Received: 17 June 2021 | Revised: 21 April 2022 | Accepted: 6 July 2022

DOI: 10.1111/rssb.12546

ORIGINAL ARTICLE

ROYAL STATISTICAL SOCIETY Series B Statistical Methodology

Dimension-free mixing for high-dimensional Bayesian variable selection

Quan Zhou[1] | **Jun Yang**[2] | **Dootika Vats**[3] | **Gareth O. Roberts**[4] | **Jeffrey S. Rosenthal**[5]

[1]Department of Statistics, Texas A&M University, College Station, Texas, USA

[2]Department of Statistics, University of Oxford, Oxford, UK

[3]Department of Mathematics and Statistics, Indian Institute of Technology Kanpur, Kanpur, India

[4]Department of Statistics, University of Warwick, Coventry, UK

[5]Department of Statistical Sciences, University of Toronto, Toronto, Ontario, Canada

Correspondence
Jeffrey S. Rosenthal, Department of Statistical Sciences, University of Toronto, Toronto, ON, Canada.
Email: jeff@math.toronto.edu

Funding information
Engineering and Physical Sciences Research Council, Grant/Award Numbers: EP/R018561/1, EP/R034710/1; Natural Sciences and Engineering Research Council of Canada, Grant/Award Number: RGPIN-2019-04142; Science and Engineering Research Board, Grant/Award Number: SPG/2021/001322

Abstract

Yang et al. proved that the symmetric random walk Metropolis–Hastings algorithm for Bayesian variable selection is rapidly mixing under mild high-dimensional assumptions. We propose a novel Markov chain Monte Carlo (MCMC) sampler using an informed proposal scheme, which we prove achieves a much faster mixing time that is independent of the number of covariates, under the assumptions of Yang et al. To the best of our knowledge, this is the first high-dimensional result which rigorously shows that the mixing rate of informed MCMC methods can be fast enough to offset the computational cost of local posterior evaluation. Motivated by the theoretical analysis of our sampler, we further propose a new approach called 'two-stage drift condition' to studying convergence rates of Markov chains on general state spaces, which can be useful for obtaining tight complexity bounds in high-dimensional settings. The practical advantages of our algorithm are illustrated by both simulation studies and real data analysis.

KEYWORDS

add-delete-swap sampler, drift condition, finite Markov chain, genome-wide association study, informed MCMC, rapid mixing

1 | INTRODUCTION

Consider a variable selection problem where we observe an $n \times p$ design matrix X and a response vector y; each column of X represents a covariate. The goal is to identify the set of all 'influential'

covariates which have non-negligible effects on y; we denote this set by γ. We are mostly interested in a high-dimensional setting where p is much larger than the sample size n but most of the covariates have either zero or negligible effects. Due to this sparsity assumption, we can choose some threshold s_0, which may grow with n, and assume that the unknown parameter γ takes value in the space

$$\mathcal{M}(s_0) = \{\gamma \subseteq \{1, 2, \ldots, p\} : |\gamma| \leq s_0\},$$

where $|\cdot|$ denotes the cardinality of a set. By assigning a prior distribution on $\mathcal{M}(s_0)$ and then updating it using the data, we can compute the posterior distribution of γ, denoted by $\pi_n(\gamma)$ (Chipman et al., 2001). One advantage of the Bayesian approach is that we can make inferences by averaging over π_n, a property known as model averaging (Kass & Raftery, 1995). This is different from methods such as penalized regression, where we aim to find a single best model that minimises some loss function. For theoretical results on Bayesian variable selection in high-dimensional settings, see Johnson and Rossell (2012), Narisetty and He (2014), Castillo et al. (2015), Jeong and Ghosal (2021), among many others.

1.1 | Background and main contributions of this work

The calculation of π_n is usually performed by Markov chain Monte Carlo (MCMC) sampling, including both Metropolis–Hastings (MH) and Gibbs algorithms (Brown et al., 1998; George & McCulloch, 1993, 1997; Guan & Stephens, 2011); see O'Hara and Sillanpää (2009) for a review. For problems with extremely large p, the efficiency of the MCMC sampler largely depends on how we propose the next state given current state γ. Zanella (2020) considered the so-called locally informed proposal schemes on general discrete state spaces, which assign a proposal weight to each neighboring state γ' using some function of $\pi_n(\gamma')/\pi_n(\gamma)$. Though variable selection was not discussed explicitly in Zanella (2020), similar ideas are utilized in most state-of-the-art MCMC methods for variable selection. Examples include the tempered Gibbs sampler of Zanella and Roberts (2019) and the ASI (adaptively scaled individual adaptation) proposal of Griffin et al. (2021), both of which require calculating $\pi_n(\gamma')$ (up to the normalizing constant) for each $\gamma' \in \mathcal{N}_1(\gamma) = \{\gamma' : |\gamma \triangle \gamma'| = 1\}$, where $\triangle$ denotes the symmetric set difference. In a similar spirit, the Hamming ball sampler of Titsias and Yau (2017) performs an exact sampling according to π_n within a randomly selected subset of the neighbourhood of γ. For non-MCMC algorithms, we note that the design of the shotgun stochastic search (Hans et al., 2007; Shin et al., 2018) bears a striking resemblance to informed proposals.

Intuitively, informed proposals rely on the following idea: avoid visiting states with low posterior probabilities by carefully tuning the proposal probabilities. Though this seems very appealing, it naturally comes at the computational cost of evaluating the local posterior landscape around the current state. For instance, an informed proposal that draws the next state from $\mathcal{N}_1(\cdot)$ has complexity linear in p. Whether such local evaluation of π_n is worthwhile is theoretically unclear, and convergence analysis of informed sampling algorithms (for variable selection) is very challenging because the landscape of π_n is hard to characterise, especially in high-dimensional asymptotic regimes. Indeed, even for the 'uninformed' random-walk MH algorithm (denoted by RW-MH henceforth), its mixing rate has only been obtained recently by Yang et al. (2016) under mild high-dimensional assumptions. The order of their upper bound on the mixing time is approximately $pns_0^2 \log p$ (see Remark 2), which shows that RW-MH is rapidly mixing (i.e. the mixing time

is polynomial in n and p). Then, the question is whether informed MCMC methods can achieve sufficiently fast mixing rates that can at least offset the additional computation cost.

In this work, we rigorously derive a positive answer to the above question. We consider a novel informed MH algorithm, named LIT-MH (Metropolis–Hastings with Locally Informed and Thresholded proposal distributions), which assigns bounded proposal weights to the standard add-delete-swap moves. Under the high-dimensional assumptions made in Yang et al. (2016), LIT-MH achieves a mixing time that does not depend on p. To the best of our knowledge, this is the first dimension-free mixing time result for an informed MCMC algorithm in a 'general' high-dimensional setting. (There exist similar results for special cases where the posterior distribution has independent coordinates or the design matrix is orthogonal, which are not very useful for real high-dimensional problems; see, e.g., Zanella & Roberts, 2019 and Griffin et al., 2021.) To prove the mixing rate of LIT-MH, unlike most existing approaches based on path methods, we propose a 'two-stage drift condition' method, which provides theoretical insights into the behaviour of MCMC methods for variable selection. General results for the two-stage drift condition are derived, which can be useful to other problems where multiple drift conditions hold on different parts of the state space. Simulation studies show that LIT-MH can efficiently explore the posterior distribution under various settings. A real data example is also provided, where five genetic variants associated with cup-to-disc ratio (CDR) are identified.

1.2 | Motivation for the LIT-MH algorithm

One may expect that by using an informed proposal scheme that assigns larger proposal probabilities to states with larger posterior, the resulting MH algorithm requires less iterations than RW-MH to find high posterior regions. This is not always true, and surprisingly, it is even possible that such an informed MH algorithm is slowly mixing while RW-MH is rapidly mixing.

Consider MH algorithms for variable selection that always propose the next state from $\mathcal{N}_1(\cdot)$; that is, we can either add or remove a covariate (we will consider swap moves later in Section 2.1). Suppose we assign proposal weight $\pi_n(\gamma')^\nu$ to each $\gamma' \in \mathcal{N}_1(\gamma)$ for some constant $\nu \geq 0$. That is, we can express the proposal matrix $\boldsymbol{K}_\nu$ as

$$\boldsymbol{K}_\nu(\gamma, \gamma') = \frac{\pi_n(\gamma')^\nu}{\sum_{\tilde{\gamma} \in \mathcal{N}_1(\gamma)} \pi_n(\tilde{\gamma})^\nu} \mathbb{1}_{\mathcal{N}_1(\gamma)}(\gamma'), \tag{1}$$

where $\mathbb{1}$ denotes the indicator function. When $\nu = 0$, $\boldsymbol{K}_\nu(\gamma, \cdot)$ becomes the uniform distribution on the set $\mathcal{N}_1(\gamma)$, which is uninformed. It seems desirable to choose some $\nu > 0$ so that with high probability we propose adding an influential covariate or removing a non-influential one. We give a toy low-dimensional example below, which shows that for any $\nu > 0$, the MH algorithm using $\boldsymbol{K}_\nu$ as the proposal can fail to work well when the sample size is sufficiently large.

Example 1. Suppose that there are only two influential covariates, X_1 and X_2, and $\pi_n(\{1,2\}) \gg \pi_n(\{i\}) \gg \pi_n(\emptyset) \gg \pi_n(\{j\})$ for $i = 1, 2$ and $3 \leq j \leq p$. Thus, if we start an MH algorithm at the null model, we want the chain to first move to $\{1\}$ or $\{2\}$ and then move to $\{1,2\}$. By using some $\nu > 0$, we can make the proposal probability $\boldsymbol{K}_\nu(\emptyset, \{1\} \cup \{2\})$ close to 1. Let $\boldsymbol{P}_\nu$ denote the transition matrix of the MH algorithm with proposal $\boldsymbol{K}_\nu$ given in (1). To bound the transition probability from $\emptyset$ to $\{1\}$, observe that $\boldsymbol{K}_\nu(\{1\}, \emptyset) \leq \pi_n(\emptyset)^\nu / \pi_n(\{1,2\})^\nu$, since $\{1,2\}$ is a neighbor of $\{1\}$. It then follows from the Metropolis rule that

$$\begin{aligned}\boldsymbol{P}_\nu(\emptyset,\{1\}) &= \boldsymbol{K}_\nu(\emptyset,\{1\})\min\left\{1,\ \frac{\pi_n(\{1\})}{\pi_n(\emptyset)}\frac{\boldsymbol{K}_\nu(\{1\},\emptyset)}{\boldsymbol{K}_\nu(\emptyset,\{1\})}\right\}\\ &\le \frac{\pi_n(\{1\})}{\pi_n(\emptyset)}\boldsymbol{K}_\nu(\{1\},\emptyset) \le \left\{\frac{\pi_n(\{1\})}{\pi_n(\emptyset)}\right\}^{1-\nu}\left\{\frac{\pi_n(\{1\})}{\pi_n(\{1,2\})}\right\}^{\nu}.\end{aligned} \tag{2}$$

An analogous bound holds for $\boldsymbol{P}_\nu(\emptyset,\{2\})$. It is clear from (2) that if $\nu > 1$, $\boldsymbol{P}_\nu(\emptyset,\{1\})$ can be exceedingly small.

Next, we construct a concrete example to show that even if $\nu \in (0,1]$, $\boldsymbol{P}_\nu$ may still have very poor mixing. Fix some $\nu \in (0,1]$. Let X_j denote the j th column of X. Suppose the design matrix satisfies $X_j^\top X_j = n$ for each $j \in [p]$, $X_1^\top X_2 = (\nu-1)n$, and $X_i^\top X_j = 0$ for any other $i < j$. Assume that the response vector y is generated by $y = X_1 + X_2 + z$ where z is a deterministic error vector such that $z^\top z = n$ and $X^\top z = 0$. Choose some $s_0 \ge 2$, and let the posterior distribution be given by (4) with hyperparameters $\kappa, g > 0$ (see Section 2.1 for details). Fix p, ν, κ, g and let n tends to infinity. In Section S5.1 in Appendix A, we show that

$$\boldsymbol{K}_\nu(\emptyset,\{1\}\cup\{2\}) = 1 - O(e^{-a_1 n}), \quad \boldsymbol{P}_\nu(\emptyset,\{1\}\cup\{2\}) = O(e^{-a_2 n}),$$

where $a_1, a_2 > 0$ are some constants that only depend on ν and g. Hence, the chain must be slowly mixing since $\boldsymbol{P}_\nu(\emptyset,\emptyset) = 1 - O(e^{-a_1 n}) - O(e^{-a_2 n})$. That is, the informed MH chain can get stuck at the null model for exponentially many iterations, where we keep proposing adding X_1 or X_2 but getting rejected. In contrast, one can use the path method of Yang et al. (2016) to show that RW-MH is rapidly mixing (proof is omitted).

This toy example reveals that the real challenge in developing informed MH algorithms is to bound the acceptance probability of informed proposals. From (2), we can see that in order to make $\boldsymbol{P}_\nu(\emptyset,\{1\})$ large, we need $\boldsymbol{K}_\nu(\emptyset,\{1\})$ to be sufficiently large and $\boldsymbol{K}_\nu(\{1\},\emptyset)$ not to be too small. This motivates us to use proposal weights that are bounded both from above and from below so that the proposal probability of any neighboring state is bounded as well. Further, we partition the neighbourhood of each γ according to the proposal type (e.g. addition, deletion or swap) and then perform proposal weighting in each subset separately, which also helps control the acceptance probability of informed proposal moves.

1.3 | Two-stage drift condition

Drift-and-minorization methods have been used to show rapid mixing of various MCMC algorithms (Fort et al., 2003; Johndrow et al., 2020; Qin & Hobert, 2019; Rosenthal, 1995; Roy & Hobert, 2007; Vats, 2017; Yang & Rosenthal, 2022); see Jones and Hobert (2001) for a review. These methods are particularly useful for studying Gibbs sampling algorithms on continuous state spaces. One possible reason is that to establish the drift condition, we need to bound the expected change in the drift function in the next MCMC iteration, which is easier if the next sample is drawn from a smooth full conditional distribution. For problems like high-dimensional variable selection, the posterior landscape is highly irregular and difficult to characterise. The convergence analysis becomes even more challenging for informed MH algorithms since the proposal distribution usually involves normalizing constants that do not admit simple expressions.

We prove the dimension-free mixing rate of the LIT-MH algorithm using a novel drift condition. But unlike traditional drift-and-minorization methods which only involve a single drift condition, we establish two drift conditions on two disjoint subsets of the state space separately.

Our method is motivated by the forward-backward stepwise selection (An et al., 2008) and the insights obtained in Yang et al. (2016). Let γ^* denote the model consisting of all influential covariates. We say a model γ is overfitted if $\gamma^* \subseteq \gamma$; otherwise, we say γ is underfitted. Under certain mild conditions, we expect that the posterior probability mass will concentrate on γ^*. It is then tempting to use a single drift condition that measures the distance between γ and γ^*. Unfortunately, this approach may not work. The most important reason is that for an underfitted γ, non-influential covariates may appear to be influential due to the correlation with some truly influential covariate(s) missing in γ (and similarly, some influential covariates may appear to be non-influential). Nevertheless, as in the stepwise variable selection, once the model becomes overfitted, we expect that all non-influential covariates can be easily removed. This observation suggests that we can partition $\mathcal{M}(s_0)$ into underfitted and overfitted models. On the set of overfitted models, we may construct a drift function using the distance from γ^*, and the corresponding drift condition should reflect that the chain tends to move towards γ^* by removing non-influential covariates. On the set of underfitted models, we need a different drift condition capturing the tendency of the chain to add (possibly truly non-influential) covariates, in order to explain the variation in the response variable.

We propose to use this two-stage drift condition as a general method for convergence analysis of Markov chains; all related results will be derived for general state spaces in Section 4. The flexibility of this approach could be useful to other problems where the state space has a complex topological structure. To derive a bound on the mixing time using the two-stage drift condition, we use regeneration theory as in the classical drift-and-minorization methods (Roberts & Tweedie, 1999), but it is more difficult in our case to bound the tail probability of the regeneration time. In our proof, we split the path of the Markov chain into disjoint segments using an auxiliary sequence of geometric random variables and then apply a union bound argument of Rosenthal (1995).

The use of the two-stage drift condition is critical to proving the dimension-free mixing of LIT-MH. In Yang et al. (2016), the convergence rate of RW-MH is analysed by using canonical paths (Sinclair, 1992), a method widely used for Markov chains on discrete spaces (Levin et al., 2017, chapter 14). A key step of their proof is to identify, for any $\gamma \neq \gamma^*$, a 'high-probability' path from γ to γ^* ('high-probability' means that each step of the path has a sufficiently large transition probability). A potential limitation of this approach is that for some $\gamma \neq \gamma^*$, there may exist a large number of 'high-probability' paths leading to γ^*, and if we only consider one of them, the resulting mixing time bound may be loose. This is indeed the case for our LIT-MH algorithm. In order to obtain a sharp bound on the mixing time, we need to invoke the drift condition to take into account all possible moves, and the method of canonical paths will fail to yield a dimension-free estimate for the mixing time of LIT-MH.

1.4 | Organization of the paper

In Section 2.1 we formally introduce the Bayesian variable selection problem. Key results of Yang et al. (2016) for the RW-MH algorithm are reviewed in Section 2.2, and our LIT-MH algorithm is introduced in Section 2.3. In Section 3, we construct two drift conditions for LIT-MH and then derive the mixing time bound in Theorem 1. In Section 4, we consider the two-stage drift condition in a general setting, for which the main result is presented in Theorem 2. Simulation studies are presented in Section 5, with some results provided in Section S4 in Appendix A. A real data example is provided in Section 6, where we apply the LIT-MH algorithm to genome-wide

association studies on glaucoma. Section 7 concludes the paper with some discussion on the implementation and generalisation of LIT-MH and its differences from other MCMC methods. All technical proofs are relegated to Appendix A.

2 | RW-MH AND LIT-MH ALGORITHMS FOR VARIABLE SELECTION

We first define some notation. Let $[p] = \{1, 2, \dots, p\}$. For $\gamma \subseteq [p]$, let X_γ denote the submatrix of X with columns indexed by γ, and β_γ denote the subvector with entries indexed by γ. Recall that $|\cdot|$ denotes the cardinality of a set.

2.1 | Model, prior and local proposals

Consider a sparse linear regression model,

$$y = X_\gamma \beta_\gamma + e, \quad e \sim \mathrm{MN}(0,\ \phi^{-1} I_n),$$

where MN denotes the multivariate normal distribution and I_n is the identity matrix. Hence, γ can be understood as the set of nonzero entries of β. We follow Yang et al. (2016) to consider the following prior:

$$\begin{aligned} &\text{(g-prior)} && \beta_\gamma | \gamma \sim \mathrm{MN}(0,\ g\phi^{-1}(X_\gamma^\top X_\gamma)^{-1}), \\ &\text{(precision prior)} && \pi_0(\phi) \propto \phi^{-1}, \\ &\text{(sparsity prior)} && \pi_0(\gamma) \propto p^{-\kappa_0|\gamma|} \mathbb{1}_{\mathcal{M}(s_0)}(\gamma), \\ &\text{(choice of } g\text{)} && 1 + g = p^{2\kappa_1}, \end{aligned} \tag{3}$$

where $\kappa_0, \kappa_1 > 0$ are hyperparameters, π_0 denotes the prior probability density/mass function, and we recall s_0 is the maximum model size we allow. After integrating out β, the marginal posterior probability of $\gamma \subseteq [p]$ can be computed by

$$\pi_n(\gamma) \propto p^{-\kappa|\gamma|} \left(g^{-1} y^\top y + y^\top P_\gamma^\perp y\right)^{-n/2} \mathbb{1}_{\mathcal{M}(s_0)}(\gamma), \tag{4}$$

where $\kappa = \kappa_0 + \kappa_1$ and $P_\gamma^\perp$ denotes the projection matrix:

$$P_\gamma^\perp = I_n - P_\gamma, \quad P_\gamma = X_\gamma (X_\gamma^\top X_\gamma)^{-1} X_\gamma^\top.$$

For two models γ, γ', let $B(\gamma, \gamma')$ denote their posterior probability ratio. It follows from (4) that

$$B(\gamma, \gamma') = \frac{\pi_n(\gamma')}{\pi_n(\gamma)} = p^{\kappa(|\gamma| - |\gamma'|)} \left\{ 1 + \frac{y^\top (P_\gamma - P_{\gamma'}) y}{g^{-1} y^\top y + y^\top P_\gamma^\perp y} \right\}^{-n/2}. \tag{5}$$

For MH algorithms on the space $\mathcal{M}(s_0)$, the most common approach is to use a proposal scheme consisting of three types of local moves, 'addition', 'deletion' and 'swap', which induces an

irreducible Markov chain on $\mathcal{M}(s_0)$. Explicitly, for every $\gamma \in \mathcal{M}(s_0)$, define the addition, deletion and swap neighborhoods of γ by

$$\begin{aligned}
\mathcal{N}_{\mathrm{a}}(\gamma) &= \{\gamma' \in \mathcal{M}(s_0) : \gamma' = \gamma \cup \{j\} \text{ for some } j \notin \gamma\},\\
\mathcal{N}_{\mathrm{d}}(\gamma) &= \{\gamma' \in \mathcal{M}(s_0) : \gamma' = \gamma \setminus \{k\} \text{ for some } k \in \gamma\},\\
\mathcal{N}_{\mathrm{s}}(\gamma) &= \{\gamma' \in \mathcal{M}(s_0) : \gamma' = (\gamma \cup \{j\}) \setminus \{k\} \text{ for some } j \notin \gamma,\ k \in \gamma\}.
\end{aligned} \tag{6}$$

The three sets are disjoint, and for each $\gamma \in \mathcal{M}(s_0)$, $|\mathcal{N}_{\mathrm{a}}(\gamma) \cup \mathcal{N}_{\mathrm{d}}(\gamma)| \le p$ (the equality holds if $|\gamma| < s_0$) and $|\mathcal{N}_{\mathrm{s}}(\gamma)| \le ps_0$. The definitions of these neighbourhoods can be generalised by allowing changing more covariates at one time. Let $\mathcal{N}(\gamma) = \mathcal{N}_{\mathrm{a}}(\gamma) \cup \mathcal{N}_{\mathrm{d}}(\gamma) \cup \mathcal{N}_{\mathrm{s}}(\gamma)$ denote the union of the three neighbourhoods.

2.2 | Rapid mixing of the RW-MH algorithm

Consider a proposal scheme defined by a transition probability matrix $\boldsymbol{K}_{\mathrm{rw}} : \mathcal{M}(s_0) \times \mathcal{M}(s_0) \to [0,1]$ such that

$$\boldsymbol{K}_{\mathrm{rw}}(\gamma, \gamma') = \frac{h_{\mathrm{a}}(\gamma)\mathbb{1}_{\mathcal{N}_{\mathrm{a}}(\gamma)}(\gamma')}{|\mathcal{N}_{\mathrm{a}}(\gamma)|} + \frac{h_{\mathrm{d}}(\gamma)\mathbb{1}_{\mathcal{N}_{\mathrm{d}}(\gamma)}(\gamma')}{|\mathcal{N}_{\mathrm{d}}(\gamma)|} + \frac{h_{\mathrm{s}}(\gamma)\mathbb{1}_{\mathcal{N}_{\mathrm{s}}(\gamma)}(\gamma')}{|\mathcal{N}_{\mathrm{s}}(\gamma)|}, \tag{7}$$

where $h_{\mathrm{a}}(\gamma), h_{\mathrm{d}}(\gamma), h_{\mathrm{s}}(\gamma)$ are non-negative constants that sum to 1; thus, $h_{\mathrm{a}}(\gamma)$ is the probability of proposing an addition move given current state γ. When $h_{\mathrm{a}}, h_{\mathrm{d}}, h_{\mathrm{s}}$ are all constants independent of γ, we refer to the resulting MH algorithm as asymmetric RW-MH. If $h_{\mathrm{a}}(\gamma) = |\mathcal{N}_{\mathrm{a}}(\gamma)|/2p, h_{\mathrm{d}}(\gamma) = |\mathcal{N}_{\mathrm{d}}(\gamma)|/2p$ and $h_{\mathrm{s}}(\gamma) = 1/2$, the resulting MH algorithm is called symmetric RW-MH, since $\boldsymbol{K}_{\mathrm{rw}}(\gamma, \gamma') = \boldsymbol{K}_{\mathrm{rw}}(\gamma', \gamma)$ for any $\gamma, \gamma' \in \mathcal{M}(s_0)$. We will now explain the main idea of the proof for the rapid mixing of RW-MH given in Yang et al. (2016), which will be useful later for studying the mixing time of LIT-MH. Note that though Yang et al. (2016) only considered symmetric RW-MH, their argument can be easily extended to prove the rapid mixing of asymmetric RW-MH.

A key step in the mixing time analysis of RW-MH is to characterise the shape of π_n. To this end, suppose that the true model is given by $y = X\beta^* + z$ where $z \sim \mathrm{MN}(0, \sigma_z^2 I_n)$ and define

$$\gamma^* = \{j \in [p] : |\beta_j^*| \ge \beta_{\min}\}, \quad s^* = |\gamma^*|, \tag{8}$$

where $\beta_{\min} > 0$ is some threshold. Covariates in γ^* are called 'influential', and we assume $s^* \le s_0$ so that identification of γ^* is possible. We are interested in a high-dimensional asymptotic regime where p can grow much faster than n but $s_0 \log p = O(n)$; $s_0, s^*, \gamma^*, \beta_{\min}$ are also allowed to depend on n.

Assume the entries of β^* corresponding to non-influential covariates are very close to zero so that $y - X_{\gamma^*}\beta^*_{\gamma^*}$ can be effectively treated as the noise. Denote by $y_{\mathrm{s}} = X_{\gamma^*}\beta^*_{\gamma^*}$ the signal part of y. For an overfitted model γ, the variation of the signal y_{s} is fully explained, so non-influential covariates in γ tend to be 'useless': they may happen to explain some noise, but the degree can be controlled by concentration inequalities and mild eigenvalue conditions on the design matrix X. Provided that the penalty on the model size is sufficiently large, we should be able to remove non-influential covariates from an overfitted model. If γ is underfitted (i.e. $\gamma^* \not\subseteq \gamma$), the analysis becomes much more complicated due to the correlation among the covariates. Suppose for some

$j \in \gamma^*$, $|\beta_j^*|$ is very large but X_j is not selected in the model γ. Then, non-influential covariates in $\gamma^c \setminus \gamma^*$ that are slightly correlated with X_j may be added to the model, and similarly, influential covariates in $\gamma^* \cap \gamma$ may be removed from γ due to correlation with X_j. However, a known result from forward-backward stepwise selection (An et al., 2008) guarantees that an underfitted model will eventually become overfitted in order to fully explain the signal, as long as the sample size n and $\beta_{\min}$ in (8) are sufficiently large relative to the collinearity in X.

The above reasoning implies that the following condition, under certain mild assumptions, would hold true with high probability for a sufficiently large sample size, which was proved in Yang et al. (2016, lemma 4).

Condition 1. There exist $\gamma^* \in \mathcal{M}(s_0)$ and constants $c_0, c_1 > 0$ (not depending on γ) such that the following three conditions are satisfied. (We say γ is overfitted if $\gamma^* \subseteq \gamma$, and underfitted if $\gamma^* \nsubseteq \gamma$.)

(1a) For any overfitted $\gamma \in \mathcal{M}(s_0)$ and $j \notin \gamma$, $B(\gamma, \gamma \cup \{j\}) \le p^{-c_0}$.
(1b) For any underfitted $\gamma \in \mathcal{M}(s_0)$, there exists some $j \in \gamma^* \setminus \gamma$, which may not be unique, such that $B(\gamma, \gamma \cup \{j\}) \ge p^{c_1}$.
(1c) For any underfitted γ with $|\gamma| = s_0$, there exist some $j \in \gamma^* \setminus \gamma$ and $k \in \gamma \setminus \gamma^*$, which may not be unique, such that $B(\gamma, (\gamma \cup \{j\}) \setminus \{k\}) \ge p^{c_1}$.

Though in the statement of lemma 4 of Yang et al. (2016), they set $c_0 = 2$ and $c_1 = 3$, their argument actually proved Condition 1 for at least $c_0 = 2$ and $c_1 = 4$, which suffices for the analysis to be conducted in later sections. Indeed, by modifying the universal constants in their assumptions, the same argument can prove the claim for any fixed $c_0, c_1 > 0$; see Section S3 in Appendix A, where we state this result as Theorem S2 in Appendix A and provide a sketch of the proof. In the proof, we treat the design matrix as fixed and do not make assumptions on how the columns of X are generated. In particular, the design matrix may include interaction terms which can account for potentially non-linear relationship between the response and explanatory variables.

If Condition 1 holds, given any $\gamma \neq \gamma^*$, we can increase the posterior probability by a local move: if γ is overfitted, we can remove a non-influential covariate by Condition (1a); if γ is underfitted, we can find an addition or swap move according to Condition (1b) and (1c). Thus, Condition 1 essentially assumes that π_n is unimodal on $\mathcal{M}(s_0)$ with respect to the add-delete-swap neighbourhood relation; see the definition below.

Definition 1. Given a function $\mathcal{N} : \mathcal{M}(s_0) \to 2^{\mathcal{M}(s_0)}$, we say γ is a local mode (w.r.t. $\mathcal{N}$) if $\pi_n(\gamma) \ge \max_{\gamma' \in \mathcal{N}(\gamma)} \pi_n(\gamma)$, and we say π_n is unimodal (w.r.t. $\mathcal{N}$) if there is only one local mode w.r.t. $\mathcal{N}$.

Another important consequence of Condition 1 is that, as long as $c_1 > 2c_0 > 2$, tails of π_n 'decay fast', since for any integer $k \ge 1$, we have $\pi_n(S_k) \le p^{1-c_0} \pi_n(S_{k-1})$ where $S_k = \{\gamma \in \mathcal{M}(s_0) : |\gamma \triangle \gamma^*| = k\}$ denotes the set of all models that have a Hamming distance of k from γ^*. This fact is a byproduct of the rapid mixing proof and implies that as n tends to infinity, $\pi_n(\gamma^*) \to 1$ in probability with respect to the true data-generating probability measure, a property that is often known as strong model selection consistency and has been proved for other spike-and-slab priors (Narisetty & He, 2014). To prove the rapid mixing of RW-MH, we define an operator $\mathcal{T} : \mathcal{M}(s_0) \to \mathcal{M}(s_0)$ such that $\mathcal{T}(\gamma^*) = \gamma^*$, and for any $\gamma \in \mathcal{M}(s_0) \setminus \{\gamma^*\}$, $\mathcal{T}(\gamma) \in \mathcal{N}(\gamma)$ and $B(\gamma, \mathcal{T}(\gamma)) \ge p^{c_0 \wedge c_1}$. Then, as shown in Yang et al. (2016), one can construct a canonical path ensemble, which yields a bound on the spectral gap of the transition matrix of the RW-MH

chain (Diaconis & Stroock, 1991; Sinclair, 1992). It is noteworthy that π_n can still be highly 'irregular' under Condition 1 in the sense that its p coordinates may have a very complicated dependence structure due to the collinearity in the design matrix.

2.3 | The LIT-MH algorithm

The proposal distribution of the RW-MH algorithm is not 'informed' in the sense that it is constructed without using information from π_n. But as explained in Section 1.2, a naive informed proposal scheme may lead to worse performance due to exceedingly small acceptance probabilities.

We consider a more general setup where the proposal weighting can be performed for each type of proposal separately. By modifying the transition matrix in (7), define $\boldsymbol{K}_{\text{lit}} : \mathcal{M}(s_0) \times \mathcal{M}(s_0) \to [0, 1]$ by

$$\begin{aligned} \boldsymbol{K}_{\text{lit}}(\gamma, \gamma') &= \sum_{\star = \text{'a', 'd', 's'}} \frac{h_\star(\gamma) w_\star(\gamma'|\gamma)}{Z_\star(\gamma)} \mathbb{1}_{\mathcal{N}_\star(\gamma)}(\gamma'), \\ Z_\star(\gamma) &= \sum_{\tilde{\gamma} \in \mathcal{N}_\star(\gamma)} w_\star(\tilde{\gamma}|\gamma), \end{aligned} \tag{9}$$

where $w_\star(\gamma'|\gamma) \in [0, \infty)$ denotes the proposal weight of $\gamma' \in \mathcal{N}_\star(\gamma)$ given current state γ. In words, we first sample the type of move with probabilities given by $h_{\text{a}}(\gamma), h_{\text{d}}(\gamma)$ and $h_{\text{s}}(\gamma)$. If an addition move is to be proposed, we sample a state $\gamma' \in \mathcal{N}_{\text{a}}(\gamma)$ with weight $w_{\text{a}}(\gamma'|\gamma)$. We propose to use

$$w_\star(\gamma'|\gamma) = p^{\ell_\star} \vee B(\gamma, \gamma') \wedge p^{L_\star}, \quad \text{for } \star = \text{'a', 'd', 's'}, \tag{10}$$

where $-\infty \le \ell_\star \le L_\star \le \infty$ are some constants that may depend on the type of move. This proposal scheme has two desirable properties. First, states with larger posterior probabilities are more likely to be proposed. Second, for any $\gamma' \in \mathcal{N}_\star(\gamma)$, we can bound its proposal probability from below by

$$\boldsymbol{K}_{\text{lit}}(\gamma, \gamma') = \frac{h_\star(\gamma) w_\star(\gamma'|\gamma)}{Z_\star(\gamma)} \ge \frac{h_\star(\gamma)}{|\mathcal{N}_\star(\gamma)|} p^{\ell_\star - L_\star}.$$

More generally, these two properties still hold if we replace $B(\gamma, \gamma')$ in (10) with $f(B(\gamma, \gamma'))$ for any monotone function $f : (0, \infty) \to (0, \infty)$; some related discussion will be given in Section 7.2.

Calculating the normalizing constant $Z_\star$ requires evaluating $B(\gamma, \gamma')$ for every $\gamma' \in \mathcal{N}_\star(\gamma)$. We use the method described in Zanella and Roberts (2019, supplement B), and we note that the Cholesky decomposition of $X_\gamma^\top X_\gamma$ can be obtained by efficient updating algorithms, which only have complexity $O(|\gamma|^2)$ (George & McCulloch, 1997; Smith & Kohn, 1996). Assuming that $X^\top X$ and $X^\top y$ are pre-computed, the complexity of each addition or deletion move has complexity $O(p|\gamma|^2)$ for LIT-MH and complexity $O(|\gamma|^2)$ for RW-MH. We will discuss the implementation of swap moves in Section 7.1. For extremely large p, one may first use marginal regression (i.e. simple linear regression of y against X_j for each j) to select a subset of potentially influential covariates (Fan & Lv, 2008). Denoting this subset by S, we then replace the weighting function w_{a} in (10) by

$$\tilde{w}_{\mathrm{a}}(\gamma'|\gamma) = \begin{cases} p^{\ell_{\mathrm{a}}} \vee B(\gamma,\gamma') \wedge p^{L_{\mathrm{a}}}, & \text{if } \gamma' \in \mathcal{N}_{\mathrm{a}}(\gamma),\ \gamma' \setminus \gamma \in S, \\ p^{\ell_{\mathrm{a}}}, & \text{if } \gamma' \in \mathcal{N}_{\mathrm{a}}(\gamma),\ \gamma' \setminus \gamma \notin S. \end{cases} \tag{11}$$

The function $\tilde{w}_{\mathrm{s}}$ can be defined similarly. Note that the calculation of w_{d} is much easier since $|\mathcal{N}_{\mathrm{d}}(\gamma)| = |\gamma| \le s_0$. In Section 6, we will see that such a practical implementation of the LIT-MH algorithm works well for a real dataset with $p = 328{,}129$.

3 | DIMENSION-FREE MIXING OF LIT-MH

In this section, we prove that, if the parameters of LIT-MH are properly chosen, the algorithm can achieve a dimension-free mixing rate when π_n satisfies Condition 1 (the actual data-generation mechanism and the interpretation of γ^* as the set of all influential covariates as given in (8) are irrelevant to our proof as long as Condition 1 is satisfied). To simplify the analysis, we only consider swaps when $|\gamma| = s_0$. For any γ with $|\gamma| < s_0$, with probability $1/2$ we propose to add a covariate, and with probability $1/2$ we propose to remove one; that is, we let $h_{\mathrm{a}}(\gamma) = h_{\mathrm{d}}(\gamma) = 1/2$, $h_{\mathrm{s}}(\gamma) = 0$ for the proposal matrix $\boldsymbol{K}_{\mathrm{lit}}$ given in (9). If $|\gamma| = s_0$, we let $h_{\mathrm{s}}(\gamma) = h_{\mathrm{d}}(\gamma) = 1/2$ and $h_{\mathrm{a}}(\gamma) = 0$. Thus, $\boldsymbol{K}_{\mathrm{lit}}$ can be written as

$$\begin{aligned} \boldsymbol{K}_{\mathrm{lit}}(\gamma,\gamma') &= \frac{w_{\mathrm{a}}(\gamma|\gamma')}{2Z_{\mathrm{a}}(\gamma)} \mathbb{1}_{\mathcal{N}_{\mathrm{a}}(\gamma)}(\gamma') + \frac{w_{\mathrm{d}}(\gamma|\gamma')}{2Z_{\mathrm{d}}(\gamma)} \mathbb{1}_{\mathcal{N}_{\mathrm{d}}(\gamma)}(\gamma'), \quad \text{if } |\gamma| < s_0, \\ \boldsymbol{K}_{\mathrm{lit}}(\gamma,\gamma') &= \frac{w_{\mathrm{s}}(\gamma|\gamma')}{2Z_{\mathrm{s}}(\gamma)} \mathbb{1}_{\mathcal{N}_{\mathrm{s}}(\gamma)}(\gamma') + \frac{w_{\mathrm{d}}(\gamma|\gamma')}{2Z_{\mathrm{d}}(\gamma)} \mathbb{1}_{\mathcal{N}_{\mathrm{d}}(\gamma)}(\gamma'), \quad \text{if } |\gamma| = s_0. \end{aligned} \tag{12}$$

This is different from the symmetric RW-MH algorithm, where the probability of proposing a deletion move is only $O(s_0/p)$ since states in $\mathcal{N}_{\mathrm{a}}(\gamma) \cup \mathcal{N}_{\mathrm{d}}(\gamma)$ are proposed randomly with equal probability.

For the weighting functions $w_{\mathrm{a}}, w_{\mathrm{d}}, w_{\mathrm{s}}$, we assume

$$\begin{aligned} w_{\mathrm{a}}(\gamma'|\gamma) &= B(\gamma,\gamma') \wedge p^{c_1}, \\ w_{\mathrm{d}}(\gamma'|\gamma) &= 1 \vee B(\gamma,\gamma') \wedge p^{c_0}, \\ w_{\mathrm{s}}(\gamma'|\gamma) &= ps_0 \vee B(\gamma,\gamma') \wedge p^{c_1}, \end{aligned} \tag{13}$$

where B is the posterior probability ratio given in (5) and c_0, c_1 are constants given in Condition 1. Clearly, they are special cases of the general form given in (10). Other choices of the threshold values may also yield the same mixing rate. For example, one may use $w_{\mathrm{a}}(\gamma'|\gamma) = B(\gamma,\gamma')$, and the proof will be essentially the same. However, for w_{d} and w_{s}, the use of two-sided thresholds is critical.

3.1 | Two-stage drift condition for LIT-MH

Let $\boldsymbol{K}_{\mathrm{lit}}$ be as defined in (12) and (13) and $\boldsymbol{P}_{\mathrm{lit}}$ denote the corresponding transition matrix, which is given by

$$\boldsymbol{P}_{\text{lit}}(\gamma,\gamma') = \begin{cases} \boldsymbol{K}_{\text{lit}}(\gamma,\gamma')\min\left\{1,\ \frac{\pi_n(\gamma')\boldsymbol{K}_{\text{lit}}(\gamma',\gamma)}{\pi_n(\gamma)\boldsymbol{K}_{\text{lit}}(\gamma,\gamma')}\right\}, & \text{if } \gamma' \neq \gamma, \\ 1-\sum_{\gamma'\neq\gamma}\boldsymbol{P}_{\text{lit}}(\gamma,\gamma'), & \text{if } \gamma' = \gamma. \end{cases} \tag{14}$$

For any function f, let $(\boldsymbol{P}_{\text{lit}}f)(\gamma) = \sum_{\gamma'} f(\gamma')\boldsymbol{P}_{\text{lit}}(\gamma,\gamma')$. If for some set $\mathsf{A} \subset \mathcal{M}(s_0)$, function $V : \mathcal{M}(s_0) \to [1,\infty)$ and constant $\lambda \in (0,1)$,

$$(\boldsymbol{P}_{\text{lit}}V)(\gamma) \leq \lambda V(\gamma), \quad \forall\ \gamma \in \mathsf{A}, \tag{15}$$

we say the LIT-MH chain satisfies a drift condition on A, which implies that the entry time of the LIT-MH chain into A^c has a 'thin-tailed' distribution (see Lemma S1 in Appendix A).

To analyse the convergence rate of the LIT-MH algorithm, we will establish two drift conditions, one for underfitted models and the other for overfitted models. The two conditions jointly imply that, if initialized at an underfitted model, the LIT-MH chain tends to first move to some overfitted model and then move to γ^*. Then, general results for the two-stage drift condition to be proved in Section 4 (see Theorem 2 and Corollary 2) can be used to derive a bound on the mixing time of LIT-MH. Define

$$\mathcal{O} = \mathcal{O}(\gamma^*, s_0) = \{\gamma \in \mathcal{M}(s_0) : \gamma^* \subseteq \gamma\},$$

which denotes the set of all overfitted models in $\mathcal{M}(s_0)$. The two drift functions we choose are given by

$$V_1(\gamma) = \left(1 + \frac{y^\top P_\gamma^\perp y}{g^{-1}y^\top y}\right)^{1/\log(1+g)}, \quad V_2(\gamma) = e^{|\gamma\setminus\gamma^*|/s_0}, \tag{16}$$

where we recall $1+g = p^{2\kappa_1}$ defined in (3). If the current model $\gamma \notin \mathcal{O}$, we expect that $V_1(\gamma)$ tends to decrease in the next iteration since some covariates can be added to explain the variation of the signal. If $\gamma \in \mathcal{O}\setminus\{\gamma^*\}$, $V_2(\gamma)$ tends to decrease since non-influential covariates in γ can be removed. For convenience, we introduce the notation

$$R_i(\gamma,\gamma') = \frac{V_i(\gamma')}{V_i(\gamma)} - 1, \quad i = 1,2.$$

We summarize the properties of functions V_1, V_2, R_1, R_2 in the following lemma.

Lemma 1. *Assume $s_0 \geq 1$. For any $\gamma,\gamma' \in \mathcal{M}(s_0)$, the following statements hold.*

(i) $1 \leq V_1(\gamma) \leq e$ *and* $1 \leq V_2(\gamma) \leq e$.
(ii) *For any $j \notin \gamma$, $R_1(\gamma, \gamma\cup\{j\}) \leq 0$; for any $k \in \gamma$, $R_1(\gamma,\gamma\setminus\{k\}) \geq 0$.*
(iii) *For any $j \in (\gamma\cup\gamma^*)^c$ and $k \in \gamma\setminus\gamma^*$,*

$$R_2(\gamma,\gamma\cup\{j\}) \leq \frac{2}{s_0}, \quad R_2(\gamma,\gamma\setminus\{k\}) \leq -\frac{1}{2s_0}.$$

Proof. See Section S5.2 in Appendix A.

Since $\boldsymbol{P}_{\text{lit}}(\gamma,\gamma) = 1 - \sum_{\gamma'\neq\gamma}\boldsymbol{P}_{\text{lit}}(\gamma,\gamma')$, some algebra yields that, for $i = 1,2$,

$$\begin{aligned}\frac{(\boldsymbol{P}_{\text{lit}}V_i)(\gamma)}{V_i(\gamma)} &= 1 + \sum_{\gamma' \neq \gamma} R_i(\gamma, \gamma')\boldsymbol{P}_{\text{lit}}(\gamma, \gamma'), \\ &= 1 + \sum_{\star = \text{'a', 'd', 's'}} \sum_{\gamma' \in \mathcal{N}_\star(\gamma)} R_i(\gamma, \gamma')\boldsymbol{P}_{\text{lit}}(\gamma, \gamma').\end{aligned} \tag{17}$$

Therefore, we only need to bound the sum of $R_i(\gamma, \gamma')\boldsymbol{P}_{\text{lit}}(\gamma, \gamma')$ for three types of proposals separately. Since by (12), the proposal probability of any move is bounded by $1/2$, we have, for any $\gamma' \neq \gamma$,

$$\boldsymbol{P}_{\text{lit}}(\gamma, \gamma') = \min\left\{\boldsymbol{K}_{\text{lit}}(\gamma, \gamma'),\ B(\gamma, \gamma')\boldsymbol{K}_{\text{lit}}(\gamma', \gamma)\right\} \leq B(\gamma, \gamma')/2. \tag{18}$$

Consider the case of overfitted models first. Let $\gamma \in \mathcal{M}(s_0)$ be overfitted. By Condition (1a), if we remove any non-influential covariate from γ, we will get a model with a much larger posterior probability; by Condition (1b), if we remove any influential covariate from γ, we will get an underfitted model with a much smaller posterior probability. As a result, when a deletion move is proposed, the covariate to be removed will be non-influential with high probability. The resulting change in V_2 can be bounded using Lemma 1(iii). Note that we also need to bound the probability of adding this covariate back so that we can show the acceptance probability of the desired deletion move is large. The case of adding a non-influential covariate is easier to handle; one just needs to use Lemma 1(iii) and the inequality in (18). The argument for swap moves is essentially a combination of those for addition and deletion moves. The bounds we find for the summation term in (17) are given in Lemma S3, from which we obtain the drift condition for overfitted models.

Proposition 1. *Suppose Condition 1 holds for some $c_0 \geq 2$ and $c_1 \geq 1$. For any overfitted model γ such that $\gamma \neq \gamma^*$ and $|\gamma| \leq s_0$,*

$$\frac{(\boldsymbol{P}_{\text{lit}}V_2)(\gamma)}{V_2(\gamma)} = 1 - \frac{1}{4s_0} + O\left(\frac{1}{ps_0}\right).$$

Proof. It follows from (17) and the bounds provided in Lemma S3.

Remark 1. By Proposition 1 and Lemma 1(i), if we consider the LIT-MH chain restricted to the set $\mathcal{O}$, the mixing time has order at most s_0. The order of this bound is sharp. Consider the worst case where $\gamma^* = \emptyset$ and $|\gamma| = s_0$. Then we need approximately $2s_0$ steps to remove all the covariates in γ.

Next, consider the set of all underfitted models. Comparing the expression of V_1 in (16) with that of π_n in (4), we see that a lower bound on $B(\gamma, \gamma')$ can yield an upper bound on $R_1(\gamma, \gamma')$. This is proven in Lemma S5 in Section S5 in Appendix A. Just like in the analysis of overfitted models, we will bound $R_1(\gamma, \gamma')\boldsymbol{P}_{\text{lit}}(\gamma, \gamma')$ for three types of proposals separately. In particular, by Lemma 1(ii), we need to bound the increase in V_1 when we remove any covariate and show that the expected decrease in V_1 is sufficiently large when we use the addition move (or swap move). However, the calculation is much more complicated than in the overfitted case. By Condition (1b), we know that there exists at least one model in $\mathcal{N}_{\text{a}}(\gamma)$ which has a much smaller value of V_1; denote this model by $\gamma \cup \{j^*\}$. But in an extreme case, we may have $B(\gamma, \gamma \cup \{j\}) \geq p^{c_1}$ for every $j \notin \gamma$. This happens when, for some $j^* \in [p]$, $|\beta^*_{j^*}|$ is extremely large and every non-influential covariate is slightly correlated with X_{j^*}. Hence, the proposal probability, $\boldsymbol{K}_{\text{lit}}(\gamma, \gamma \cup \{j^*\})$, may be as small as $O(p^{-1})$, and if we only consider the best addition move, the bound on the mixing time

will have a factor of p. This is the main reason why the path method used by Yang et al. (2016) is unable to yield a dimension-free mixing time bound for LIT-MH. To overcome this problem, we will directly bound the sum of $R_1(\gamma, \gamma')\boldsymbol{P}_{\text{lit}}(\gamma, \gamma')$ over all possible addition moves and take into account 'good' moves other than $\gamma \cup \{j^*\}$. The same technique is also needed for the analysis of swap moves. The following proposition gives the drift condition for underfitted models, where we recall $\kappa = \kappa_0 + \kappa_1$.

Proposition 2. *Suppose that $n = O(p)$, $s_0 \log p = O(n)$, $\kappa = O(s_0)$, and Condition 1 holds for some c_1 such that*

$$(c_0 + 1) \vee 4 \le c_1 \le n\kappa_1 - \kappa.$$

For any underfitted model $\gamma \in \mathcal{M}(s_0)$,

$$\frac{(\boldsymbol{P}_{\text{lit}} V_1)(\gamma)}{V_1(\gamma)} \le 1 - \frac{c_1}{8n\kappa_1} + o\left(\frac{1}{n\kappa_1}\right).$$

Proof. It follows from (17) and Lemma S6.

3.2 | Mixing time of the LIT-MH algorithm

The remaining challenge is to find a mixing time bound for the LIT-MH chain by combining the two drift conditions derived in Propositions 1 and 2. This is a very interesting problem in its own right and will be investigated in full generality in the next section. Applying Corollary 2 (which will be presented in Section 4.2), we find the following mixing time bound for the LIT-MH algorithm.

Theorem 1. *Consider the Markov chain LIT-MH defined by (12), (13) and (14) with stationary distribution π_n given in (4). Define the mixing time of LIT-MH by*

$$T_{\text{mix}} = \sup_{\gamma \in \mathcal{M}(s_0)} \min\{t \ge 0 : \|\boldsymbol{P}_{\text{lit}}^t(\gamma, \cdot) - \pi_n(\cdot)\|_{\text{TV}} \le 1/4\},$$

where $\|\cdot\|_{\text{TV}}$ denotes the total variation distance. Suppose that $n = O(p)$, $s_0 \log p = O(n)$ and $\kappa = O(s_0)$. If Condition 1 holds for $c_0 = 2$ and $4 \le c_1 \le n\kappa_1 - \kappa$, then, for sufficiently large n, we have

$$T_{\text{mix}} \le 800 \max\left\{\frac{n\kappa_1}{c_1},\ 3s_0\right\}.$$

Proof. See Section S5.5 in Appendix A.

Remark 2. The assumptions we have made in Theorem 1 are mild and are essentially the same as those of Yang et al. (2016). First, it is known that $s_0 \log p = O(n)$ is a necessary condition for estimation consistency in high-dimensional sparse regression models; Yang et al. (2016) assumed that $s_0 \log p \le n/32$. Second, the prior parameter choice $\kappa = O(s_0)$ is very reasonable. Indeed, if κ grows faster than $n/\log p$, for the data-generating model considered in Section 2.2, the threshold $\beta_{\min}$ in (8) needs to go to infinity for consistent model selection,

which would be of little interest in most applications. Note that we can always let κ_1 be a fixed positive constant, and then the mixing time bound in Theorem 1 is at most $O(n)$. One can even prove Condition 1 for some c_1 growing with n (e.g. by letting $\beta_{\min}$ in (8) be sufficiently large), in which case it is possible for our bound to only grow at rate s_0. For comparison, the upper bound on the mixing time of the symmetric RW-MH algorithm given in Yang et al. (2016, theorem 2) is $O(ps_0^2(n\kappa_1 + s_0\kappa_1 + s_0\kappa_0)\log p)$.

4 | GENERAL RESULTS FOR THE TWO-STAGE DRIFT CONDITION

For the LIT-MH algorithm, we have established two drift conditions, one for underfitted models and the other for overfitted models. In this section, we derive some general results for using such a two-stage drift condition to bound the mixing time of a Markov chain (not necessarily the LIT-MH chain), which we denote by $(\mathsf{X}_t)_{t\in\mathbb{N}}$ where $\mathbb{N} = \{0, 1, 2, \ldots\}$. We only need to require the following assumption on $(\mathsf{X}_t)_{t\in\mathbb{N}}$, and note that the underlying state space may not be discrete.

Assumption A. $(\mathsf{X}_t)_{t\in\mathbb{N}}$ is a Markov chain defined on a state space $(\mathcal{X}, \mathcal{E})$ where the σ-algebra $\mathcal{E}$ is countably generated. The transition kernel $\boldsymbol{P}$ is reversible with respect to a stationary distribution π, and $\boldsymbol{P}$ has non-negative spectrum.

Remark 3. There is little loss of generality by assuming reversibility and non-negative spectrum for MCMC algorithms. First, both Metropolis–Hastings and random-scan Gibbs algorithms (in the classical sense) are always reversible, though some non-reversible versions have been proposed in recent years (Bierkens, 2016; Bierkens et al., 2019; Bouchard-Côté et al., 2018; Fearnhead et al., 2018; Gagnon & Doucet, 2020). Second, for any transition kernel $\boldsymbol{P}$, its lazy version $\boldsymbol{P}_{\text{lazy}} = (\boldsymbol{P} + \boldsymbol{I})/2$ always has non-negative spectrum. As noted in Baxendale (2005), these two assumptions can yield better bounds on the convergence rates.

For any non-negative measurable function f, let $(\boldsymbol{P}^t f)(x) = \mathbb{E}_x[f(\mathsf{X}_t)]$, where $\mathbb{E}_x$ denotes the expectation with respect to the probability measure for $(\mathsf{X}_t)_{t\in\mathbb{N}}$ with $\mathsf{X}_0 = x$. For a non-empty measurable set $\mathsf{C} \subset \mathcal{X}$, we say (X_t) satisfies a drift condition on $\mathcal{X} \setminus \mathsf{C}$ if

$$(\boldsymbol{P}V)(x) \le \lambda V(x), \quad \forall\, x \notin \mathsf{C}, \tag{19}$$

for some function $V : \mathcal{X} \to [1, \infty)$ and constant $\lambda \in (0, 1)$. If $\mathsf{C} = \{x^*\}$ is a singleton set, (19) will be referred to as a 'single element' drift condition. This happens when the state x^* has a large stationary probability mass and the chain has a tendency to move towards x^*.

4.1 | Convergence rates with the two-stage drift condition

Motivated by the variable selection problem, we consider a setting where $(\mathsf{X}_t)_{t\in\mathbb{N}}$ satisfies two 'nested' drift conditions. Let A be a measurable subset of $\mathcal{X}$ and x^* be a point in A. First, we use a drift condition on A^c to describe the tendency of the chain to move towards A, if it is currently outside A. Second, we assume once the chain enters A, it drifts towards x^*, which can be described by a single element drift condition on $\mathsf{A} \setminus \{x^*\}$. We refer to such a construction as a two-stage drift condition, for which the main result is provided in the following theorem.

Theorem 2. *Let $(\mathsf{X}_t)_{t\in\mathbb{N}}, \mathcal{X}, \boldsymbol{P}, \pi$ be as given in Assumption A. Suppose that there exist two drift functions $V_1, V_2 : \mathcal{X} \to [1,\infty)$, constants $\lambda_1, \lambda_2 \in (0,1)$, a set $\mathsf{A} \in \mathcal{E}$ and a point $x^* \in \mathsf{A}$ such that*

(i) $\boldsymbol{P}V_1 \le \lambda_1 V_1$ on A^c, and
(ii) $\boldsymbol{P}V_2 \le \lambda_2 V_2$ on $\mathsf{A} \setminus \{x^*\}$.

Further, suppose that A *satisfies the following conditions for some finite constants $M \ge 2$ and $K \ge 1$.*

(iii) *For any $x \in \mathsf{A}$, $V_1(x) = 1$, and if $\boldsymbol{P}(x, \mathsf{A}^c) > 0$, $\mathbb{E}_x[V_1(\mathsf{X}_1)|\mathsf{X}_1 \in \mathsf{A}^c] \le M/2$.*
(iv) *For any $x \in \mathsf{A}$, $V_2(x) \le K$, and if $\boldsymbol{P}(x, \mathsf{A}^c) > 0$, $\mathbb{E}_x[V_2(\mathsf{X}_1)|\mathsf{X}_1 \in \mathsf{A}^c] \ge V_2(x)$.*
(v) *For any $x \in \mathsf{A}$, $\boldsymbol{P}(x, \mathsf{A}^c) \le q$ for some constant $q < \min\{1-\lambda_1,\ (1-\lambda_2)/K\}$.*

Then, for every $x \in \mathcal{X}$ and $t \in \mathbb{N}$, we have

$$\|\boldsymbol{P}^t(x,\cdot) - \pi\|_{\mathrm{TV}} \le 4\alpha^{t+1}\left(1 + \frac{V_1(x)}{M}\right),$$

where α is a constant in $(1-q/4, 1)$ and can be computed by

$$\alpha = \frac{1+\rho^r}{2} = \frac{1+M^r/u}{2}, \quad \rho = \frac{qK}{1-\lambda_2}, \quad u = \frac{1}{1-q/2}, \quad r = \frac{\log u}{\log(M/\rho)}.$$

Proof. See Section S2 in Appendix A.

Remark 4. To interpret the two drift conditions, (i) and (ii), it may help to think of $\log V_1$ as the 'distance' to the set A, and $\log V_2$ as the 'distance' to the point x^*. Both conditions (iii) and (iv) then become natural. Indeed, there is no loss of generality by assuming that $V_1 = 1$ on A. Given any other drift function V_1' which satisfies (i) on A^c, we can always define V_1 by letting $V_1 = 1$ on A and $V_1 = V_1'$ on A^c, which still satisfies all the assumptions made in the theorem. The constant M can be simply chosen to be $2\sup_{x\in\mathcal{X}} V_1(x)$, if it is finite.

Remark 5. Consider the distribution of the hitting time $\tau^* = \min\{t \ge 0 : \mathsf{X}_t = x^*\}$, which, by Theorem S1 in Appendix A, can be used to bound the mixing time of the chain. The sample path from an arbitrary point $x \in \mathsf{A}^c$ to x^* can be broken into disjoint segments in A^c and A. Though the length of each segment has a finite expectation due to the two drift conditions, the number of these segments largely depends on the parameter q, and $\mathbb{E}[\tau^*]$ may be infinite if the chain can easily escape from the set A. This is why we need condition (v). Consider some $x \in \mathsf{A}$ and $y \in \mathsf{A}^c$ such that $\boldsymbol{P}(x,y) > 0$. For Markov chains that move locally, $\boldsymbol{P}(x,y) > 0$ implies that x and y are very 'close'. By the reversibility of the chain, we have $\boldsymbol{P}(x,y) \le \pi(y)/\pi(x)$. Hence, to check condition (v), it suffices to bound the ratio $\pi(y)/\pi(x)$, which is often straightforward for neighbouring states x, y. When applying Theorem 2, we should be careful with the choice of q. Even if $\boldsymbol{P}(x, \mathsf{A}^c) = 0$ for all $x \in \mathsf{A}$, we should try other positive values of q so that $1-\alpha$ can be maximised. As a rule of thumb, we can choose some q that has the same order as $\min\{1-\lambda_1,\ (1-\lambda_2)/K\}$; see Corollary 1.

4.2 | Convergence rates in the high-dimensional setting

For MCMC algorithms, $\mathcal{X}$ is the parameter space and its dimension is conventionally denoted by p. For high-dimensional problems, $p = p(n)$ grows to infinity and typically, we have the drifting parameters $\lambda_1, \lambda_2 \uparrow 1$ and the convergence rate $(1-\alpha) \downarrow 0$, where $\lambda_1, \lambda_2, \alpha$ are as given in Theorem 2. To show the chain is rapidly mixing, we need to find a finite constant $c > 0$ such that $p^{-c} = O(1-\alpha)$. The following result extends Theorem 2 to the high-dimensional setting.

Corollary 1. *Consider a sequence of Markov chains where each* $(\mathsf{X}_t)_{t\in\mathbb{N}}$ *(implicitly indexed by* n*) satisfies the assumptions in Theorem 2. Assume that* $\lambda_1, \lambda_2 \to 1$ *and* $q \le \min\{1-\lambda_1, (1-\lambda_2)/CK\}$ *for some universal constant* $C > 1$*. Then, we have*

$$\|\boldsymbol{P}^t(x,\cdot) - \pi\|_{\mathrm{TV}} \le 4\alpha^{t+1}\left(1 + \frac{V_1(x)}{M}\right),$$

for some α *such that (*$\sim$ *denotes asymptotic equivalence)*

$$1-\alpha \sim \frac{(1-\lambda^*)\log C}{4\log(MC)}, \quad \text{where } 1-\lambda^* = \min\left\{1-\lambda_1, \frac{1-\lambda_2}{CK}\right\}.$$

Proof. Observe that without loss of generality we can assume $1-\lambda_1 = (1-\lambda_2)/CK = 1-\lambda^*$ and $q = 1-\lambda^* = o(1)$. Then, the constants defined in Theorem 2 satisfy $\rho = C^{-1}$ and $r \sim q/(2\log(MC))$, from which the result follows.

Corollary 2. *For* $\epsilon \in (0, 1/2)$*, define the* ϵ*-mixing time of the chain* $(\mathsf{X}_t)_{t\in\mathbb{N}}$ *by*

$$T_{\mathrm{mix}}(\epsilon) = \sup_{x\in\mathcal{X}} \min\{t \ge 0 : \|\boldsymbol{P}^t(x,\cdot) - \pi(\cdot)\|_{\mathrm{TV}} \le \epsilon\}.$$

In the setting of Corollary 1 with $M = 2\sup_{x\in\mathcal{X}} V_1(x)$*, for sufficiently large* n*, we have*

$$T_{\mathrm{mix}}(\epsilon) \lesssim \frac{4\log(6/\epsilon)}{\log C}\log(CM)\max\left\{\frac{1}{1-\lambda_1}, \frac{CK}{1-\lambda_2}\right\}.$$

Proof. This follows from a straightforward calculation using $-\log(\alpha) \sim 1-\alpha$.

Remark 6. In Corollaries 1 and 2, we do not make assumptions on the growth rates of M and K. In particular, if $M = p^c$ for some constant $c \ge 0$, it will only introduce an additional $\log p$ factor to the the mixing time.

4.3 | Comparison with drift-and-minorization methods

The two-stage drift condition can be seen as a generalisation of the single element drift condition since eventually the chain will arrive at the central state x^*. But, from a different angle, it also resembles the classical drift-and-minorization methods, which assume that there exist a drift function $V : \mathcal{X} \to [1,\infty)$, a 'small' set $\mathsf{S} \in \mathcal{E}$, a probability measure ψ on $(\mathcal{X}, \mathcal{E})$, constants $\lambda \in (0,1)$, $\xi > 0$ and $b < \infty$ such that

$$\text{(drift condition)} \quad \boldsymbol{P}V \le \lambda V \mathbb{1}_{\mathsf{S}^c} + b\mathbb{1}_{\mathsf{S}},$$
$$\text{(minorization condition)} \quad \boldsymbol{P}(x,\cdot) \ge \xi\psi(\cdot) \text{ for } x \in \mathsf{S}.$$

Both coupling arguments and regeneration theory can be used to compute a bound on $\|\boldsymbol{P}^t(x,\cdot) - \pi\|_{\text{TV}}$; see, for example, Rosenthal (1995); Roberts and Tweedie (1999). By the minorization condition, each time the chain is in S, we can let the whole process regenerate according to ψ with probability ξ. By the drift condition, the return times into S have geometrically decreasing tails. Jointly, the two conditions imply that the first time that the chain regenerates has a 'thin-tailed' distribution. The proof for our result with the two-stage drift condition uses a similar idea. The set A in Theorem 2 can be seen as the small set, and each time the chain enters A, there is some positive probability that the chain will hit x^* and thus regenerates before leaving A. Essentially, we have replaced the minorization condition on the small set with another drift condition, which is still used to bound the regeneration probability when the chain visits the small set.

The above comparison between the two-stage drift condition and drift-and-minorization method suggests that one may want to consider the following more general setting. Suppose there exist a sequence of sets $\mathcal{X} = \mathsf{A}_0 \supseteq \mathsf{A}_1 \supseteq \cdots \supseteq \mathsf{A}_k$ such that for each $i = 0, 1, \ldots, k-1$, a drift condition holds on $\mathsf{A}_i \setminus \mathsf{A}_{i+1}$ showing that the chain tends to drift from $\mathsf{A}_i \setminus \mathsf{A}_{i+1}$ into A_{i+1}. When A_k is a singleton set, one can mimic the proof of Theorem 2 to combine the k drift conditions and derive a quantitative bound on the mixing time. When A_k is a set on which we can establish a minorization condition or we have a mixing time bound for the Markov chain restricted to A_k, the main idea of our proof still applies, though some details may need nontrivial modification. For a concrete example, consider the posterior distribution of β in our variable selection problem described in Section 2.1. To sample from $\pi_n(\beta)$, we just need to modify any MH algorithm targeting $\pi_n(\gamma)$ by sampling from the conditional posterior distribution $\pi_n(\beta|\gamma)$ at the end of each iteration. Let $\|\beta\|_0$ denote the number of non-zero entries of β. Then, assuming Condition 1 holds, we can define $\mathcal{A}_0 = \{\beta \in \mathbb{R}^p : \|\beta\|_0 \le s_0\}$, $\mathcal{A}_1 = \{\beta \in \mathcal{A}_0 : \forall j \in \gamma^*,\ \beta_j \ne 0\}$ (the set of all possible values of β for an overfitted model), and $\mathcal{A}_2 = \{\beta \in \mathcal{A}_1 : \forall j \notin \gamma^*,\ \beta_j = 0\}$ (the set of all possible values of β for the model γ^*). The two drift conditions proved in Section 3.1 show that the MH algorithm tends to drift from $\mathcal{A}_0 \setminus \mathcal{A}_1$ into $\mathcal{A}_1$ and from $\mathcal{A}_1 \setminus \mathcal{A}_2$ into $\mathcal{A}_2$. By combining them with standard results in the literature for the mixing time of an MH algorithm targeting $\pi_n(\beta|\gamma = \gamma^*)$ (which is just a multivariate normal distribution), one can derive the mixing time bound for the MH chain.

4.4 | Applications of the two-stage drift condition

Our use of the two-stage drift condition in Section 3 is largely motivated by Condition 1, which characterises the different behaviours of underfitted and overfitted models. Though this makes the two-stage drift condition look very specific to the variable selection problem, there are actually many discrete-state-space problems other than variable selection where the mixing time of MH algorithms can be conveniently analysed by using multiple drift conditions.

First, many model selection problems can be written as a set of sparse linear regression models. For example, in structure learning problems, the goal is to infer the underlying Bayesian network (i.e. directed acyclic graph) of a p-variate distribution. These problems are often formulated as structural equation models where the causal relationships among p coordinate variables are described by p sparse linear regression models. When the ordering is known, structure learning of Bayesian networks becomes very similar to variable selection, and one can extend and

prove Condition 1 in a way completely analogous to that in Yang et al. (2016). The application of the two-stage drift condition and the construction of LIT-MH are also straightforward. When the ordering is unknown, the problem becomes much more complicated due to the existence of Markov equivalent Bayesian networks, but it is still possible to generalise Condition 1 (Zhou & Chang, 2021), and the two-stage drift condition can be constructed accordingly.

Moreover, in recent years, general high-dimensional consistency results have been obtained for a large variety of Bayesian model selection problems. Gao et al. (2020) proved optimal posterior contraction rates for a general class of structured linear models, including as special cases stochastic block model, multi-task learning, dictionary learning and wavelet estimation. All these examples share common features with the variable selection problem; for example, one can naturally define a model to be underfitted or overfitted according as it has the best model nested within it. It seems very promising that the methodology developed in Gao et al. (2020) can be used to prove results similar to Condition 1 in general settings, from which we may further establish a two-stage drift condition, one for underfitted models and the other for overfitted ones.

In addition to 'underfitted/overfitted' schemes, for some problems, we may partition the state space using a different strategy. For example, consider a change-point detection problem where we need to infer both the number and locations of change points. Suppose that we use an informative prior which favours models with equal-sized segments. Then, if we incorrectly infer the number of change points, the locations of change points cannot be accurately estimated either due to the prior. In such cases, a possible approach to mixing time analysis is to construct one drift condition showing that the chain first drifts towards models with true number of change points and another drift condition showing that once the number of change points is correctly inferred, the chain is able to tune the locations of change points towards their true values. We note that similar ideas may also be applied to more complicated spatial clustering models, such as those based on spanning trees (Lee et al., 2021; Luo et al., 2021). Compared with using a single drift condition on the whole space, the two-stage approach often leads to an easier and more constructive proof.

5 | SIMULATION STUDIES

For our simulation studies, we implement the RW-MH and LIT-MH algorithms as follows. Assume the proposal scheme has the form given in (9). In each iteration we propose an addition, deletion or swap move with fixed probabilities 0.4, 0.4 and 0.2, respectively; that is, we set $h_a(\gamma) = h_d(\gamma) = 0.4$, $h_s(\gamma) = 0.2$ in (9). This is slightly different from the setting in Section 3, but our mixing time bound still applies up to some constant factor. We consider four choices of the weighting functions w_a and w_d.

$$
\begin{aligned}
&\text{RW-MH:} && w_a(\gamma'|\gamma) = 1, && w_d(\gamma'|\gamma) = 1.\\
&\text{LIT-MH-1:} && w_a(\gamma'|\gamma) = p^{-1} \vee B(\gamma,\gamma') \wedge p, && w_d(\gamma'|\gamma) = p^{-1} \vee B(\gamma,\gamma') \wedge 1,\\
&\text{LIT-MH-2:} && w_a(\gamma'|\gamma) = p^{-2} \vee B(\gamma,\gamma') \wedge p^2, && w_d(\gamma'|\gamma) = p^{-2} \vee B(\gamma,\gamma') \wedge p,\\
&\text{LB-MH-1:} && w_a(\gamma'|\gamma) = \sqrt{B(\gamma,\gamma')}, && w_d(\gamma'|\gamma) = \sqrt{B(\gamma,\gamma')}.
\end{aligned}
$$

LIT-MH-2 is more aggressive than LIT-MH-1 in the sense that the proposal distribution is more concentrated on the neighboring states with very large posterior probabilities. The last one is inspired by the locally balanced proposals of Zanella (2020). Since the proposal weights are unbounded in LB-MH-1, we expect its acceptance probability to be smaller than that of LIT-MH

algorithms. For comparison, we also consider the original locally balanced MH algorithm of Zanella (2020) with proposal

$$\boldsymbol{K}_{\mathrm{lb}}(\gamma,\gamma')=\frac{\sqrt{B(\gamma,\gamma')}}{Z(\gamma)}\mathbb{1}_{\mathcal{N}_{\mathrm{a}}(\gamma)\cup\mathcal{N}_{\mathrm{d}}(\gamma)}(\gamma'),\ \text{ where }\ Z(\gamma)=\sum_{\tilde{\gamma}\in\mathcal{N}_{\mathrm{a}}(\gamma)\cup\mathcal{N}_{\mathrm{d}}(\gamma)}\sqrt{B(\gamma,\tilde{\gamma})}.$$

Denote this algorithm by LB-MH-2. It differs from LB-MH-1 in that we do not distinguish types of proposal moves when calculating proposal weights. We will discuss LB-MH-1 and LB-MH-2 in Sections 7.2 and 7.3.

The use of the parameter s_0 is unnecessary in our simulation studies since all sampled models have size much smaller than n. For computational convenience, we do not consider swap moves for LB-MH-2 and, for the other algorithms, we implement swap moves by compounding one addition and one deletion move, which makes the swap proposal only 'partially informed'; details are given in Section 7.1. When we describe the modality of π_n in simulation results, we are always referring to the 'single-flip' neighbourhood relation $\mathcal{N}_{\mathrm{a}}(\cdot)\cup\mathcal{N}_{\mathrm{d}}(\cdot)$. The code is written in C++ in order to maximise the computational efficiency; see Section S6 in Appendix A.

5.1 | Finding models with high posterior probabilities

For the first simulation study, we consider the settings used in Yang et al. (2016) with random design matrices. Let all rows of X be i.i.d., and the ith row vector, $x_{(i)}$, be generated in the following two ways.

$$\text{Independent design: } x_{(i)}\overset{i.i.d.}{\sim}\mathrm{MN}(0,I_p),$$
$$\text{Correlated design: } x_{(i)}\overset{i.i.d.}{\sim}\mathrm{MN}(0,\Sigma),\ \Sigma_{jk}=e^{-|j-k|}.$$

The response vector y is simulated by $y=X\beta^*+z$ with $z\sim\mathrm{MN}(0,I_n)$. The first 10 entries of β^* are given by

$$\beta^*_{[10]}=\mathrm{SNR}\sqrt{\frac{\log p}{n}}(2,-3,2,2,-3,3,-2,3,-2,3),$$

where $\mathrm{SNR}>0$ denotes the signal-to-noise ratio. All the other entries of β^* are set to zero. The un-normalized posterior probability of a model γ is calculated using (4) with $\kappa_0=2$ and $\kappa_1=3/2$. Simulation experiments are conducted for $\mathrm{SNR}=1,2,3$ and $(n,p)=(500,1000)$ or $(1000,5000)$. For each setting, we simulate 100 datasets, and for each dataset, we run RW-MH for 10^5 iterations and each informed algorithm for 2000 iterations. All algorithms are initialized with a randomly generated model $\gamma^{(0)}$ with $|\gamma^{(0)}|=10$. Let $\gamma_{\mathrm{true}}=\{1,2,\ldots,10\}$ denote the true set of covariates with non-zero regression coefficients, and let $\hat{\gamma}_{\max}$ be the model with the largest posterior probability that has been sampled by any of the five algorithms. If an algorithm has never sampled $\hat{\gamma}_{\max}$, the run is counted as a failure.

Results are summarized in Table 1. We first note that when $\mathrm{SNR}=3$, LB-MH-2 performs much worse than RW-MH and almost always gets stuck at some sub-optimal local mode. This is consistent with the observation made in Example 1, which will be further discussed in Section 7.2. Due to its poor performance, we exclude LB-MH-2 from all the remaining numerical experiments.

TABLE 1 Simulation study I. For each setting, we simulate 100 datasets

Number of iterations			RW-MH 100,000	LIT-MH-1 2,000	LIT-MH-2 2,000	LB-MH-1 2,000	LB-MH-2 2,000
$n = 500$, $p = 1000$, independent design	SNR = 3 (100)	Time	7.8	1.3	1.3	1.2	1.2
		Success	100	100	100	100	12
		$H_{\max}$	8004(13773)	20(30)	210(578)	39(53)	2000(2000+)
		$t_{\max}$	0.62	0.012	0.13	0.023	1.1
	SNR = 2 (100)	Time	7.8	1.3	1.3	1.2	1.6
		Success	100	100	100	100	95
		$H_{\max}$	9316(17884)	20(30)	60(385)	38(53)	94(1972)
		$t_{\max}$	0.73	0.013	0.038	0.023	0.077
	SNR = 1 (0)	Time	5.3	0.96	0.95	0.9	1.2
		Success	100	100	100	100	100
		$H_{\max}$	33(9326)	22(35)	21(39)	21(36)	9(15)
		$t_{\max}$	0.0014	0.010	0.0095	0.0088	0.0060
$n = 500$, $p = 1000$, correlated design	SNR = 3 (98)	Time	7.7	1.3	1.3	1.2	1.2
		Success	89	99	98	98	4
		$H_{\max}$	17230(10^5+)	29(726)	50(390)	45(689)	2000(2000+)
		$t_{\max}$	1.3	0.018	0.032	0.028	1.1
	SNR = 2 (42)	Time	7.4	1.2	1.2	1.1	1.2
		Success	57	79	81	85	57
		$H_{\max}$	62308(10^5+)	94(2000+)	72(2000+)	124(2000+)	1200(2000+)
		$t_{\max}$	4.5	0.061	0.046	0.083	0.84

(Continues)

TABLE 1 (Continued)

Number of iterations			RW-MH 100,000	LIT-MH-1 2,000	LIT-MH-2 2,000	LB-MH-1 2,000	LB-MH-2 2,000
	SNR = 1 (0)	Time	3.7	0.8	0.79	0.76	2
		Success	100	100	100	100	100
		H_{max}	26(4722)	22(32)	21(30)	22(34)	9(10)
		t_{max}	0.00074	0.0085	0.0084	0.0081	0.010
$n = 1000, p = 5000$, independent design	SNR = 3 (100)	Time	39	9.2	9.3	8.9	13
		Success	99	100	100	100	4
		H_{max}	35291(65746)	20(32)	36(53)	43(53)	2000(2000+)
		t_{max}	14	0.091	0.17	0.19	12
	SNR = 2 (100)	Time	39	9.2	9.4	8.9	13
		Success	98	100	91	100	93
		H_{max}	39746(70458)	20(31)	469(2000+)	33(48)	160(2000+)
		t_{max}	15	0.089	2.2	0.15	1.1
	SNR = 1 (0)	Time	34	8.5	8.6	8.3	8.3
		Success	97	100	100	100	100
		H_{max}	20140(76960)	21(44)	20(44)	20(43)	12(22)
		t_{max}	7.5	0.089	0.086	0.081	0.056

(Continues)

TABLE 1 (Continued)

Number of iterations			RW-MH 100,000	LIT-MH-1 2,000	LIT-MH-2 2,000	LB-MH-1 2,000	LB-MH-2 2,000
$n = 1000$, $p = 5000$, correlated design	SNR = 3 (100)	Time	41	10	10	9.9	8.9
		Success	94	100	99	100	0
		$H_{\max}$	51178(10^5+)	20(32)	65(1181)	41(54)	2000(2000+)
		$t_{\max}$	21	0.10	0.33	0.20	8.7
	SNR = 2 (84)	Time	41	10	10	9.8	10
		Success	42	89	94	97	20
		$H_{\max}$	10^5+(10^5+)	37(2000+)	64(2000+)	50(1194)	2000(2000+)
		$t_{\max}$	38	0.19	0.33	0.24	8.5
	SNR = 1 (0)	Time	20	7.2	7.2	7	16
		Success	100	100	100	100	100
		$H_{\max}$	25(13474)	23(36)	22(33)	22(32)	9(10)
		$t_{\max}$	0.0041	0.082	0.078	0.076	0.069

Notes: 'Time' is the average wall time usage measured in seconds. 'Success' is the number of successful runs; a run is successful if $\hat{\gamma}_{\max}$ has been sampled (see the main text). $H_{\max}$ is the median number of iterations needed to sample $\hat{\gamma}_{\max}$; the number in the parenthesis is the 95% quantile. $t_{\max}$ is the median wall time (in seconds) needed to sample $\hat{\gamma}_{\max}$. The number in parentheses next to the signal-to-noise ratio value is the number of replicates out of 100 where $\hat{\gamma}_{\max} = \gamma_{\text{true}}$.

For SNR = 2 and 3, informed algorithms always find the model $\hat{\gamma}_{\max}$ much faster than RW-MH. Remarkably, the median wall time needed for LIT-MH-1 to sample $\hat{\gamma}_{\max}$ (denoted by $t_{\max}$ in the table) is less than 0.2 second in all scenarios. When SNR = 1, the best model is often the null model, in which case RW-MH can also find $\hat{\gamma}_{\max}$ easily (since we fix $h_{\mathrm{d}}(\gamma) = 0.4$, it only takes RW-MH about 25 iterations to propose removing the 10 covariates in $\gamma^{(0)}$). When the SNR is either very strong (SNR = 3) or very weak (SNR = 1), all algorithms except LB-MH-2 can identify $\hat{\gamma}_{\max}$ in most runs. Similar findings were made in Yang et al. (2016), and this is because when SNR is very large, Condition 1 is likely to be satisfied with $\gamma^* = \gamma_{\mathrm{true}}$, and when SNR is very small, Condition 1 is likely to be satisfied with $\gamma^* = \emptyset$. For correlated designs with SNR = 2, the posterior landscape tends to be multi-modal, and all algorithms may get stuck at local modes. But the informed algorithms still have much better performance than RW-MH: each informed algorithm, except LB-MH-2, is able to sample $\hat{\gamma}_{\max}$ in $\geq 80\%$ of the runs, while RW-MH has a much larger failure rate and finds $\hat{\gamma}_{\max}$ much more slowly. We also observe that in most settings, LIT-MH-1 is (significantly) more efficient than LIT-MH-2 and LB-MH-1. This suggests that it is helpful to truncate the weighting function w_{a} and w_{d} to a relatively small range, which is consistent with our theory. Since $|\gamma^{(0)}| = |\gamma_{\mathrm{true}}| = 10$ (and the two sets are disjoint in most cases), it takes at least 20 addition and deletion proposals to move from $\gamma^{(0)}$ to γ_{true}. According to Table 1, in high SNR settings, $\hat{\gamma}_{\max}$ usually coincides with γ_{true}, and the median number of iterations needed by LIT-MH-1 to reach $\hat{\gamma}_{\max}$ is about 20 (except in the correlated design case with $n = 500, p = 1000$), suggesting that the performance of LIT-MH-1 is close to being 'optimal' in the sense that any other local MH sampler cannot find $\hat{\gamma}_{\max}$ in a smaller number of iterations.

Since to implement informed MCMC algorithms, we need to evaluate π_n for every possible addition or deletion move in each iteration, we can use the MCMC sample paths to empirically study to what extent Condition 1 is satisfied. As predicted by the theory developed in Yang et al. (2016), we find that Condition 1 is more likely to be violated when SNR is small or the design matrix contains highly correlated covariates. See Sections S4.1 and S4.2 in Appendix A for details.

5.2 | Rao-Blackwellization for LIT-MH

During MCMC, given the current model γ, we can estimate β using the conditional posterior mean $\hat{\beta}(\gamma) = \mathbb{E}[\beta|\gamma, y]$. For LIT-MH algorithms, we can obtain a Rao-Blackwellized estimator, $\hat{\beta}_{\mathrm{RB}}(\gamma)$, as follows. For each $j \in [p]$, let $\gamma_j = \mathbb{1}_\gamma(j)$ indicate whether covariate j is selected in γ, and let $\gamma_{-j} = (\gamma_1, \ldots, \gamma_{j-1}, \gamma_{j+1}, \ldots, \gamma_p)$ denote the status of all the other $p-1$ covariates. By the law of total expectation, $\mathbb{E}[\beta_j|y] = \mathbb{E}[\,\mathbb{E}[\beta_j|\gamma_{-j}, y]\,]$. For informed proposal schemes, we can get $\mathbb{E}[\beta_j|\gamma_{-j}, y]$ for every $j \in [p]$ with little additional computational cost, since

$$\mathbb{E}[\beta_j|\gamma_{-j}, y] = \frac{\pi_n(\gamma \cup \{j\})\,\mathbb{E}[\beta_j|\gamma_{-j}, y, \gamma_j = 1]}{\pi_n(\gamma \cup \{j\}) + \pi_n(\gamma \setminus \{j\})},$$

and all the three terms on the right-hand side (one of them is just $\pi_n(\gamma)$) have already been obtained when we calculate the normalizing constants, $Z_{\mathrm{a}}(\gamma)$ and $Z_{\mathrm{d}}(\gamma)$. The estimator $\hat{\beta}_{\mathrm{RB}}(\gamma)$ is then obtained by estimating each entry using $\mathbb{E}[\beta_j|\gamma_{-j}, y]$.

Let $\mathrm{MSE}(\hat{\beta}) = p^{-1}\|\hat{\beta} - \beta^*\|_2^2$ denote the error of an estimator $\hat{\beta}$. For the simulation study described in Section 5.1, we observe that $\mathrm{MSE}(\hat{\beta}_{\mathrm{RB}}(\gamma))$ always decreases to a nearly optimum level within a few iterations. See Figure 1 for the trajectories of $\mathrm{MSE}(\hat{\beta}_{\mathrm{RB}}(\gamma))$ of LIT-MH-1 and $\mathrm{MSE}(\hat{\beta}(\gamma))$ of RW-MH, averaged over 100 datasets. (Trajectories of LIT-MH-2 and LB-MH-1 are

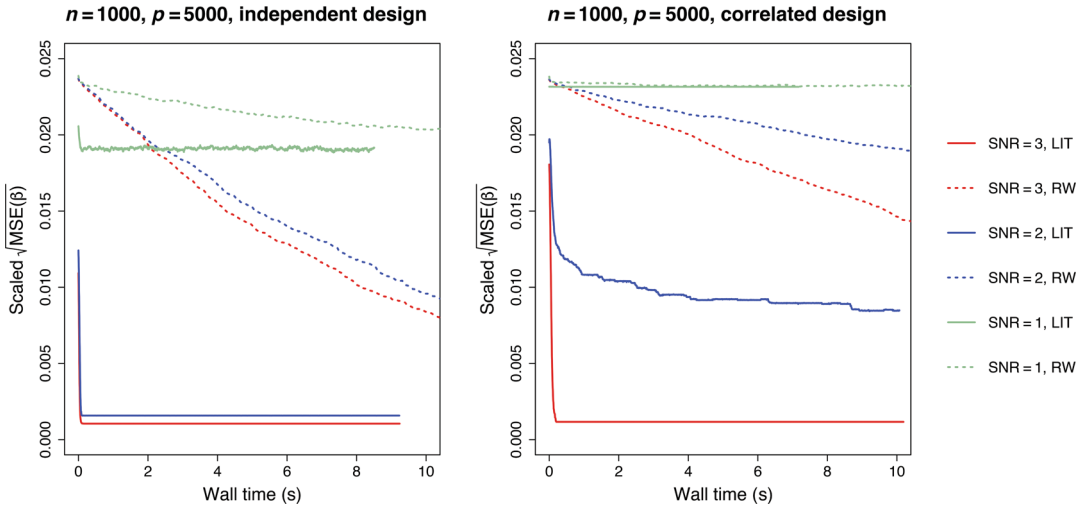

FIGURE 1 Trajectories of MSE($\hat{\beta}$). Solid lines represent the LIT-MH-1 algorithm, which uses $\hat{\beta} = \hat{\beta}_{\mathrm{RB}}(\gamma)$; dotted lines represent the RW-MH algorithm, which uses $\hat{\beta} = \hat{\beta}(\gamma)$. The y-axis is $\{\mathrm{MSE}(\hat{\beta})\}^{1/2}/\mathrm{SNR}$ averaged over 100 datasets.

omitted since they are very similar to that of LIT-MH-1.) The advantage of LIT-MH over RW-MH becomes even more substantial. More investigation is needed to justify the use of $\hat{\beta}_{\mathrm{RB}}(\gamma)$, but our analysis at least shows that the computation of the proposal weights can be made use of in multiple ways.

5.3 | Exploring multi-modal posterior distributions

Condition 1 represents the ideal case where the posterior distribution is unimodal, but in reality multi-modality is the norm. In our second simulation study, we consider a more realistic simulation scheme which gives rise to multi-modal posterior distributions. The design matrix X is still assumed to have i.i.d. rows, but each row is sampled from $\mathrm{MN}(0, \Sigma_{d,p})$ where $\Sigma_{d,p} = \mathrm{diag}(\Sigma_d, \ldots, \Sigma_d)$ is block-diagonal. Each block Σ_d has dimension $d \times d$, and $(\Sigma_d)_{jk} = e^{-|j-k|/3}$. We fix $n = 1000$, $p = 5000$ and $d = 20$. The response y is still simulated by $y = X\beta^* + z$ with $z \sim \mathrm{MN}(0, I_n)$. But we generate β^* by first sampling γ^* from the uniform distribution on the set $\{\gamma \subset [p] : |\gamma| = 100\}$ and then sampling $\beta^*_{\gamma^*} \sim \mathrm{MN}(0, \sigma_\beta^2 I_{100})$. We use $\sigma_\beta = 0.1, 0.2, 0.3, 0.4, 0.5$ to simulate posterior distributions with varying degrees of multi-modality. For the hyperparameters, we choose $\kappa_0 = 1$ and $\kappa_1 = 1/2$. We observe that the posterior multi-modality is most severe for $\sigma_\beta = 0.2$.

For each setting, we simulate 20 datasets, and for each dataset, we run RW-MH for 2×10^5 iterations and each informed algorithm for 2000 iterations. All four algorithms are initialized with the model obtained by forward-backward stepwise selection. We use effective sample size (ESS) to measure the sampling efficiency. To calculate ESS, we consider two one-dimensional 'summary statistics'. Let $(\gamma^{(k)}, \beta^{(k)})$ denote the sample collected in the k-th MCMC iteration, where $\beta^{(k)}$ is drawn from the conditional posterior distribution of β given $\gamma^{(k)}$. Let $T_1^{(k)} = \|\hat{\beta}(\gamma^{(k)}) - \beta^*\|_2^2$ where $\hat{\beta}(\gamma) = \mathbb{E}[\beta|\gamma, y]$, and let $T_2^{(k)} = \|X\beta^{(k)}\|_2^2$. Note that $T_1^{(k)}$ only depends on $\gamma^{(k)}$, and thus the ESS of T_1 roughly reflects the efficiency of sampling γ, while the ESS of T_2 reflects the efficiency

of sampling β. The ESS estimates presented in Table 2 are calculated by using the `R` package `coda` (Plummer et al., 2006), and we provide nonparametric ESS estimates in Section S4.4 in the supplement. Given that the interest is in estimating the posterior means of β and γ, following Vats et al. (2019), we may calculate a multivariate ESS by appealing to a multivariate central limit theorem using $\{\gamma^{(k)}\}$ or $\{\beta^{(k)}\}$. However, the resulting Monte Carlo covariance matrix is quite large, and the estimation of such high-dimensional matrices is an ongoing problem in the literature; see Jin and Tan (2021). In Section S4.4 in Appendix A, we propose a method for constructing a low-dimensional summary of γ, and present corresponding multivariate ESS estimation results.

From Table 2, we see that given a fixed total number of iterations, the acceptance rate of RW-MH decreases quickly for larger σ_β, while it remains roughly unchanged around 0.5 for LIT-MH algorithms. In all scenarios, LIT-MH-1 and LIT-MH-2 have much larger effective sample sizes (per second) of both statistics T_1 and T_2 than RW-MH, indicating that LIT-MH algorithms can explore multi-modal distributions and collect posterior samples much more efficiently than RW-MH. Comparing ESS(T_1) of LIT-MH and that of RW-MH, we see that the advantage of using LIT-MH for sampling γ is huge for small values of σ_β. For LB-MH-1, we note that it always has smaller acceptance rate and effective sample sizes than the two LIT-MH algorithms. This is probably due to the use of an unbounded weighting function, which will be further discussed in Section 7.3.

6 | ANALYSIS OF REAL GWAS DATA

We have obtained access to two GWAS (genome-wide association study) data sets on glaucoma from dbGaP (the database of Genotypes and Phenotypes) with accession no. phs000308.v1.p1 and phs000238.v1.p1. Both datasets only contain individuals of Caucasian descent, and they were generated using the same genotyping array. We remove individuals whose self-reported race is Hispanic Caucasian and those with abnormal intraocular pressure or CDR measurements. We choose the response variable y to be the standardised CDR measurement averaged over two eyes. After merging the two datasets, we discard variants with minor allele frequency less than 0.05 or missing rate greater than 0.01 and variants that fail the Hardy–Weinberg equilibrium test (p-value less than 10^{-6} in control samples). Finally, we use PLINK to prune variants with pairwise correlation > 0.75 and end up with $n = 5{,}418$ and $p = 328{,}129$. Each entry of the matrix X takes value in $\{0, 1, 2\}$, representing the number of copies of the minor allele. For the hyperparameters, we choose κ_0 and κ_1 such that $g = 100$ and $\pi_0(\gamma) \propto (20/p)^{|\gamma|}$. These choices are motivated by practical considerations. First, the prior on γ reflects that a priori we believe there are about 20 variants associated with y. For a complex trait such as CDR, this is a very conservative estimate. Second, assuming a causal variant X_j has minor allele frequency 0.5, $g = 100$ implies that the prior effect size $(g/X_j^\top X_j)^{1/2} \approx 0.2$, which is recommended for Bayesian analysis of GWAS data (Stephens & Balding, 2009).

We first conduct five parallel runs (with different random seeds) of the RW-MH algorithm, each consisting of 1 million iterations. Then we build a set, denoted by $S_\delta \subseteq [p]$, which includes all variants with posterior probabilities (estimated using the RW-MH output) greater than δ. When implementing the addition proposals for LIT-MH, we use (11) with $S = S_\delta$. One can also use marginal regression to build the set S_δ (Fan & Lv, 2008), which would yield very similar results. We consider $\delta = 10^{-4}, 5 \times 10^{-4}$ and 10^{-3}. For each choice, we conduct five parallel runs of the LIT-MH-1 algorithm. Some summary statistics of the output are provided in

TABLE 2 Simulation study II

	Number of iterations	RW-MH 200,000	LIT-MH-1 2,000	LIT-MH-2 2,000	LB-MH-1 2,000
$\sigma_\beta = 0.1$ Mean model size = 6.1	Time	78.1	9.95	10.0	9.30
	Local modes	1.85	2.30	2.15	2.00
	Acc. Rate	0.012	0.495	0.566	0.228
	ESS(T_1)/Time	0.706	16.8	15.1	6.46
	ESS(T_2)/Time	4.83	34.5	29.3	11.6
$\sigma_\beta = 0.2$ Mean model size = 26.4	Time	79.1	16.0	15.9	14.6
	Local modes	2.60	6.20	6.25	3.90
	Acc. Rate	0.0060	0.602	0.580	0.320
	ESS(T_1)/Time	0.414	4.72	3.76	2.40
	ESS(T_2)/Time	4.67	19.9	18.5	12.7
$\sigma_\beta = 0.3$ Mean model size = 50.2	Time	80.4	27.9	27.6	24.7
	Local modes	2.40	5.05	4.45	3.65
	Acc. Rate	0.0037	0.578	0.571	0.296
	ESS(T_1)/Time	0.360	2.49	2.82	1.48
	ESS(T_2)/Time	3.57	19.8	18.1	9.79
$\sigma_\beta = 0.4$ Mean model size = 63.9	Time	81.2	37.0	36.8	32.6
	Local modes	2.00	3.85	5.20	3.65
	Acc. Rate	0.0027	0.541	0.546	0.261
	ESS(T_1)/Time	0.333	2.57	1.85	1.02
	ESS(T_2)/Time	3.02	17.5	14.3	7.92
$\sigma_\beta = 0.5$ Mean model size = 71.6	Time	81.8	42.5	42.7	36.8
	Local modes	1.80	2.75	2.60	2.65
	Acc. Rate	0.0021	0.485	0.536	0.217
	ESS(T_1)/Time	0.526	3.62	3.28	1.36
	ESS(T_2)/Time	2.45	15.0	15.1	5.78

Notes: 'Mean model size' is the mean size of γ sampled by LIT-MH-1 (which is almost the same as that for the other three samplers). 'Time' is the wall time usage measured in seconds. 'Local modes' is the number of unique local modes sampled in the MCMC; we say γ is a local mode if $\pi_n(\gamma) > \pi_n(\gamma')$ for any $\gamma' \in \mathcal{N}_a(\gamma) \cup \mathcal{N}_d(\gamma)$. 'Acc. rate' is the acceptance rate. ESS(T_1) and ESS(T_2) are the estimated effective sample sizes of T_1 and T_2 (see the main text for details). All statistics are averaged over 20 datasets.

Table 3. For all four algorithms, the mean size of sampled models is 13. According to ESS(T_2) per minute, LIT-MH-1 algorithms are much more efficient than RW-MH in terms of sampling β, which may be surprising since RW-MH also has acceptance rate 0.27 and thence a much larger total number of accepted moves than LIT-MH algorithms. This indicates that LIT-MH can achieve greater sampling efficiency by significantly reducing the autocorrelation in MCMC samples.

TABLE 3 Cup-to-disc ratio analysis using GWAS data

Algorithm	$\lvert S_\delta \rvert$	Iterations	Time	Acc. rate	ESS(T_2)/Time
RW-MH	NA	1,000,000	428	0.273	1.95
LIT-MH-1 ($\delta = 0.0001$)	7255	8,000	48.7	0.714	8.52
LIT-MH-1 ($\delta = 0.0005$)	1410	40,000	56.3	0.635	26.8
LIT-MH-1 ($\delta = 0.001$)	715	80,000	82.6	0.603	33.5

Notes: $\lvert S_\delta \rvert$ is the number of variants for which we need to evaluate the posterior probabilities when proposing addition moves (see the main text for details). 'Iterations' is the number of MCMC iterations for each run; for each algorithm, we conduct five independent runs. 'Time' is the average wall time usage measured in minutes. 'Acc. rate' is the average acceptance rate. ESS(T_2) is the effective sample size calculated using the statistic T_2 as in Table 2.

TABLE 4 Top 10 signals in the cup-to-disc ratio analysis

Variant name	Location	PIP	Known hit	References
rs1063192	9p23.1	0.989	Yes	Osman et al. (2012)
rs653178	12q24.12	0.972	No	
rs10483727	14q23.1	0.888	Yes	Bailey et al. (2016)
rs319773	17q11.2	0.532	No	
rs2275241	9q33.3	0.531	Yes	Craig et al. (2020)
rs4557053	20p12.3	0.222	No	
rs10491971	12p13.32	0.144	No	
rs4901977	14q23.1	0.112	Yes	Springelkamp et al. (2014)
rs587409	13q34	0.111	Yes	Khawaja et al. (2018)
rs314300	7q22.1	0.107	No	

Notes: 'Location' is the cytogenetic location of the variant in human genome. 'PIP' is the posterior inclusion probability estimate averaged over all LIT-MH-1 runs. 'Known hit' indicates whether the variant is known to be associated with ocular traits; if yes, a reference is provided in the last column.

Next, we examine the estimate of the posterior inclusion probability (PIP), $\mathbb{E}[\mathbb{1}_\gamma(j)|y]$, for each $j \in [p]$. In Table 4, we list the 10 variants with the largest PIPs averaged over all runs of LIT-MH-1. Among them five are known GWAS hits for ocular traits (or ocular disorders) located in four different regions. For these five hits (which we may assume to be true signals), the PIP estimate in each individual run of RW-MH exhibits a very high variability. For example, in Table 5, we see that only the fourth run of RW-MH yields a PIP estimate greater than 0.1 for *rs587409*. Further, if one uses 0.1 as the threshold, each RW-MH run can miss at least two of the five hits. This observation suggests that, for large datasets, we often need to run RW-MH for an extremely large number of iterations so that the results can be 'replicable'. In contrast, the individual PIP estimates from 15 LIT-MH-1 runs are much more stable. The only exception is the variant *rs4901977*. This is because *rs4901977* is located closely to *rs10483727*, and thus the two variants are correlated, which makes it challenging to identify both variants at the same time.

TABLE 5 Posterior inclusion probabilities of the five hits in Table 4

	PIPs of RW-MH runs					
Variant name	**Run 1**	**Run 2**	**Run 3**	**Run 4**	**Run 5**	**PIP range in 15 LIT-MH-1 runs**
rs1063192	0	0.729	0.309	0.753	0.330	[0.839, 1]
rs10483727	0.483	0.147	0	0.852	0	[0.308, 1]
rs2275241	0.383	0	0	0	0.426	[0.491, 0.569]
rs4901977	0.486	0.239	0.678	0	0.214	[0, 0.695]
rs587409	0	0.003	0.065	0.117	0.032	[0.083, 0.132]

Notes: 'PIP' is posterior inclusion probability estimate. The last column gives the minimum and maximum PIP estimates obtained from 15 LIT-MH-1 runs.

7 | DISCUSSION

7.1 | On the swap moves of LIT-MH

Both the parameter s_0 and swap moves are used in our mixing time analysis of LIT-MH for merely technical reasons. As shown in Yang et al. (2016), rapid mixing on the space $\mathcal{M}(p)$ is usually impossible since sharp local modes can easily occur among very large models, suggesting that the use of s_0 is necessary for theoretical analysis. Then, swap moves are introduced to ensure that the chain cannot get trapped at an underfitted model with size s_0. However, in practice, even if we let $s_0 = p$ and run the chain on $\mathcal{M}(p)$, the chain is very unlikely to visit those models with size much larger than s^* since they have negligible posterior probabilities (Narisetty & He, 2014). In other words, assuming that both s^* and $|\gamma^{(0)}|$ are small, Condition 1 actually implies that we will 'observe' the chain is 'rapidly mixing' by using only addition and deletion moves.

The above reasoning suggests that an approximate implementation of informed swap moves will not significantly affect the overall performance of LIT-MH. One way to realize a 'partially informed' swap move is to treat it as a composition of one addition and one deletion. Given current state γ, we first use an informed addition move to propose some $\tilde{\gamma} \in \mathcal{N}_{\mathrm{a}}(\gamma)$, and then use an informed deletion move to propose $\gamma' \in \mathcal{N}_{\mathrm{d}}(\tilde{\gamma})$. The acceptance probability of γ' is calculated by

$$1 \wedge \frac{\pi_n(\gamma')\boldsymbol{K}_{\mathrm{lit}}(\gamma', \tilde{\gamma})\boldsymbol{K}_{\mathrm{lit}}(\tilde{\gamma}, \gamma)}{\pi_n(\gamma)\boldsymbol{K}_{\mathrm{lit}}(\gamma, \tilde{\gamma})\boldsymbol{K}_{\mathrm{lit}}(\tilde{\gamma}, \gamma')}.$$

One can check that the resulting transition matrix is reversible with respect to π_n. In our implementation of LIT-MH, we further impose the constraint that $\gamma' \neq \gamma$ when sampling γ' from $\mathcal{N}_{\mathrm{d}}(\tilde{\gamma})$ and adjust the Hastings ratio accordingly. Note that to implement an addition proposal, we need to calculate both $\boldsymbol{K}_{\mathrm{lit}}(\gamma, \tilde{\gamma})$ and $\boldsymbol{K}_{\mathrm{lit}}(\tilde{\gamma}, \gamma)$, which requires evaluating π_n for p models. Similarly, for the deletion proposal, we also need to evaluate π_n for p models. Hence, in our implementation, each swap proposal involves $2p$ evaluations of π_n. This is much more efficient than implementing an informed swap proposal exactly as described in (12), which requires evaluating π_n for $2(p - |\gamma|)|\gamma|$ models.

7.2 | LIT-MH on general discrete state spaces

Zanella (2020) considered 'locally balanced proposals' for general discrete-state-space problems. Let π be a distribution defined on a general discrete state space $\mathcal{X}$. For each x, let $\mathcal{N}(x) \subset \mathcal{X}$ denote its neighborhood. A locally balanced proposal scheme can be written as

$$\boldsymbol{K}_{\mathrm{lb}}(x, x') = \frac{f\left(\frac{\pi(x')}{\pi(x)}\right)}{Z_f(x)} \mathbb{1}_{\mathcal{N}(x)}(x'), \quad Z_f(x) = \sum_{y \in \mathcal{N}(x)} f\left(\frac{\pi(y)}{\pi(x)}\right), \tag{20}$$

where the 'balancing function' $f : (0, \infty) \to (0, \infty)$ must satisfy $f(b) = bf(b^{-1})$ for any $b > 0$. Examples of balancing functions include $f(b) = \sqrt{b}$ and $f(b) = 1 \vee b$. Consider an MH algorithm with proposal $\boldsymbol{K}_{\mathrm{lb}}$. A seemingly desirable property of balancing functions is that the acceptance probability of a proposal move from x to x' is given by

$$\mathrm{acc}(x, x') = \min\left\{1, \frac{Z_f(x)}{Z_f(x')}\right\}, \tag{21}$$

for any $x' \in \mathcal{N}(x)$. If $Z_f(x) \approx Z_f(x')$, this method should work well. Indeed, Zanella (2020) argued that if $\sup_{x,x' : x' \in \mathcal{N}(x)} Z_f(x)/Z_f(x') \to 1$, such a locally balanced proposal is asymptotically optimal. But, for problems like variable selection (which was not considered in Zanella, 2020), the behaviour of the function $x \mapsto Z_f(x)$ is very difficult to predict, and Table 1 confirms that for $f(b) = \sqrt{b}$, the informed MH algorithm with proposal (20) completely fails when the SNR is sufficiently large.

Motivated by Condition 1, consider some π that satisfies the following condition: there exist $x^* \in \mathcal{X}$, $\mathcal{T} : \mathcal{X} \to \mathcal{X}$, and $b_0 > 1$ such that for any $x \neq x^*$, $\mathcal{T}(x) \in \mathcal{N}(x)$ and $\pi(\mathcal{T}(x))/\pi(x) \geq b_0$. Define $\omega(x) = \pi(\mathcal{T}(x))/\pi(x)$ for each $x \neq x^*$. Note that by (20), $\boldsymbol{K}_{\mathrm{lb}}(x, \mathcal{T}(x)) = f(\omega(x))/Z_f(x)$. It follows from (21) that for any x such that $\mathcal{T}(x) \neq x^*$,

$$\boldsymbol{K}_{\mathrm{lb}}(x, \mathcal{T}(x))\mathrm{acc}(x, \mathcal{T}(x)) \leq \frac{f(\omega(x))}{Z_f(\mathcal{T}(x))} \leq \frac{f(\omega(x))}{f(\omega(\mathcal{T}(x)))}, \tag{22}$$

since $Z_f(\mathcal{T}(x)) \geq f(\omega(\mathcal{T}(x)))$. The ratio $f(\omega(x))/f(\omega(\mathcal{T}(x)))$ can be exceedingly small since it is possible that $\omega(\mathcal{T}(x))$ is much larger than $\omega(x)$. As we have seen in Example 1, for variable selection, $\omega(\mathcal{T}(x)) \gg \omega(x)$ can easily happen if there are correlated covariates and the sample size is large. Since collinearity is common for high-dimensional data, this analysis suggests that for general model selection problems, locally balanced MH schemes with proposal given by (20) may not have good performance when both n and p are large.

The main idea behind our LIT-MH algorithm can still be applied in this general setting. Pick constants $\bar{f} > \underline{f} > 0$ and modify (20) by

$$\tilde{\boldsymbol{K}}_{\mathrm{lb}}(x, x') = \frac{\underline{f} \vee f(\frac{\pi(x')}{\pi(x)}) \wedge \bar{f}}{\tilde{Z}_f(x)} \mathbb{1}_{\mathcal{N}(x)}(x'), \quad \tilde{Z}_f(x) = \sum_{y \in \mathcal{N}(x)} \underline{f} \vee f\left(\frac{\pi(y)}{\pi(x)}\right) \wedge \bar{f}.$$

This modification guarantees that a proposal move from x to x' has acceptance probability 1 as long as $\pi(x')/\pi(x)$ is sufficiently large, as shown in the following lemma.

Lemma 2. *Let π be an arbitrary probability distribution on $\mathcal{X}$. Consider the MH algorithm with proposal $\tilde{\boldsymbol{K}}_{\mathrm{lb}}$ given in (22) where $f : (0,\infty) \to (0,\infty)$ is an arbitrary non-decreasing function. Suppose there exists $b < \infty$ such that*

$$f\left(b^{-1}\right) \le \underline{f}, \text{ and } b \ge \frac{\overline{f}}{\underline{f}} \max_{x\in\mathcal{X}} |\mathcal{N}(x)|.$$

Then, for any $x, x' \in \mathcal{N}(x)$ such that $\pi(x')/\pi(x) \ge b$, the proposal move from x to x' has acceptance probability 1.

Proof. See Section S5.6 in Appendix A.

In Zanella (2020), one motivation for the locally balanced proposal was to mimic the behaviour of Metropolis-adjusted Langevin algorithms defined on continuous state spaces (Roberts & Rosenthal, 1998). However, for model selection problems with large sample sizes, the local posterior landscape can change drastically when we move from x to some $x' \in \mathcal{N}(x)$, which may result in strange behaviour of the MH chain (i.e. keep proposing some state x' with $\pi(x') \gg \pi(x)$ and getting rejected). One key observation of this work is that once we truncate the function f in (20), the mapping $x \mapsto \tilde{Z}_f(x)$ becomes much 'smoother' than $x \mapsto Z_f(x)$. Since there is almost no difference in computational cost between the two proposals $\boldsymbol{K}_{\mathrm{lb}}$ and $\tilde{\boldsymbol{K}}_{\mathrm{lb}}$, it is apparently always desirable to use the 'stabilized version' $\tilde{\boldsymbol{K}}_{\mathrm{lb}}$ in practice.

7.3 | On the LB-MH-1 algorithm for variable selection

The discussion in Section 7.2 explains why LB-MH-2 fails to perform well in our simulation study. Next, consider the LB-MH-1 algorithm, which also uses the balancing function $f(b) = \sqrt{b}$ to weight neighboring states. The only difference is that in LB-MH-1, we perform the proposal weighting for addition and deletion moves separately. It may be surprising that this simple modification improves the sampling performance substantially in our simulation studies.

To explain this, assume Condition 1 holds. By Condition (1b), as long as γ is underfitted, there exists some $\gamma' \in \mathcal{N}_{\mathrm{a}}(\gamma)$ such that $\pi_n(\gamma')/\pi_n(\gamma) \ge p^{c_1}$, and thus the proposal probability $\boldsymbol{K}_{\mathrm{lb}}(\gamma, \gamma')$ is large. Further, we have

$$B(\gamma,\gamma')\frac{\boldsymbol{K}_{\mathrm{lb}}(\gamma',\gamma)}{\boldsymbol{K}_{\mathrm{lb}}(\gamma,\gamma')} = \frac{\sum_{\tilde{\gamma}\in\mathcal{N}_{\mathrm{a}}(\gamma)} \sqrt{B(\gamma,\tilde{\gamma})}}{\sum_{\tilde{\gamma}\in\mathcal{N}_{\mathrm{d}}(\gamma')} \sqrt{B(\gamma',\tilde{\gamma})}} \ge \frac{p^{(c_1-\kappa)/2}}{s_0},$$

where the inequality follows from $B(\gamma', \tilde{\gamma}) \le p^{\kappa}$ for any $\tilde{\gamma} \in \mathcal{N}_{\mathrm{d}}(\gamma')$ and $|\mathcal{N}_{\mathrm{d}}(\gamma')| \le s_0$. So if the SNR is sufficiently large so that $c_1 > \kappa + 2$, the proposal will always be accepted. A similar argument shows that for an overfitted model γ, a proposal to remove a non-influential covariate will be always accepted if the constant c_0 in Condition (1a) is greater than 1. This heuristic argument explains why, unlike LB-MH-2, LB-MH-1 does not get trapped at a model because of extremely small acceptance probabilities of informed proposal moves. However, it is not clear whether LB-MH-1 can attain a dimension-free mixing time for high-dimensional variable selection, and even if it is possible, it would require stronger assumptions on the true model so that $c_1 > \kappa$.

The simulation study in Section 5 also shows that LB-MH-1 under-performs the two LIT-MH algorithms.

7.4 | Closing remarks

Theorem 1 provides the theoretical guarantee for the use of informed MCMC methods for high-dimensional problems, the proof of which relies on a novel 'two-stage drift condition' argument. Simulation studies show that LIT-MH is indeed much more efficient than the uninformed version, no matter whether the posterior distribution is multi-modal. As noted in Zanella (2020), one can further boost LIT-MH using parallel computing (Lee et al., 2010): the calculation of $\pi_n(\gamma')/\pi_n(\gamma)$ for each $\gamma' \in \mathcal{N}(\gamma)$ can be easily parallelized.

One major advantage of LIT-MH is its simplicity, which makes it both theoretically and practically appealing. The adaptive MCMC methods proposed by Griffin et al. (2021) have the usual sensitivities of possibly adapting to wrong information and require running multiple chains in the adaptation phase. The tempered Gibbs sampler of Zanella and Roberts (2019), which is one of the most efficient existing MCMC methods (see Supplement B.6 therein), is conceptually very similar to our method in that it selects the coordinate to update using local information of π_n. But, as a consequence of this informed updating scheme, the tempered Gibbs sampler requires the calculation of an importance weight in each iteration, which may reduce the efficiency of the sampler when the weight is unbounded. LIT-MH has a provable mixing time bound and, due to its simple design, can be combined with other MCMC techniques such as tempering, blocking and adaptive proposals. But whether further sophistication enhances the sampler's efficiency needs more investigation, which we leave for future work.

ACKNOWLEDGEMENTS

Gareth O. Roberts was supported by EPSRC grants EP/R018561/1 and EP/R034710/1, Jeffrey S. Rosenthal was supported by NSERC grant RGPIN-2019-04142, and Dootika Vats was supported by SERB grant SPG/2021/001322. We thank the submitters and participants of the two dbGaP studies (phs000308.v1.p1 and phs000238.v1.p1), which were funded by NIH.

DATA AVAILABILITY STATEMENT

Data available from the database of Genotypes and Phenotypes (dbGaP): https://www.ncbi.nlm.nih.gov/gap/.

ORCID

Quan Zhou https://orcid.org/0000-0003-3190-3598
Gareth O. Roberts https://orcid.org/0000-0002-4894-2537
Jeffrey S. Rosenthal https://orcid.org/0000-0002-5118-6808

REFERENCES

An, H., Huang, D., Yao, Q. Zhang, C. H. (2008) *Stepwise searching for feature variables in high-dimensional linear regression* (Technical report). Department of Statistics, The London School of Economics and Political Science.

Bailey, J.N., Loomis, S.J., Kang, J.H., Allingham, R., Gharahkhani, P., Khor, C.C. et al. (2016) Genome-wide association analysis identifies TXNRD2, ATXN2 and FOXC1 as susceptibility loci for primary open-angle glaucoma. *Nature Genetics*, 48(2), 189–194.

Baxendale, P.H. (2005) Renewal theory and computable convergence rates for geometrically ergodic Markov chains. *The Annals of Applied Probability*, 15(1B), 700–738.

Bierkens, J. (2016) Non-reversible metropolis-Hastings. *Statistics and Computing*, 26(6), 1213–1228.

Bierkens, J., Fearnhead, P. & Roberts, G. (2019) The ZIG-ZAG process and super-efficient sampling for Bayesian analysis of big data. *The Annals of Statistics*, 47(3), 1288–1320.

Bouchard-Côté, A., Vollmer, S.J. & Doucet, A. (2018) The bouncy particle sampler: A nonreversible rejection-free Markov chain Monte Carlo method. *Journal of the American Statistical Association*, 113(522), 855–867.

Brown, P.J., Vannucci, M. & Fearn, T. (1998) Multivariate Bayesian variable selection and prediction. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 60(3), 627–641.

Castillo, I., Schmidt-Hieber, J. & Van der Vaart, A. (2015) Bayesian linear regression with sparse priors. *Annals of Statistics*, 43(5), 1986–2018.

Chipman, H., George, E.I., McCulloch, R.E., Clyde, M., Foster, D.P. & Stine, R.A. (2001) The practical implementation of Bayesian model selection. *Lecture Notes-Monograph Series*, 38, 65–134.

Craig, J.E., Han, X., Qassim, A., Hassall, M., Cooke Bailey, J.N., Kinzy, T.G. et al. (2020) Multitrait analysis of glaucoma identifies new risk loci and enables polygenic prediction of disease susceptibility and progression. *Nature Genetics*, 52(2), 160–166.

Diaconis, P. & Stroock, D. (1991) Geometric bounds for eigenvalues of Markov chains. *The Annals of Applied Probability*, 1, 36–61.

Fan, J. & Lv, J. (2008) Sure independence screening for ultrahigh dimensional feature space. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(5), 849–911.

Fearnhead, P., Bierkens, J., Pollock, M. & Roberts, G.O. (2018) Piecewise deterministic Markov processes for continuous-time Monte Carlo. *Statistical Science*, 33(3), 386–412.

Fort, G., Moulines, É., Roberts, G.O. & Rosenthal, J.S. (2003) On the geometric ergodicity of hybrid samplers. *Journal of Applied Probability*, 40, 123–146.

Gagnon, P. & Doucet, A. (2020) Nonreversible jump algorithms for Bayesian nested model selection. *Journal of Computational and Graphical Statistics*, 30, 312–323.

Gao, C., van der Vaart, A.W. & Zhou, H.H. (2020) A general framework for Bayes structured linear models. *Annals of Statistics*, 48(5), 2848–2878.

George, E.I. & McCulloch, R.E. (1993) Variable selection via Gibbs sampling. *Journal of the American Statistical Association*, 88(423), 881–889.

George, E.I. & McCulloch, R.E. (1997) Approaches for Bayesian variable selection. *Statistica Sinica*, 7, 339–373.

Griffin, J.E., Łatuszyński, K.G. & Steel, M.F.J. (2021) In search of lost mixing time: adaptive Markov chain Monte Carlo schemes for Bayesian variable selection with very large p. *Biometrika*, 108(1), 53–69.

Guan, Y. & Stephens, M. (2011) Bayesian variable selection regression for genome-wide association studies and other large-scale problems. *The Annals of Applied Statistics*, 5, 1780–1815.

Hans, C., Dobra, A. & West, M. (2007) Shotgun stochastic search for "large *p*" regression. *Journal of the American Statistical Association*, 102(478), 507–516.

Jeong, S. & Ghosal, S. (2021) Unified Bayesian theory of sparse linear regression with nuisance parameters. *Electronic Journal of Statistics*, 15(1), 3040–3111.

Jin, R. & Tan, A. (2021) Fast Markov chain Monte Carlo for high-dimensional Bayesian regression models with shrinkage priors. *Journal of Computational and Graphical Statistics*, 30(3), 632–646.

Johndrow, J.E., Orenstein, P. & Bhattacharya, A. (2020) Scalable approximate MCMC algorithms for the horseshoe prior. *Journal of Machine Learning Research*, 21(73), 1–61.

Johnson, V.E. & Rossell, D. (2012) Bayesian model selection in high-dimensional settings. *Journal of the American Statistical Association*, 107(498), 649–660.

Jones, G.L. & Hobert, J.P. (2001) Honest exploration of intractable probability distributions via Markov chain Monte Carlo. *Statistical Science*, 16, 312–334.

Kass, R.E. & Raftery, A.E. (1995) Bayes factors. *Journal of the American Statistical Association*, 90(430), 773–795.

Khawaja, A.P., Cooke Bailey, J.N., Wareham, N.J., Scott, R.A., Simcoe, M., Igo, R.P. et al. (2018) Genome-wide analyses identify 68 new loci associated with intraocular pressure and improve risk prediction for primary open-angle glaucoma. *Nature Genetics*, 50(6), 778–782.

Lee, A., Yau, C., Giles, M.B., Doucet, A. & 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(4), 769–789.

Lee, C., Luo, Z.T. & Sang, H. (2021) T-LoHo: a Bayesian regularization model for structured sparsity and smoothness on graphs. *Advances in Neural Information Processing Systems*, 34, 598–609.

Levin, D.A., Peres, Y. & Wilmer, E.L. (2017) *Markov Chains and Mixing Times*, Vol. 107. Providence, Rhode Island: American Mathematical Society.

Luo, Z., Sang, H. & Mallick, B. (2021) A Bayesian contiguous partitioning method for learning clustered latent variables. *Journal of Machine Learning Research*, 22, 1748–1799.

Narisetty, N.N. & He, X. (2014) Bayesian variable selection with shrinking and diffusing priors. *The Annals of Statistics*, 42(2), 789–817.

O'Hara, R.B. & Sillanpää, M.J. (2009) A review of Bayesian variable selection methods: what, how and which. *Bayesian. Analysis*, 4(1), 85–117.

Osman, W., Low, S.-K., Takahashi, A., Kubo, M. & Nakamura, Y. (2012) A genome-wide association study in the Japanese population confirms 9p21 and 14q23 as susceptibility loci for primary open angle glaucoma. *Human Molecular Genetics*, 21(12), 2836–2842.

Plummer, M., Best, N., Cowles, K. & Vines, K. (2006) CODA: convergence diagnosis and output analysis for MCMC. *R News*, 6(1), 7–11.

Qin, Q. & Hobert, J.P. (2019) Convergence complexity analysis of Albert and Chib's algorithm for Bayesian probit regression. *Annals of Statistics*, 47(4), 2320–2347.

Roberts, G.O. & Rosenthal, J.S. (1998) Optimal scaling of discrete approximations to Langevin diffusions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 60(1), 255–268.

Roberts, G.O. & Tweedie, R.L. (1999) Bounds on regeneration times and convergence rates for Markov chains. *Stochastic Processes and Their Applications*, 80(2), 211–229.

Rosenthal, J.S. (1995) Minorization conditions and convergence rates for Markov chain Monte Carlo. *Journal of the American Statistical Association*, 90(430), 558–566.

Roy, V. & Hobert, J.P. (2007) Convergence rates and asymptotic standard errors for Markov chain Monte Carlo algorithms for Bayesian probit regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(4), 607–623.

Shin, M., Bhattacharya, A. & Johnson, V.E. (2018) Scalable Bayesian variable selection using nonlocal prior densities in ultrahigh-dimensional settings. *Statistica Sinica*, 28(2), 1053.

Sinclair, A. (1992) Improved bounds for mixing rates of Markov chains and multicommodity flow. *Combinatorics, Probability and Computing*, 1(4), 351–370.

Smith, M. & Kohn, R. (1996) Nonparametric regression using Bayesian variable selection. *Journal of Econometrics*, 75(2), 317–343.

Springelkamp, H., Höhn, R., Mishra, A., Hysi, P.G., Khor, C.C., Loomis, S.J. et al. (2014) Meta-analysis of genome-wide association studies identifies novel loci that influence cupping and the glaucomatous process. *Nature. Communications*, 5(1), 1–7.

Stephens, M. & Balding, D.J. (2009) Bayesian statistical methods for genetic association studies. *Nature Reviews Genetics*, 10(10), 681–690.

Titsias, M.K. & Yau, C. (2017) The Hamming ball sampler. *Journal of the American Statistical Association*, 112(520), 1598–1611.

Vats, D. (2017) Geometric ergodicity of Gibbs samplers in Bayesian penalized regression models. *Electronic Journal of Statistics*, 11(2), 4033–4064.

Vats, D., Flegal, J.M. & Jones, G.L. (2019) Multivariate output analysis for Markov chain Monte Carlo. *Biometrika*, 106(2), 321–337.

Yang, J. & Rosenthal, J.S. (2022) Complexity results for MCMC derived from quantitative bounds. *The Annals of Applied Probability*, to appear.

Yang, Y., Wainwright, M.J. & Jordan, M.I. (2016) On the computational complexity of high-dimensional Bayesian variable selection. *The Annals of Statistics*, 44(6), 2497–2532.

Zanella, G. (2020) Informed proposals for local MCMC in discrete spaces. *Journal of the American Statistical Association*, 115(530), 852–865.

Zanella, G. & Roberts, G. (2019) Scalable importance tempering and Bayesian variable selection. *Journal of the Royal Statistical Society Series B*, 81(3), 489–517.
Zhou, Q. & Chang, H. (2021) Complexity analysis of Bayesian learning of high-dimensional DAG models and their equivalence classes. *arXiv preprint arXiv:2101.04084*, 2021.

SUPPORTING INFORMATION

Additional supporting information may be found in the online version of the article at the publisher's website.

How to cite this article: Zhou, Q., Yang, J., Vats, D., Roberts, G.O. & Rosenthal, J.S. (2022) Dimension-free mixing for high-dimensional Bayesian variable selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 1–34. Available from: https://doi.org/10.1111/rssb.12546