n \doteq \pi$. Common choices for these intermediate distributions include the *homogeneous* sequence $\pi_j = \pi$ for $j > 0$, or the *tempered* version $\pi_j = \pi^{(1-\beta_j)}\pi_0^{\beta_j}$ for a fixed schedule $1 = \beta_0 > \cdots > \beta_n = 0$. From these components, one defines a step-wise importance weighting function

$$R_j(x_{j-1}, x_j) \doteq \frac{\pi_j(x_j)B_j(x_j, x_{j-1})}{\pi_{j-1}(x_{j-1})F_j(x_{j-1}, x_j)}, \qquad (5)$$

which can be used to define a sequence importance weights

$$w_j(x_{0:j}) = R_j(x_{j-1}, x_j)w_{j-1}(x_{0:j-1})$$

recursively, starting from $w_0(x_0) = 1$. This weighting is used to cancel any bias that would otherwise be introduced by simulating the forward Markov chain. The full SMCS procedure is given in Algorithm 1.

The key advantage of Algorithm 1 is that it only requires sampling from $\pi_0(\cdot)$ and $F_j(x_{j-1}, \cdot)$, not from $\pi_n(\cdot)$ which might be intractable. For simplicity, the pseudocode omits the optional resampling step that has been extensively developed in the literature [6, 4, 3]. It is also worth noting that

Algorithm 1 SMCS Procedure

for $i \in \{1, ..., N\}$
  Sample $X_0^{(i)} \sim \pi_0(\cdot)$; set $w_0^{(i)} = 1$
end for
for $i \in \{1, ..., N\}$
  for $j \in \{1, ..., n\}$
    Sample $X_j^{(i)} \sim F_j(X_{j-1}^{(i)}, \cdot)$;
    Set $w_j(X_{0:j}^{(i)}) = w_{j-1}(X_{0:j-1}^{(i)}) R_j(X_{j-1}^{(i)}, X_j^{(i)})$;
  end for
  Let $X^{(i)} = X_n^{(i)}$
end for
return estimates from Eq. (2) or Eq. (3) (alternatively Eq. (6) or Eq. (7)) using $\{X^{(i)}\}$ and $\{w_n(X_{0:n}^{(i)})\}$

in the case where *homogenous* intermediate distributions are used the following unbiased estimators are available

$$\mathcal{I}_{SMCS}^N \doteq N^{-1} \sum_{i=1}^{N} n^{-1} \sum_{j=1}^{n} w_j^{(i)} h(X_j^{(i)}), \tag{6}$$

$$\mathcal{Z}_{SMCS}^N \doteq N^{-1} \sum_{i=1}^{N} n^{-1} \sum_{j=1}^{n} w_j^{(i)}. \tag{7}$$

Despite its generality, the most commonly deployed form of SMCS is the pre-dated *annealed importance sampling* method (AIS) [7], where one defines the forward kernel using any MCMC transition to ensure $F_j(x, x')\pi_j(x) = F_j(x', x)\pi_j(x')$. The backward kernel is similarly defined as $B_j(x, x') = F_j(x', x)\pi_j(x')/\pi_j(x)$. These choices lead to convenient cancellations, since the weights in Eq. (5) then become $R_j(x_{j-1}, x_j) = \pi_j(x_{j-1})/\pi_{j-1}(x_{j-1})$.

Note, despite its use of MCMC moves, AIS does not require its local chain to approach stationarity to ensure unbiasedness. It does, however, require the chain to mix rapidly to yield worthwhile variance reduction. Naturally, in multimodal problems, like that illustrated in Fig. 1, MCMC kernels achieving rapid mixing are difficult to formulate and AIS may exhibit poor performance.

3 Antithetic Markov Chain Sampling

As an alternative we propose the *antithetic Markov chain sampling* (AMCS) approach that, like SMCS, extends importance sampling through the use of local Markov transitions. Roughly speaking, the algorithm first draws a single sample from the proposal π_0, simulates two independent Markov chains to produce a set of points, evaluates the target function on each, then returns the resulting average.

More precisely, the local Markov chains are simulated using two Markov transition kernels, a *positive* kernel, K_+, and a *negative* kernel K_-. Additionally, these chains are terminated by probabilistic stopping rules, referred to as (positive and negative) *acceptance functions*, A_+ and A_-,

Algorithm 2 AMCS Procedure

1: **for** $i \in \{1, ..., N\}$
2: Sample $X_0^{(i)} \sim \pi_0(\cdot)$;
3: **for** $j = 1, 2, ...$
4: Sample $X_j^{(i)} \sim K_+(X_{j-1}^{(i)}, \cdot)$;
5: With probability $1 - A_+(X_{j-1}^{(i)}, X_j^{(i)})$ **break loop** and set $M_+^{(i)} = j$;
6: **end for**
7: **for** $j = -1, -2, ...$
8: Sample $X_j^{(i)} \sim K_-(X_{j+1}^{(i)}, \cdot)$;
9: With probability $1 - A_-(X_{j+1}^{(i)}, X_j^{(i)})$ **break loop** and set $M_-^{(i)} = j$;
10: **end for**
11: **end for**
12: **return** estimates from Eq. (8) or Eq. (9)

that specify the probability of accepting each move in the respective directions. The kernels must be efficiently evaluable, simulable, and must also satisfy a *joint symmetry* property together with the acceptance functions.

Definition 1. *The Markov kernels and acceptance functions* (K_+, A_+) *and* (K_-, A_-) *are said to be* jointly symmetric *if for any* $x, x' \in \mathbb{R}^d$ *the following holds*

$$K_+(x, x')A_+(x, x') = K_-(x', x)A_-(x', x).$$

Given these components, we formulate the AMCS procedure given in Algorithm 2. The procedure first draws N samples from π_0, then for each sample the algorithm simulates a *positive* chain until termination (lines 3-6), then simulates a *negative* chain (lines 7-10) before returning the trajectory. To ensure that the algorithm terminates we also require the following assumption.

Assumption 1. *The acceptance functions* A_+ *and* A_- *are assumed to terminate any chain within a finite number of transitions; i.e.* $M_+ < \infty$ *and* $M_- > -\infty$ *almost surely.*

After using AMCS to produce N *trajectories* and indices, $\left\{ \left(X_{M_-^{(1)}}^{(1)}, ..., X_{M_+^{(1)}}^{(1)}\right), \ ..., \ \left(X_{M_-^{(N)}}^{(N)}, ..., X_{M_+^{(N)}}^{(N)}\right) \right\}$ we approximate the desired quantity with the estimators

$$\mathcal{I}_{AMCS}^N \doteq N^{-1} \sum_{i=1}^{N} \frac{1}{\bar{M}^{(i)}} \sum_{j=M_-^{(i)}+1}^{M_+^{(i)}-1} \frac{h\left(X_j^{(i)}\right)\pi\left(X_j^{(i)}\right)}{\pi_0\left(X_0^{(i)}\right)}, \tag{8}$$

$$\mathcal{Z}_{AMCS}^N \doteq N^{-1} \sum_{i=1}^{N} \frac{1}{\bar{M}^{(i)}} \sum_{j=M_-^{(i)}+1}^{M_+^{(i)}-1} \frac{\hat{\pi}\left(X_j^{(i)}\right)}{\pi_0\left(X_0^{(i)}\right)}, \tag{9}$$

where $\bar{M}^{(i)} \doteq M_+^{(i)} - M_-^{(i)} - 1$. Note that the two endpoints X_{M_-} and X_{M_+} are not used in the estimate, we refer to all other points $(X_{M_-+1}, ..., X_{M_+-1})$ as the *accepted* points.

3.1 Unbiasedness

We will now establish the unbiasedness of the estimators given in Eq. (8) and Eq. (9). In doing so we will need to consider the joint density over the random variables $(M_-, M_+, X_{M_-}, ..., X_{M_+})$; however, to avoid the burdensome notation resulting from negative indices we simplify the subsequent notation by remapping indices to $(M, M_0, X_0, ..., X_M)$ such that $M \doteq M_+ - M_-$ and $M_0 \doteq -M_-$, $\bar{M} = M - 1$.

Since the proof of unbiasedness for either estimator follows the same progression, we will only consider the estimator Eq. (8). We begin by simplifying the inner summation by through the definition of a new random variable $J^{(i)} \sim \text{Uniform}(\{1, ..., M^{(i)} - 1\})$ and observe that

$$\begin{aligned}\mathbb{E}\left[\mathcal{I}^N_{AMCS}\right] &= \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}\frac{h(X^{(i)}_{J_i})\pi(X^{(i)}_{J_i})}{\pi_0(X^{(i)}_{M_0})}\right] \\ &= \mathbb{E}\left[\frac{h(X_J)\pi(X_J)}{\pi_0(X_{M_0})}\right]. \qquad (10)\end{aligned}$$

That is, the inner summation and the coefficient $\frac{1}{\bar{M}}$ in Eq. (8) can be interpreted as an expectation with respect to the uniform distribution, and the last equality follows from the independence between trajectories. To make use of the important symmetry properties relating X and X_J we will need the following lemma.

Lemma 1. *Suppose $X \sim \pi_0(\cdot)$ and $X' \sim K(X, \cdot)$ for symmetric Markov kernel K, that is $K(x, x') = K(x', x)$. It follows that*

$$\mathbb{E}\left[\frac{h(X')\pi(X')}{\pi_0(X)}\right] = \mathcal{I}.$$

(The lemma follows from the measure theoretic properties of Markov kernels; the full proof is given in Appendix A.)

Consequently, it remains only to show only that the process of generating a AMCS trajectory, then selecting a point uniformly at random, can be expressed as a symmetric Markov kernel. With this objective in mind we proceed by showing that the likelihood of generating a trajectory is independent of the point initiating the two chains. Specifically, we can write the density for $(M, M_0, X_0, ..., X_M)$ as

$$\begin{aligned}\gamma(m, m_0, x_0, ..., x_m) \doteq\ & \\ & (1 - A_-(x_1, x_0))K_-(x_1, x_0) \\ & \times \prod_{j=2}^{m_0} A_-(x_j, x_{j-1})K_-(x_j, x_{j-1}) \\ & \times \prod_{j=m_0}^{m-1} A_+(x_j, x_{j+1})K_+(x_j, x_{j+1}) \\ & \times (1 - A_+(x_{m-1}, x_m))K_+(x_{m-1}, x_m). \qquad (11)\end{aligned}$$

This density function meets an important symmetry condition formalized in the following lemma.

Lemma 2. *For the density γ defined in Eq. (11), given any jointly symmetric (K_+, A_+) and (K_-, A_-), sequence $(x_0, ..., x_m)$, and integers m, m_0 and m'_0 such that $m > 1$, $0 < m_0 < m$ and $0 < m'_0 < m$, it follows that*

$$\gamma(m, m_0, x_0, ..., x_m) = \gamma(m, m'_0, x_0, ..., x_m)$$

(The equality follows from the definition of joint symmetry; the full proof is given in Appendix C.)

Using the density function for a point chosen uniformly at random from a larger set of random values, given in Lemma 4 (Appendix B), we can describe the process of generating a trajectory with AMCS $(X_{M_-}, ..., X_{M_+})$ then uniformly drawing a point $X \in (X_{M_-+1}, ..., X_{M_+-1})$ as sampling from a forward transition kernel given by

$$K(x, x') = \sum_{m=2}^{\infty}\sum_{m_0=1}^{m-1}\frac{1}{m-1}\sum_{j=1, j\neq m_0}^{m-1}\gamma_j(m, m_0, x, x'), \qquad (12)$$

where $\gamma_j(m, m_0, x, x')$ is the density function γ (Eq. (11)) with x' in the jth position and the remaining x-variables excluding x and x' marginalized out. More precisely, if we let $\bar{x}^{(m_0=x, j=x')}$ denote $(x_0, ..., x_m)$ with x in position m_0 and x' in position j, and let $\bar{x}^{\setminus\{m_0, j\}}$ denote $(x_0, ..., x_m) \setminus \{x_j, x_{m_0}\}$ then the marginal density can be expressed by

$$\gamma_j(m, m_0, x, x') \doteq \int \gamma(m, m_0, \bar{x}^{(m_0=x, j=x')})d\bar{x}^{\setminus\{m_0, j\}}.$$

We now establish the symmetry of the above forward density function through the following lemma.

Lemma 3. *If the density γ satisfies the conditions in Lemma 2 the forward transition kernel in Eq. (12) satisfies*

$$K(x, x') = K(x', x).$$

(The lemma follows from reordering sums in γ and deploying Lemma 2; the full proof is given in Appendix D.)

From these three lemmas we can now establish the unbiasedness of AMCS.

Theorem 1. *Provided the transition kernels and acceptance functions satisfy the conditions of Lemma 2 and Lemma 3, for any $N > 0$ the AMCS procedure achieves*

$$\begin{aligned}\mathbb{E}\left[\mathcal{I}^N_{AMCS}\right] &= \mathcal{I} \\ \mathbb{E}\left[\mathcal{Z}^N_{AMCS}\right] &= \mathcal{Z}_{\hat{\pi}}.\end{aligned}$$

(The theorem follows directly from the symmetry of the forward transition kernel shown in Lemma 3, in conjunction with Lemma 1 and Eq. (10).)

3.2 Variance Analysis

Since the AMCS estimator is unbiased for any choice of jointly symmetric $K_{+/-}$ and $A_{+/-}$ we can now consider how these choices affect its variance. In the following development we reuse the definition of uniformly distributed index J, and additionally to save space we define $f(x, x') \doteq \frac{h(x)\pi(x)}{\pi_0(x')}$. Observe that

$$v_{AMCS} \doteq \mathbb{V}\left(\sum_{j=1}^{M-1} \frac{1}{M} f(X_j, X_{M_0})\right) = \mathbb{V}\left(\mathbb{E}\left[f(X_J, X_{M_0})|X_0, ..., X_M\right]\right),$$

where the inner expectation is take w.r.t. J. Now consider the discrepancy between the above variance expression and that of vanilla importance sampling given by $v_{IS} \doteq \mathbb{V}\left(f(X, X)\right)$ for $X \sim \pi_0(\cdot)$. To relate these quantities we make the simplifying assumption that π_0 is *locally uniform*: that is, for all $x \in \text{supp}(\pi_0)$ and all $x' \in K(x, \cdot)$ we assume $\pi_0(x) = \pi_0(x')$, where K is the Markov kernel given in Eq. (12). This assumption allows us to essentially ignore the effects of π_0, which are expected to be negligible in practice. From this assumption and the symmetry and measurability of K (Lemma 3) it follows that

$$\begin{aligned}\mathbb{V}\left(f(X_J, X_{M_0})\right) &= \iint f(x', x)^2 K(x, x')\pi_0(x)dx'dx - \mu^2 \\ &= \int f(x', x')^2 \pi_0(x')dx' \int K(x', x)dx - \mu^2 \\ &= \mathbb{V}\left(f(X, X)\right) = v_{IS}\,.\end{aligned}$$

That is, if one were to actually use a uniformly drawn sample from each trajectory the variance of the resulting estimator would be unchanged. Furthermore, using the law of total variance we also have that

$$\begin{aligned}v_{IS} &= \mathbb{E}\left[\mathbb{V}\left(f(X_J, X_{M_0})|X_0, ..., X_M\right)\right] \\ &\quad + \mathbb{V}\left(\mathbb{E}\left[f(X_J, X_{M_0})|X_0, ..., X_M\right]\right) \\ &= \mathbb{E}\left[\mathbb{V}\left(f(X_J, X_{M_0})|X_0, ..., X_M\right)\right] + v_{AMCS}\,.\end{aligned}$$

From this we can now define the *variance capture* identity

$$v_{AMCS} = v_{IS} - \mathbb{E}\left[\mathbb{V}\left(f(X_J, X_{M_0})|X_0, ..., X_M\right)\right]. \quad (13)$$

This identify shows that the variance of the AMCS estimator cannot be higher than a vanilla importance sampling estimator given the same number of samples. Additionally, the variance reduction is due entirely to the expected variance of the points *inside* a given trajectory under the uniform distribution. These intuitions motivate the use of so-called *antithetic Markov chains*, whose transition kernels $K_{+/-}$ are configured to explore the integrand in opposite directions in the hopes of capturing greater variance.

However, before proposing specific choices for $K_{+/-}$ and $A_{+/-}$ the additional computational costs for simulating the Markov chains must also be considered. For instance, if one considers a basic Monte Carlo estimator taking the empirical average of an arbitrary sequence of i.i.d. random variables, say $X^{(1)}, ..., X^{(N)}$, the variance is given by $\frac{\mathbb{V}(X)}{N}$. Alternatively, consider a procedure that has a stochastic cost associated with each sample, denoted by the random variables $D^{(1)}, ..., D^{(N)}$ such that $\delta \doteq \mathbb{E}\left[D\right]$, where it is assumed $D^{(i)} \perp\!\!\!\perp X^{(i)}$. By fixing a computational budget $C >> \delta$, standard arguments for renewal reward processes indicate that the resulting estimator will have a variance of approximately $\frac{\mathbb{V}(X)}{C/\delta} = \frac{\delta\mathbb{V}(X)}{C}$. Simply put, if technique A requires, on average, a factor of δ more computation per sample than technique B, then it must have a reduced variance by a factor of at least $1/\delta$ to be worthwhile. Substituting this forumla into Eq. (13) shows that AMCS will offer a variance reduction whenever

$$\mathbb{E}\left[\mathbb{V}\left(f(X_J, X_{M_0})|X_0, ..., X_M\right)\right] > \frac{\delta - 1}{\delta} v_{IS},$$

where $\delta = \mathbb{E}\left[M\right]$ gives the expected computational cost in terms of the number of evaluations of π and h. It is clear from this expression that the potential for savings drops of quickly as the computational costs increase. In the next section we explore ways in which the acceptance functions can be used to help keep these costs in check.

3.3 Parameterization

We now provide some specific parameterizations for $K_{+/-}$ and $A_{+/-}$ which the goal of exploiting the structure of multi-modal, peaked integrands. Perhaps the most useful observation one can make in this setting is that if the integrand is near zero at some point, it is not likely to have large values in the vicinity, hence continuing a local Markov chain simulation is not likely to be worthwhile. Conversely, if the integrand value has non-negligible magnitude it has a much higher chance of being near a mode. This observation motivates the *threshold acceptance function*

$$A_{+/-}(x, x') = \begin{cases} 1, & \text{if } |f(x)| > \varepsilon \text{ and } |f(x')| > \varepsilon \\ 0, & \text{otherwise,} \end{cases}$$

where $\varepsilon > 0$ and $f(x)$ is some function of interest (e.g. the integrand). An important side-benefit of this acceptance function is that if the first point sampled from π_0 is below the threshold, the AMCS procedure can immediately return without evaluating the integrand at any neighboring points, hence avoiding additional computational cost.

Keeping in mind the objective of "capturing" variability through the use of antithetic chains, a natural first choice for Markov kernels are the *linear* kernel densities, given by

$$\begin{aligned}K_+(x, \cdot) &= \mathcal{N}(x + v, \sigma^2 I), \\ K_-(x, \cdot) &= \mathcal{N}(x - v, \sigma^2 I),\end{aligned}$$

where $\mathcal{N}(\mu, \Sigma)$ denotes a multivariate normal distribution with mean μ and covariance Σ, $v \in \mathbb{R}^d$ is some fixed

vector, and σ^2 a fixed variance parameter. For any configuration with $\sigma^2 << ||v||^2$ one can expect the resulting Markov chain to make consistent progress in one direction and hence experience more function variability than, say, a normal random walk given the same number of steps.

For continuously differentiable integrands one can use the gradient to set the direction vector, which results in the *Langevin* Markov kernels

$$K_+(x,\cdot) = \mathcal{N}(x + \varepsilon\nabla f(x), \sigma^2 I),$$
$$K_-(x,\cdot) = \mathcal{N}(x - \varepsilon\nabla f(x), \sigma^2 I),$$

where $\varepsilon > 0$ is a step size parameter. Since the gradient points in the direction of steepest ascent this choice seems ideal for capturing variability within a trajectory. However, a potential concern is that the Langevin kernel is not exactly symmetric for nonlinear functions. While this issue can be partially addressed by ensuring the gradient vector is normalized to length 1, exact joint symmetry (Definition 1) can be attained through the use of the *symmetrizing* acceptance functions

$$A_+(x,x') = \min\left(\frac{K_-(x',x)}{K_+(x',x)}, 1\right),$$
$$A_-(x,x') = \min\left(\frac{K_+(x',x)}{K_-(x',x)}, 1\right).$$

Note that multiple acceptance functions can be combined into a single function by taking their product.

Finally, when following gradient steps in either direction one can expect to eventually settle around a local mode or plateau. Since continuing the chain is not likely to capture any additional function variation, it is beneficial to terminate the chain in these cases, which can be accomplished through the use of the *monotonic* acceptance functions

$$A_+(x,x') = \begin{cases} 1, & \text{if } f(x) + \varepsilon < f(x') \\ 0, & \text{otherwise,} \end{cases}$$
$$A_-(x,x') = \begin{cases} 1, & \text{if } f(x) - \varepsilon > f(x') \\ 0, & \text{otherwise,} \end{cases}$$

where $\varepsilon \geq 0$ is some fixed threshold. This acceptance function ensures the chains make monotonic progress.

4 Experimental Evaluation

In this section we evaluate the performance of the AMCS procedure and contrast it with that of related existing techniques, namely, vanilla *importance sampling* (IS), *annealed importance sampling* (AIS), and *greedy importance sampling* (GIS) [14]. The previously unmentioned GIS approach uses a sequence of deterministic, axis-aligned, steepest ascent moves to augment a fixed proposal. Note, this method does not use continuous gradient information and instead computes the steepest descent direction by checking all neighboring points at a fixed step-size. An important consideration is that these approaches each require a different level of computational effort to produce a single sample. In order to account for these additional costs we account for the expected number of integrand evaluations per sample (δ_M) by considering the *cost-adjusted variance* for a given estimator $\mathcal{I}_M^N$, defined as $\bar{v}_M \doteq \delta_M N \mathbb{V}\left(\mathcal{I}_M^N\right)$. Additionally, to ensure a meaningful comparison across experiments, we normalize this value by taking its ratio between the variance of the vanilla importance sampling approach to give the *relative cost-adjusted variance* given by $\bar{v}_M/\bar{v}_{IS}$. Here, a value of 0.5 indicates a 2x reduction in the number of integrand evaluations needed to attain the same error as an importance sampling estimate.[2]

For our comparisons we considered two different integration tasks, first a Bayesian k-means posterior, and finally a Bayesian posterior for a robot localization task.

4.1 Bayesian k-mixture Model

Consider the task of approximating the normalization constant ($\mathcal{Z}$), or model evidence, for a Bayesian k-mixture model. Specifically, we define the a generative model with k uniformly weighted multivariate normal distributions in $\mathbb{R}^d$ with fixed diagonal covariance matrices $\Sigma_i = \frac{i}{20}I$ for $i = \{1, ..., k\}$. The unobserved latent variables for this model are the means for each component $\mu_i \in \mathbb{R}^d$ which are assumed to be drawn from a multivariate normal prior with mean zero and identity covariance. Given n samples, $y_1, ..., y_n$, from the underlying model, the model evidence is given by integrating the un-normalized posterior

$$\mathcal{Z} = \int \prod_{i=1}^{n} \mathcal{L}(\mu_1, ..., \mu_k | y_i) p(\mu_1, ..., \mu_k) d\mu_1 ... d\mu_k,$$

where the likelihood function is given by $\mathcal{L}(\mu_1, ..., \mu_k|y_i) = \sum_{j=1}^{k} \frac{1}{k}\mathcal{N}(y_i; \mu_j, \Sigma_j)$ and the prior density the standard normal $p(\mu_1, ..., \mu_k) = \mathcal{N}([\mu_1, ..., \mu_k]; 0, I)$, where $[\mu_1, ..., \mu_k]$ denotes a dk-dimensional vector of "stacked" μ_i vectors. Using the same notation as previous sections we may write $\hat{\pi}(x) = \prod_{i=1}^{n} \mathcal{L}(x|y_i)p(x)$, where $x = [\mu_1, ..., \mu_k]$.

For the AIS approach we used 150 annealing distributions set using the "power of 4" heuristic suggested by [5], i.e. $\beta_i = ((150 - i)/150)^4$. Each annealing stage used 3 MCMC transitions, here we experimented with both slice sampling [8] and Hamiltonian transitions [9]. The Hamiltonian moves were tuned to achieve a accept/reject rate of about 80% which resulted in a step-size parameter of 0.003

[2] Note that in our analysis we do not apply additional costs for gradient evaluations since, in most settings, computations of $h(x)$ and $\nabla h(x)$ typically share the same sub-computations which can be cached and reused.

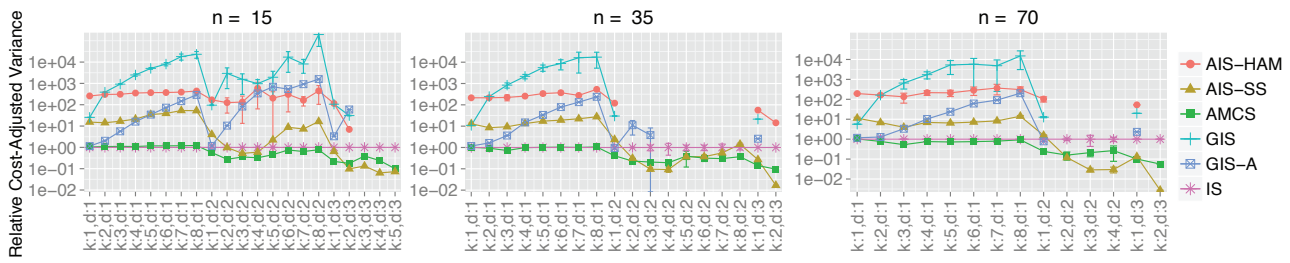

Figure 2: Cost-adjusted variance (log scale) for the different approaches on the Bayesian k-means task. Missing data points are due to the fact that trials where the final estimate (empirical mean) is incorrect by a factor of 2 or greater are automatically removed. From left to right the three plots indicate performance on the same problem but with an increasing number of observed training samples 15, 35, and 70 respectively.

and 5 leapfrog steps. Additionally, for AIS and the remaining methods we use the prior as the proposal density, $\pi_0 = p$, and the posterior as the target.

For AMCS we used Langevin local moves with monotonic, symmetrizing, and threshold acceptance functions. For the Langevin moves we used a step-size parameter $\varepsilon = 0.015$ and $\sigma^2 = 3\text{E}^{-5}$. The threshold acceptance functions were configured using a preliminary sampling approach. In particular, we let $f = \hat{\pi}$ and set the threshold parameter to a value that accepted roughly 1.5% of the data points on a small sub-sample (2000 points). These points were not used in the final estimate but in practice they can be incorporated without adverse effects. For the GIS approach we used step-size parameter $\varepsilon = 0.015$, also, we experimented with a modified version (GIS-A) by incorporating an acceptance function borrowed from the AMCS approach.

The results for this problem are shown in Fig. 2 as the number of "training" points (n), the dimensionality of these points (d), and number of mixture components (k) are altered. For each of these different settings the parameters for the sampling approaches remain fixed. Simulations were run for a period of 8 hours for each method and each setting of d, n, and k giving a total running time of 106 CPU days running on a cluster with 2.66GHz processors. However, even in this time many of the methods were not able to return a meaningful estimate after execution, these results are therefore omitted from the figure.

It is clear from these simulations that GIS (both variants) and AIS with Hamiltonian moves (AIS-HAM) are simply not effective for this task. While the AIS approach with slice sampling moves (AIS-SS) and the AMCS approach had more varied performance. In particular, the experiments indicate that AIS-SS can offer tremendous savings over both IS and AMCS for higher dimensional problems and problems with more training samples. However, this advantage comes at a price as the method performed up to 10-20x worse than even simple importance sampling in cases where the proposal was remotely close to the target. AMCS, on the other hand, was considerably more robust to changes in the target since for each setting it performed at least as good as vanilla importance sampling while offering a considerable advantage in more challenging settings.

To summarize, depending on the problem at hand, and the practitioner's appetite for risk, the most appropriate approach for this problem is likely either AMCS or AIS-SS. However, in many cases the practitioner may be interested in a large set of potential problem settings where it is too labor intensive to determine which method, and parameter settings, are most appropriate for each case. In such scenarios it may be worthwhile to consider an *adaptive* approach to select approaches automatically. In particular, recent work has shown that the task of allocating computation to a fixed set of Monte Carlo estimators with the goal of minimizing the variance reduces to the well known *stochastic multi-armed bandit* setting for which many effective adaptation schemes exist [10, 2]. Adaptive approaches of this form highlight the advantages of having a diverse suite of Monte Carlo integration approaches.

4.2 Problem 2: Robot Localization

We next consider approximating the normalization constant of a Bayesian posterior for the (simulated) *kidnapped robot problem* [16] where an autonomous robot is placed at an unknown location and must recover its position using relative sensors, such as a laser range finder, and a known map. This posterior distribution is notoriously difficult to work with when the sensors are highly accurate which creates a highly peaked distribution; a phenomenon referred to as *the curse of accurate sensors*. Here, we assume the prior distribution over the robot's (x,y) position and orientation, denoted $x \in \mathbb{R}^3$, is a uniform distribution.

In our simulations the robot's observations are given by a laser range finder which returns distance measurements at n positions spaced evenly in a 360° field of view (see Fig. 3). The sensor model for each individual sensor, that is, the likelihood of observing a measurement y given the true ray-traced distance from position x: $d(x)$, is given by the mixture $\mathcal{L}(y|d(x)) = 0.95\mathcal{N}(y; d(x), \sigma^2) +$

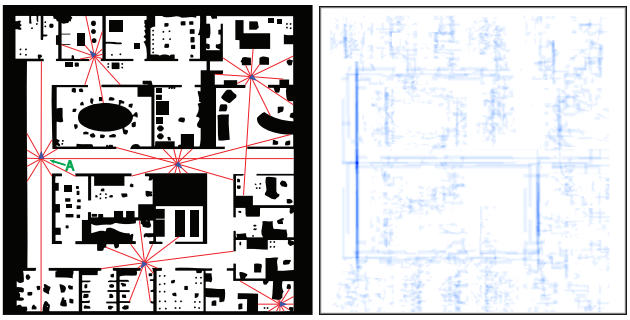

Figure 3: Left, the map used for the robot simulator with 6 different robot poses and corresponding laser measurements (for $n = 12$). Right, a 2d image where %blue is proportional to the log-likelihood function using the observations shown at position 'A', here pixel locations correspond to robot (x, y) position while the orientation remains fixed.

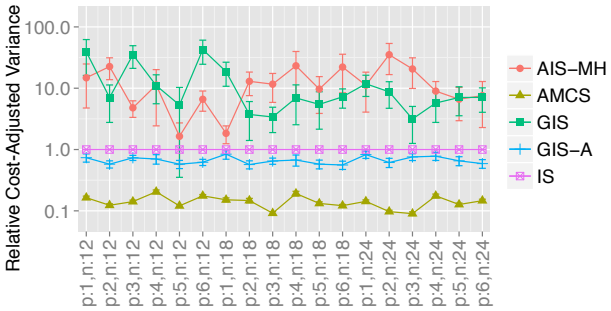

Figure 4: Relative cost-adjusted variance for the different approaches on the robot localization task for 6 different positions (p) and 3 different laser configurations (n = #laser readings).

$0.05\mathcal{U}(y; 0, M)$, where $\sigma^2 = 4$cm and the maximum ray length $M = 25$m.[3] This sensor model is used commonly in the literature (see [16]) and is meant to capture the noise inherent in laser measurements (normal distribution) as well as moving obstacles or failed measurements (uniform distribution). Given a set of observed measurements $y_1, ..., y_n$ then, we have the un-normalized posterior distribution $\hat{\pi}(x) = \prod_{i=1}^{n} \mathcal{L}(y_i|d_i(x))p(x)$, where p denotes the density of the uniform prior.

The log-posterior distribution for a fixed observation and orientation is shown in the right of Fig. 3 and a similar 3d plot in Fig. 1. This distribution poses challenges for Monte Carlo integration approaches because it is highly multimodal and individual integrand values require an expensive ray-tracing procedure to compute, which underscores the importance of efficient sampling approaches. Additionally, due to the sharp map edges and properties of the observation model, the posterior distribution is highly non-continuous and non-differentiable. This prevents the use of gradient-based local moves (for AMCS and AIS) and severely limits the effectiveness of annealing.

For this problem we experimented with AIS using 100 annealing distributions each featuring 3 Metropolis-Hastings MCMC steps with proposal $q(x, \cdot) = \mathcal{N}(x, \sigma^2 I)$ with $\sigma^2 = 4$cm. For AMCS we used the prior as a proposal density, *linear* Markov kernels with $v = [2\text{cm}, 2\text{cm}, 0.2\text{cm}]$ and $\sigma^2 = 2\text{E}^{-3}$cm and threshold acceptance function with threshold set to be larger than 4% of points on a 2000 pont sub-sample. For GIS we used the same proposal, step-sizes (v), and (optionally) the same threshold acceptance function as AMCS. IS used the prior for a proposal.

The error rates for the different sampling approaches for 6 different positions (see Fig. 3) and 3 different laser configurations, $n = 12, 18, 24$, are shown in Fig. 4. Unlike the previous task the results here are fairly straightforward and indicate that AMCS consistently offers an 8 to 10 times improvement over vanilla importance sampling. The cost-adjusted variance of the GIS approach can be significantly improved through the use of threshold acceptance functions but only marginally better than IS. Also, it is clear that AIS is simply not an effective approach for this task as it is roughly 10 times less efficient than simple IS and 100 times less efficient than AMCS. This is primarily due to the fact that the unmodified proposal density has some reasonable chance of landing near a region of some likelihood. Consequently, taking a large number of MCMC transitions is not a cost-effective way to improve the proposal, this detail exacerbated by landscape of the posterior distribution which inhibits efficient MCMC mixing.

[3] Measurements assume that the map (Fig. 3) is 10x10 meters.

5 Conclusion

We have introduced an alternative importance sampling approach that, like sequential Monte Carlo sampling, augments a fixed proposal density through the addition of local Markov chains. The approach differs from existing SMCS techniques in two fundamental ways: first, through the inclusion of fixed stopping rules for the Markov chains, and second, through the simulation of two antithetic chains from each point. The resulting estimator is unbiased and can be shown to have reduced variance through a straightforward analysis stemming from the law of total variance. The same analysis provides insight into the use of antithetic Markov transitions that lead to large changes in the value of the integrand, such as gradient ascent/descent moves.

We evaluated the performance of the proposed approach on two real-world machine learning tasks, where significant improvements over the state of the art could be observed under common conditions. This work provides a useful alternative to existing Monte Carlo integration approaches that exhibits complementary strengths.

References

[1] Alexandros Beskos, Ajay Jasra, and Alexandre Thiery. On the convergence of adaptive sequential Monte Carlo methods. *arXiv preprint arXiv:1306.6462*, 2013.

[2] S Bubeck, N Cesa-Bianchi, et al. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and trends in machine learning*, 5(1):1–122, 2012.

[3] P. Del Moral, A. Doucet, and A. Jasra. Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68:411–436, 2006.

[4] A. Doucet, N. de Freitas, and N. Gordon. *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, 2001.

[5] M. Kuss and C. Rasmussen. Assessing approximate inference for binary Gaussian process classification. *The Journal of Machine Learning Research*, 6:1679–1704, 2005.

[6] J. Liu. *Monte Carlo Strategies in Scientific Computing*. Springer, 2001.

[7] R. Neal. Annealed importance sampling. Technical report, University of Toronto, 2001.

[8] R. Neal. Slice sampling. *Annals of Statistics*, pages 705–741, 2003.

[9] R. Neal. *Handbook of Markov Chain Monte Carlo*, chapter MCMC using Hamiltonian dynamics. Chapman & Hall / CRC Press, 2010.

[10] J. Neufeld, A. György, D. Schuurmans, and Cs. Szepesvári. Adaptive Monte Carlo via bandit allocation. In *International Conference on Machine Learning (ICML)*, 2014.

[11] James R Norris. *Markov chains*. Cambridge University Press, 1998.

[12] M. Powell and J. Swann. Weighted uniform sampling, a Monte Carlo technique for reducing variance. *IMA Journal of Applied Mathematics*, 1966.

[13] C. Robert. *Handbook of Computational Statistics (revised)*, chapter 11, Bayesian Computational Methods, http://arxiv.org/abs/1002.2702. Springer, 2010.

[14] F. Southey, D. Schuurmans, and A. Ghodsi. Regularized greedy importance sampling. In *Advances in Neural Information Processing Systems*, 2002.

[15] S. Thrun, W. Burgard, and D. Fox. Monte Carlo localization with mixture proposal distribution. In *AAAI Conference on Artificial Intelligence*, 2000.

[16] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. MIT Press, 2005.

Appendix

A Proof of Lemma 1

$$\begin{aligned}
\mathbb{E}\left[\frac{h(X')\pi(X')}{\pi(X)}\right] &= \int\int \frac{h(x')\pi(x')}{\pi_0(x)} K(x,x')\pi_0(x)dx'dx \\
&= \int\int h(x')\pi(x')K(x,x')dxdx' \qquad \text{(by Fubini's theorem)} \\
&= \int h(x')\pi(x') \int K(x',x)dxdx' \qquad \text{(by symmetry of } K\text{)} \\
&= \int h(x')\pi(x')dx'.
\end{aligned}$$

Here Fubini's theorem makes use of the facts that, since K is a Markov kernel, $\forall A \in \mathcal{B}$, $K(\cdot, A)$ is measurable and $\forall x \in \mathcal{X}, K(x,\cdot)$ is a probability measure (thus measurable). See Theorem 6.4.2, [11] for details.

B Lemma 4

We now consider the p.d.f. for a random element chosen uniformly from a set of random variables (i.e. $X^{(i)}_{J^{(i)}}$).

Lemma 4. *Given random variables $(X_0, ..., X_n)$ distributed according to a joint density g, and a random variable $J \in \{0, ..., n\}$ such that $\mathbb{P}\{J = j\} = 1/(n-1)\ \forall_{j\in\{1,...,n-1\}}$ and $\mathbb{P}\{J = 0\} = \mathbb{P}\{J = n\} = 0$, the variable $Y = \sum_{j=0}^{n} \mathbb{I}\{J = j\}X_j$ has p.d.f. $p(y) = \frac{1}{n-1}\sum_{j=1}^{n-1} g_j(y)$ where g_j is the j^{th} marginal density of g.*

Proof. For any bounded measurable function f we have

$$\mathbb{E}[f(Y)] = \mathbb{E}[\mathbb{E}[f(Y)|J]] = \sum_{j=0}^{n} \mathbb{P}\{J = j\}\,\mathbb{E}[f(X_j)] = \sum_{j=1}^{n-1} \frac{1}{n-1}\mathbb{E}[f(X_j)] = \frac{1}{n-1}\sum_{j=1}^{n-1}\int f(x)g_j(x)dx = \int f(x)p(x)dx,$$

where $p(x) = \frac{1}{n-1}\sum_{j=1}^{n-1} g_j(x)$. □

C Proof of Lemma 2

Letting $m' = m_0 - k$ for any integer offset k such that $0 < k + m_0 < m$, observe that

$$\begin{aligned}
\gamma(m, m_0, x_0, ..., x_m) &= (1 - A_-(x_1, x_0))K_-(x_1, x_0) \prod_{j=2}^{m_0} A_-(x_j, x_{j-1})K_-(x_j, x_{j-1}) \\
&\quad (1 - A_+(x_{m-1}, x_m))K_+(x_{m-1}, x_m) \prod_{j=m_0}^{m-1} A_+(x_j, x_{j+1})K_+(x_j, x_{j+1}) \\
&= (1 - A_-(x_1, x_0))K_-(x_1, x_0) \prod_{j=2}^{m_0-k} A_-(x_j, x_{j-1})K_-(x_j, x_{j-1}) \\
&\quad (1 - A_+(x_{m-1}, x_m))K_+(x_{m-1}, x_m) \prod_{j=m_0-k}^{m-1} A_+(x_j, x_{j+1})K_+(x_j, x_{j+1}) \\
&= \gamma(m, m_0 - k, x_0, ..., x_m),
\end{aligned}$$

where the second equality follows from the fact that (K_+, A_+) and (K_-, A_-) are jointly symmetric.

D Proof of Lemma 3

To prove the theorem we make use of Lemma 2, which asserts that for any fixed sequence of points one can shift the "starting index" m_0 without altering the value of probability density γ. The proof then amounts to a straightforward reindexing of the summations in K. Observe that

$$
\begin{aligned}
K(x, x') &= \sum_{m=2}^{\infty} \frac{1}{m-1} \sum_{m_0=1}^{m-1} \sum_{j=1, j\neq m_0}^{m-1} \gamma_j(m, m_0, x, x') \\
&= \sum_{m=2}^{\infty} \frac{1}{m-1} \sum_{m_0=1}^{m-1} \sum_{j=1, j\neq m_0}^{m-1} \int \gamma(m, m_0, \bar{x}^{(m_0=x, j=x')}) d\bar{x}^{\setminus\{m_0, j\}} \\
&= \sum_{m=2}^{\infty} \frac{1}{m-1} \sum_{j=1}^{m-1} \sum_{m_0=1, m_0\neq j}^{m-1} \int \gamma(m, m_0, \bar{x}^{(m_0=x, j=x')}) d\bar{x}^{\setminus\{m_0, j\}} \\
&= \sum_{m=2}^{\infty} \frac{1}{m-1} \sum_{m_0=1}^{m-1} \sum_{j=1, j\neq m_0}^{m-1} \int \gamma(m, j, \bar{x}^{(m_0=x', j=x)}) d\bar{x}^{\setminus\{m_0, j\}} && \text{(by renaming } j \text{ and } m_0\text{, i.e. swapping them)} \\
&= \sum_{m=2}^{\infty} \frac{1}{m-1} \sum_{m_0=1}^{m-1} \sum_{j=1, j\neq m_0}^{m-1} \int \gamma(m, m_0, \bar{x}^{(m_0=x', j=x)}) d\bar{x}^{\setminus\{m_0, j\}} && \text{(by Lemma 2 where } k = j - m_0\text{)} \\
&= \sum_{m=2}^{\infty} \frac{1}{m-1} \sum_{m_0=1}^{m-1} \sum_{j=1, j\neq m_0}^{m-1} \gamma_j(m, m_0, x', x) \\
&= K(x', x),
\end{aligned}
$$

which establishes the result.