Journal of the American Statistical Association

ISSN: 0162-1459 (Print) 1537-274X (Online) Journal homepage: https://www.tandfonline.com/loi/uasa20

Informed Proposals for Local MCMC in Discrete Spaces

Giacomo Zanella

To cite this article: Giacomo Zanella (2020) Informed Proposals for Local MCMC in Discrete Spaces, Journal of the American Statistical Association, 115:530, 852-865, DOI: 10.1080/01621459.2019.1585255

To link to this article: https://doi.org/10.1080/01621459.2019.1585255

JOURNAL OF THE AMERICAN STATISTICAL ASSOCIATION
2020, VOL. 115, NO. 530, 852–865: Theory and Methods
https://doi.org/10.1080/01621459.2019.1585255

Informed Proposals for Local MCMC in Discrete Spaces

Giacomo Zanella

Department of Decision Sciences, BIDSA and IGIER, Bocconi University, Milan, Italy

ABSTRACT

There is a lack of methodological results to design efficient Markov chain Monte Carlo (MCMC) algorithms for statistical models with discrete-valued high-dimensional parameters. Motivated by this consideration, we propose a simple framework for the design of informed MCMC proposals (i.e., Metropolis–Hastings proposal distributions that appropriately incorporate local information about the target) which is naturally applicable to discrete spaces. Using Peskun-type comparisons of Markov kernels, we explicitly characterize the class of asymptotically optimal proposal distributions under this framework, which we refer to as *locally balanced* proposals. The resulting algorithms are straightforward to implement in discrete spaces and provide orders of magnitude improvements in efficiency compared to alternative MCMC schemes, including discrete versions of Hamiltonian Monte Carlo. Simulations are performed with both simulated and real datasets, including a detailed application to Bayesian record linkage. A direct connection with gradient-based MCMC suggests that locally balanced proposals can be seen as a natural way to extend the latter to discrete spaces. Supplementary materials for this article are available online.

ARTICLE HISTORY
Received December 2017
Accepted January 2019

KEYWORDS
Bayesian discrete models; Bayesian record linkage; Informed Metropolis–Hastings schemes; Markov chain Monte Carlo; Peskun ordering.

1. Introduction

Markov chain Monte Carlo (MCMC) algorithms are one the most widely used methodologies to sample from complex and intractable probability distributions, especially in the context of Bayesian statistics (Robert and Casella 2005). Given a distribution of interest $\pi(x)$ on a discrete state space $\mathcal{X}$, MCMC methods simulate a Markov chain $\{X_t\}_{t=1}^{\infty}$ having π as stationary distribution and then use the states visited by X_t as Monte Carlo samples from π. Under mild assumptions, the Ergodic theorem guarantees that the resulting sample averages are consistent estimators for arbitrary expectations under π. Many MCMC schemes used in practice fall within the Metropolis–Hastings (MH) framework (Metropolis et al. 1953; Hastings 1970). Given a current state $x \in \mathcal{X}$, the MH algorithm samples a proposed state y according to some proposal distribution $Q(x,\cdot)$ and then accepts it with probability $a(x,y) = \min\left\{1, \frac{\pi(y)Q(y,x)}{\pi(x)Q(x,y)}\right\}$ or otherwise rejects it and stays at x. The resulting transition kernel

$$P(x,y) = Q(x,y)a(x,y) + \delta_x(y)\sum_{z\in\mathcal{X}}(1-a(x,z))Q(x,z)$$

is π-reversible and can be used for MCMC purposes. Although the MH algorithm can be applied to virtually any target distribution, its efficiency depends drastically on the proposal distribution Q and its interaction with the target π. Good choices of Q will speed up the Markov chain's convergence while bad choices will slow it down in a potentially dramatic way.

1.1. Random Walk Versus Informed Schemes

Random walk MH schemes use symmetric proposal distributions satisfying $Q(x,y) = Q(y,x)$, such as uniform distributions over neighboring states. Although these schemes are easy to implement, the new state y is proposed "blindly" (i.e., using no information about π) and this can lead to bad mixing and slow convergence. In continuous frameworks, where $\mathcal{X} = \mathbb{R}^n$ and $\pi(x)$ are a probability density function, various *informed* MH proposal distributions have been designed to obtain better convergence than continuous RW schemes. For example the Metropolis-adjusted Langevin algorithm (MALA, e.g., Roberts and Rosenthal 1998) exploits the gradient of the target to bias the proposal distribution toward high probability regions by setting $Q_\sigma(x,\cdot) = N(x + \frac{\sigma^2}{2}\nabla(\log\pi)(x), \sigma^2\mathbb{I}_n)$, where $N(x,\sigma^2\mathbb{I}_n)$ denotes a normal distributions centered at x with isotropic variance $\sigma^2\mathbb{I}_n$. The MALA proposal is derived by discretizing the π-reversible Langevin diffusion X_t given by $dX_t = \frac{\sigma^2}{2}\nabla(\log\pi)(X_t)dt + \sigma\, dB_t$, so that Q_σ is approximately π-reversible for small values of σ. More elaborate gradient-based informed proposals have been devised, such as Hamiltonian Monte Carlo (HMC, e.g., Neal 2011; Girolami and Calderhead 2011), and other schemes (Welling and Teh 2011; Titsias and Papaspiliopoulos 2018; Durmus et al. 2017), resulting in substantial improvements of MCMC performances both in theory and in practice. However, most of these proposal distributions are derived as discretization of continuous-time diffusion processes or measure-preserving flows, and are based on derivatives and Gaussian distributions. Currently, it is not clear how to appropriately extend such methods to frameworks where $\mathcal{X}$ is a discrete space. As a consequence, practitioners using MCMC to target measures on discrete spaces often rely on symmetric and uninformed proposal distributions, which can induce slow convergence.

CONTACT Giacomo Zanella giacomo.zanella@unibocconi.it Department of Decision Sciences, Bocconi University, Via Roentgen 1, 20136, Milan, Italy.
Supplementary materials for this article are available online. Please go to www.tandfonline.com/r/JASA.

1.2. Informed Proposals in Discrete Spaces

A simple way to circumvent the problem described above is to map discrete spaces to continuous ones and then apply informed schemes in the latter, typically using HMC (Zhang et al. 2012; Pakman and Paninski 2013; Nishimura, Dunson, and Lu 2017). Although useful in some scenarios, the main limitation of this approach is that the embedding of discrete spaces into continuous ones is not always feasible and can potentially destroy the natural topological structure of the discrete space under consideration (e.g., spaces of trees, partitions, permutations,…), thus resulting in highly multimodal and irregular target distributions that are hard to explore. An alternative approach was recently proposed in Titsias and Yau (2017), where informed proposals are obtained by introducing auxiliary variables and performing Gibbs sampling in the augmented space. The resulting scheme, named the Hamming ball sampler, requires no continuous space embedding and is directly applicable to generic discrete spaces, but the potentially strong correlation between the auxiliary variables and the chain state can severely slow down convergence.

In this work we formulate the problem of designing informed MH proposal distributions in an original way. Our formulation has the merit of being simple and naturally applicable to discrete space. The theoretical results hint to a simple and practical class of informed MH proposal distributions that are well designed for high-dimensional discrete problems, which we refer to as *locally balanced proposals*. Experiments on both simulated and real data show orders of magnitude improvements in efficiency compared to both random walk MH and the alternative informed schemes described above.

1.3. Paper Structure

In Section 2, we define the class of informed proposal distributions considered (which are obtained as a product of some "base" uninformed kernel and a multiplicative biasing term) and provide a first heuristic discussion on how to choose appropriate biasing terms. In Section 3, we characterize, under regularity assumptions on the target, the asymptotically optimal class of biasing terms in terms of Peskun ordering as the dimensionality of the state space increases. We also provide some easy-to-verify sufficient conditions for the regularity assumptions to hold. In Section 4.1, we consider a simple binary target distribution in order to compare different proposals within the class of asymptotically optimal ones and identify the one leading to the smallest mixing time, which turns out to be related to the Barker's algorithm (Barker 1965), while in Section 4.2 we discuss generic guidelines for practical implementations. Section 5 discusses the connection with classical gradient-based MCMC and MALA in particular. In Section 6, we perform simulation studies on classic discrete models (permutation spaces and Ising model), while in Section 7 we consider a more detailed application to Bayesian record linkage problems. Finally, in Section 8, we discuss possible extensions and future works. Supplementary material includes proofs, additional details on the simulations studies and R code to reproduce simulations.

2. Locally Balanced Proposals

Let π be a target probability distribution on some finite state space $\mathcal{X}$, and $K(x,y)$ be the uninformed symmetric kernel that we would use to generate proposals in a random walk MH scheme, such as the uniform distribution over neighboring states. Suppose that we want to modify $K(x,y)$ and incorporate information about π in order to bias the proposal toward high-probability states. In this work, we will consider a specific class of informed proposal distributions, which we refer to as *pointwise informed* proposals. These proposals have the following structure:

$$Q_g(x,y) \quad = \quad \frac{g\left(\frac{\pi(y)}{\pi(x)}\right)K(x,y)}{Z_g(x)}, \tag{1}$$

where g is a continuous function from $(0,\infty)$ to itself and $Z_g(x)$ is the normalizing constant

$$Z_g(x) = \sum_{z\in\mathcal{X}} g\left(\frac{\pi(z)}{\pi(x)}\right)K(x,z)\,. \tag{2}$$

The distribution Q_g in (1) inherits the topological structure of K and incorporates information regarding π through the multiplicative term $g\left(\frac{\pi(y)}{\pi(x)}\right)$. Although the scheme in Equation (1) is not the only way to design informed MH proposals, it is an interesting framework to consider. In particular it includes the uninformed choice $Q(x,y) = K(x,y)$ when $g(t) = 1$, and the localized version of π, $Q(x,y) \propto K(x,y)\pi(y)$ when $g(t) = t$. Given Equation (1), the question of interest is how to choose the function g.

2.1. Heuristics: Local Moves Versus Global Moves

In this section, we provide some heuristic arguments on why a good choice of g depends substantially on whether we consider a "local move" or a "global move" regime. To facilitate the discussion, we assume that the proposal has a scale parameter σ such that $K_\sigma(x,y)$ converges weakly to the delta measure in x as $\sigma \downarrow 0$ while it converges to the uniform distribution on $\mathcal{X}$ as $\sigma \uparrow \infty$. See Remark 1 for more discussion on such assumptions.

Consider first the apparently natural choice of using the localized version of π:

$$Q_\pi(x,y) = \frac{\pi(y)K_\sigma(x,y)}{\varphi_\sigma(x)}\,, \tag{3}$$

where $\varphi_\sigma(x)$ is the normalizing constant. This coincides with the pointwise informed proposal based on $g(t) = t$. Equation (3) and the symmetry of K_σ implies that

$$\frac{Q_\pi(x,y)}{\pi(y)\varphi_\sigma(y)} = \frac{Q_\pi(y,x)}{\pi(x)\varphi_\sigma(x)}\,,$$

which means that Q_π is reversible with respect to $\pi(x)$
$\varphi_\sigma(x)/\sum_{y\in\mathcal{X}}\pi(y)\varphi_\sigma(y)$. Note that $\varphi_\sigma(x)$ coincides with the convolution between π and K_σ which we denote by $\varphi_\sigma(x) = (\pi * K_\sigma)(x) = \sum_{y\in\mathcal{X}}\pi(y)K_\sigma(x,y)$. Therefore, from the assumptions on K_σ, we have that $\varphi_\sigma(x)$ converges to $\pi(x)$ as $\sigma \downarrow 0$, while it converges to $1/|\mathcal{X}|$ as $\sigma \uparrow \infty$. It follows that the

invariant measure of Q_π looks very different in the two opposite limiting regimes because

$$\frac{\pi(x)\varphi_\sigma(x)}{\sum_{y\in\mathcal{X}}\pi(y)\varphi_\sigma(y)} \to \begin{cases} \pi(x) & \text{if } \sigma \uparrow \infty \text{ (Global moves)} \\ \pi(x)^2/\sum_{y\in\mathcal{X}}\pi(y)^2 & \text{if } \sigma \downarrow 0 \quad \text{(Local moves)} \end{cases}.$$

Therefore, for big values of σ, Q_π will be approximately π-reversible and thus it would be a good proposal distribution for the MH algorithm. For this reason, we refer to Q_π as *globally balanced* proposal. On the contrary, for small values of σ, Q_π would *not* be a good MH proposal because its invariant distribution converges to $\pi(x)^2$ which is potentially very dissimilar from the target $\pi(x)$.

Following the previous arguments, it is easy to correct for this behavior with a different choice of g. For example, considering $g(t) = \sqrt{t}$, we obtain the proposal

$$Q_{\sqrt{\pi}}(x,y) = \frac{\sqrt{\pi(y)}K_\sigma(x,y)}{(\sqrt{\pi} * K_\sigma)(x)}.$$

Arguing as before it is trivial to see that $Q_{\sqrt{\pi}}$ is reversible with respect to $\sqrt{\pi(x)}(\sqrt{\pi} * K_\sigma)(x)$, which converges to $\pi(x)$ as $\sigma \downarrow 0$. We thus refer to $Q_{\sqrt{\pi}}$ as an example of *locally balanced proposal* with respect to π, according to the following definition.

Definition 1. (Locally balanced kernels) A family of Markov transition kernels $\{Q_\sigma\}_{\sigma>0}$ is *locally balanced* with respect to a distribution π if each Q_σ is reversible with respect to some distribution π_σ such that π_σ converges weakly to π as $\sigma \downarrow 0$.

The choice $g(t) = \sqrt{t}$ is not the only one leading to locally balanced proposals, as shown by the following theorem.

Theorem 1. A pointwise informed proposal Q_g is locally-balanced with respect to a general π if and only if

$$g(t) = t\,g(1/t) \qquad \forall\, t > 0\,. \tag{4}$$

Motivated by Theorem 1 we refer to functions g satisfying Equation (4) as *balancing functions*. One would expect locally balanced kernels to be suitable MH proposal distributions when targeting π in a local move regime (i.e., for small σ). Intuitively, if Q_σ is approximately π-reversible, the MH correction has less job to do (namely correcting for the difference between π_σ and π) and thus the algorithm can propose longer moves without rejecting them, thus improving the algorithm's efficiency. This is, for example, the intuition behind the MALA algorithm discussed in Section 1.1, which is designed so that Q_σ is approximately π-reversible for small values of σ. This intuition is confirmed by the results presented in the next section, which show that locally balanced proposals produce MH algorithms that are asymptotically optimal within the class of pointwise informed proposals when the space dimensionality increases.

Remark 1. The limiting assumption $\lim_{\sigma\downarrow 0} K_\sigma(x,\cdot) = \delta_x(\cdot)$, where $\delta_x(\cdot)$ denotes the delta measure in x, is more natural in continuous spaces, for example $\mathcal{X} = \mathbb{R}^d$ and $Q_\sigma(x,\cdot) = N(x,\sigma^2\mathbb{I}_n)$, rather than discrete ones. In fact for many (but not all) base kernels K_σ in discrete spaces, it holds $K_\sigma(x,\cdot) = \delta_x(\cdot)$ for all σ below some threshold. Thus, a value of σ above the threshold must be used to avoid degenerate proposal distributions. However, even in such cases, the practically used values of σ are small enough for the relevant aspects of the kernel behavior to be captured by the $\sigma \downarrow 0$ asymptotics. In particular, as we will see below, the key aspect is the behavior of $(\pi * K_\sigma)(x)$ for small values of σ, which is the same in continuous and discrete spaces.

3. Peskun Optimality of Locally Balanced Proposals

In this section, we use Peskun ordering to compare the efficiency of MH schemes generated by pointwise informed proposals defined in Equation (1). Unlike Section 2.1, where we considered the local limit $\sigma \downarrow 0$, we now consider a scenario with fixed base kernel $K(x,y)$.

3.1. Background on Peskun Ordering

Peskun ordering provides a comparison result for Markov chains convergence properties. It measure the efficiency of MCMC algorithms in terms of *asymptotic variance* and *spectral gap*. The notion of asymptotic variance describes how the correlation among MCMC samples affects the variance of the sample mean estimators. Given a π-stationary transition kernel P and a function $h : \mathcal{X} \to \mathbb{R}$, the asymptotic variance $\mathrm{var}_\pi(h,P)$ is defined as

$$\begin{aligned}\mathrm{var}_\pi(h,P) &= \lim_{T\to\infty} T\,\mathrm{var}\left(\frac{\sum_{t=1}^T h(X_t)}{T}\right) \\ &= \lim_{T\to\infty} T^{-1}\,\mathrm{var}\left(\sum_{t=1}^T h(X_t)\right),\end{aligned}$$

where $X_1, X_2, \ldots$is a Markov chain with transition kernel P started in stationarity (i.e., with $X_1 \sim \pi$) . The smaller $\mathrm{var}_\pi(h,P)$ the more efficient the corresponding MCMC algorithm is in estimating the expectation of h under π. The spectral gap of a Markov transition kernel P is defined as $\mathrm{Gap}(P) = 1-\lambda_2$, where λ_2 is the second largest eigenvalue of P, and always satisfies $\mathrm{Gap}(P) \geq 0$. The value of $\mathrm{Gap}(P)$ is closely related to the convergence properties of P, with values close to 0 corresponding to slow convergence and values distant from 0 corresponding to fast convergence (see, e.g., (Levin, Peres, and Wilmer 2009, Ch.12-13) for a review of spectral theory for discrete Markov chains).

Theorem 2. Let P_1 and P_2 be π-reversible Markov transition kernels on $\mathcal{X}$ such that $P_1(x,y) \geq c\,P_2(x,y)$ for all $x \neq y$ and a fixed $c > 0$. Then it holds

$$\begin{aligned}&(a) \quad \mathrm{var}_\pi(h,P_1) \leq \frac{\mathrm{var}_\pi(h,P_2)}{c} \\ &\qquad\qquad + \frac{1-c}{c}\mathrm{var}_\pi(h) \qquad \forall h : \mathcal{X} \to \mathbb{R}, \\ &(b) \quad \mathrm{Gap}(P_1) \geq c\,\mathrm{Gap}(P_2),\end{aligned}$$

where $\mathrm{var}_\pi(h)$ denotes the variance of $h(X)$ with $X \sim \pi$.

The case $c = 1$ of Theorem 2 is known as Peskun ordering (Peskun 1973; Tierney 1998). Theorem 2 implies that if

$P_1(x,y) \geq c\,P_2(x,y)$ for all $x \neq y$, then P_1 is "c times more efficient" than P_2 in terms of spectral gap and asymptotic variances (ignoring the $\mathrm{var}_\pi(h)$ term which is typically much smaller than $\mathrm{var}_\pi(h, P_2)$ in non trivial applications).

3.2. Peskun Comparison Between Pointwise Informed Proposals

To state Theorem 3, we define the following constant:

$$c_g \quad = \quad \sup_{(x,y)\in \mathrm{R}} \frac{Z_g(y)}{Z_g(x)}\,, \tag{5}$$

where $\mathrm{R} = \{(x,y) \in \mathcal{X}\times\mathcal{X} : \pi(x)K(x,y) > 0\}$ and $Z_g(x)$ is defined by (2). By construction, it holds $c_g \geq 1$ because switching x and y the fraction in Equation (5) gets inverted.

Theorem 3. Let $g : (0,\infty) \to (0,\infty)$. Define $\tilde{g}(t) = \min\{g(t), t\,g(1/t)\}$ and let P_g and $P_{\tilde{g}}$ be the MH kernels obtained from the pointwise informed proposals Q_g and $Q_{\tilde{g}}$ defined as in (1). It holds

$$P_{\tilde{g}}(x,y) \quad \geq \quad \frac{1}{c_g c_{\tilde{g}}} P_g(x,y) \qquad \forall x \neq y\,. \tag{6}$$

The function $\tilde{g}(t) = \min\{g(t), t\,g(1/t)\}$ satisfies $\tilde{g}(t) = t\,\tilde{g}(1/t)$ by definition. Therefore, Theorems 2 and 3 imply that for any $g : \mathbb{R}_+ \to \mathbb{R}_+$ there is a corresponding balancing function $\tilde{g}$ which leads to a more efficient MH algorithm modulo the factor $\frac{1}{c_g c_{\tilde{g}}}$. As we discuss in the next section, in many models of interest $\frac{1}{c_g c_{\tilde{g}}}$ converges to 1 as the dimensionality of $\mathcal{X}$ increases. When the latter is true, we can deduce that locally balanced proposals are asymptotically optimal, in the sense that the potential improvement over locally balanced proposals goes to 0 as the dimensionality increases. Note that, even when $\frac{1}{c_g c_{\tilde{g}}}$ converges to 1, Theorem 3 does not imply that $P_{\tilde{g}}$ Peskun-dominates P_g for fixed target distributions, but rather that P_g cannot significantly improve over $P_{\tilde{g}}$ in large dimensions.

A similar approach can be used to obtain upper bounds on how much one can gain going from g to $\tilde{g}$. Define

$$b_g \quad = \quad \sup_{(x,y)\in \mathrm{R}} \frac{g(t_{xy})}{t_{xy}\,g(t_{yx})} \quad \geq \quad 1\,, \tag{7}$$

where t_{xy} denotes the ratio $\frac{\pi(y)}{\pi(x)}$. It holds $b_g \geq 1$ by the same argument used for c_g. The constant b_g represents how "unbalanced" the function g is: the bigger b_g the less balanced g is according to Equation (4) (if $b_g = 1$ then g satisfies Equation (4)). As shown by the following theorem, b_g provides an upper bound on the improvement achievable by using locally balanced proposals.

Theorem 4. Under the same assumptions and notation of Theorem 3, it holds

$$P_g(x,y) \quad \geq \quad \frac{1}{c_g c_{\tilde{g}} b_g} P_{\tilde{g}}(x,y) \qquad \forall x \neq y\,. \tag{8}$$

Theorem 4 implies that a locally balanced proposal $Q_{\tilde{g}}$ is at most $c_g c_{\tilde{g}} b_g$ times more efficient than the original proposal Q_g. In asymptotic regimes where $c_g c_{\tilde{g}}$ converges to 1, this means an asymptotic improvement upper bounded by b_g. The value of b_g depends also on how "rough" the target is. For example, if $g(t) = 1$ or $g(t) = t$, then $b_g = \sup_{(x,y)\in \mathrm{R}} \frac{\pi(y)}{\pi(x)}$, while if the target is uniform, then $b_g = 1$ for any g. This suggests that the improvement of balanced proposals will increase with the target's roughness, while it will be negligible if the target is close to uniform. This is supported by the simulation study in Section 6.

3.3. High-Dimensional Regime

Suppose now that the distribution of interest $\pi^{(n)}$ is indexed by a positive integer n which represents the dimensionality of the underlying state space $\mathcal{X}^{(n)}$. Similarly, also the base kernel $K^{(n)}$ and the constants $c_g^{(n)}$ defined by Equation (5) depend on n. In many discrete contexts, as the dimensionality goes to infinity, the size of a single move of $K^{(n)}$ becomes smaller and smaller with respect to the size of $\mathcal{X}^{(n)}$ and does not change significantly the landscape around the current location. In those cases, we expect the following to hold:

$$c_g^{(n)} \to 1 \qquad \text{as } n \to \infty \tag{A}$$

for every well-behaved g (e.g., bounded on compact subsets of $(0,\infty)$). When (A) holds, the factor $\frac{1}{c_g c_{\tilde{g}}}$ in the Peskun comparison of Theorem 3 converges to 1 and locally balanced proposals are asymptotically optimal, as discussed above.

We now provide an example of sufficient conditions for (A). We consider sample spaces with Cartesian products, $\mathcal{X}^{(n)} = \times_{i=1}^n \mathcal{X}_i$ where each $\mathcal{X}_i$ is a finite set with at most M elements, and a base kernel $K(x,\cdot) = \mathrm{Unif}(N(x))$ uniformly distributed over a ball of radius σ in the Hamming metric, $N(x) = B_\sigma(x)$. We denote by d_n the maximum degree of the conditional independence graph of $x_1, \ldots, x_n$ where $(x_1, \ldots, x_n) \sim \pi^{(n)}$.

Theorem 5. If $\frac{\pi^{(n)}(y)}{\pi^{(n)}(x)} \leq C$ for some $C < \infty$ and every $x \in \mathcal{X}$ and $y \in N(x)$, then

$$1 \leq c_g^{(n)} \leq 1 + \mathcal{O}\left(\frac{d_n+1}{n}\right) \qquad \text{as } n \to \infty\,. \tag{9}$$

Therefore, if $d_n/n \to 0$ as $n \to \infty$, (A) is satisfied.

The notation $c_g^{(n)} \leq 1 + \mathcal{O}\left(\frac{d_n+1}{n}\right)$ in Equation (9) stands for $\limsup_{n\to\infty} \frac{c_g^{(n)}-1}{(d_n+1)/n} < \infty$. Equation (9) implies

$$1 \;\geq\; \frac{1}{c_g^{(n)} c_{\tilde{g}}^{(n)}} \;\geq 1 - \mathcal{O}\left(\frac{d_n+1}{n}\right) \qquad \text{as } n \to \infty\,. \tag{10}$$

Combining Equation (10) with Theorem 3 we obtain bounds on the rate at which locally balanced proposals are asymptotically optimal.

Remark 2. One could replace $N(x) = B_\sigma(x)$ with $N(x) = B_\sigma(x)\backslash\{x\}$ or $N(x) = \{y \in \mathcal{X}^{(n)} : \|y - x\| = k\}$ for some positive integer k and the result of Theorem 5 would still be valid, with minor modifications of the proof required.

Consider the following two models, which will be used as illustrative examples in the following sections.

Example 1 (Independent binary components). Consider $\mathcal{X}^{(n)} = \{0,1\}^n$ and, denoting the elements of $\mathcal{X}^{(n)}$ as $\mathbf{x}_{1:n} = (x_1, \dots, x_n)$, the target distribution is

$$\pi^{(n)}(\mathbf{x}_{1:n}) = \prod_{i=1}^{n} p_i^{1-x_i}(1-p_i)^{x_i},$$

where each p_i is a probability value in $(0,1)$. The base kernel $K^{(n)}(\mathbf{x}_{1:n}, \cdot)$ is the uniform distribution on the neighborhood $N(\mathbf{x}_{1:n})$ defined as

$$N(\mathbf{x}_{1:n}) = \left\{ \mathbf{y}_{1:n} = (y_1, \dots, y_n) : \sum_{i=1}^{n} |x_i - y_i| = 1 \right\}.$$

Example 2 (Ising model). Consider the state space $\mathcal{X}^{(n)} = \{-1,1\}^{V_n}$, where (V_n, E_n) is the $n \times n$ square lattice graph with, for example, periodic boundary conditions. For each $\mathbf{x} = (x_i)_{i \in V_n}$, the target distribution is defined as

$$\pi^{(n)}(\mathbf{x}) = \frac{1}{Z} \exp\left(\sum_{i \in V_n} \alpha_i x_i + \lambda \sum_{(i,j) \in E_n} x_i x_j \right), \tag{11}$$

where $\alpha_i \in \mathbb{R}$ are biasing terms representing the propensity of x_i to be positive, $\lambda > 0$ is a global interaction term and Z is a normalizing constant. The neighboring structure is the one given by single-bit flipping

$$N(\mathbf{x}) = \left\{ \mathbf{y} \in \mathcal{X}^{(n)} : \sum_{i \in V_n} |x_i - y_i| = 2 \right\}. \tag{12}$$

In the context of Examples 1 and 2, condition (A) follows directly from Theorem 5.

Corollary 1. If $\inf_{i \in \mathbb{N}} p_i > 0$ and $\sup_{i \in \mathbb{N}} p_i < 1$, then Example 1 satisfies (A). Similarly for Example 2 if $\inf_{i \in V_n, n \in \mathbb{N}} \alpha_i > -\infty$ and $\sup_{i \in V_n, n \in \mathbb{N}} \alpha_i < \infty$.

Having a Cartesian product or a conditional independence graph with degree growing sublinearly in n is not necessary for (A) to hold, as shown by the following example.

Example 3 (Weighted permutations). Let

$$\pi^{(n)}(\rho) = \frac{1}{Z} \prod_{i=1}^{n} w_{i\rho(i)} \qquad \rho \in \mathcal{S}_n, \tag{13}$$

where $\{w_{ij}\}_{i=i,j}^{n}$ are positive weights, Z is the normalizing constant $\sum_{\rho \in \mathcal{S}_n} \prod_{i=1}^{n} w_{i\rho(i)}$ and $\mathcal{S}_n$ is the space of permutations of n elements (i.e., bijections from $\{1, \dots, n\}$ to itself). We consider local moves that pick two indices and switch them. The induced neighboring structure is $\{N(\rho)\}_{\rho \in \mathcal{S}_n}$ with

$$N(\rho) = \Big\{ \rho' \in \mathcal{S}_n : \rho' = \rho \circ (i,j) \text{ for some } i,j \in \{1, \dots, n\} \text{ with } i \neq j \Big\}, \tag{14}$$

where $\rho' = \rho \circ (i,j)$ is defined by $\rho'(i) = \rho(j)$, $\rho'(j) = \rho(i)$ and $\rho'(l) = \rho(l)$ for $l \neq i$ and $l \neq j$.

Proposition 1. If $\inf_{i,j \in \mathbb{N}} w_{ij} > 0$ and $\sup_{i,j \in \mathbb{N}} w_{ij} < \infty$, then Example 3 satsifies (A).

Example 1 is an illustrative toy example that we analyze explicitly in Section 4.1. Instead, the target measures in Examples 2 and 3 are nontrivial distributions occurring in many applied scenarios (see, e.g., Sections 6 and 7), and MCMC schemes are among the most commonly used approaches to obtain approximate samples from those. Such examples will be used for illustrations in Sections 6.2 and 6.3. Corollary 1 and Proposition 1, combined with Theorem 3, imply that for Examples 1–3 locally balanced proposal are asymptotically optimal within the class of pointwise informed proposals.

4. Optimal Choice of Balancing Function

In Section 3, we showed that, under the regularity assumption (A), locally balanced proposals are asymptotically optimal among pointwise informed proposals. It is thus natural to ask if there is an optimal proposal among the locally balanced ones or, equivalently, if there is an optimal balancing function g among the ones satisfying $g(t) = tg(1/t)$ (see Table 1). In general no choice of balancing function Peskun-dominates the others, not even asymptotically. Therefore, the comparison between different balancing functions is more subtle than the one between the functions that satisfy $g(t) = tg(1/t)$ and the ones that do not. In the following sections, we identify explicitly the optimal balancing function in a simplified scenario (Section 4.1) and provide guidelines on the choice of balancing functions in generic contexts (Section 4.2).

Remark 3. The MH algorithm accepts each proposed state y with probability $a(x,y) = g(t(x,y))$, where $t(x,y) = \frac{\pi(y)Q(y,x)}{\pi(x)Q(x,y)}$ and $g(t) = 1 \wedge t$. We refer to $a(x,y)$ as acceptance probability function (APF). It is well known that there are many possible choices of APF: as long as $a(x,y) = t(x,y)\, a(y,x)$, the resulting kernel P is π-reversible and can be used for MCMC purposes. The class of APFs is closely related to the class of balancing functions. Writing $a(x,y)$ as $g(t(x,y))$, the condition $a(x,y) = t(x,y)\, a(y,x)$ translates to $g(t) = tg(1/t)$, which coincides with Equation (4). Therefore, by Theorem 1, each APF $a(x,y) = g(t(x,y))$ corresponds to a balancing function g. However, the

Table 1. Examples of locally balanced proposals Q_g obtained from different balancing functions g. The symbols $\wedge$ and $\vee$ represent minimum and maximum operators, respectively.

	$g(t) = \sqrt{t}$	$g(t) = \frac{t}{1+t}$	$g(t) = 1 \wedge t$	$g(t) = 1 \vee t$
$Q_g(x,y) \propto$	$\sqrt{\pi(y)}K(x,y)$	$\frac{\pi(y)}{\pi(x)+\pi(y)}K(x,y)$	$\left(1 \wedge \frac{\pi(y)}{\pi(x)}\right)K(x,y)$	$\left(1 \vee \frac{\pi(y)}{\pi(x)}\right)K(x,y)$

family of balancing functions is broader than the family of APFs because $a(x, y)$ represents a probability and thus needs to be bounded by 1, while balancing functions don't. For example, $g(t) = \sqrt{t}$ or $1 \vee t$ are valid balancing functions but are not upper bounded by 1 and thus they are not valid APFs. Moreover, in the context of APFs, it is well known that the MH choice $g(t) = 1 \wedge t$ is optimal and Peskun-dominates all other choices (Peskun 1973; Tierney 1998), while this is no longer true in the context of balancing functions.

4.1. The Optimal Proposal for Independent Binary Variables

In this section we compare the efficiency of different locally balanced proposals in the independent binary components case of Example 1. It turns out that in this specific case the Barker balancing function $g(t) = \frac{t}{1+t}$ leads to the smallest mixing time.

From Example 1, each move from $\mathbf{x}_{1:n}$ to a neighboring state $\mathbf{y}_{1:n} \in N(\mathbf{x}_{1:n})$ is obtained by flipping one component of $\mathbf{x}_{1:n}$, say the ith bit, either from 0 to 1 or from 1 to 0. We denote the former by $\mathbf{y}_{1:n} = \mathbf{x}_{1:n} + \mathbf{e}^{(i)}_{1:n}$ and the latter by $\mathbf{y}_{1:n} = \mathbf{x}_{1:n} - \mathbf{e}^{(i)}_{1:n}$. The pointwise informed proposal $Q^{(n)}_g$ defined in Equation (1) can then be written as

$$Q^{(n)}_g(\mathbf{x}_{1:n}, \mathbf{y}_{1:n}) = \frac{1}{Z^{(n)}_g(\mathbf{x}_{1:n})} \begin{cases} g(\frac{p_i}{1-p_i}) & \text{if } \mathbf{y}_{1:n} = \mathbf{x}_{1:n} + \mathbf{e}^{(i)}_{1:n}, \\ g(\frac{1-p_i}{p_i}) & \text{if } \mathbf{y}_{1:n} = \mathbf{x}_{1:n} - \mathbf{e}^{(i)}_{1:n}, \\ 0 & \text{if } \mathbf{y}_{1:n} \notin N(\mathbf{x}_{1:n}). \end{cases} \tag{15}$$

To compare the efficiency of $Q^{(n)}_g$ for different choices of g we proceed in two steps. First, we show that, after appropriate time-rescaling, the MH chain of interest converges to a tractable continuous time process as $n \to \infty$ (Theorem 6). Second, we find which choice of g induces the fastest mixing on the limiting continuous-time process. Similar asymptotic approaches are well-established in the literature to compare MCMC schemes (see, e.g., Roberts, Gelman, and Gilks 1997).

To simplify the following discussion, we first rewrite $Q^{(n)}_g$ as

$$Q^{(n)}_g(\mathbf{x}_{1:n}, \mathbf{y}_{1:n}) = \frac{1}{Z^{(n)}_g(\mathbf{x}_{1:n})} \begin{cases} \nu_i\, c_i\, (1-p_i) & \text{if } \mathbf{y}_{1:n} = \mathbf{x}_{1:n} + \mathbf{e}^{(i)}_{1:n}, \\ \nu_i\, (1-c_i)\, p_i & \text{if } \mathbf{y}_{1:n} = \mathbf{x}_{1:n} - \mathbf{e}^{(i)}_{1:n}, \\ 0 & \text{if } \mathbf{y}_{1:n} \notin N(\mathbf{x}_{1:n}), \end{cases} \tag{16}$$

where $c_i \in (0,1)$ and $\nu_i > 0$ are the solution of $\nu_i c_i(1-p_i) = g(\frac{p_i}{1-p_i})$ and $\nu_i(1-c_i)p_i = g(\frac{p_i}{1-p_i})$. Given Equation (16), finding the optimal g corresponds to finding the optimal values for the two sequences $(\nu_1, \nu_2, \dots)$ and $(c_1, c_2, \dots)$. In the following we assume $\inf_{i\in\mathbb{N}} p_i > 0$, $\sup_{i\in\mathbb{N}} p_i < 1$ and the existence of $\lim_{n\to\infty} \frac{\sum_{i=1}^n \nu_i p_i(1-p_i)}{n} > 0$. The latter is a mild assumption to avoid pathological behavior of the sequence of p_i's and guarantee the existence of a limiting process.

Let $\mathbf{X}^{(n)}(t)$ be the MH Markov chain with proposal $Q^{(n)}_g$ and target $\pi^{(n)}$. For any real time t and positive integer $k \le n$, we define

$$S^{(n)}_{1:k}(t) = \left(X^{(n)}_1(\lfloor nt \rfloor), \dots, X^{(n)}_k(\lfloor nt \rfloor)\right),$$

with $\lfloor nt \rfloor$ being the largest integer smaller than nt. Note that $S^{(n)}_{1:k} = (S^{(n)}_{1:k}(t))_{t\ge 1}$ is a continuous-time (non-Markovian) stochastic process on $\{0,1\}^k$ describing the first k components of $(\mathbf{X}^{(n)}(t))_{t\ge 1}$.

Theorem 6. Let $\mathbf{X}^{(n)}(1) \sim \pi^{(n)}$ for every n. For any positive integer k, it holds

$$S^{(n)}_{1:k} \overset{n\to\infty}{\Longrightarrow} S_{1:k},$$

where $\Rightarrow$ denotes weak convergence and $S_{1:k}$ is a continuous-time Markov chain on $\{0,1\}^k$ with jumping rates given by

$$A(\mathbf{x}_{1:k}, \mathbf{y}_{1:k}) = \begin{cases} e_i(\mathbf{v}, c_i) \cdot (1-p_i) & \text{if } \mathbf{y}_{1:k} = \mathbf{x}_{1:k} + \mathbf{e}^{(i)}_{1:k}, \\ e_i(\mathbf{v}, c_i) \cdot p_i & \text{if } \mathbf{y}_{1:k} = \mathbf{x}_{1:k} - \mathbf{e}^{(i)}_{1:k}, \\ 0 & \text{if } \mathbf{y}_{1:k} \notin N(\mathbf{x}_{1:k}) \text{ and } \mathbf{y}_{1:k} \neq \mathbf{x}_{1:k}, \end{cases} \tag{17}$$

where

$$e_i(\mathbf{v}, c_i) = \frac{1}{\bar{Z}(\mathbf{v})}\, \nu_i\, ((1-c_i) \wedge c_i) \tag{18}$$

with $\bar{Z}(\mathbf{v}) = \lim_{n\to\infty} \frac{\sum_{i=1}^n \nu_i p_i(1-p_i)}{n}$.

We can now use Theorem 6 and the simple form of the limiting process $S_{1:k}$ to establish what is the asymptotically optimal proposal $Q^{(n)}_g$. In fact Equation (17) implies that, in the limiting process $S_{1:k}$, each bit is flipping independently of the others, with flipping rate of the ith bit being proportional to $e_i(\mathbf{v}, c_i)$. Moreover, from Equation (18), we see that the parameter c_i influences only the behavior of the ith component. Therefore, each c_i can be independently optimized by maximizing $e_i(\mathbf{v}, c_i)$, which leads to $c_i = \frac{1}{2}$ for every i. By definition of c_i, the condition $c_i = \frac{1}{2}$ corresponds to $g(\frac{p_i}{1-p_i}) = \frac{p_i}{1-p_i} g(\frac{1-p_i}{p_i})$. Therefore, requiring $c_i = \frac{1}{2}$ for all i corresponds to using a balancing function g satisfying $g(t) = tg(1/t)$. This is in accordance with the results of Section 3 and with the intuition that locally balanced proposal are asymptotically optimal in high dimensions.

Let us now consider the parameters $(\nu_1, \nu_2, \dots)$. Given $c_i = \frac{1}{2}$, different choices of $(\nu_1, \nu_2, \dots)$ correspond to different locally balanced proposals. From Equation (18), we see that each ν_i affects the flipping rate of all components through the normalizing constant $\bar{Z}(\mathbf{v})$, making the optimal choice of ν_i less trivial. Intuitively, the parameter ν_i represents how much effort we put into updating the ith component, and increasing ν_i reduces the effort put into updating other components. To discriminate among various choices of $(\nu_1, \nu_2, \dots)$, we look for the choice that minimizes the mixing time of $S_{1:k}$ for k going to infinity. Although this is not the only possible criterion to use, it is a reasonable and natural one. As we discuss in Supplement A, the latter is achieved by minimizing the mixing time of the slowest bit, which corresponds to choosing ν_i constant over i. Intuitively, this means that we are sharing the sampling effort equally across components. It follows that the asymptotically optimal proposal $Q^{(n)}_{g_{\text{opt}}}$ is

$$Q^{(n)}_{g_{\text{opt}}}(\mathbf{x}_{1:n}, \mathbf{y}_{1:n}) \propto \begin{cases} (1-p_i) & \text{if } \mathbf{y}_{1:n} = \mathbf{x}_{1:n} + \mathbf{e}^{(i)}_{1:n}, \\ p_i & \text{if } \mathbf{y}_{1:n} = \mathbf{x}_{1:n} - \mathbf{e}^{(i)}_{1:n}, \\ 0 & \text{if } \mathbf{y}_{1:n} \notin N(\mathbf{x}_{1:n}). \end{cases} \tag{19}$$

The latter can be written as

$$Q^{(n)}_{g_{\text{opt}}}(\mathbf{x}_{1:n}, \mathbf{y}_{1:n}) \propto \frac{\pi^{(n)}(\mathbf{y}_{1:n})}{\pi^{(n)}(\mathbf{x}_{1:n}) + \pi^{(n)}(\mathbf{y}_{1:n})} \mathbb{1}_{N(\mathbf{x}_{1:n})}(\mathbf{y}_{1:n})$$

which means that the optimal balancing function is

$$g_{opt}(t) = \frac{t}{1+t}.$$

Interestingly, the latter choice has been previously considered in the context of acceptance probability functions (see Remark 3) and is commonly referred to as Barker choice (Barker 1965).

4.2. Guidelines for the Choice of Balancing Function

The analytic results of the previous section are derived in the specific context of independent binary distribution. To facilitate practical implementations, we now provide some heuristic guidelines for the choice of balancing function in generic contexts. Section 1.3 in Supplement B contains a more detailed discussion and some illustrations. In our experience, the difference in efficiency between locally balanced proposals and other pointwise informed proposals is larger than the one among different locally balanced proposals. Nonetheless, a bad choice of g, even if Equation (4) is satisfied, can lead to poor performances. We focus our attention on functions g that are continuous and non-decreasing. The latter is a natural requirement as it corresponds to assuming that states with a higher target probability are more likely to be proposed.

The main recommendation is to always use balancing functions such that $\lim_{t\to 0} g(t) = 0$. In fact the performances of the resulting sampler can deteriorate significantly if g is chosen so that $\lim_{t\to 0} g(t) > 0$, for example, $g(t) = 1 + t$ or $g(t) = \max\{1, t\}$. Intuitively, the reason is that if $\lim_{t\to 0} g(t) > 0$ neighboring states $y \in N(x)$ such that $\pi(y)/\pi(x) \ll 1$ will still receive substantial probability of being proposed. This will lead either to most moves being rejected or to the chain often making a move and the reverse move straight after, leading to poor mixing. This behavior is illustrated in Section 1.3 of Supplement B through simulations.

The results of the latter simulations also support the use of $g(t) = t/(1+t)$, which was theoretically shown to be optimal in the context of Example 1 in Section 4.1. In fact we always found this choice either to lead to the fastest mixing or to a mixing roughly equivalent to competitors, see Supplement B. The choice $g(t) = \sqrt{t}$ is also an interesting one. In fact it is quite intuitive (see, e.g., Section 2.1) and it is the only balancing function that factorizes, that is, $g(t_1 t_2) = g(t_1)g(t_2)$ for all $t_1, t_2 > 0$. In cases were such property simplifies calculations or computation, the choice $g(t) = \sqrt{t}$ may be preferred. In the simulation study of Section 6, we use $g(t) = \sqrt{t}$ and $g(t) = t/(1+t)$, which both led to good mixing in the examples we tried.

We found that the behavior of $g(t)$ for large values of t has a significant impact on the sampler during burn-in (i.e., the transient phase). In particular, unbounded g functions, such as $g(t) = \sqrt{t}$, often lead to faster convergence during burn-in compared to bounded g functions, such as $g(t) = t/(1+t)$. The reason is that for unbounded g functions the behavior of the sampler during burn-in is more similar to a greedy optimization algorithm (indeed as the derivative $g'(t)$ increases, the proposal distribution collapses to a point mass on the highest probability state in the neighborhood) and results in faster convergence towards a mode of the target distribution. Thus one may prefer $g(t) = \sqrt{t}$ over $g(t) = t/(1+t)$ if burn-in speed is the main criterion of interest.

In Supplement B, we also consider the choice $g(t) = \min\{1, t\}$. The latter coincides with $\tilde{g}(t) = \min\{g(t), t\, g(1/t)\}$ when $g(t) = 1$ or $g(t) = t$ and thus Theorem 3 implies that the sampler based on $g(t) = \min\{1, t\}$ cannot perform substantially worse than the uninformed random walk proposal and the globally balanced proposal in high dimensions. In our experiments, the use of $g(t) = \min\{1, t\}$ led to behaviors that are very similar to the ones induced by $g(t) = t/(1+t)$. In fact, Proposition 1.1 in Supplement B shows that the efficiencies induced by $g(t) = \min\{1, t\}$ and $g(t) = t/(1+t)$ can differ by at most a factor of 2. Combining the results in Section 6 and Supplement B, one can see that $g(t) = \min\{1, t\}$ significantly outperforms $g(t) = 1$ and $g(t) = t$ in all the examples under consideration, which provides an empirical illustration of Theorem 3.

5. Connection to MALA and Gradient-Based MCMC

In the context of continuous state spaces, such as $\mathcal{X} = \mathbb{R}^n$, it is typically not feasible to sample efficiently from pointwise informed proposals as defined in Equation (1). A natural thing to do in this context is to replace the intractable term in $g(\frac{\pi(y)}{\pi(x)})$, that is, the target $\pi(y)$, with some local approximation around the current location x. For example, using a first-order Taylor expansion $e^{\log \pi(y)} \approx e^{\log \pi(x) + (\nabla \log \pi(x))^T (y-x)}$, we obtain a family of first-order informed proposals of the form

$$Q^{(1)}_g(x, dy) \propto g\left(e^{(\nabla \log \pi(x))^T (y-x)}\right) K(x, dy), \qquad (20)$$

for K symmetric and g satisfying Equation (4). Interestingly, the well-known MALA proposal (e.g., Roberts and Rosenthal 1998) can be obtained from Equation (20) by choosing $g(t) = \sqrt{t}$ and a Gaussian kernel $K(x, \cdot) = N(x, \sigma^2 \mathbb{I}_n)$. Therefore, we can think at MALA as a specific instance of locally balanced proposal with first-order Taylor approximation. This simple and natural connection between locally balanced proposals and classical gradient-based schemes hints to many possible extensions of the latter, such as modifying the balancing function g or kernel K or considering a different approximation for $\pi(y)$. The flexibility of the resulting framework could help to increase the robustness and efficiency of gradient-based methods. Recently, Titsias and Papaspiliopoulos (2018) considered different but related approaches to improve gradient-based MCMC schemes for latent Gaussian models, achieving state-of-the-art sampling algorithms for various applications compared to both MALA and HMC. Given the focus of this article on discrete spaces, we do not pursue this avenue here, leaving this research lines to future work (see Section 8).

6. Simulation Studies

In this section, we perform simulation studies using the target distributions of Examples 2 and 3. All computations are per-

formed using the R programming language with code available in the online supplementary material. The aim of the simulation study is two-folded: first comparing informed schemes with random walk ones, and secondly comparing different constructions of informed schemes among themselves.

6.1. MCMC Schemes Under Consideration

We compare seven schemes: random walk MH (RW), a globally balanced proposal (GB), two locally balanced proposals (LB1 and LB2), the Hamming Ball sampler (HB) proposed in Titsias and Yau (2017), the discrete HMC algorithm (D-HMC) proposed in Pakman and Paninski (2013) and the Multiple-try Metropolis (MTM) scheme (Liu, Liang, and Wong 2000). The first four schemes (RW-GB-LB1-LB2) are MH algorithms with pointwise informed proposals of the form $Q_g(x,y) \propto g\left(\frac{\pi(y)}{\pi(x)}\right) K(x,y)$, with $g(t)$ equal to 1, t, $\sqrt{t}$ and $\frac{t}{1+t}$, respectively. HB is a data augmentation scheme that, given the current location x_t, first samples an auxiliary variable $u \sim K(x_t,\cdot)$ and then samples the new state $x_{t+1} \sim Q_\pi(u,\cdot)$, where $Q_\pi(u,y) \propto \pi(y)K(u,y)$ is defined as in Equation (3). No acceptance–reject step is required as the chain is already π-reversible, being interpretable as a two stage Gibbs sampler on the extended state space (x,u) with target $\pi(x)K(x,u)$. MTM is a popular MCMC scheme that proceeds as follows: first k values $y_1,\ldots,y_k$ are sampled from $K(x,\cdot)$, where x is the current state of the chain; then one of the y_i's is chosen from $y_1,\ldots,y_k$ with probability proportional to $\pi(y_i)$; finally an accept/reject step is performed requiring to sample $k-1$ points around the chosen y_i (see Liu, Liang, and Wong 2000 for full details and variations). In our simulations, we used $k = 100$. We also considered $k = 10$ and $k = 1000$ which led to comparable or worse performances, depending on the target under consideration. To have a fair comparison, all the six schemes described so far use the same base kernel K, defined as $K(x,\cdot) = \text{Unif}(N(x))$ with neighboring structures $\{N(x)\}_{x\in\mathcal{X}}$ defined in Examples 2 and 3. Finally, D-HMC is a sampler specific to binary target distributions (thus applicable to Example 2 but not to Example 3) constructed by first embedding the binary space in a continuous space and then applying HMC in the latter. For its implementation, we followed Pakman and Paninski (2013), using a Gaussian distribution for the momentum variables and an integration time equal to 2.5π. We will be talking about acceptance rates for all schemes, even if HB and D-HMC are not constructed as MH schemes. For HB, we define the acceptance rate as the proportion of times that the new state x_{t+1} is different from the previous state x_t (indeed the sampling procedure $u \sim K(x_t,\cdot)$ and $x_{t+1} \sim Q_\pi(u,\cdot)$ does often return $x_{t+1} = x_t$). For D-HMC, we define the acceptance rate as the proportion of times that a proposal to flip a binary component in the HMC flow is accepted (using the Pakman and Paninski 2013 terminology, the proportion of times that the particle crosses a potential wall rather than bouncing back). Such definitions of acceptance rate are related to the notion of mutation rate previously used in the context of discrete space sampling problems (see, e.g., Schäfer and Chopin 2013) and will help us diagnose which algorithms exhibit pathologically low acceptance rate.

6.2. Sampling Permutations

Consider the setting of Example 3, with target density $\pi^{(n)}(\rho) \propto \prod_{i=1}^n w_{i\rho(i)}$ defined in Equation (13) and base kernel $K(\rho,\cdot)$ being the uniform distribution on the neighborhood $N(\rho)$ defined in (14). The distribution $\pi^{(n)}$ arises in many applied scenarios (e.g., Dellaert et al. 2003; Oh, Russell, and Sastry 2009; Zanella 2015) and sampling from the latter is a nontrivial task that is often accomplished with MCMC algorithms (see, e.g., Jerrum and Sinclair (1996) for related complexity results).

For our simulations, we first consider the case of iid. weights $\{w_{ij}\}_{i,j=1}^n$ with $\log(w_{ij}) \sim N(0,\lambda^2)$. Here, n and λ provide control on the dimensionality and the smoothness of the target distribution, respectively. For example, when $\lambda = 0$ the target distribution is uniform and the six schemes under consideration (D-HMC not applicable) collapse to the same transition kernel, which is $K(\rho,\cdot)$ itself (modulo HB performing two steps per iteration). On the other hand, as λ increases the difference between RW and informed schemes becomes more prominent (Table 2).

In fact, for "rough" distributions, most states proposed by RW have small probability under the target and get rejected. Despite being more robust than RW, also GB and HB suffer from high rejection rates as λ increase. This results in a low efficiency, measured as effective sample size per unit of computation time (see Table 2). On the contrary, LB1 and LB2 are robust to the increase in roughness and their efficiency does not deteriorate as λ increases. Table 2 suggests that for flatter targets (i.e., small values of λ) the computational overhead required

Table 2. Acceptance rates and effective sample sizes per second (ESS/time) when targeting Example 3 with $\log(w_{ij}) \overset{iid}{\sim} N(0,\lambda^2)$ for $n = 500$ and varying λ.

		RW	GB	LB1	LB2	HB	MTM
$\lambda = 1$	Acc.rate	0.319	0.322	0.998	0.999	1	0.372
	ESS/time	10.88	0.32	0.7	0.58	1.24	0.49
$\lambda = 2$	Acc.rate	0.05	0.015	0.991	0.996	0.828	0.403
	ESS/time	1.05	0.01	0.54	0.39	0.52	0.45
$\lambda = 3$	Acc.rate	0.008	0	0.985	0.989	0.239	0.053
	ESS/time	0.13	0	0.51	0.53	0.08	0.02
$\lambda = 4$	Acc.rate	0.004	0	0.915	0.98	0.048	0
	ESS/time	0.33	0	1.08	0.97	0.06	0
$\lambda = 5$	Acc.rate	0.003	0	0.861	0.969	0.014	0
	ESS/time	0.03	0	0.62	0.72	0.01	0

to use informed proposal schemes is not worth the effort, as a plain RW proposal achieves the highest efficiency (ESS/time). However, as the roughness increases and the target distribution becomes more challenging to explore, informed proposals (in particular LB1 and LB2) become drastically more efficient (see, e.g., Figure 1).

Note that GB and MTM are extremely sensitive to the starting state: if the latter is unlikely under the target (e.g., a uniformly at random permutation) the chain gets stuck and reject almost all moves, while if started in stationarity (i.e., from a permutation approximately drawn from the target) the chain has a more stable behavior (see Figure 1). The similarity between GB and MTM is not a coincidence. In fact they both incorporate information from the target in a similar way, namely through a multiplicative term proportional to $\pi(y)$.

Finally, consider a more structured case where, rather than iid weights, we sample $w_{ij} \sim \exp(-\chi^2_{|i-j|})$. The resulting matrix $\{w_{ij}\}_{i,j=1}^n$ has a banded-like structure, with weights getting smaller with distance from the diagonal. As shown in Figure 2, the results are analogous to the iid case.

6.3. Ising Model

Consider now the Ising model described in Example 2. The latter is a classic model used in many scientific areas, for example statistical mechanics and spatial statistics. In this simulation study, we consider target distributions motivated by Bayesian image analysis, where one seeks to partition an image into objects and background. In the simplest version of the problem, each pixel i needs to be classified as object ($x_i = 1$) or background ($x_i = -1$). One approach to such task is to define a Bayesian model, using the Ising model (or the Potts model in more general multi-objects contexts) as a prior distribution to induce positive correlation among neighboring pixels. The resulting posterior distribution is made of a prior term $\exp(\lambda \sum_{(i,j)\in E_n} x_i x_j)$ times a likelihood term $\exp(\sum_{i\in V_n} \alpha_i x_i)$, which combined produce a distribution of the form Equation (11). See Section 1.2 of Supplement B for more details on the derivation of such distributions and Moores et al. (2015a) for recent applications to computed tomography.

Similarly to Section 6.2, we considered a sequence of target distributions, varying the concentration of the target distribution (controlled by the strength of spatial correlation λ and signal-to-noise ratio in the likelihood terms α_i). Figure 3 displays the convergence behavior of the seven MCMC schemes under consideration for four levels of target concentration. Target 1 refers to a close-to-uniform distribution while Target 4 refers to the most concentrated distribution (see Supplement B for full details on the set up for λ and the α_i's). It can be seen

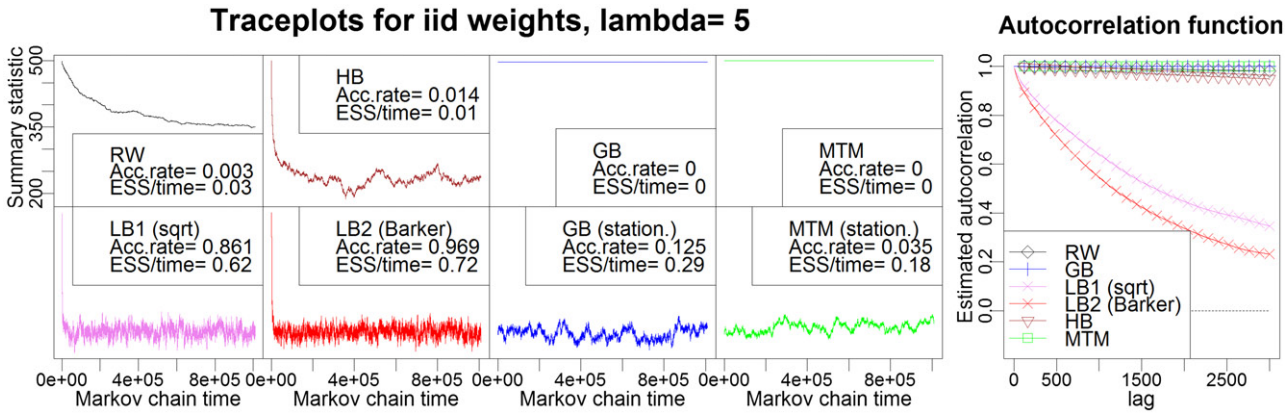

Figure 1. Setting: $n = 500$ and $\log(w_{ij}) \overset{iid}{\sim} N(0, \lambda^2)$ with $\lambda = 5$. Left: traceplots of a summary statistic (Hamming distance from fixed permutation). Right: estimated autocorrelation functions.

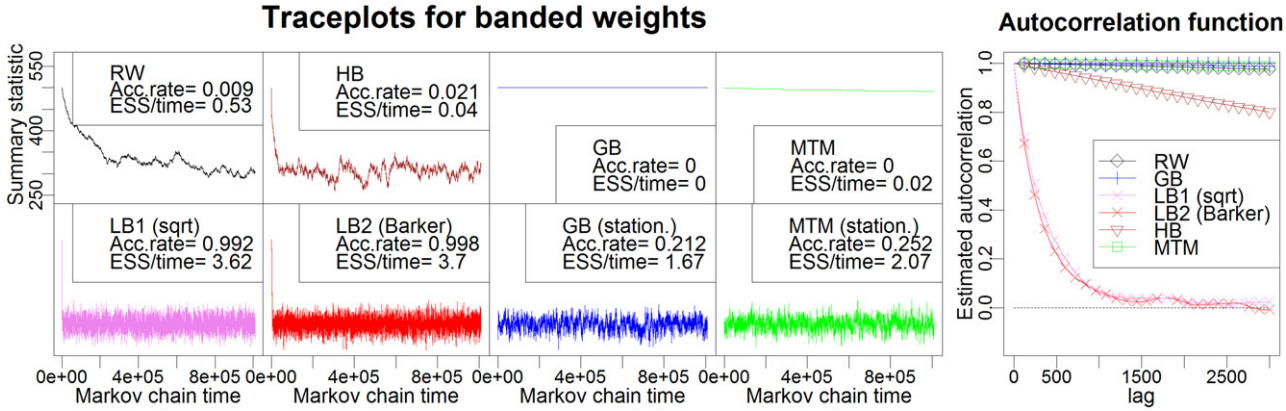

Figure 2. Setting: $n = 500$ and $w_{ij} \sim \exp(-\chi^2_{|i-j|})$. Left: traceplots of a summary statistic (Hamming distance from fixed permutation). Right: estimated autocorrelation functions.

that LB1 and LB2 converge significantly faster than alternative schemes and that the difference gets larger as the target gets further from the uniform one. Like in the matching example, GB and MTM converge extremely slowly and often get stuck at the starting configuration.

Table 3 describes the samplers' behavior in stationarity, where all schemes are started from a configuration approximately sampled from the target. Broadly speaking, the results are similar to the ones of Section 6.2, with LB1 and LB2 being drastically more efficient than alternative schemes, especially for highly nonuniform targets. Despite the pathological behavior during burn-in, MTM performs well in stationarity, although it still less efficient than LB1 and LB2. Figure 4 reports traceplots and autocorrelation functions for Target 3 (see Supplement B for plots for all targets).

In addition, we compare the samplers under consideration with the Swendsen–Wang (SW) algorithm (Swendsen and Wang 1987), which is a specialized scheme for the Ising model performing global updates. SW is somehow complementary to the single-site updating schemes considered here. In fact previous works suggest that SW performs better then schemes updating one variable at a time when the target is multimodal (e.g., in the absence of informative likelihood terms and with moderately strong interaction term λ), while it performs poorly in cases where the likelihood terms $\{\alpha_i\}_{i\in V_n}$ dominate, like the image analysis context considered here (see, e.g., Hurn (1997) and Moores, Pettitt, and Mengersen (2015b) for more discussion). Our results (see last column of Table 3) are consistent with previous work, as we observe SW to perform well for targets where the likelihood is weak (Target 1) while it performs poorly for targets with strong likelihood (Target 4). Interestingly, LB1 and LB2 are always at least competitive with SW, although it should be noted that no target under consideration here exhibits strong multimodality. In general, any pointwise informed proposal (including locally balanced and globally balanced ones) will struggle to move across well-separated modes, in particular modes such that it is hard to go from one mode to the other with moves supported by the base kernel $K(x, y)$.

6.4. Computational Cost Versus Statistical Efficiency Trade-off

In many scenarios, as the dimensionality of the state space increases, also the computational cost of sampling from the pointwise informed proposals as defined in Equation (1) increases. For example, in discrete space settings it is common to have a base kernel $K(x, \cdot)$ which is a uniform distribution on some neighborhood $N(x) \subseteq \mathcal{X}$ whose size grows with the dimensionality of $\mathcal{X}$. In these cases, when the size of $N(x)$ becomes too large, it may be inefficient to use an informed proposal on the whole neighborhood $N(x)$. Rather, it may be more efficient to first select a sub-neighborhood $N'(x) \subseteq N(x)$ at random, and then apply locally balanced proposals to the selected subspace. For example, if the state space under consideration has a Cartesian product structure, then one could update a subset of variables given the others in a *block-wise* fashion, analogously to the Gibbs Sampling context. By choosing appropriately the size of $N'(x)$ one can obtain an appropriate trade-off between computational cost (a small $N'(x)$ induces an informed proposal that is cheap to sample) and statistical efficiency (a large $N'(x)$ produces better informed moves as it considers more candidate states at each step). Such an approach is illustrated in Section 7 and Supplement B on a record linkage application. See also Titsias and Yau (2017) for additional discussion on block-wise implementations and the resulting cost-vs-efficiency trade-off in the context of the Hamming Ball

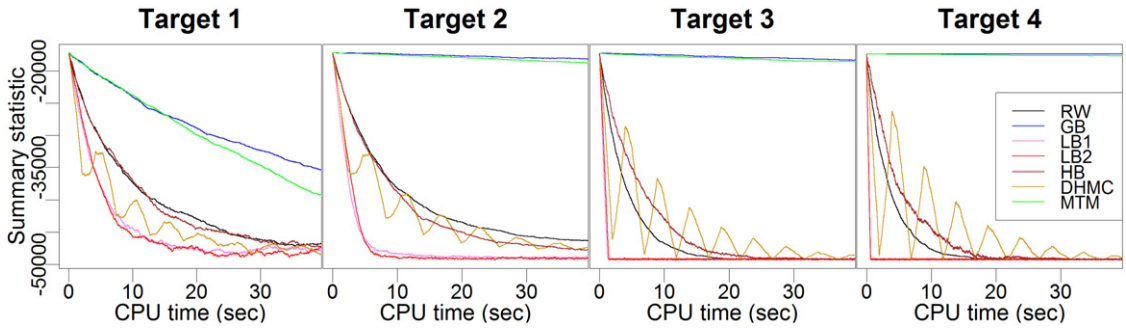

Figure 3. Convergence behavior of the seven MCMC schemes under consideration when targeting Example 2 for $n = 500$ and various levels of concentration of the target.

Table 3. Acceptance rates and effective sample sizes per second (ESS/time) in stationarity when targeting Example 2 with $n = 500$. See Supplement B for details on λ and the α_i's.

		RW	GB	LB1	LB2	HB	MTM	DHMC	SW
Target 1	Acc.rate	0.09	0.024	1	1	1	0.233	0.081	
	ESS/time	0.09	0.02	0.3	0.34	0.2	0.11	0.11	0.38
Target 2	Acc.rate	0.006	< 0.001	0.998	1	0.68	0.374	0.006	
	ESS/time	0.07	0.04	0.34	0.63	0.15	0.79	0.14	0.08
Target 3	Acc.rate	0.001	0.003	0.99	0.998	0.49	0.106	0.001	
	ESS/time	0.18	0.1	3.68	26.31	0.22	3.19	0.26	0.28
Target 4	Acc.rate	< 0.001	0.002	0.949	0.996	0.28	0.042	< 0.001	
	ESS/time	0.35	0.48	24.5	85.98	0.4	6.14	0.44	0.24

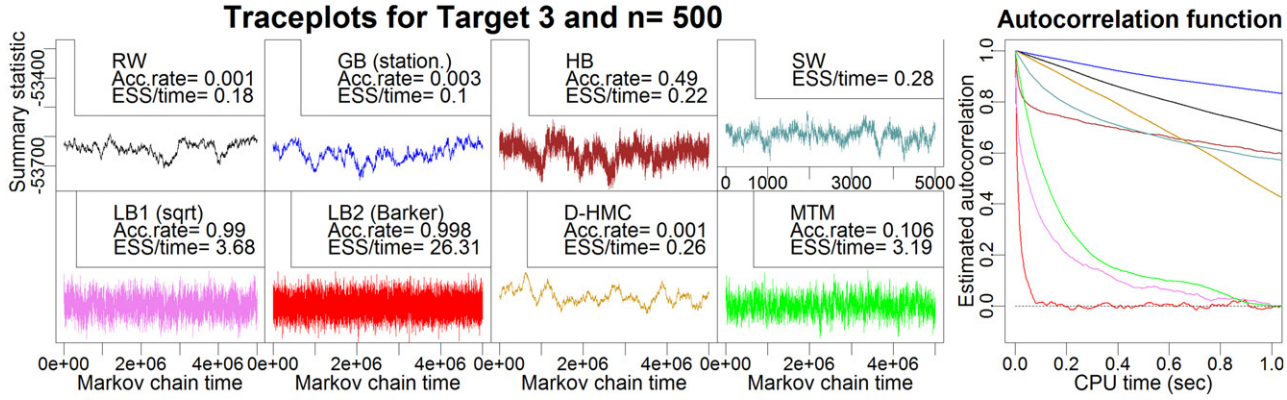

Figure 4. Setting: Ising model with $n = 500$, $\lambda = 1$ and α_i's described in Supplement B. Left: traceplots of the summary statistic $\sum_i x_i$. The D-HMC plot displays the whole trajectory, including the path during integration. Right: estimated autocorrelation functions.

sampler and Section 8 for related comments in future works discussion.

Another issue related to cost-vs-efficiency trade-offs is the one of choosing the value of σ. In fact, in many contexts the size of the neighborhood depends on some parameter σ, for example, when $N(x)$ is a ball of radius σ in some metric, for example, Hamming distance. In such cases, using a large value of σ increases the computational cost of each MCMC step, but also increases the number of candidate states under consideration, thus hopefully increasing mixing. Typically, there is a minimal value for σ, for example $\sigma = 1$ in Example 2 (corresponding to flipping one bit) or $\sigma = 2$ for Example 3 (corresponding to swapping two indices). Increasing σ above the minimal value may be advantageous when the target exhibits strong multimodality, as a large σ allows the sampler to jump across modes avoiding low probability states. The issue is carefully discussed in Titsias and Yau (2017). In all our experiments, it was not worth increasing σ above the minimal value as the improvement in mixing was not enough to compensate the additional computational cost.

7. Application to Bayesian Record Linkage

Record linkage, also known as entity resolution, is the process of identifying which records refer to the same entity across two or more databases with potentially repeated entries. Such operation is typically performed to remove duplicates when merging different databases. If records are potentially noisy and unique identifiers are not available, statistical methods are needed to perform reliable record linkage operations. While traditional record linkage methodologies are based on the early work Fellegi and Sunter (1969), Bayesian approaches to record linkage are receiving increasing attention in recent years (Tancredi and Liseo 2011; Steorts, Hall, and Fienberg 2016; Sadinle 2017; Johndrow, Lum, and Dunson 2018). Such approaches are particularly interesting because they provide uncertainty statements on the linkage procedure that can be naturally propagated to subsequent inferences, such as population-size estimation (Tancredi and Liseo 2011). Despite recent advances in Bayesian modeling for record linkage problems (see, e.g., Zanella et al. (2016)), the practical applicability of such methodology is still limited. The main reason is that the MCMC schemes that are typically used (e.g., random walk metropolis or Gibbs Sampling) struggle to explore efficiently the discrete space of possible linkage configurations. In this section, we use locally balanced proposals to derive improved samplers for Bayesian record linkage.

We consider bipartite record linkage tasks, where one seeks to merge two databases with duplicates occurring across databases but not within. This is the most commonly considered case in the record linkage literature (see Sadinle 2017 and references therein). Denote by $\mathbf{x} = (x_1, \ldots, x_{n_1})$ and $\mathbf{y} = (y_1, \ldots, y_{n_2})$ the two databases under consideration. In this context, the parameter of interest is a partial matching between $\{1, \ldots, n_1\}$ and $\{1, \ldots, n_2\}$, where i is matched with j if and only if x_i and y_j represent the same entity. We represent such a matching with a n_1-dimensional vector $\mathbf{M} = (M_1, \ldots, M_{n_1})$, where $M_i = j$ if x_i is matched with y_j and $M_i = 0$ if x_i is not matched with any y_j. In Section 2 of Supplement B, we specify a Bayesian model for bipartite record linkage, assuming the number of entities and the number of duplicates to follow Poisson distributions a priori, and assuming the joint distribution of $(\mathbf{x}, \mathbf{y})$ given $\mathbf{M}$ to follow a spike-and-slab categorical distribution (often called hit-miss model in the record linkage literature Copas and Hilton 1990). The unknown parameters of the model are the partial matching $\mathbf{M}$ and two real-valued hyperparameters λ and p_{match}, representing the expected number of entities and the probability of having a duplicate for each entity.

We perform posterior inferences in a Metropolis-within-Gibbs fashion, where we alternate sampling $(p_{\text{match}}, \lambda)|\mathbf{M}$ and $\mathbf{M}|(p_{\text{match}}, \lambda)$, see Section 2.3 of Supplement B for full details on the sampler. While it is straightforward to sample from the two-dimensional distribution $(p_{\text{match}}, \lambda)|\mathbf{M}$, see (2.3) and (2.4) in Supplement B for explicit full conditionals, providing an efficient way to update the high-dimensional discrete object $\mathbf{M}$ is more challenging and this is where we exploit locally balanced proposals. The typical schemes used in the literature to update such discrete parameters are either Gibbs Samplers on augmented spaces (e.g., (Tancredi and Liseo 2011; Sadinle 2017)) or MH schemes with add/delete/swap moves (e.g., Green and Mardia 2006; Steorts, Hall, and Fienberg 2016; Zanella et al. 2016; McVeigh and Murray 2017; Briscolini et al. 2018). We focus on the latter as they have been reported to be more

scalable (see, e.g., Steorts, Hall, and Fienberg 2016) and they fit more naturally in our framework. The MH scheme evolves by choosing uniformly at random two indices $(i, j) \in \{1, \ldots, n_x\} \times \{1, \ldots, n_y\}$ and then proposing a corresponding move (either add, delete, single-swap, or double-swap). Section 2.3 of Supplement B describes the set of allowed moves and the resulting base kernel $K(\mathbf{M}, \mathbf{M}')$. Following the notation of Section 6, we denote the resulting scheme as random walk (RW) and we compare it with three alternative schemes, namely the globally balanced proposal (GB), locally balanced proposal with Barker weights (LB), and the Hamming ball sampler (HB). See Section 6 for their description.

We consider a dataset derived from the Italian Survey on Household and Wealth, which is a biennial survey conducted by the Bank of Italy. The dataset is publicly available (e.g., through the *Italy* R package) and consists of two databases, the 2008 survey (covering 13,702 individuals) and the 2010 one (covering 13,733 individuals). For each individual, the survey recorded 11 variables, such as year of birth, sex, and working status. First, following Steorts (2015), we perform record linkage for each Italian region separately. This results in 20 separate record linkage tasks with roughly 1300 individuals each on average. We ran the four MCMC schemes under consideration for each record linkage task and compare their performances. Figure 5(a) shows the number of matches over MCMC iterations for region 1 (other regions show a similar qualitative behavior). We can see that LB and HB converge rapidly to the region of substantial posterior probability, while RW and GB exhibit an extremely slow convergence. HB, however, converges and mixes significantly slower than LB (see e.g., the autocorrelation functions in 5(b)). To provide a quantitative comparison between the performances of the four schemes, we consider as efficiency measure the effective sample sizes per unit of computation time relative to RW. Effective sample sizes are computed using the coda R package (Plummer et al. 2006) and averaged over five summary statistics (each summary statistics being the Hamming distance from a matching randomly drawn from the posterior). From Table 4, we can see that LB provides roughly two orders of magnitude improvement in efficiency over RW and GB and one order of magnitude improvement over HB. Indeed from Figure 5(c), we can see that, given the same computational budget, LB manages to provide much more reliable estimates of the posterior probabilities of each couple being matched compared to HB. Computations were performed using the *R* programming language. For each region we ran LB for 35,000 iterations (enough to produce reliable estimates of posterior pairwise matching probabilities like the one in Figure 5(c)), requiring on average around 120 s per region.

Next, we consider the task of performing recording linkage on the whole Italy dataset, without dividing it first into regions. In fact the latter operation (often called deterministic blocking in the record linkage literature) is typically done for computational reasons but has the drawback of excluding a priori possible matches across groups (in this case regions). We apply LB to the whole dataset using the block-wise implementation discussed in Section 6.4 (more details in Supplement B). Standard output analysis suggests that the LB chain is converging fast (Figure 6, left) and mixing well in the region of substantial probability (Figure 6, center). Comparing independent runs of the algorithms suggests that we are obtaining reliable estimates of the nearly one billion posterior pairwise matching probabilities (Figure 6, right). The simulation needed to obtain the probability estimates in Figure 6 (right) took less than 40 min to run with a plain *R* implementation on a single desktop computer.

8. Discussion and Future Work

In this work, we discussed a fundamental and yet not satisfactorily answered question in the MCMC methodology literature, which is how to design "informed" MH proposals in discrete spaces. We proposed a simple and original framework (point-

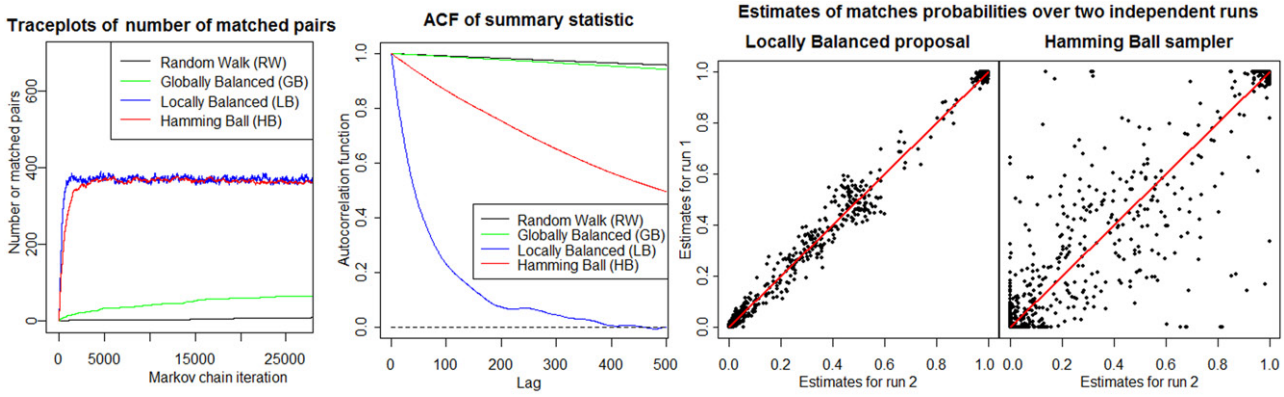

Figure 5. Output analysis for the four MCMC schemes under comparison applied to the Italy dataset (region 1).

Table 4. Relative efficiency (defined as ESS/time) of GB, LB, and HB compared to RW .

Region	1	2	3	4	5	6	7	8	Average
GB vs. RW	0.8	2.9	0.7	1.7	1.4	0.5	1.0	0.9	0.94
LB vs. RW	60.6	209	36.2	127	48.9	104	172	33.3	94.0
HB vs. RW	6.3	20.3	3.7	15.3	7.2	10.9	15.6	3.1	9.96

NOTE: The table reports the value for the first 8 regions and the average over all 20 regions.

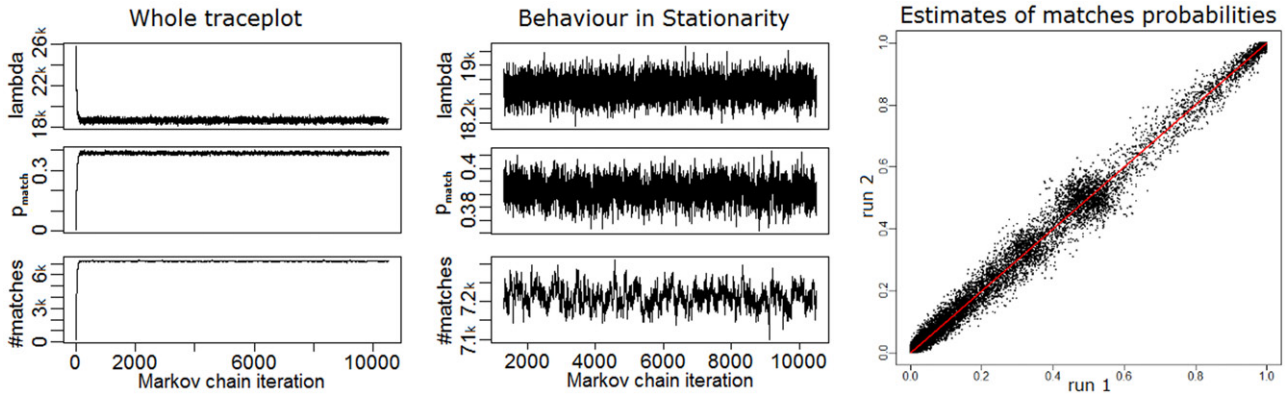

Figure 6. Output analysis for locally balanced MCMC applied to the full Italy dataset (see Section 7).

wise informed proposals and balancing functions) that provides useful and easy-to-implement methodological guidance. Under regularity assumptions, we were able to prove the asymptotic optimality of locally balanced proposals in high-dimensional regimes and to identify the optimal elements within such class. The theoretical results of Sections 2–4 are confirmed by simulations (Sections 6 and 7), where we observe orders of magnitude improvements in efficiency over alternative schemes for both simulated and real data scenarios. We envisage that locally balanced proposals could be helpful in various contexts to be incorporated as a building block in more elaborate Monte Carlo algorithms. The proposed methodology can be applied to arbitrary statistical models with discrete-valued parameters, such as Bayesian Nonparametric models or model selection problems.

The present work offers many directions for possible extensions and future work. For example, the connection to MALA in Section 5 could be exploited to improve the robustness of gradient-based MCMC schemes and reduce the notoriously heavy burden associated with their tuning procedures. Also, it would be interesting to extend our study to the popular context of multiple-try Metropolis (MTM) schemes (see Section 6). In fact, the MTM weight function, used to select the proposed point among candidate ones, plays a role that is very similar to the one of multiplicative biasing terms in pointwise informed proposals and we expect our results to translate quite naturally to that context. We already started exploring this connection and our preliminary results suggest that multiple-try schemes based on balancing functions can be more efficient and dramatically more robust than standard MTM. The Multiple-Try framework, however, is more complex to analyze compared to the one of locally balanced proposals because of the additional randomness arising from the random selection of candidate points and because the interaction between the algorithm efficiency and the number of candidate points is subtle. Thus, it seems harder to obtain theoretical results such as the ones of Section 3 in that context.

In terms of implementation, it would be interesting to explore in more depth the trade-off between computational cost per iteration and statistical efficiency of the resulting Markov chain. Beyond the use of block-wise implementations discussed in Section 6.4, another approach to reduce the cost per iteration would be to replace the informed term $g(\frac{\pi(y)}{\pi(x)})$ in the proposal with some cheap-to-evaluate approximation, while still using the exact target in the MH accept/reject step. Also, the computations required to sample from locally balanced proposals are trivially parallelizable and specific hardware for parallel computations, such as Graphics Processing Units (GPUs), could be used to reduce the computational overhead required by using informed proposals (Lee et al. 2010).

From the theoretical point of view, it would be interesting to provide guidance for a regime which is intermediate between local and global, maybe by designing appropriate interpolations between locally balanced and globally balanced proposals. This could be useful to design schemes that adaptively learn the appropriate level of interpolation needed. Also, one could work on extending Theorem 5 in order to provide sufficient conditions for (A) that hold in more general contexts. Finally, throughout the article, we assumed the base kernel K_σ to be symmetric and it would be interesting to extend the results to the case of general base kernels.

Supplementary Material

A - Proofs: contains proofs of Theorems 1-6 and Proposition 1. (pdf)
B - Details on simulations: additional details and results for the discussion in Section 4.2, the simulation studies in Section 6 and the Bayesian record linkage application in Section 7. (pdf)
C - Code: *R* code to reproduce the simulation studies in Section 6. (zip)

Acknowledgments

The author is grateful to Samuel Livingstone, Omiros Papaspiliopoulos and Gareth Roberts for stimulating discussions, and thanks the Editor, Associate Editor and referees for useful and constructive comments.

Funding

This work was supported in part by an *Engineering and Physical Sciences Research Council* (EPSRC) Doctoral Prize fellowship and by the *European Research Council* (ERC) through StG "N-BNP" 306406.

References

Barker, A. A. (1965), "Monte Carlo Calculations of the Radial Distribution Functions for a Proton–Electron Plasma," *Australian Journal of Physics*, 18, 119–134. [853,858]

Briscolini, D., Di Consiglio, L., Liseo, B., Tancredi, A., and Tuoto, T. (2018), "New Methods for Small Area Estimation With Linkage Uncertainty," *International Journal of Approximate Reasoning*, 94, 30–42. [862]

Copas, J., and Hilton, F. (1990), "Record Linkage: Statistical Models for Matching Computer Records," *Journal of the Royal Statistical Society*, Series A, 153, 287–320. [862]

Dellaert, F., Seitz, S., Thorpe, C., and Thrun, S. (2003), "EM, MCMC, and Chain Flipping for Structure From Motion With Unknown Correspondence," *Machine Learning*, 50, 45–71. [859]

Durmus, A., Roberts, G. O., Vilmart, G., Zygalakis, K. C. (2017), "Fast langevin Based Algorithm for MCMC in High Dimensions," *The Annals of Applied Probability*, 27, 2195–2237. [852]

Fellegi, I. P., and Sunter, A. B. (1969), "A Theory for Record Linkage," *Journal of the American Statistical Association*, 64, 1183–1210. [862]

Girolami, M., and Calderhead, B. (2011), "Riemann Manifold Langevin and Hamiltonian Monte Carlo Methods," *Journal of the Royal Statistical Society*, Series B, 73, 123–214. [852]

Green, P. J., and Mardia, K. V. (2006), "Bayesian Alignment Using Hierarchical Models, With Applications in Protein Bioinformatics," *Biometrika*, 93, 235–254. [862]

Hastings, W. K. (1970), "Monte Carlo Sampling Methods Using Markov Chains and Their Applications," *Biometrika*, 57, 97–109. [852]

Hurn, M. (1997), "Difficulties in the Use of Auxiliary Variables in Markov Chain Monte Carlo Methods," *Statistics and Computing*, 7, 35–44. [861]

Jerrum, M., and Sinclair, A. (1996), "The Markov Chain Monte Carlo method: An Approach to Approximate Counting and Integration," in *Approximation Algorithms for NP-hard Problems*, Boston: PWS Publishing Co., pp. 482–520. [859]

Johndrow, J. , Lum, K., and Dunson, D. (2018), " Theoretical Limits of Microclustering for Record Linkage," *Biometrika*, 105, 431–446. [862]

Lee, A., Yau, C., Giles, M. B., Doucet, A., and Holmes, C. C. (2010), "On the Utility of Graphics Cards to Perform Massively Parallel Simulation of Advanced Monte Carlo Methods," *Journal of Computational and Graphical Statistics*, 19, 769–789. [864]

Levin, D., Peres, Y., and Wilmer, E. (2009), *Markov Chains and Mixing Times*, Providence, RI: American Mathematical Society. [854]

Liu, J. S., Liang, F., and Wong, W. H. (2000), "The Multiple-Try Method and Local Optimization in Metropolis Sampling," *Journal of the American Statistical Association*, 95, 121–134. [859]

McVeigh, B. S., and Murray, J. S. (2017), "Practical Bayesian Inference for Record Linkage," arXiv preprint arXiv:1710.10558. [862]

Metropolis, N., Rosenbluth, A. W. , Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953), "Equation of State Calculations by Fast Computing Machines," *The Journal of Chemical Physics*, 21, 1087–1092. [852]

Moores, M. T., Hargrave, C. E., Deegan, T., Poulsen, M., Harden, F., and Mengersen, K. (2015a), "An External Field Prior for the Hidden Potts Model With Application to Cone-Beam Computed Tomography," *Computational Statistics & Data Analysis*, 86, 27–41. [860]

Moores, M. T., Pettitt, A. N., and Mengersen, K. "Scalable Bayesian Inference for the Inverse Temperature of a Hidden Potts Model," arXiv preprint arXiv:1503.08066, 2015b. [861]

Neal, R. (2011), "MCMC Using Hamiltonian Dynamics," in *Handbook of Markov Chain Monte Carlo*, eds. S. Brooks, A. Gelman, G. Jones, and X. Meng, New York: CRC Press, pp. 113–162. [852]

Nishimura, A., Dunson, D., and Lu, J. (2017), "Discontinuous Hamiltonian Monte Carlo for Sampling Discrete Parameters," arXiv preprint arXiv:1705.08510. [853]

Oh, S., Russell, S., and Sastry, S. "Markov Chain Monte Carlo Data Association for Multi-Target Tracking," *IEEE Transactions on Automatic Control*, 54, 481–497, 2009. [859]

Pakman, A., and Paninski, L. (2013), "Auxiliary-Variable Exact Hamiltonian Monte Carlo Samplers for Binary Distributions," in *Advances in Neural Information Processing Systems*, 2490–2498. [853,859]

Peskun, P. (1973), "Optimum Monte-Carlo Sampling Using Markov Chains," *Biometrika*, 60, 607–612. [854,857]

Plummer, M., Best, N., Cowles, K., and Vines. K. (2006), "Coda: Convergence Diagnosis and Output Analysis for MCMC," *R News*, 6, 7–11, available at *https://journal.r-project.org/archive/*. [863]

Robert, C., and Casella, G. (2005), *Monte Carlo Statistical Methods*, Heidelberg: Springer-Verlag Berlin. [852]

Roberts, G., and Rosenthal, J. (1998), "Optimal Scaling of Discrete Approximations to Langevin Diffusions," *Journal of the Royal Statistical Society*, Series B, 60, 255–268. [852,858]

Roberts, G., Gelman, A., and Gilks, W. (1997), "Weak Convergence and Optimal Scaling of Random Walk Metropolis Algorithms," *The Annals of Applied Probability*, 7, 110–120. [857]

Sadinle, M. (2017), "Bayesian Estimation of Bipartite Matchings for Record Linkage," *Journal of the American Statistical Association*, 112, 1–13. [862]

Schäfer, C., and Chopin, N. (2013), "Sequential Monte Carlo on Large Binary Sampling Spaces," *Statistics and Computing*, 23, 163–184. [859]

Steorts, R. C. (2015) "Entity Resolution With Empirically Motivated Priors," *Bayesian Analysis*, 10, 849–875. [863]

Steorts, R. C., Hall, R., and Fienberg, S. E. (2016), "A Bayesian Approach to Graphical Record Linkage and Deduplication," *Journal of the American Statistical Association*, 111, 1660–1672. [862,863]

Swendsen, R. H., and Wang, J.-S. (1987), "Nonuniversal Critical Dynamics in Monte Carlo Simulations," *Physical Review Letters*, 58, 86. [861]

Tancredi, A., and Liseo, B. (2011), "A Hierarchical Bayesian Approach to Record Linkage and Population Size Problems," *The Annals of Applied Statistics*, 5, 1553–1585. [862]

Tierney, L. (1998), "A Note on Metropolis–Hastings Kernels for General State Spaces," *Annals of Applied Probability*, 8, 1–9. [854,857]

Titsias, M. K., and Papaspiliopoulos, O. (2018), "Auxiliary Gradient-based Sampling Algorithms," *Journal of the Royal Statistical Society*, Series B, 80, 749–767. [852,858]

Titsias, M. K., and Yau, C. (2017), "The Hamming Ball Sampler," *Journal of the American Statistical Association*, 112, 1–14. [853,859,861,862]

Welling, M., and Teh, Y. (2011), "Bayesian Learning Via Stochastic Gradient Langevin Dynamics," *Proceedings of the 28th International Conference on Machine Learning* (ICML-11), Bellevue, WA, 681–688. [852]

Zanella, G. (2015), "Random Partition Models and Complementary Clustering of Anglo-Saxon Place-Names," *The Annals of Applied Statistics*, 9, 1792–1822. [859]

Zanella, G., Betancourt, B., Wallach, H., Miller, J., Zaidi, A., and Steorts, R. C. (2016), "Flexible Models for Microclustering With Application to Entity Resolution," *Advances in Neural Information Processing Systems*, 1417–1425. [862]

Zhang, Y., Ghahramani, Z., Storkey, A. J., and Sutton, C. A. (2012), "Continuous Relaxations for Discrete Hamiltonian Monte Carlo," in *Advances in Neural Information Processing Systems*, eds. F. Pereira, C. J. C. Burges, L. Bottou and K. Q. Weinberger, Lake Tahoe, NV: Curran Associates, Inc., 3194–3202. [853]