The G-Wishart Weighted Proposal Algorithm: Efficient Posterior Computation for Gaussian Graphical Models

Willem van den Boom*
Yong Loo Lin School of Medicine, National University of Singapore
and
Alexandros Beskos
Department of Statistical Science, University College London
Alan Turing Institute, UK
and
Maria De Iorio
Yong Loo Lin School of Medicine, National University of Singapore
Singapore Institute for Clinical Sciences, A*STAR
Department of Statistical Science, University College London

Abstract

Gaussian graphical models can capture complex dependency structures among variables. For such models, Bayesian inference is attractive as it provides principled ways to incorporate prior information and to quantify uncertainty through the posterior distribution. However, posterior computation under the conjugate G-Wishart prior distribution on the precision matrix is expensive for general non-decomposable graphs. We therefore propose a new Markov chain Monte Carlo (MCMC) method named the G-Wishart weighted proposal algorithm (WWA). WWA's distinctive features include delayed acceptance MCMC, Gibbs updates for the precision matrix and an informed proposal distribution on the graph space that enables embarrassingly parallel computations. Compared to existing approaches, WWA reduces the frequency of the relatively expensive sampling from the G-Wishart distribution. This results in faster MCMC convergence, improved MCMC mixing and reduced computing time. Numerical studies on simulated and real data show that WWA provides a more efficient tool for posterior inference than competing state-of-the-art MCMC algorithms. Supplemental materials for the article are available online.

*Email: willem@yale-nus.edu.sg. This work was supported by the Singapore Ministry of Education Academic Research Fund Tier 2 under Grant MOE2019-T2-2-100.

Keywords: Exchange algorithm; Hyper Inverse Wishart distribution; Locally balanced proposal; Reversible jump MCMC; Scalable Bayesian computations

1 Introduction

Gaussian graphical models (GGMs, Dempster, 1972; Lauritzen, 1996) are a powerful tool to investigate conditional independence structure among variables which are represented by the nodes of a graph. Graph estimation is often very challenging given the dimensionality of the graph space. In the frequentist literature, different strategies have been proposed to bypass this problem. For example, Friedman et al. (2007) propose the graphical lasso, which involves the estimation of the precision matrix of a Multivariate Gaussian vector through penalised likelihood methods. Then, from the estimation of the covariance, the graph is constructed by drawing an edge between variables whose partial correlation is estimated as different from zero. Alternatively, nodewise regression (Zhou et al., 2011) approximates the joint distribution of the variables by considering individual regressions of each variable on the others. This leads to a computationally efficient set-up which is also amenable to parallelisation, at the cost of not being founded upon a probabilistically consistent model.

On the other hand, in the Bayesian framework, graph estimation requires the specification of a prior on the space of graphs and, conditionally on the graph, a prior on the precision matrix. Bayesian inference allows for principled uncertainty quantification, handling complexity through the specification of (conditionally independent) submodules and superior performance compared to frequentist methods. The posterior distribution on the graph space provided by the Bayesian framework provides principled and interpretable measures of uncertainty such as posterior edge inclusion probabilities. This contrasts with many frequentist methods such as the graphical lasso which conceptually focus on the precision matrix. Then, graph estimation happens through thresholding or penalisation of this matrix, which do not provide interpretable measures of uncertainty on the graph space. Additionally, the modular nature of MCMC enables its use in extended models that incorporate GGMs such as sparse seemingly unrelated regressions (e.g., Wang, 2010; Bhadra and Mallick, 2013) while propagating the uncertainty of the GGM to other parts of the model. Finally, the empirical comparisons in Mohammadi and Wit (2015) show that Bayesian inference with the G-Wishart prior considered in this work often outperforms frequentist methods such as the graphical lasso in terms of graph structure recovery and precision matrix estimation.

The previous points provide the rationale for this work, as the goal is to better enable and speed up Bayesian inference in GGMs. Posterior inference is usually performed through Markov chain Monte Carlo (MCMC, e.g., Wang and Li, 2012; Hinne et al., 2014) and more recently sequential Monte Carlo (SMC, Tan et al., 2017; van den Boom et al., 2021). However, these methods are often associated with high computational cost. A possible solution is to constrain the analysis to decomposable graphs (e.g., Giudici and Green, 1999; Letac and Massam, 2007; Scott and Carvalho, 2008; Wang, 2010; Bornn and Caron, 2011; Bhadra and Mallick, 2013) or to related subsets of the graph space (Khare et al., 2018) as the associated distributions are tractable. The assumption of decomposability is hard to justify from an applied perspective and increasingly restrictive as the number of nodes increases. In the large data limit where the posterior concentrates, decomposability results in spurious edges which constitute a minimal triangulation of the true graph (Fitch et al., 2014; Niu et al., 2021). In practice, this implies that up to half of the edges in the estimated graph can be spurious, even if the posterior is highly concentrated.

To avoid the assumption of decomposable graphs, Roverato (2002) introduces the G-Wishart prior distribution for the precision matrix conditional on a graph, which is an extension of the Hyper Inverse Wishart distribution (Dawid and Lauritzen, 1993) employed in the case of decomposable graphs. This allows for more flexibility at the cost of more expensive computations. Given the difficulties in exploring the posterior space, stochastic search methods in a Bayesian model have been developed (Jones et al., 2005; Scott and Carvalho, 2008; Lenkoski and Dobra, 2011), which aim to identify graphs with high posterior probability. Nevertheless, MCMC algorithms have received most attention in the literature (e.g., Wang and Li, 2012; Cheng and Lenkoski, 2012; Lenkoski, 2013; Mohammadi and Wit, 2015) as they allow for full posterior inference.

Inference with the G-Wishart distribution is challenging. For instance, Wang (2015), Gan et al. (2018), Li et al. (2019) and Sagar et al. (2021) obtain major improvements in computational efficiency by replacing the G-Wishart distribution with shrinkage priors on the precision matrix that enable fast Gibbs sampling updates or EM algorithms. Moreover, building on MCMC methods, SMC (Tan et al., 2017) and unbiased Monte Carlo approx-

imation (van den Boom et al., 2021) have also been considered as these techniques are *embarrassingly parallel*.

Still working with a G-Wishart prior, we propose an MCMC algorithm by carefully addressing the major computational bottlenecks in the current literature. Our work builds on advances in MCMC algorithms, in particular, work on delayed Metropolis-Hastings acceptance (Christen and Fox, 2005) and informed proposals (Zanella, 2019) which we extend to the graph literature. First, we propose a delayed acceptance MCMC step to reduce the number of times we need to sample from the G-Wishart distribution. Such sampling involves iterating an $O(pd^3)$ algorithm to convergence where p is the number of nodes and d the degree of the graph (Lenkoski, 2013). Moreover, we introduce Gibbs updates for the precision matrix enabled by a node reordering which further reduce the need to sample from the G-Wishart distribution. Finally, we develop an informed proposal distribution for graphs which enables the use of parallel computing environments still in an MCMC framework. As the main distinctive features of the proposed method relate to its proposal distribution on the graph space, we refer to it as the G-Wishart weighted proposal algorithm (WWA). We show that WWA improves computation significantly and allows for exploration of larger graph spaces.

The paper is structured as follows. Section 1.1 introduces Bayesian GGMs based on the G-Wishart prior, while Section 1.2 reviews related literature on posterior inference. Section 2 describes the proposed WWA and contextualises it. Section 3 presents simulation studies to investigate the performance of WWA and compares it with the state of the art. In Section 4, we consider a real data application. We conclude the paper in Section 5.

1.1 Model Description

Object of inference is a graph $G = (V, E)$ defined by a set of edges $E \subset \{(i, j) \mid 1 \leq i < j \leq p\}$ that represent links among the nodes in $V = \{1, \ldots, p\}$. In the GGM framework, we have an $n \times p$ data matrix Y with independent rows Y_i, $i = 1, \ldots, n$, corresponding to a p-dimensional random vector with its elements represented by nodes on the graph. Each Y_i is distributed according to a Multivariate Gaussian distribution $\mathcal{N}(0_{p\times 1}, K^{-1})$ with precision

matrix K. We assume that the precision matrix K depends on the graph G: we have that $K_{ij} = 0$ if nodes i and j are not connected, while if there is an edge between two nodes in the graph then the corresponding element of the precision matrix is different from zero with probability one. Thus, $K \in M^+(G)$ where $M^+(G)$ is the cone of positive-definite matrices K with $K_{ij} = 0$ for $(i, j) \notin E$. The graph G determines the conditional independence structure of the p variables in Y_i, $i = 1, \ldots, n$, since $K_{ij} = 0$ implies that the i-th and j-th columns of Y are independent conditionally on the others by properties of the Multivariate Gaussian distribution.

A popular choice as prior for the precision matrix K conditional on the graph G is the G-Wishart distribution $\mathcal{W}_G(\delta, D)$ as it induces conjugacy and allows one to work with non-decomposable graphs (Roverato, 2002). It is parameterised by degrees of freedom $\delta > 2$ and a positive-definite rate matrix D. Its density is

$$p(K \mid G) = \frac{1}{I_G(\delta, D)} |K|^{\delta/2-1} \exp\left\{-\tfrac{1}{2}\mathrm{tr}(K^\top D)\right\}, \quad K \in M^+(G),$$

where $I_G(\delta, D)$ is a normalising constant. Due to conjugacy, $K \mid G, Y \sim \mathcal{W}_G(\delta^\star, D^\star)$ where $\delta^\star = \delta + n$, $D^\star = D + Y^\top Y$. The model is completed by specifying a prior $p(G)$ on the graph space. We highlight that the following development does not assume any particular form for $p(G)$.

1.2 Posterior Distribution

The goal is to compute the posterior distribution (e.g., Atay-Kayis and Massam, 2005)

$$p(G \mid Y) \propto p(G) \int_{M^+(G)} p(K \mid G)\, p(Y \mid K)\, dK = \frac{p(G)\, I_G(\delta^\star, D^\star)}{(2\pi)^{np/2} I_G(\delta, D)}.$$

The normalising constant $I_G(\delta, D)$ does not have a simple analytical form for general non-decomposable G, making evaluation of a Metropolis-Hastings acceptance probability infeasible. To overcome this problem, Monte Carlo (Atay-Kayis and Massam, 2005) and Laplace (Moghaddam et al., 2009; Lenkoski and Dobra, 2011) approximations of $I_G(\delta, D)$ have been developed. Alternatively, Uhler et al. (2018) provide a recursive expression for $I_G(\delta, D)$, but it results in a computationally efficient procedure only for specific types of graphs.

Another line of work avoids direct evaluation of $I_G(\delta, D)$ through application of the exchange algorithm (Murray et al., 2006) within a broader MCMC. Wang and Li (2012) and Cheng and Lenkoski (2012) employ this strategy in which sampling from the G-Wishart is performed through the edgewise or the maximum clique block Gibbs sampler described in Wang and Li (2012). More recently, Hinne et al. (2014); Mohammadi and Wit (2015); van den Boom et al. (2021) propose methodology based on the exchange algorithm where sampling from $\mathcal{W}_G(\delta, D)$ is performed through the exact G-Wishart sampler by Lenkoski (2013), making the algorithm more accurate in terms of exploration of posterior space.

2 WWA: G-Wishart Weighted Proposal Algorithm

In this section, we introduce WWA which advances existing literature by (i) speeding MCMC convergence, (ii) improving the mixing of the chain and (iii) reducing computing time.

Algorithm 1 describes the double conditional Bayes factor (DCBF) sampler from Hinne et al. (2014). We use DCBF as a prototype for Bayesian algorithms suggested in the literature that allow for posterior inference on non-decomposable graphs in the context of GGMs with the G-Wishart prior. In the next three sections, we describe how our strategy allows us to overcome the main bottlenecks of such approaches.

First, we briefly explain the derivation of the acceptance probability of the DCBF sampler. We defer a more extensive and general explanation to Section S2 of the Appendix where we derive the WWA acceptance probabilities. Let $K^e = (\Phi^e)^\top \Phi^e$, where Φ^e is an upper triangular matrix and K^e is obtained from K by reordering the nodes such that the edge involved in the proposed graph change corresponds to nodes in the last two rows (columns) of K^e. Let Φ^e_{ij}, $i, j \in \{1, \ldots, p\}$, denote the elements of the Cholesky decomposition, and define $\Phi^e_{-f} := \Phi^e \setminus \{\Phi^e_{p-1,p}, \Phi^e_{pp}\}$. Then, consider as target the distribution $p(G, \Phi^e_{-f} \mid Y)$ as implied by the full posterior $p(G, K \mid Y) \propto p(G)\, p(K \mid G)\, p(Y \mid K)$. To compute the acceptance probability in Step 4 of Algorithm 1, we need the the expression (see Cheng and

Algorithm 1 (Hinne et al., 2014) A single DCBF MCMC Step.

Input: Graph G.

Output: MCMC update for G that preserves the posterior $p(G \mid Y)$.

For each edge $e \in \{(i,j) \mid 1 \leq i < j \leq p\}$, do the following:

1. Let $\widetilde{G} = (V, \widetilde{E})$ where $\widetilde{E} = E \cup \{e\}$ if $e \notin E$ and $\widetilde{E} = E \setminus \{e\}$ otherwise.
2. Reorder the nodes in G and $\widetilde{G}$ so that e connects nodes $p-1$ and p. Rearrange D, $D^\star$ accordingly. Denote all resulting quantities after reordering by a superscript e.
3. Draw $K^e \mid G, Y \sim \mathcal{W}_{G^e}(\delta^\star, D^{\star,e})$ and $\widetilde{K}^{0,e} \mid \widetilde{G} \sim \mathcal{W}_{\widetilde{G}^e}(\delta, D^e)$. Compute their respective upper triangular Cholesky decompositions Φ^e and $\widetilde{\Phi}^{0,e}$.
4. Set $G = \widetilde{G}$ w.p. $1 \wedge R_{\text{exchange}}$ where R_{exchange} is given by Equation (4).

Lenkoski, 2012, for a derivation)

$$\frac{p(Y, \Phi^e_{-f} \mid \widetilde{G})}{p(Y, \Phi^e_{-f} \mid G)} = N(\Phi^e_{-f}, D^{\star,e})^{|\widetilde{E}|-|E|} \frac{I_G(\delta, D)}{I_{\widetilde{G}}(\delta, D)} \tag{1}$$

where $N(\Phi^e_{-f}, D^{\star,e})$ is an analytically available quantity:

$$N(\Phi^e_{-f}, D^{\star,e}) := \Phi^e_{p-1,p-1} \sqrt{\frac{2\pi}{D^{\star,e}_{pp}}} \exp\left\{ \frac{D^{\star,e}_{pp}}{2} \left(\frac{\Phi^e_{p-1,p-1} D^{\star,e}_{p-1,p}}{D^{\star,e}_{pp}} - \frac{\sum_{i=1}^{p-2} \Phi^e_{i,p-1} \Phi^e_{ip}}{\Phi^e_{p-1,p-1}} \right)^2 \right\} \tag{2}$$

The ratio in (1) is not of direct use due to the intractable normalising constants $I_G(\delta, D)$, $I_{\widetilde{G}}(\delta, D)$. The exchange algorithm (Murray et al., 2006) avoids the computation of the normalising constant via the introduction of a Metropolis step defined on an augmented target distribution, which still admits as marginal the desired posterior $p(G \mid Y)$. Specifically, as shown in Step 3 of Algorithm 1, the exchange algorithm requires simulating $\widetilde{K}^{0,e}$ from the G-Wishart prior based on the proposed graph $\widetilde{G}$. Let $\widetilde{\Phi}^{0,e}_{-f}$ be defined analogously to Φ^e_{-f} and denote its distribution by $p(\widetilde{\Phi}^{0,e}_{-f} \mid \widetilde{G})$. Consider the distribution defined on the augmented space

$$p(G, \Phi^e_{-f}, \widetilde{G}, \widetilde{\Phi}^{0,e}_{-f} \mid Y) \propto p(G, \Phi^e_{-f} \mid Y)\, p(\widetilde{\Phi}^{0,e}_{-f} \mid \widetilde{G}). \tag{3}$$

DCBF proposes the deterministic exchange $G \leftrightarrow \widetilde{G}$ on the above target. Standard application of detailed balance identifies the acceptance probability as $1 \wedge R_{\text{exchange}}$ with

$$\begin{aligned} R_{\text{exchange}} &= \frac{p(\widetilde{G}, \Phi^e_{-f}, G, \widetilde{\Phi}^{0,e}_{-f} \mid Y)}{p(G, \Phi^e_{-f}, \widetilde{G}, \widetilde{\Phi}^{0,e}_{-f} \mid Y)} = \frac{p(Y, \Phi^e_{-f} \mid \widetilde{G})\, p(\widetilde{G})\, p(\widetilde{\Phi}^{0,e}_{-f} \mid G)}{p(Y, \Phi^e_{-f} \mid G)\, p(G)\, p(\widetilde{\Phi}^{0,e}_{-f} \mid \widetilde{G})} \\ &= \frac{p(\widetilde{G})}{p(G)} \left\{ \frac{N(\Phi^e_{-f}, D^{\star,e})}{N(\widetilde{\Phi}^{0,e}_{-f}, D^e)} \right\}^{|\widetilde{E}|-|E|} \end{aligned} \tag{4}$$

where the last equality follows from (1).

2.1 Full Conditionals for K

In Algorithm 1, sampling from the G-Wishart distributions in Step 3 is computationally expensive. Moreover, sampling from $\mathcal{W}_{G^e}(\delta^\star, D^{\star,e})$ can be considerably slower than sampling from $\mathcal{W}_{\widetilde{G}^e}(\delta, D^e)$ under the default hyperparameter choice $D = I_p$. To avoid repeated sampling of the full matrix K, WWA updates only the elements in the Cholesky decomposition of K that are affected by the change in the graph. WWA makes use of the following conditional distributions. First, for $\Phi^e_{p-1,p}$,

$$\Phi^e_{p-1,p} \mid G,\, \Phi^e_{-f},\, Y \sim \mathcal{N}\left(\frac{-\Phi^e_{p-1,p-1} D^{\star,e}_{p-1,p}}{D^{\star,e}_{p,p}}, \frac{1}{D^{\star,e}_{pp}} \right), \qquad e \in E, \tag{5a}$$

$$\Phi^e_{p-1,p} \mid G,\, \Phi^e_{-f},\, Y = -\frac{1}{\Phi^e_{p-1,p-1}} \sum_{l=1}^{p-2} \Phi^e_{l,p-1} \Phi^e_{lp}, \qquad e \notin E, \tag{5b}$$

where Equation (5a) follows from Equation (5) of van den Boom et al. (2021) and Equation (5b) is Equation (10) of Roverato (2002). For Φ^e_{pp}, we derive in Section S1 of the Appendix

$$D^{\star,e}_{pp} \, (\Phi^e_{pp})^2 \mid G,\, \Phi^e_{p-1,p},\, \Phi^e_{-f},\, Y \sim \chi^2(\delta^\star). \tag{6}$$

Note that the idea of updating only $\Phi^e_{p-1,p}$ and Φ^e_{pp} has already been mentioned in Cheng and Lenkoski (2012) but as part of an approximate rather than an exact MCMC algorithm.

2.2 Approximations

We consider the approximation for the ratio of intractable normalising constants derived by Mohammadi et al. (2021) under the default prior choice $D = I_p$. That is,

$$\frac{I_G(\delta, D)}{I_{\widetilde{G}}(\delta, D)} \approx \left\{ \frac{\Gamma\left(\frac{\delta+d_{\widetilde{G}}}{2}\right)}{2\sqrt{\pi}\,\Gamma\left(\frac{\delta+d_{\widetilde{G}}+1}{2}\right)} \right\}^{|\widetilde{E}|-|E|} =: \widehat{I_G/I_{\widetilde{G}}} \tag{7}$$

where $d_{\widetilde{G}}$ is the number of paths of length two linking the endpoints of edge e.

Notice that one can avoid working with the extended target in (3), and apply the exchange step $G \leftrightarrow \widetilde{G}$ directly on the target $p(G, \Phi^e_{-f} \mid Y)$ with acceptance probability $1 \wedge R$ where

$$R = \frac{p(\widetilde{G}, \Phi^e_{-f} \mid Y)}{p(G, \Phi^e_{-f} \mid Y)} \equiv \frac{p_u(\widetilde{G}, \Phi^e_{-f} \mid Y)}{p_u(G, \Phi^e_{-f} \mid Y)} \times \frac{I_G(\delta, D)}{I_{\widetilde{G}}(\delta, D)} \tag{8}$$

for the analytically available unnormalised densities $p_u(\cdot, \cdot \mid \cdot)$ defined in the obvious way via (1). From (7), (8), one can obtain the approximation

$$\widehat{R} \equiv \widehat{R}(G, \widetilde{G}, K) := \frac{p_u(\widetilde{G}, \Phi^e_{-f} \mid Y)}{p_u(G, \Phi^e_{-f} \mid Y)} \times \widehat{I_G/I_{\widetilde{G}}} \tag{9}$$

We make use of this approximation both within the development of our informed proposal and for the introduction of a delayed acceptance step within WWA. Combining (1), (8) leads to an explicit expression for $\widehat{R}$:

$$\widehat{R} = \frac{p(\widetilde{G})}{p(G)} \left\{ N(\Phi^e_{-f}, D^{\star,e}) \, \frac{\Gamma\left(\frac{\delta+d_{\widetilde{G}}}{2}\right)}{2\sqrt{\pi}\,\Gamma\left(\frac{\delta+d_{\widetilde{G}}+1}{2}\right)} \right\}^{|\widetilde{E}|-|E|}$$

2.3 Informed Proposal

WWA improves MCMC convergence and mixing per G-Wishart sample through the use of a proposal distribution that is informed by the target. We will first describe a simple modification of the proposal that is blind to the target before proceeding to the description of the informed approach.

First, notice that at every MCMC iteration, Algorithm 1 scans through all the edges. At each such substep, it proposes graph $\widetilde{G}$ with the edge removed if it is present in the

current graph G or vice versa. This is similar in rationale to specifying a uniform proposal distribution $q(\widetilde{G} \mid G)$ on which edge to flip. That is, $q(\widetilde{G} \mid G) = 1/m_{\max}$ for $\widetilde{G} \in \mathsf{nbd}(G)$ where $m_{\max} = p(p-1)/2$ denotes the maximum number of edges and the neighbourhood $\mathsf{nbd}(G)$ is the set of $m_{\max}$ graphs that differ from G by exactly one edge. A downside of the uniform proposal is that the probability of removing an edge equals $|E|/m_{\max}$ which is usually small, especially when a shrinkage prior on graphs is used. A possible solution is offered by Dobra et al. (2011) who first propose to remove or add an edge with probability 0.5 and then pick an edge uniformly at random from the appropriate subset. Obviously, for $|E| \in \{0,\, m_{\max}\}$, we propose to add and remove an edge accordingly. This results in the proposal

$$q(\widetilde{G} \mid G) = \begin{cases} \frac{1}{m_{\max}}, & |E| = 0,\, m_{\max}, \\ \frac{1}{2|E|}, & |\widetilde{E}| < |E| \neq m_{\max}, \\ \frac{1}{2(m_{\max} - |E|)}, & |\widetilde{E}| > |E| \neq 0, \end{cases} \tag{10}$$

for $\widetilde{G} \in \mathsf{nbd}(G)$.

Second, and most importantly, WWA makes use of a proposal for the graph that learns from the target. Locally balanced proposals (Zanella, 2019) provide inspiration to further improve $q(\widetilde{G} \mid G)$ defined above. Such proposals are informed by an *embarrassingly parallel* scan through the neighbourhood of the current discrete state in a Markov chain. Specifically, denote the current and proposed states by x and $\widetilde{x}$, respectively, the target distribution by $\pi(x)$ and some baseline proposal by $q(\widetilde{x} \mid x)$. Then, an informed proposal $Q(\widetilde{x} \mid x)$ is defined by

$$Q(\widetilde{x} \mid x) \propto g\left\{\frac{\pi(\widetilde{x})}{\pi(x)}\right\} q(\widetilde{x} \mid x), \tag{11}$$

for some balancing function $g(t)$. Here, the transition kernel $Q(\widetilde{x} \mid x)$ is locally balanced if and only if $g(t) = t\, g(1/t)$ (Zanella, 2019).

The role of the balancing function is to redirect the proposal towards candidates of high posterior probability. The aggressiveness of the redirection is determined by the shape of $g(t)$. In practice, best MCMC mixing results from a balance between information from the neighbourhood scan, which concentrates the proposal, and the diffuseness of the proposal.

We employ the balancing function $g(t) = t/(1+t)$ as suggested by Zanella (2019), although WWA is well defined for any $g(t)$. In our experiments, we also consider $g(t) = \sqrt{t}$ as its unboundedness might help convergence. We find that this alternative choice results in both worse convergence and mixing (results not shown). This reduced performance probably stems from $g(t) = \sqrt{t}$ resulting in a too concentrated proposal. Zanella (2019) derives $g(t) = t/(1+t)$ as the optimal choice in an example, but also notes that many similarly behaving balancing functions lead to virtually identical MCMC performance.

The use of an informed proposal in a Metropolis-Hastings acceptance probability requires the normalising constant of $Q(\widetilde{x} \mid x)$ in (11). Computing the constant involves computing $\pi(\widetilde{x})/\pi(x)$ for all $\widetilde{x}$ in the support of $q(\widetilde{x} \mid x)$. This task is *embarrassingly parallel*.

WWA develops an informed proposal in the context of GGMs with $q(\widetilde{G} \mid G)$ in Equation (10) as baseline proposal. We note that the ratio $\pi(\widetilde{x})/\pi(x)$ in (11) only serves to improve the proposal. Instead, we use the analytically available approximation of the ratio of targets $\widehat{R}$ in (9). Notice that this ratio involves the current precision matrix K. That is, we have

$$\begin{aligned} Q(\widetilde{G} \mid G, K) &:= C(G,K) \cdot g\left\{ \frac{p_u(\widetilde{G}, \Phi^e_{-f} \mid Y)}{p_u(G, \Phi^e_{-f} \mid Y)} \times \widehat{I_G/I_{\widetilde{G}}} \right\} q(\widetilde{G} \mid G) \\ &= C(G,K) \cdot g\left\{ \widehat{R}(G, \widetilde{G}, K) \right\} q(\widetilde{G} \mid G), \end{aligned} \tag{12}$$

for a normalising constant $C(G,K)$. Thus, the informed proposal for $\widetilde{G}$ has the form $Q(\widetilde{G} \mid G, K)$, differently from Zanella (2019) where the informed proposal only depends on the discrete state. Moreover, in the GGM context, an update on the graph leads to an update on the precision matrix (with a distribution defined on a continuous space). Such considerations are carefully addressed in Section S2 of the Appendix to ensure correctness of the deduced MCMC.

2.4 Delayed Acceptance

We make use of the approximation $\widehat{R}$ in (9) for the introduction of a delayed acceptance (DA) step (Christen and Fox, 2005) within WWA. We develop the DA approach by applying the idea of targeting $p(G, \Phi^e_{-f} \mid Y)$ with a proposed exchange $G \leftrightarrow \widetilde{G}$ based on the kernel

$Q(\widetilde{G} \mid G, K)$ in (12), and a simultaneous exchange between $K \leftrightarrow \widetilde{K}$ where $\widetilde{K}$ involves the constituent elements $\widetilde{\Phi}^e_{-f} = \Phi^e_{-f}$, $\widetilde{\Phi}^e_{pp} = \Phi^e_{pp}$ (i.e., the same as the corresponding elements of K) and sampling $\widetilde{\Phi}^e_{p-1,p}$ according to (5).

Following the DA idea, the approximation $\widehat{R}$ in (9) will be used in place of the ratio of targets. That is, we 'promote' a proposed $\widetilde{G}$ with acceptance probability $1 \wedge \widehat{R}_{\text{DA}}$ where

$$\begin{aligned} \widehat{R}_{\text{DA}} &:= \frac{p_u(\widetilde{G}, \Phi^e_{-f} \mid Y)}{p_u(G, \Phi^e_{-f} \mid Y)} \times \widehat{I_G / I_{\widetilde{G}}} \times \frac{Q(G \mid \widetilde{G}, \widetilde{K})}{Q(\widetilde{G} \mid G, K)} \\ &= \widehat{R} \times \frac{Q(G \mid \widetilde{G}, \widetilde{K})}{Q(\widetilde{G} \mid G, K)} \end{aligned} \tag{13}$$

This promotion step provides a speed-up over the exchange algorithm of acceptance probability $1 \wedge R_{\text{exchange}}$ by not having to sample from $\mathcal{W}_{\widetilde{G}}(\delta, D)$ at the cost of targeting the wrong distribution. Use of the complete machinery of the DA MCMC in the WWA algorithm corrects for this inconsistency (see Algorithm 2).

DA has been originally developed to employ approximate posteriors within an exact MCMC, while we directly approximate the acceptance ratio. Specifically, our use of DA involves first a Metropolis-Hastings step with approximate acceptance ratio. Then, the outcome is treated as proposal in a second (delayed) accept-reject step that uses the exact ratio in (4). This has the advantage that one only needs to perform the exchange algorithm when an acceptance in the approximate Metropolis-Hastings is achieved. This implies that 'poor' proposed graphs get rejected quickly and more computational effort is spent on regions of the space with high posterior probability.

2.5 The Complete WWA Algorithm

Algorithm 2 details the WWA algorithm. A proposed $\widetilde{G}$ is associated with a $\widetilde{K}$ as the graph imposes a sparsity pattern on the precision matrix. To ensure detailed balance, $\widetilde{K}$ appears in the computation of the reverse probability $Q(G \mid \widetilde{G}, \widetilde{K})$. As such, the MCMC update needs to be joint on K and G where the dimensionality $p + |E|$ of the continuous state space of K varies with G, since every time we remove or add an edge, the number of free parameters in K changes. In Section S2 of the Appendix, the resulting acceptance probability is derived

Algorithm 2 A Single WWA MCMC Step.

Input: Graph G.

Output: MCMC update for G such that the invariant distribution is the posterior $p(G \mid Y)$.

1. Draw $K \mid G, Y \sim \mathcal{W}_G(\delta^\star, D^\star)$.

2. Repeat the following single-edge update a number of times n_E:

   (a) Sample $\widetilde{G}$ from the informed proposal $Q(\widetilde{G} \mid G, K)$ given by (12) with $g(t) = t/(1+t)$.

   (b) Denote the edge in which G and $\widetilde{G}$ differ by e. Reorder the nodes in G and $\widetilde{G}$ so that e connects nodes $p-1$ and p. Rearrange K, D and $D^\star$ accordingly. Denote the resulting quantities by a superscript e.

   (c) Denote the upper triangular Cholesky decomposition of K^e by Φ^e. Update Φ^e_{pp} according to (6).

   (d) Generate a $\widetilde{K}^e$ corresponding with $\widetilde{G}^e$ from K by setting $\widetilde{\Phi}^e_{-f} = \Phi^e_{-f}$, $\widetilde{\Phi}^e_{pp} = \Phi^e_{pp}$, and sampling $\widetilde{\Phi}^e_{p-1,p}$ according to (5).

   (e) Compute $Q(G \mid \widetilde{G}, \widetilde{K})$.

   (f) 'Promote' $\widetilde{G}$ to be considered for delayed acceptance w.p. $1 \wedge \widehat{R}_{\mathrm{DA}}$, where $\widehat{R}_{\mathrm{DA}}$ is given by (13). If $\widetilde{G}$ is promoted:

      i. Sample $\widetilde{K}^{0,e} \mid \widetilde{G} \sim \mathcal{W}_{\widetilde{G}^e}(\delta, D^e)$.

      ii. Set $G = \widetilde{G}$ and $K = \widetilde{K}$ w.p. $1 \wedge R_{\mathrm{DA}}$ where

$$R_{\mathrm{DA}} = R_{\mathrm{exchange}} \frac{(1 \wedge \widehat{R}_{\mathrm{DA}}^{-1})\, Q(G \mid \widetilde{G}, \widetilde{K})}{(1 \wedge \widehat{R}_{\mathrm{DA}})\, Q(\widetilde{G} \mid G, K)}$$

      where R_{exchange} is given by (4).

via reversible jump MCMC (Green, 1995) to account for the transdimensionality.

In Section S2 of the Appendix, we derive the acceptance probabilities involved in Step 2f. The detailed balance condition on an extended space implies an acceptance probability resulting in the correct invariant distribution on the variable of interest G. The construction of the extended space uses ideas from the exchange algorithm (Murray et al., 2006). We also apply delayed acceptance which does not affect the invariant distribution (Christen and Fox, 2005, Theorem 1) if $\widehat{R}_{\mathrm{DA}} > 0$, which implies $\widehat{I_G/I_{\widetilde{G}}} > 0$ by (13). Ultimately, the update is a Metropolis-Hastings step with an invariant distribution that has as marginal for G the target distribution $p(G \mid Y)$. Section S6 of the Appendix confirms empirically that WWA recovers $p(G \mid Y)$. Any approximation that satisfies $\widehat{I_G/I_{\widetilde{G}}} > 0$ leads to an MCMC that converges to $p(G \mid Y)$. For instance, Section S3 of the Appendix considers $\widehat{I_G/I_{\widetilde{G}}} = 1$ instead of Equation (7).

The relative computational cost of sampling from $\mathcal{W}_G(\delta^\star, D^\star)$ in Step 1 becomes negligible if the number of single edge updates n_E is sufficiently large, e.g. $n_E = p$. Then, the embarrassingly parallel computation of the informed proposal in Steps 2a and 2e, and the sampling from $\mathcal{W}_{\widetilde{G}}(\delta, D)$ in Step 2(f)i carry the vast majority of computational cost (in most applications more than 90%).

To efficiently sample from the G-Wishart distribution in WWA, we combine graph decomposition with the G-Wishart sampler of Lenkoski (2013). The main idea is as follows: first, we split the graph into connected components as sampling of the rows and columns of the precision matrix can be done independently for each connected component. Note that each independent component can be sampled from a G-Wishart of appropriate dimension, as the entire precision matrix can be rewritten as a block matrix. G-Wishart sampling for a connected component proceeds using a perfectly ordered clique minimal separator decomposition (see, for example, Berry et al., 2010, for an introduction to graph decomposition) as detailed in Wang and Carvalho (2010). Note that Carvalho et al. (2007) first mention the idea of sampling from the G-Wishart exploiting a decomposition of the graph. We opt for the MCSM-Atom-Tree algorithm of Berry et al. (2014) to compute a perfectly ordered clique minimal separator decomposition at negligible cost. The

decomposition splits the graphs in complete (i.e. cliques) and incomplete prime graphs. In the first case, we can use a standard Wishart sampler, while the latter requires sampling from a G-Wishart. WWA uses the G-Wishart sampler of Lenkoski (2013) for the incomplete prime graphs. The rejection sampler of Wang and Carvalho (2010) is an alternative for small incomplete prime graphs where it can be faster, but any speed-up would be negligible as the main computational cost derives from sampling large incomplete prime graphs. We empirically show in Section S4 of the Appendix that graph decomposition can substantially speed up sampling from the G-Wishart distribution for sparse graphs, and that most graphs are not sparse and do not have an effective decomposition for $p \geq 20$ nodes, in which case the method from Lenkoski (2013) without graph decomposition performs similarly.

The G-Wishart sampler of Lenkoski (2013) is an iterative algorithm initialised at a Wishart random variate. We compare this approach with initialising at a $\bar{G}$-Wishart random variate for some decomposable graph $\bar{G}$ that contains all edges of G, i.e. $E \subset \bar{E}$. Such initialisation also results in a G-Wishart sampler. However, we found that initialising with a minimal triangulation $\bar{G}$ of G results in similar or increased computational cost, depending on the G, compared to the method from Lenkoski (2013) (results not shown).

WWA requires a reordering of the nodes in Step 2b as explained in Section 2.1. In Step 2c, we update Φ^e_{pp}, which is not necessary for a valid MCMC as its distribution does not depend on the edge e being in the graph or not. Nonetheless, WWA includes it as its computational cost is negligible and it improves mixing, especially for a large number of single edge updates n_E. The CL algorithm in Cheng and Lenkoski (2012) also includes the step.

2.6 Related Work

In this section, we contextualise WWA in reference to previous work on MCMC for graphs. Wang and Li (2012) and Cheng and Lenkoski (2012) also consider doing a single edge update many times for each full update of K as in Step 2 of Algorithm 2. Additionally, they describe a two stage procedure that resembles delayed acceptance. In Section S5 of the Appendix, we describe the CL algorithm of Cheng and Lenkoski (2012). Its first accept-reject step

(for individual edges) uses Barker's algorithm (Barker, 1965) with acceptance ratio R_{exchange} given in (4) where the term $N(\widetilde{\Phi}^{0,e}_{-f}, D^e)$ is set to one. To correct for this approximation, they, then, introduce a second Metropolis-Hastings accept-reject step. This combination of steps is, in fact, a delayed acceptance, but both Wang and Li (2012) and Cheng and Lenkoski (2012) do not explicitly justify it as such. Effectively, they approximate the ratio of normalising constants by one while WWA uses (7).

WWA's computation of the informed proposal $Q(\widetilde{G} \mid G, K)$ is embarrassingly parallel. In this respect, so is the calculation of birth and death rates in Mohammadi and Wit (2015). Moreover, The R package `BDgraph` (Mohammadi and Wit, 2019) approximates these rates using (7) by default, which is the same approximation used for $Q(\widetilde{G} \mid G, K)$ in Step 2a of Algorithm 2. Unlike WWA, `BDgraph` does not correct for the fact that it uses an approximation.

The embarrassingly parallel search through the neighbourhood $\mathsf{nbd}(G)$, which constitutes $Q(\widetilde{G} \mid G, K)$, is reminiscent of the parallel computation enabled by shotgun stochastic search (SSS, Hans et al., 2007). Jones et al. (2005) apply SSS for stochastic search on the graph space using an approximate likelihood for GGMs. WWA similarly enables parallel computing in an MCMC framework while still using the exact likelihood.

3 Simulation Studies

We compare WWA (Algorithm 2) with DCBF (Algorithm 1) as DCBF can be considered the state of the art for MCMC in GGMs with a G-Wishart distribution as shown by Hinne et al. (2014). Additional comparisons in Section S6 of the Appendix consider also the CL algorithm of Cheng and Lenkoski (2012) and `BDgraph` (Mohammadi and Wit, 2019) which, unlike DCBF, do not have the exact posterior $p(G \mid Y)$ as invariant distribution. The comparisons show that the CL algorithm can provide accurate estimates of the posterior edge inclusion probabilities despite being an approximate MCMC. The computational efficiency of WWA's exact MCMC is comparable to the approximate MCMC of the CL algorithm in a simulation with $p = 100$ nodes. `BDgraph`'s approximations have a larger effect on inclusion probability estimates. For completeness, we mention here that we have

not included a comparison with the WL algorithm from Wang and Li (2012) as Cheng and Lenkoski (2012) show that their CL algorithm substantially outperforms the WL algorithm in terms of computational speed and accuracy of posterior approximation.

We choose $n_E = p$ for the number of single edge updates in Algorithm 2. To bring the computational cost of Algorithm 1 more in line with WWA and for a fairer comparison, we slightly modify DCBF: instead of executing the steps in Algorithm 1 for all $m_{\max}$ possible edges, we execute them for p edges drawn uniformly at random with replacement.

To measure MCMC efficiency, we use as metric the cost of an independent sample, which is the computing time required for a unit increase in the effective sample size (Fang et al., 2020):

$$\begin{aligned}\text{cost of an independent sample} &= \frac{\text{number of MCMC steps}}{\text{effective sample size}} \times \text{cost per step} \\ &= \text{integrated autocorrelation time} \times \text{cost per step}\end{aligned}$$

This captures MCMC mixing and adjusts for computational cost. MCMC convergence can additionally be a computational bottleneck, especially if an effective initialisation is not available. Therefore, we also discuss convergence issues in Section 4. The integrated autocorrelation time is computed for the number of edges $|E|$ using the R package `LaplacesDemon` (Statisticat, LLC., 2020). The cost of the embarrassingly parallel computation of the informed proposal in Steps 2a and 2e of Algorithm 2 is assessed based on 128 CPU cores. To make computing times comparable, all methods are implemented in C++ and use the same routines as much as possible, for instance, to sample from the G-Wishart distribution.

3.1 Cycle Graphs

A major improvement in computational cost of WWA over DCBF derives from reducing the number of times we need to sample from the G-Wishart distribution. The gains associated with WWA will thus be larger if G-Wishart sampling is slower. This situation arises, for example, when G contains large incomplete prime graphs. To highlight this point, we consider cycle graphs, which are themselves incomplete prime graphs. We follow Section 6.2 of Wang and Li (2012) to simulate data from cycle graphs. In the G-Wishart prior, we

set $\delta = 3$ and $D = I_p$. The edges are a priori independent with edge inclusion probability $\rho = 2/(p-1)$. That is, $p(G) = \rho^{|E|}(1-\rho)^{m_{\max}-|E|}$. We simulate $n = \frac{3}{2}p$ random vectors Y_i from $\mathcal{N}(0_{p\times 1},\, K^{-1})$ with a precision matrix K given by $K_{ii} = 1$ for $i = 1, \ldots, p$, $K_{ij} = 0.5$ for $|i-j| = 1$, $K_{1p} = K_{p1} = 0.4$ and all other elements being equal to zero. We simulate data for $p = 10, 20, 40$. The performance of the two algorithms is assessed over 32 replicates of the simulations.

We run the MCMC for 11,000 iterations, discarding the first 1,000 as burn-in. We initialise the graph at the true cycle for all algorithms. We compare the performance of WWA with (i) DCBF; (ii) WWA without the delayed acceptance and the informed proposal; (iii) WWA with delayed acceptance but without the informed proposal; and (iv) WWA with the informed proposal but without delayed acceptance. When we do not use the informed proposal, we set $q(\widetilde{G} \mid G)$ equal to (10). When we do not perform delayed acceptance in WWA, we use the acceptance probability in Step 4 of Algorithm 1 directly. These extra comparisons provide insight into the role of the different innovations of WWA.

Figure 1 shows that WWA provides more efficient posterior computation than DCBF on these simulated data. This difference increases with the number of nodes with WWA being 39 times more efficient than DCBF for $p = 40$ nodes. The MCMC without the informed proposal outperforms WWA for $p = 40$. This is probably a result of the informed proposal and the delayed acceptance both relying on the same approximation in (7): the informed proposal, compared to the base proposal $q(\widetilde{G} \mid G)$, increases the acceptance ratio of the first approximate accept-reject step in DA MCMC, but this increased acceptance does not translate to a proportional increase in the overall acceptance ratio. These effects are compounded by the approximation in (7) becoming less accurate for larger graphs (Letac and Massam, 2007). The result is that Step 2(f)i of Algorithm 2, which involves the relatively expensive sampling from $\mathcal{W}_{\widetilde{G}}(\delta, D)$, is evaluated more often without a corresponding improvement in MCMC mixing, i.e. the gain in mixing from the informed proposal does not compensate for this extra sampling.

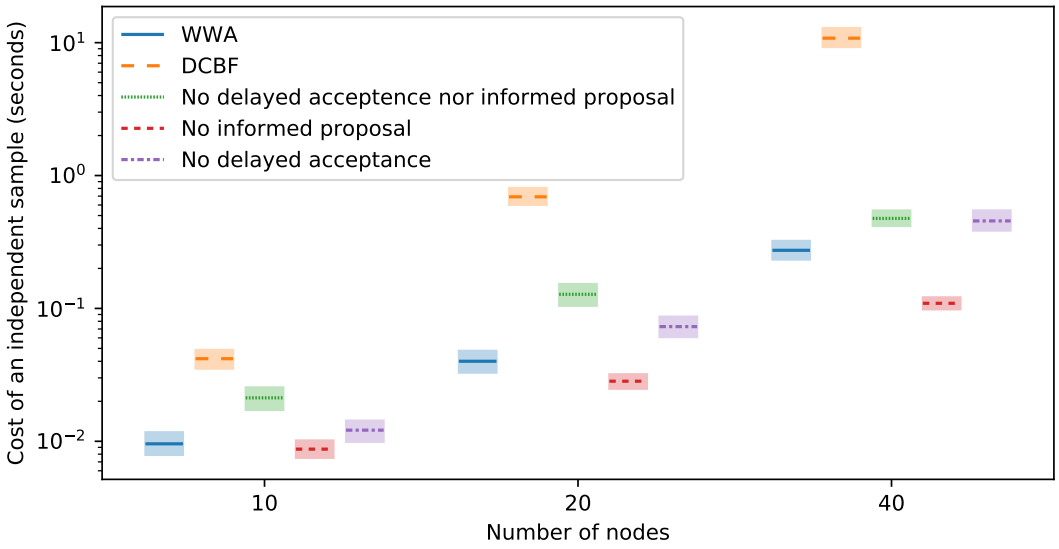

Figure 1: Cost of posterior computations versus the number of nodes for the data simulated from a cycle graph. The lines represent means over the 32 repetitions for DCBF and WWA as well as different specifications of WWA. The shaded areas are 95% bootstrapped confidence intervals.

3.2 Uniformly Sampled Graphs

In this section, we compare the performance of the different algorithms on data simulated from the Bayesian model described in Section 1.1 with $n = 2p$ and a uniform prior on graphs: $p(G) = 2^{-m_{\max}}$. In particular, for each replicate, we generate a graph G from this uniform distribution $p(G)$, sample a precision matrix K from $\mathcal{W}_G(\delta, D) = \mathcal{W}_G(3, I_p)$ and data from $\mathcal{N}(0_{p\times 1}, K^{-1})$. We show results for 32 replicates and for $p = 10, 20, 40$. MCMC is initialised at the true graph G. The remaining set-up of this simulation study follows Section 3.1.

Also in this simulation scenario, WWA provides more efficient posterior computation than DCBF as shown in Figure 2. Again, this difference increases with the number of nodes with WWA being 3.3 times more efficient than DCBF for $p = 40$ nodes. As in Section 3.1, the MCMC without the informed proposal outperforms WWA for $p = 40$.

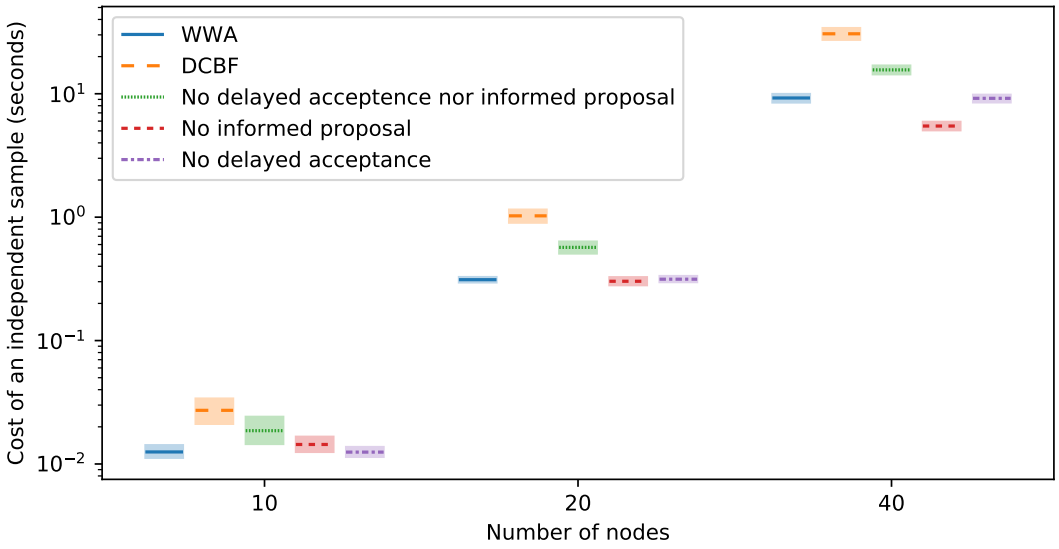

Figure 2: Cost of posterior computations versus the number of nodes for the data simulated from a uniformly sampled graph. The lines represent means over the 32 repetitions for DCBF and WWA as well as different specifications of WWA. The shaded areas are 95% bootstrapped confidence intervals.

4 Application to Gene Expression Data

We consider the real data application from Section 4.2 of Mohammadi and Wit (2015) where a more extensive data description is available. The data consist of gene expressions in B-lymphocyte cells (Stranger et al., 2007) from $n = 60$ individuals. They are quantile-normalised to marginally follow a standard Gaussian distribution, a process also known as rank normalisation. We consider two data sets Y, namely those consisting of the $p = 50$ and $p = 100$ most variable gene expressions. The prior on (G, K) is the same as in Section 3.2, which coincides with an uninformative prior.

For $p = 50$, WWA and DCBF are initialised at a graphical lasso estimate of the graph G (Friedman et al., 2007) and are run for 16,000 iterations of which the last 10,000 are used to estimate the cost of an independent sample. For the data set with $p = 100$, the number of possible graphs is $2^{m_{\max}} = 1.3 \cdot 10^{1,490}$, and the precision matrix is not identifiable in the likelihood since $n < p$. Although this is theoretically not a problem in the Bayesian

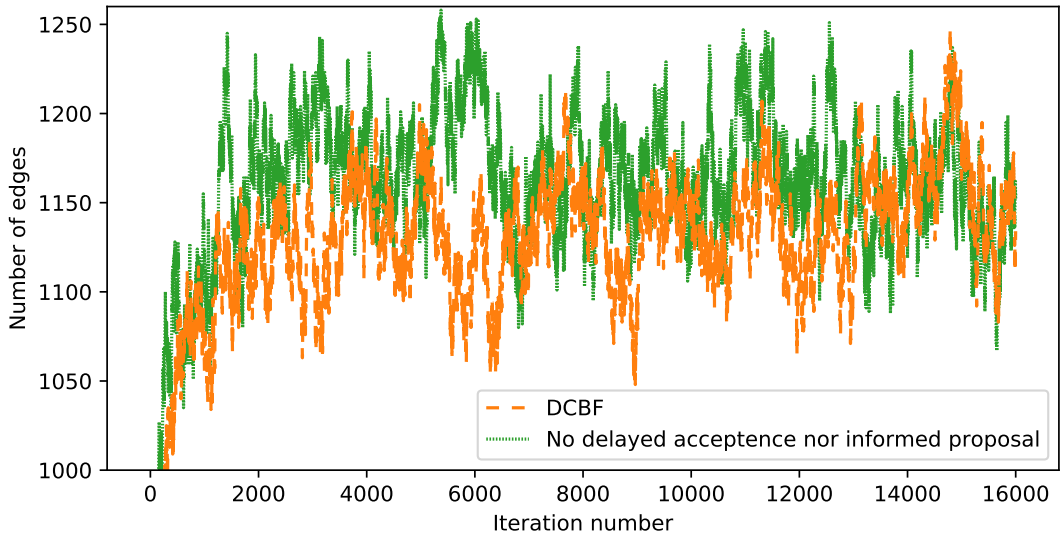

Figure 3: Trace plots for the number of edges in the gene expression application with $p = 100$ nodes.

framework because of prior regularisation, likelihood unidentifiability is known to cause problems for MCMC convergence. Also, the approximation in (7) favours sparse graphs as it is consistently biased in this direction (Mohammadi et al., 2021, page 13), a tendency which is rather strong when the posterior is not concentrated as in the case of $p = 100$. As a result, using the informed proposal or delayed acceptance based on (7) results in bad MCMC convergence and mixing. Therefore, we use Algorithm 2 without the informed proposal nor delayed acceptance because the bias in the approximation would dominate the information deriving from the posterior which is flat in this example.

In terms of speed, the first 6,000 burn-in iterations take 11 minutes for WWA versus 36 minutes for DCBF with $p = 50$, and 4.1 hours for the proposed algorithm versus 9.0 hours for DCBF with $p = 100$. We compute the improved $\widehat{R}$ of Vehtari et al. (2021) as a diagnostic of convergence on the last 10,000 iterations. $\widehat{R}$ converges to one as the number of iterations tends to infinity and $\widehat{R} > 1.01$ indicates lack of convergence. For $p = 50$, when the precision matrix is likelihood identifiable as well, convergence is reached quickly by both algorithms with $\widehat{R} = 1.003$ for WWA and $\widehat{R} = 1.006$ for DCBF. For $p = 100$, the trace

plots in Figure 3 show an increasing trend for the number of edges for DCBF, indicating slower convergence confirmed by $\widehat{R} = 1.068$. This contrasts with the proposed algorithm where $\widehat{R} = 1.001$ and which seems to converge within 2,000 iterations. The introduced methodology yields faster MCMC convergence, both in terms of number of iterations and especially in terms of computing time.

WWA is also superior to DCBF in terms of MCMC mixing. For $p = 100$, we run DCBF for 10,000 iterations initialised at the last iteration of the proposed algorithm to avoid DCBF's convergence issues while assessing MCMC mixing. The cost of an independent sample is 6.2 seconds for WWA versus 24 seconds for DCBF with $p = 50$, and 7.4 minutes for the proposed algorithm versus 14 minutes for DCBF with $p = 100$.

5 Discussion

In this work, we propose WWA, a novel algorithm for GGMs which significantly improves on existing MCMC based on the G-Wishart prior. The main contributions involve delayed acceptance, Gibbs updates for the precision matrix, and the use of parallel computing to inform the proposal. As a result, WWA outperforms the state-of-the-art alternative in terms of MCMC mixing, convergence and computing time.

Here, we discuss possible improvements and extensions to WWA. As the number of nodes increases, normalising the informed proposal takes longer, with the added computational cost potentially outweighing the improvement in MCMC mixing. A potential extension to tune this computation versus mixing trade-off is blocking (Zanella, 2019). It constrains the support of the informed proposal to a subset or 'block' of the neighbourhood of a graph, reducing the computational cost of normalising the informed proposal.

Another improvement to the informed proposal would be a faster or more accurate approximation than (9). The computational bottleneck of the current approximation is calculating the Cholesky decomposition Φ^e. For sparse graphs, the Cholesky decomposition can be sped up via fill-in reducing node reorderings, which increase sparsity in Φ^e, and Cholesky routines optimised for sparse matrices (Rue, 2001; Rue and Held, 2005, Section 2.4.3) as shown by Cheng and Lenkoski (2012).

The MCMC performance is limited by the fact that at most one edge is changed for each accept-reject step. A truly scalable algorithm requires larger moves in the graph space as the number of possible edges in a graph is quadratic in the number of nodes. Such larger moves require sufficiently good proposals for both the graph and the precision matrix K. Tan et al. (2017) take a first step in this direction by changing multiple edges at the same time with an approximate likelihood on the graph resulting from approximating $I_G(\delta, D)$.

In this work, we focus on GGMs because of their popularity in applications and the computational challenges associated with their estimation. Due to the modular nature of MCMC, WWA can also provide a feasible strategy in extended models such as multiple graphs (e.g., Peterson et al., 2015; Tan et al., 2017), Gaussian copulas to accommodate non-Gaussian data (e.g., Dobra and Lenkoski, 2011; Mohammadi and Wit, 2019) or sparse seemingly unrelated regressions (e.g., Wang, 2010; Bhadra and Mallick, 2013).

SUPPLEMENTARY MATERIAL

Appendix: Derivations of Equation (6) and WWA's acceptance probabilities, description of the algorithm from Cheng and Lenkoski (2012), and additional empirical results. (.pdf file)

Code: The scripts that produced the empirical results are available at `https://github.com/willemvandenboom/wwa`. (GitHub repository)

References

Atay-Kayis, A. and H. Massam (2005). A Monte Carlo method for computing the marginal likelihood in nondecomposable Gaussian graphical models. *Biometrika 92*(2), 317–335.

Barker, A. A. (1965). Monte Carlo calculations of the radial distribution functions for a proton-electron plasma. *Australian Journal of Physics 18*(2), 119.

Berry, A., R. Pogorelcnik, and G. Simonet (2010). An introduction to clique minimal separator decomposition. *Algorithms 3*(2), 197–215.

Berry, A., R. Pogorelcnik, and G. Simonet (2014). Organizing the atoms of the clique separator decomposition into an atom tree. *Discrete Applied Mathematics 177*, 1–13.

Bhadra, A. and B. K. Mallick (2013). Joint high-dimensional Bayesian variable and covariance selection with an application to eQTL analysis. *Biometrics 69*(2), 447–457.

Bornn, L. and F. Caron (2011). Bayesian clustering in decomposable graphs. *Bayesian Analysis 6*(4), 829–846.

Carvalho, C. M., H. Massam, and M. West (2007). Simulation of hyper-inverse Wishart distributions in graphical models. *Biometrika 94*(3), 647–659.

Cheng, Y. and A. Lenkoski (2012). Hierarchical Gaussian graphical models: Beyond reversible jump. *Electronic Journal of Statistics 6*, 2309–2331.

Christen, J. A. and C. Fox (2005). Markov chain Monte Carlo using an approximation. *Journal of Computational and Graphical Statistics 14*(4), 795–810.

Dawid, A. P. and S. L. Lauritzen (1993). Hyper Markov laws in the statistical analysis of decomposable graphical models. *The Annals of Statistics 21*(3), 1272–1317.

Dempster, A. P. (1972). Covariance selection. *Biometrics 28*(1), 157–175.

Dobra, A. and A. Lenkoski (2011). Copula Gaussian graphical models and their application to modeling functional disability data. *The Annals of Applied Statistics 5*(2A), 969–993.

Dobra, A., A. Lenkoski, and A. Rodriguez (2011). Bayesian inference for general Gaussian graphical models with application to multivariate lattice data. *Journal of the American Statistical Association 106*(496), 1418–1433.

Fang, Y., Y. Cao, and R. D. Skeel (2020). Quasi-reliable estimates of effective sample size. *IMA Journal of Numerical Analysis* (draa077), 1–18.

Fitch, A. M., M. B. Jones, and H. Massam (2014). The performance of covariance selection methods that consider decomposable models only. *Bayesian Analysis 9*(3), 659–684.

Friedman, J., T. Hastie, and R. Tibshirani (2007). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics 9*(3), 432–441.

Gan, L., N. N. Narisetty, and F. Liang (2018). Bayesian regularization for graphical models with unequal shrinkage. *Journal of the American Statistical Association 114*(527), 1218–1231.

Giudici, P. and P. J. Green (1999). Decomposable graphical Gaussian model determination. *Biometrika 86*(4), 785–801.

Green, P. J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika 82*(4), 711–732.

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

Hinne, M., A. Lenkoski, T. Heskes, and M. van Gerven (2014). Efficient sampling of Gaussian graphical models using conditional Bayes factors. *Stat 3*(1), 326–336.

Jones, B., C. Carvalho, A. Dobra, C. Hans, C. Carter, and M. West (2005). Experiments in stochastic computation for high-dimensional graphical models. *Statistical Science 20*(4), 388–400.

Khare, K., B. Rajaratnam, and A. Saha (2018). Bayesian inference for Gaussian graphical models beyond decomposable graphs. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 80*(4), 727–747.

Lauritzen, S. L. (1996). *Graphical Models*. Oxford Statistical Science Series. The Clarendon Press, Oxford University Press, New York.

Lenkoski, A. (2013). A direct sampler for G-Wishart variates. *Stat 2*(1), 119–128.

Lenkoski, A. and A. Dobra (2011). Computational aspects related to inference in Gaussian graphical models with the G-Wishart prior. *Journal of Computational and Graphical Statistics 20*(1), 140–157.

Letac, G. and H. Massam (2007). Wishart distributions for decomposable graphs. *The Annals of Statistics 35*(3), 1278–1323.

Li, Y., B. A. Craig, and A. Bhadra (2019). The graphical horseshoe estimator for inverse covariance matrices. *Journal of Computational and Graphical Statistics 28*(3), 747–757.

Moghaddam, B., E. Khan, K. P. Murphy, and B. M. Marlin (2009). Accelerating Bayesian structural inference for non-decomposable Gaussian graphical models. In *Advances in Neural Information Processing Systems*, Volume 22. Curran Associates, Inc.

Mohammadi, A. and E. C. Wit (2015). Bayesian structure learning in sparse Gaussian graphical models. *Bayesian Analysis 10*(1), 109–138.

Mohammadi, R., H. Massam, and G. Letac (2021). Accelerating Bayesian structure learning in sparse Gaussian graphical models. *Journal of the American Statistical Association*, 1–14. Advance online publication.

Mohammadi, R. and E. C. Wit (2019). BDgraph: An R package for Bayesian structure learning in graphical models. *Journal of Statistical Software 89*(3), 1–30.

Murray, I., Z. Ghahramani, and D. J. C. MacKay (2006). MCMC for doubly-intractable distributions. In *Proceedings of the Twenty-Second Conference on Uncertainty in Artificial Intelligence*, UAI'06, Arlington, Virginia, USA, pp. 359–366. AUAI Press.

Niu, Y., D. Pati, and B. K. Mallick (2021). Bayesian graph selection consistency under model misspecification. *Bernoulli 27*(1), 636–672.

Peterson, C., F. C. Stingo, and M. Vannucci (2015). Bayesian inference of multiple Gaussian graphical models. *Journal of the American Statistical Association 110*(509), 159–174.

Roverato, A. (2002). Hyper inverse Wishart distribution for non-decomposable graphs and its application to Bayesian inference for Gaussian graphical models. *Scandinavian Journal of Statistics 29*(3), 391–411.

Rue, H. (2001). Fast sampling of Gaussian Markov random fields. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 63*(2), 325–338.

Rue, H. and L. Held (2005). *Gaussian Markov Random Fields*. New York: Chapman and Hall/CRC.

Sagar, K., S. Banerjee, J. Datta, and A. Bhadra (2021). Precision matrix estimation under the horseshoe-like prior-penalty dual. arXiv:2104.10750v1.

Scott, J. G. and C. M. Carvalho (2008). Feature-inclusion stochastic search for Gaussian graphical models. *Journal of Computational and Graphical Statistics 17*(4), 790–808.

Statisticat, LLC. (2020). *LaplacesDemon: Complete Environment for Bayesian Inference*. R package version 16.1.4.

Stranger, B. E., A. C. Nica, M. S. Forrest, A. Dimas, C. P. Bird, C. Beazley, C. E. Ingle, M. Dunning, P. Flicek, D. Koller, S. Montgomery, S. Tavaré, P. Deloukas, and E. T. Dermitzakis (2007). Population genomics of human gene expression. *Nature Genetics 39*(10), 1217–1224.

Tan, L. S. L., A. Jasra, M. De Iorio, and T. M. D. Ebbels (2017). Bayesian inference for multiple Gaussian graphical models with application to metabolic association networks. *The Annals of Applied Statistics 11*(4), 2222–2251.

Uhler, C., A. Lenkoski, and D. Richards (2018, February). Exact formulas for the normalizing constants of Wishart distributions for graphical models. *The Annals of Statistics 46*(1), 90–118.

van den Boom, W., A. Jasra, M. De Iorio, A. Beskos, and J. G. Eriksson (2021). Unbiased approximation of posteriors via coupled particle Markov chain Monte Carlo. arXiv:2103.05176v1.

Vehtari, A., A. Gelman, D. Simpson, B. Carpenter, and P.-C. Bürkner (2021). Rank-normalization, folding, and localization: An improved $\widehat{R}$ for assessing convergence of MCMC (with discussion). *Bayesian Analysis 16*(2), 667–718.

Wang, H. (2010). Sparse seemingly unrelated regression modelling: Applications in finance and econometrics. *Computational Statistics & Data Analysis 54*(11), 2866–2877.

Wang, H. (2015). Scaling it up: Stochastic search structure learning in graphical models. *Bayesian Analysis 10*(2), 351–377.

Wang, H. and C. M. Carvalho (2010). Simulation of hyper-inverse Wishart distributions for non-decomposable graphs. *Electronic Journal of Statistics 4*, 1470–1475.

Wang, H. and S. Z. Li (2012). Efficient Gaussian graphical model determination under G-Wishart prior distributions. *Electronic Journal of Statistics 6*, 168–198.

Zanella, G. (2019). Informed proposals for local MCMC in discrete spaces. *Journal of the American Statistical Association 115*(530), 852–865.

Zhou, S., P. Rütimann, M. Xu, and P. Bühlmann (2011). High-dimensional covariance estimation based on Gaussian graphical models. *The Journal of Machine Learning Research 12*, 2975–3026.

Appendix to "The G-Wishart Weighted Proposal Algorithm: Efficient Posterior Computation for Gaussian Graphical Models" published in the Journal of Computational and Graphical Statistics

Willem van den Boom, Alexandros Beskos and Maria De Iorio

S1 Derivation of Equation (6)

By Equation (20) of Roverato (2002),
$p(\Phi^e \mid G, Y) \propto \exp\left[-\frac{1}{2}\text{tr}\{(\Phi^e)^\top \Phi^e D^{\star,e}\}\right] \prod_{i=1}^p (\Phi^e_{ii})^{\delta^\star + \nu_i^{G^e} - 1}$ where
$\nu_i^{G^e} = |\{j \mid (i,j) \in E^e \text{ and } i < j\}|$. This density factorises as
$p(\Phi^e \mid G, Y) \propto f(\Phi^e) \exp\left\{-\frac{1}{2}(\Phi^e_{pp})^2 D^{\star,e}_{pp}\right\} (\Phi^e_{pp})^{\delta^\star - 1}$ where $f(\Phi^e)$ is constant with respect to Φ^e_{pp}. Thus, $p(\Phi^e_{pp} \mid G, \Phi^e_{p-1,p}, \Phi^e_{-f}, Y) \propto \exp\left\{-\frac{1}{2}(\Phi^e_{pp})^2 D^{\star,e}_{pp}\right\} (\Phi^e_{pp})^{\delta^\star - 1}$. The transformation of variables $\Phi^e_{pp} \to (\Phi^e_{pp})^2$ yields
$p\{(\Phi^e_{pp})^2 \mid G, \Phi^e_{p-1,p}, \Phi^e_{-f}, Y\} \propto \exp\left\{-\frac{1}{2}(\Phi^e_{pp})^2 D^{\star,e}_{pp}\right\} \{(\Phi^e_{pp})^2\}^{\delta^\star/2 - 1}$ from which Equation (6) follows.

S2 Derivation of WWA's Acceptance Probabilities

This section derives in detail the acceptance ratios $\widehat{R}_{\text{DA}}$ and R_{DA} used in Algorithm 2. The informed proposal $Q(\widetilde{G} \mid G, K)$ depends on the value of $\Phi^e_{p-1,p}$. Thus, we need to treat the single edge update as a joint update of $\Phi^e_{p-1,p}$ and G to derive its acceptance probability. The update is transdimensional since $\Phi^e_{p-1,p}$ is fixed or free depending on G per (5). We therefore consider reversible jump MCMC (Green, 1995) for this derivation. Ultimately, the exchange algorithm (Murray et al., 2006), used to avoid intractable normalising constants,

circumvents the need for a reversible jump as exchanging variables does not affect their joint dimensionality, but the intermediate reversible jump acceptance probability is used to derive the approximate acceptance ratio $\widehat{R}_{\mathrm{DA}}$ for use with delayed acceptance MCMC.

S2.1 Reversible Jump

Recall $\Phi^e_{-f} = \Phi^e \setminus \{\Phi^e_{p-1,p}, \Phi^e_{pp}\}$. By construction, $\widetilde{\Phi}^e_{-f} = \Phi^e_{-f}$. WWA uses the joint proposal

$$q(\widetilde{G}, \widetilde{\Phi}^e_{p-1,p} \mid G, K) = Q(\widetilde{G} \mid G, K)\, q(\widetilde{\Phi}^e_{p-1,p} \mid \widetilde{G}, \Phi^e_{-f})$$

Algorithm 2 chooses $q(\widetilde{\Phi}^e_{p-1,p} \mid \widetilde{G}, \Phi^e_{-f}) = p(\widetilde{\Phi}^e_{p-1,p} \mid \widetilde{G}, \widetilde{\Phi}^e_{-f}, Y)$ given by (5) which is thus a Dirac delta function if $e \notin \widetilde{G}$. Therefore, the previous display reduces to

$$q(\widetilde{G}, \widetilde{\Phi}^e_{p-1,p} \mid G, K) = \begin{cases} Q(\widetilde{G} \mid G, K), & \widetilde{\Phi}^e_{p-1,p} = \phi^e \text{ and } e \notin \widetilde{E}, \\ p(\widetilde{\Phi}^e_{p-1,p} \mid \widetilde{G}, \widetilde{\Phi}^e_{-f}, Y)\, Q(\widetilde{G} \mid G, K), & e \in \widetilde{E}, \end{cases} \tag{S1}$$

where ϕ^e equals the right-hand side of (5b), $Q(\widetilde{G} \mid G, K)$ is a probability mass function and $p(\widetilde{\Phi}^e_{p-1,p} \mid \widetilde{G}, \widetilde{\Phi}^e_{-f}, Y)$ is a density with respect to Lebesgue measure.

The target distribution is

$$p(G, \Phi^e_{p-1,p} \mid \Phi^e_{-f}, Y) = \begin{cases} p(G \mid \Phi^e_{-f}, Y), & \Phi^e_{p-1,p} = \phi^e \text{ and } e \notin E, \\ p(\Phi^e_{p-1,p} \mid G, \Phi^e_{-f}, Y)\, p(G \mid \Phi^e_{-f}, Y), & e \in E. \end{cases} \tag{S2}$$

This target, including $\Phi^e_{p-1,p}$ and Φ^e_{-f}, depends on e and thus on the proposed $\widetilde{G}$. Such dependence on the proposed graph does not affect the MCMC's validity or the acceptance probability as the resulting Markov kernel can be interpreted as a mixture of kernels which each have $p(G, K \mid Y)$ as invariant distribution (Tierney, 1994, Section 2.4).

Reversible jump involves a dimension-matching map. In our setting, we consider the mapping $(\Phi^e_{p-1,p},\, \widetilde{\Phi}^e_{p-1,p}) \to (\widetilde{\Phi}^e_{p-1,p},\, \Phi^e_{p-1,p})$ which has matched dimensions since both sides contain exactly one free element. Moreover, the absolute value of the determinant of the Jacobian of this map equals one. Therefore, the acceptance probability follows as $1 \wedge R_{\mathrm{RJ}}$ where (Green, 1995, Equation (7))

$$R_{\mathrm{RJ}} = \frac{p(\widetilde{G}, \widetilde{\Phi}^e_{p-1,p} \mid \widetilde{\Phi}^e_{-f}, Y)\, q(G, \Phi^e_{p-1,p} \mid \widetilde{G}, \widetilde{K})}{p(G, \Phi^e_{p-1,p} \mid \Phi^e_{-f}, Y)\, q(\widetilde{G}, \widetilde{\Phi}^e_{p-1,p} \mid G, K)}. \tag{S3}$$

Bayes' rule and $\widetilde{\Phi}^e_{-f} = \Phi^e_{-f}$ yields

$$\frac{p(\widetilde{G} \mid \widetilde{\Phi}^e_{-f}, Y)}{p(G \mid \Phi^e_{-f}, Y)} = \frac{p(Y, \widetilde{G}, \widetilde{\Phi}^e_{-f})}{p(Y, G, \Phi^e_{-f})} = \frac{p(Y, \widetilde{\Phi}^e_{-f} \mid \widetilde{G})\, p(\widetilde{G})}{p(Y, \Phi^e_{-f} \mid G)\, p(G)}.$$

Combining the last two displays, and inserting Equations (S1) and (S2) provide

$$\begin{aligned} R_{\mathrm{RJ}} &= \frac{p(\widetilde{G}, \Phi^e_{-f} \mid Y)\, Q(G \mid \widetilde{G}, \widetilde{K})}{p(G, \Phi^e_{-f} \mid Y)\, Q(\widetilde{G} \mid G, K)} = \frac{p(Y, \Phi^e_{-f} \mid \widetilde{G})\, p(\widetilde{G})\, Q(G \mid \widetilde{G}, \widetilde{K})}{p(Y, \Phi^e_{-f} \mid G)\, p(G)\, Q(\widetilde{G} \mid G, K)} \\ &= N(\Phi^e_{-f}, D^{\star,e})^{|\widetilde{E}|-|E|}\, \frac{p(\widetilde{G})\, I_G(\delta, D)\, Q(G \mid \widetilde{G}, \widetilde{K})}{p(G)\, I_{\widetilde{G}}(\delta, D)\, Q(\widetilde{G} \mid G, K)}, \end{aligned} \tag{S4}$$

where the last equality follows from Equation (1).

S2.2 Exchange Algorithm

Equation (S4) cannot be used directly as it contains the intractable $I_G(\delta, D)$. Using the exchange algorithm avoids evaluating this normalising constant as in Wang and Li (2012, Section 5.2) and Cheng and Lenkoski (2012, Section 2.3) by considering an augmented target distribution.

Our augmented target is a joint distribution on $(G, K, \widetilde{G}, \widetilde{\Phi}^{0,e}_{-f})$ that mimics Algorithm 2. Specifically, (G, K) follows the unaugmented target distribution $p(G, K \mid Y)$, the distribution of $\widetilde{G} \mid G, K$ is the informed proposal $Q(\widetilde{G} \mid G, K)$, and the distribution of $\widetilde{\Phi}^{0,e}_{-f} \mid G, K, \widetilde{G}$ follows from the node reordering resulting from the pair $(G, \widetilde{G})$, $\widetilde{K}^{0,e} \mid \widetilde{G} \sim \mathcal{W}_{\widetilde{G}^e}(\delta, D^e)$ and the definition $\widetilde{\Phi}^{0,e}_{-f} = \widetilde{\Phi}^{0,e} \setminus \{\widetilde{\Phi}^{0,e}_{p-1,p}, \widetilde{\Phi}^{0,e}_{pp}\}$. The augmented target distribution can thus be written as

$$\pi(G, K, \widetilde{G}, \widetilde{\Phi}^{0,e}_{-f}) = p(G, K \mid Y)\, Q(\widetilde{G} \mid G, K)\, p(\widetilde{\Phi}^{0,e}_{-f} \mid \widetilde{G}). \tag{S5}$$

Figure S1 clarifies the structure of the augmented target.

The proposal is to exchange G with $\widetilde{G}$ and update K by a value sampled from its proposal $q(\widetilde{K} \mid G, K, \widetilde{G})$, which is defined by sampling $\widetilde{\Phi}^e_{p-1,p}$ according to (5) while leaving the other elements of $\widetilde{\Phi}^e$ equal to the corresponding elements in Φ^e. The resulting Metropolis-Hastings step has the posterior of interest $p(G, K \mid Y)$ as target marginally on (G, K). The remainder

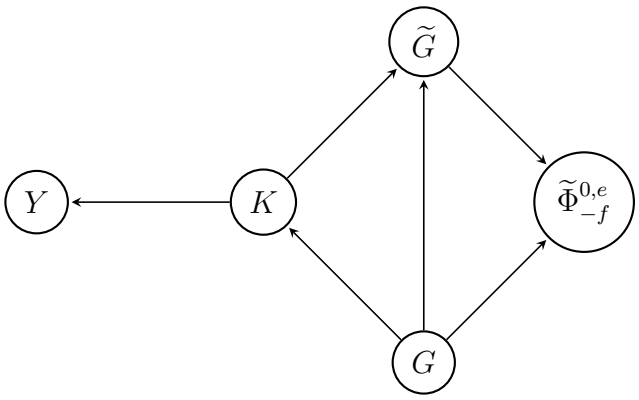

Figure S1: A directed acyclic graph representing the conditional dependence structure of the augmented target distribution (S5) used with the exchange algorithm.

of this section evaluates the corresponding acceptance probability $1 \wedge R^{\text{informed}}_{\text{exchange}}$ with

$$\begin{aligned} R^{\text{informed}}_{\text{exchange}} &= \frac{\pi(\widetilde{G}, \widetilde{K}, G, \widetilde{\Phi}^{0,e}_{-f})\, q(K \mid \widetilde{G}, \widetilde{K}, G)}{\pi(G, K, \widetilde{G}, \widetilde{\Phi}^{0,e}_{-f})\, q(\widetilde{K} \mid G, K, \widetilde{G})} \\ &= \frac{p(\widetilde{G}, \widetilde{K} \mid Y)\, Q(G \mid \widetilde{G}, \widetilde{K})\, p(\widetilde{\Phi}^{0,e}_{-f} \mid G)\, q(K \mid \widetilde{G}, \widetilde{K}, G)}{p(G, K \mid Y)\, Q(\widetilde{G} \mid G, K)\, p(\widetilde{\Phi}^{0,e}_{-f} \mid \widetilde{G})\, q(\widetilde{K} \mid G, K, \widetilde{G})} \end{aligned} \tag{S6}$$

where the last equality follows from (S5).

The distribution $q(\widetilde{K} \mid G, K, \widetilde{G})$ is defined through $\widetilde{\Phi}^{e}_{p-1,p}$ being distributed according to (5). The implied distribution on $\widetilde{K}$ follows from the transformation $\widetilde{\Phi}^{e} \to \widetilde{K}$ such that

$$q(\widetilde{K} \mid G, K, \widetilde{G}) = J(\widetilde{\Phi}^{e} \to \widetilde{K})\, q(\widetilde{\Phi}^{e}_{p-1,p} \mid \widetilde{G}, \Phi^{e}_{-f})$$

where $J(\widetilde{\Phi}^{e} \to \widetilde{K})$ is the Jacobian term resulting from the transformation of variables. Similarly,

$$p(G, K \mid Y) = J(\Phi^{e} \to K)\, p(G, \Phi^{e} \mid Y).$$

These Jacobian terms cancel in $R^{\text{informed}}_{\text{exchange}}$. Specifically, inserting the previous two displays into (S6) yields

$$R^{\text{informed}}_{\text{exchange}} = \frac{p(\widetilde{G}, \widetilde{\Phi}^{e} \mid Y)\, Q(G \mid \widetilde{G}, \widetilde{K})\, p(\widetilde{\Phi}^{0,e}_{-f} \mid G)\, q(\Phi^{e}_{p-1,p} \mid G, \widetilde{\Phi}^{e}_{-f})}{p(G, \Phi^{e} \mid Y)\, Q(\widetilde{G} \mid G, K)\, p(\widetilde{\Phi}^{0,e}_{-f} \mid \widetilde{G})\, q(\widetilde{\Phi}^{e}_{p-1,p} \mid \widetilde{G}, \Phi^{e}_{-f})}.$$

Consider the factorisation $p(G, \Phi^{e} \mid Y) = p(G, \Phi^{e}_{p-1,p} \mid \Phi^{e}_{pp}, \Phi^{e}_{-f}, Y)\, p(\Phi^{e}_{pp}, \Phi^{e}_{-f} \mid Y)$. The proposal is such that $\widetilde{\Phi}^{e}_{pp} = \Phi^{e}_{pp}$ and $\widetilde{\Phi}^{e}_{-f} = \Phi^{e}_{-f}$. We also make use of the equality

$p(G, \Phi^e_{p-1,p} \mid \Phi^e_{pp}, \Phi^e_{-f}, Y) = p(G, \Phi^e_{p-1,p} \mid \Phi^e_{-f}, Y)$ since Φ^e_{pp} is independent of $(G, \Phi^e_{-f}, \Phi^e_{p-1,p})$ by (6). Thus,

$$\frac{p(\widetilde{G}, \widetilde{\Phi}^e \mid Y)}{p(G, \Phi^e \mid Y)} = \frac{p(\widetilde{G}, \widetilde{\Phi}^e_{p-1,p} \mid \widetilde{\Phi}^e_{pp}, \widetilde{\Phi}^e_{-f}, Y)\, p(\widetilde{\Phi}^e_{pp}, \widetilde{\Phi}^e_{-f} \mid Y)}{p(G, \Phi^e_{p-1,p} \mid \Phi^e_{pp}, \Phi^e_{-f}, Y)\, p(\Phi^e_{pp}, \Phi^e_{-f} \mid Y)} = \frac{p(\widetilde{G}, \widetilde{\Phi}^e_{p-1,p} \mid \widetilde{\Phi}^e_{-f}, Y)}{p(G, \Phi^e_{p-1,p} \mid \Phi^e_{-f}, Y)}.$$

Furthermore, $Q(\widetilde{G} \mid G, K)\, q(\widetilde{\Phi}^e_{p-1,p} \mid \widetilde{G}, \Phi^e_{-f}) = q(\widetilde{G}, \widetilde{\Phi}^e_{p-1,p} \mid G, K)$ where $q(\widetilde{G}, \widetilde{\Phi}^e_{p-1,p} \mid G, K)$ is given by (S1). Combined with the previous two displays, we obtain

$$\begin{aligned} R^{\text{informed}}_{\text{exchange}} &= \frac{p(\widetilde{G}, \widetilde{\Phi}^e_{p-1,p} \mid \widetilde{\Phi}^e_{-f}, Y)\, q(G, \Phi^e_{p-1,p} \mid \widetilde{G}, \widetilde{K})\, p(\widetilde{\Phi}^{0,e}_{-f} \mid G)}{p(G, \Phi^e_{p-1,p} \mid \Phi^e_{-f}, Y)\, q(\widetilde{G}, \widetilde{\Phi}^e_{p-1,p} \mid G, K)\, p(\widetilde{\Phi}^{0,e}_{-f} \mid \widetilde{G})} \\ &= R_{\text{RJ}}\, \frac{p(\widetilde{\Phi}^{0,e}_{-f} \mid G)}{p(\widetilde{\Phi}^{0,e}_{-f} \mid \widetilde{G})} \end{aligned}$$

where the last equality follows from (S3). Inserting (S4) and recalling (4) yields

$$R^{\text{informed}}_{\text{exchange}} = R_{\text{exchange}}\, \frac{Q(G \mid \widetilde{G}, \widetilde{K})}{Q(\widetilde{G} \mid G, K)}.$$

The acceptance probabilities in Algorithm 2 follow now from the delayed acceptance method in Algorithm 1 of Christen and Fox (2005). Specifically, $\widehat{R}_{\text{DA}}$ equals R_{RJ} with the ratio of normalising constants in (S4) approximated by (7). Then, the second accept-reject step uses R_{exchange} from Equation (4) in place of the ratio of target distributions like $R^{\text{informed}}_{\text{exchange}}$ does.

S3 Effect of $\widehat{I_G/I_{\widetilde{G}}}$ Approximation on Convergence

WWA converges to $p(G \mid Y)$ for any approximation $\widehat{I_G/I_{\widetilde{G}}} > 0$ as remarked in Section 2.5. Here, we investigate in a simulation study the effect of the choice of $\widehat{I_G/I_{\widetilde{G}}}$ on the speed of convergence for WWA. The data are simulated as in Section 3.1 for $p = 40$. Then, we run WWA for 1,000 iterations both for $\widehat{I_G/I_{\widetilde{G}}}$ as given by (7) following the suggestion of Mohammadi et al. (2021) and for $\widehat{I_G/I_{\widetilde{G}}} = 1$. The MCMC chains are initialised at the empty graph.

Figure S2 visualises the resulting MCMC chains. The chain with the approximation from Mohammadi et al. (2021) converges away from the initialisation with zero edges slightly faster than $\widehat{I_G/I_{\widetilde{G}}} = 1$.

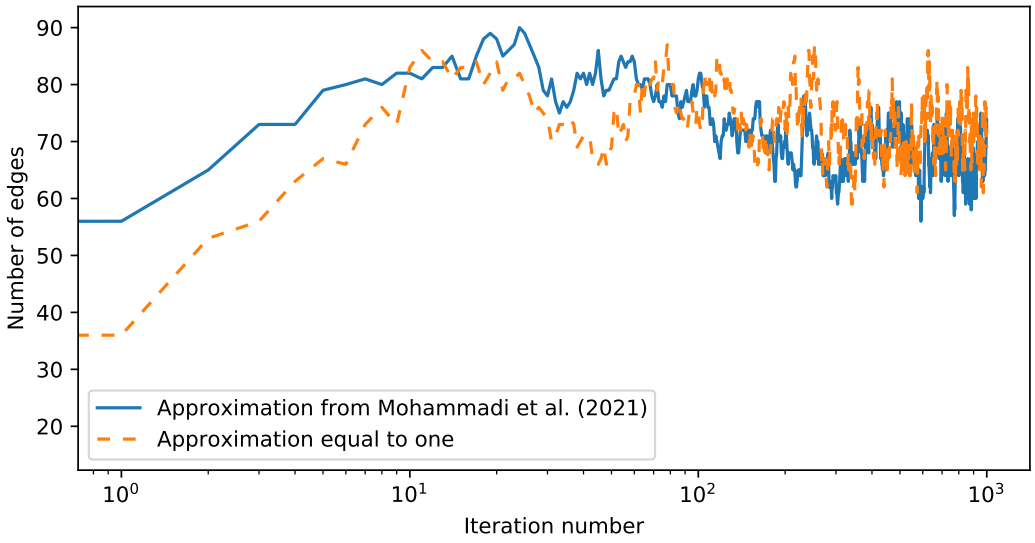

Figure S2: Trace plots for the number of edges from WWA with the approximation $\widehat{I_G/I_{\widetilde{G}}}$ as in Mohammadi et al. (2021) and with $\widehat{I_G/I_{\widetilde{G}}} = 1$ run on simulated data.

S4 Graph Decomposition and the G-Wishart Sampler

We investigate the effect of using graph decomposition on the computational cost of sampling from the G-Wishart distribution. The two G-Wishart samplers considered are the one described in Section 2.5 with graph decomposition and the one proposed by Lenkoski (2013) without graph decomposition. We record the computing time to sample from $\mathcal{W}_G(\delta, D)$ for different G with $\delta = 3$ and D as in Section 2.3 of Dobra et al. (2011). That is, $D = I_p + 100A^{-1}$ with A given by $A_{ii} = 1$ for $i = 1, \ldots, p$, $A_{ij} = 0.5$ for $|i - j| = 1$, $A_{1p} = A_{p1} = 0.4$ and all other elements of A being equal to zero. The three types of graphs G that we consider are (i) a decomposable graph created by adding $p - 3$ chords to a cycle, specifically, $E = \{(i, i+1) \mid i = 1, \ldots, p-1\} \cup \{(1, p)\} \cup \{(1, i) \mid i = 3, \ldots, p-1\}$; (ii) random graphs with independent edges with edge inclusion probability $\rho = 0.5$ and (iii) $\rho = 2/(p-1)$ such that the expected number of edges equals p. We show results for 100 replicates and for $p = 10, 20, 40, 80$.

Figure S3 shows that the G-Wishart sampler with graph decomposition can take considerably less computing time than the sampler without decomposition if the graph's

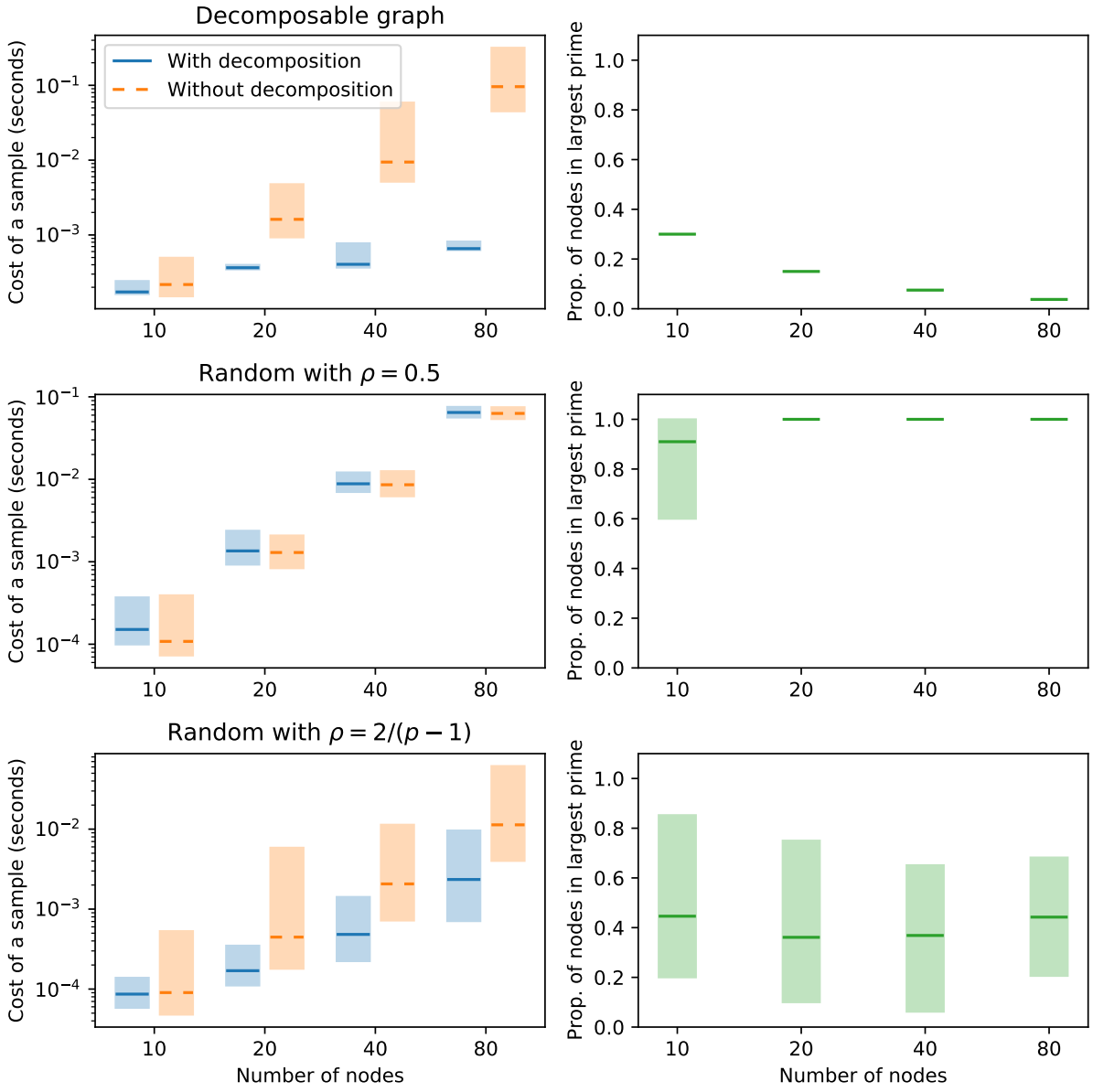

Figure S3: Cost of sampling from the G-Wishart distribution for the sampler with and without graph decomposition (left), and the proportion of nodes part of the graph's largest prime component (right) versus the number of nodes for the three types of graphs described in Section S4. The lines represent medians over 100 repetitions. The boundaries of shaded areas correspond to the 2.5% and 97.5% quantiles.

largest prime component has substantially fewer nodes than the graph itself. When no effective decomposition is available, the additional cost incurred by attempting it is negligible in these simulations. Graphs sampled uniformly from the graph space ($\rho = 0.5$) consist of a single prime component with high probability when the number of nodes is greater than or equal to 20. Thus, the use of graph decomposition does not yield a faster G-Wishart sampler for such graphs.

S5 Algorithm from Cheng and Lenkoski (2012)

We describe the CL algorithm of Cheng and Lenkoski (2012) in Algorithm S1 to aid the discussion in Section 2.6. To highlight connections with WWA, we use the delayed acceptance framework to describe the algorithm even though Cheng and Lenkoski (2012) do not use such terminology.

The first accept-reject step in Step 1c of Algorithm S1 uses $\widehat{R}_{\mathrm{CL}}$ as odds instead of acceptance ratio. This corresponds to Barker's algorithm (Barker, 1965): write $\widehat{R}_{\mathrm{CL}} = \widetilde{z}/z$ for some positive numbers z and $\widetilde{z}$ that correspond with the current and proposed state, respectively. Denote the acceptance probability by α. Accepting with odds $\widehat{R}_{\mathrm{CL}}$ means $\widehat{R}_{\mathrm{CL}} = \alpha/(1-\alpha)$. Then, $\alpha = \widetilde{z}/(z+\widetilde{z})$ which is Barker's acceptance probability. Using $\widehat{R}_{\mathrm{CL}}$ as acceptance ratio would yield the Metropolis-Hastings acceptance probability $\alpha = 1 \wedge \widehat{R}_{\mathrm{CL}}$.

S6 Quality of Posterior Approximation

Section 3 in the main text focuses on MCMC efficiency. Here, we additionally assess the performance of WWA in terms of estimation of posterior summaries such as posterior edge inclusion probabilities and accuracy of precision matrix estimation. While comparisons of effective sample size might be hard to interpret across MCMC algorithms that have different target distributions, such issues do not arise for posterior summaries. Therefore, we additionally compare with the CL algorithm detailed in Algorithm S1 and the methodology of Mohammadi and Wit (2019) as implemented in the R package `BDgraph` which do not exactly have $p(G \mid Y)$ as invariant distribution. For instance, Step 1(c)i of Algorithm S1 does

Algorithm S1 (Cheng and Lenkoski, 2012) A Single MCMC Step of the CL Algorithm.

Input: Precision matrix K and graph G.

Output: MCMC update for (G, K) such that the invariant distribution is approximately the posterior $p(G, K \mid Y)$.

1. For each edge $e \in \{(i,j) \mid 1 \leq i < j \leq p\}$, do the following:

   (a) Let $\widetilde{G} = (V, \widetilde{E})$ where $\widetilde{E} = E \cup \{e\}$ if $e \notin E$ and $\widetilde{E} = E \setminus \{e\}$ otherwise.

   (b) Reorder the nodes in G and $\widetilde{G}$ so that e connects nodes $p-1$ and p. Rearrange D, $D^\star$ and K accordingly. Denote the resulting quantities by a superscript e.

   (c) Denote the upper triangular Cholesky decomposition of K^e by Φ^e. 'Promote' $\widetilde{G}$ to be considered for delayed acceptance with odds $\widehat{R}_{\text{CL}}$ where

   $$\widehat{R}_{\text{CL}} = \frac{p(\widetilde{G})}{p(G)} N(\Phi^e_{-f}, D^{\star,e})^{|\widetilde{E}|-|E|}$$

   with $N(\Phi^e_{-f}, D^{\star,e})$ given by Equation (2). If $\widetilde{G}$ is promoted:

      i. Generate a $\widetilde{K}^{0,e}$ by running the maximum clique block Gibbs sampler (Wang and Li, 2012) initialised at K^e and with $\mathcal{W}_{\widetilde{G}^e}(\delta, D^e)$ as stationary distribution.

      ii. Compute the upper triangular Cholesky decomposition $\widetilde{\Phi}^{0,e}$ of $\widetilde{K}^{0,e}$.

      iii. Set $G = \widetilde{G}$ and $K = \widetilde{K}$ w.p. $1 \wedge N(\widetilde{\Phi}^{0,e}_{-f}, D^e)^{|E|-|\widetilde{E}|}$.

   (d) Update K by resampling $\Phi^e_{p-1,p}$ and Φ^e_{pp} according to G^e.

2. Resample K by running the maximum clique block Gibbs sampler (Wang and Li, 2012) with $\mathcal{W}_G(\delta^\star, D^\star)$ as stationary distribution.

Table S1: Posterior edge inclusion probabilities for the Iris virginica data as estimated by various MCMC methods.

Edge	SL–SW	SL–PL	SL–PW	SW–PL	SW–PW	PL–PW
WWA	0.822	1.000	0.406	0.499	0.987	0.533
DCBF	0.822	1.000	0.405	0.501	0.987	0.533
No informed proposal	0.820	1.000	0.405	0.500	0.987	0.529
CL algorithm	0.820	1.000	0.402	0.498	0.987	0.529
`BDgraph`	1.000	1.000	1.000	1.000	1.000	1.000

Abbreviations: SL, sepal length; SW, sepal width; PL, petal length; PW, petal width

not sample from $\mathcal{W}_{\widetilde{G}^e}(\delta, D^e)$ directly, resulting in an approximate version of the exchange algorithm, namely the double Metropolis-Hastings sampler (Liang, 2010), which does not preserve the invariant distribution of the Markov chain. As with DCBF in Section 3, we slightly modify the CL algorithm by executing the steps in Algorithm S1 for p edges drawn uniformly at random with replacement instead of all $m_{\max}$ possible edges for a fairer comparison with WWA.

S6.1 Fisher's Iris Virginica Data

We consider the example in Roverato (2002), Atay-Kayis and Massam (2005) and Lenkoski (2013). The data Y consist of $p = 4$ demeaned measurements in centimetres of sepal length, sepal width, petal length and petal width of $n = 50$ Iris virginica plants. The prior set-up is the same as in Section 3.2. We run WWA, DCBF, WWA with delayed acceptance but without the informed proposal, the CL algorithm and `BDgraph`. The Markov chains are initialised at the empty graph which is the default in `BDgraph`. We run the chains for 10^3 burn-in iterations followed by 10^6 recorded iterations.

Table S1 presents the resulting estimates of the posterior edge inclusion probabilities. Comparing these values with those in Table 1 of Lenkoski (2013) shows that all methods except for `BDgraph` provide accurate inclusion probabilities. The approximate nature of the CL algorithm does not substantially affect these estimates, though it seems to slightly

underestimate the inclusion probabilities for the edges between sepal length and petal width, and between sepal width and petal length. The CL algorithm has the lowest cost of an independent sample, as defined in Section 3, at 0.31 milliseconds compared to 0.52, 0.49 and 0.72 milliseconds for WWA, DCBF and WWA without the informed proposal, respectively. We do not compute cost of an independent sample for `BDgraph` as it performs continuous-time MCMC such that we cannot derive the integrated autocorrelation time in the same fashion as for the other algorithms.

S6.2 Cycle Graphs

Here, we consider the same prior set-up and data generation as in Section 3.1 for $p = 10$ and $p = 100$ nodes. We apply the same MCMC set-ups as in Section S6.1 for $p = 10$. For $p = 100$, we use 10^4 instead of 10^3 burn-in iterations. Because of their high computational cost per iteration, we do not consider WWA with the informed proposal and DCBF, and run WWA without the informed proposal and `BDgraph` for 10^5 instead of 10^6 recorded iterations for $p = 100$.

Table S2 presents the results for the following performance measures:

Max. diff. Maximum absolute difference from WWA of the posterior edge inclusion probabilities.

MSD Mean squared difference from WWA of the posterior edge inclusion probabilities.

Min. inc. prob. Minimum posterior inclusion probability for the edges included in the true underlying cycle graph.

Max. inc. prob. Maximum posterior inclusion probability for the edges excluded in the true underlying cycle graph.

$\mathbf{p}(\mathbf{G}_{\text{true}} \mid \mathbf{Y})$ Posterior probability of the true underlying cycle graph.

KL $\hat{\mathbf{K}}$ Kullback-Leibler divergence $\text{KL} = 0.5 \cdot \{\text{tr}(K_{\text{true}}^{-1}\hat{K}) - p - \log(|\hat{K}|/|K_{\text{true}}|)\}$ from the true data-generating distribution $\mathcal{N}(0_{p\times 1},\, K_{\text{true}}^{-1})$ to $\mathcal{N}(0_{p\times 1},\, \hat{K}^{-1})$ where $\hat{K}$ is the posterior mean of the precision matrix (Mohammadi and Wit, 2015).

Table S2: Performance measures with the set-up from Section 6.2 of Wang and Li (2012) for various MCMC methods.

Measure	WWA	DCBF	No informed prop.	CL algorithm	`BDgraph`
			$p = 10$ nodes		
Max. diff.	—	$6.0 \cdot 10^{-3}$	$7.7 \cdot 10^{-3}$	$6.7 \cdot 10^{-3}$	$6.3 \cdot 10^{-2}$
MSD	—	$3.5 \cdot 10^{-6}$	$6.4 \cdot 10^{-6}$	$6.1 \cdot 10^{-6}$	$3.4 \cdot 10^{-4}$
Min. inc. prob.	0.133	0.134	0.132	0.137	0.174
Max. inc. prob.	0.997	0.997	0.997	0.997	0.994
$p(G_{\text{true}} \mid Y)$	$2.1 \cdot 10^{-5}$	$8.0 \cdot 10^{-6}$	$3.6 \cdot 10^{-5}$	$2.3 \cdot 10^{-5}$	$2.8 \cdot 10^{-5}$
KL $\hat{K}$	8.2	8.2	8.2	8.2	8.4
Frobenius $\hat{K}$	63	63	63	63	63
CIS (millisec.)	4.0	18	4.8	2.9	—
			$p = 100$ nodes		
Max. diff.			—	0.24	0.24
MSD			—	$2.2 \cdot 10^{-4}$	$2.1 \cdot 10^{-4}$
Min. inc. prob.			0.761	0.755	0.834
Max. inc. prob.			0.239	0.245	0.168
$p(G_{\text{true}} \mid Y)$			$5.0 \cdot 10^{-3}$	$3.8 \cdot 10^{-3}$	$8.4 \cdot 10^{-3}$
KL $\hat{K}$			91	92	93
Frobenius $\hat{K}$			2480	2480	2480
CIS (seconds)			33	33	—

Frobenius $\hat{\mathbf{K}}$ Frobenius norm of the matrix $\hat{K} - K_{\text{true}}$.

CIS Cost of an independent sample in seconds. For the CL algorithm with $p = 100$, CIS is computed from the last 10^5 iterations for fair comparison.

The measures on the posterior inclusion probabilities do not show major differences between the MCMCs considered. The 'Min. inc. prob.', 'Max. inc. prob.' and, for $p = 10$, 'Max. diff.' of `BDgraph` differ from the other algorithms though not as extremely as the edge inclusion probabilities in Table S1. This suggests that the approximations in `BDgraph` have an impact on the estimates of the posterior edge inclusion probabilities.

The median probability graph does not recover the true underlying cycle graph for $p = 10$, suggesting that $n = 15$ observations do not provide enough information. This contrasts with $p = 100$ where $n = 150$, which recovers the cycle graph with the median probability graph as based on 'Min. inc. prob.' and 'Max. inc. prob.' in line with the results reported in Section 6.2 of Wang and Li (2012) for the same simulation set-up. The estimates of the posterior probabilities of the true graph $p(G_{\text{true}} \mid Y)$ are small which is unsurprising given the uncertainty in the posterior and the size of the graph space.

The measures of accuracy of the posterior mean $\hat{K}$ for estimation of the precision matrix K_{true} do not vary notably across algorithms. Note that Table 5 of Mohammadi and Wit (2015) reports substantially smaller 'KL $\hat{K}$' for the posterior mean of the Bayesian model that we consider than for the graphical lasso estimate of the precision matrix. Table S2 suggests that such superior performance for the precision matrix is also achieved when using approximate MCMC such as provided by `BDgraph`.

Finally, we remark that the MCMC efficiency of WWA as measured by CIS is competitive with the CL algorithm. That is, WWA performs exact MCMC with a computational efficiency comparable to the CL algorithm which does not have the correct posterior as invariant distribution.

References

Atay-Kayis, A. and H. Massam (2005). A Monte Carlo method for computing the marginal likelihood in nondecomposable Gaussian graphical models. *Biometrika 92*(2), 317–335.

Barker, A. A. (1965). Monte Carlo calculations of the radial distribution functions for a proton-electron plasma. *Australian Journal of Physics 18*(2), 119.

Cheng, Y. and A. Lenkoski (2012). Hierarchical Gaussian graphical models: Beyond reversible jump. *Electronic Journal of Statistics 6*, 2309–2331.

Christen, J. A. and C. Fox (2005). Markov chain Monte Carlo using an approximation. *Journal of Computational and Graphical Statistics 14*(4), 795–810.

Dobra, A., A. Lenkoski, and A. Rodriguez (2011). Bayesian inference for general Gaussian graphical models with application to multivariate lattice data. *Journal of the American Statistical Association 106*(496), 1418–1433.

Green, P. J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika 82*(4), 711–732.

Lenkoski, A. (2013). A direct sampler for G-Wishart variates. *Stat 2*(1), 119–128.

Liang, F. (2010). A double Metropolis-Hastings sampler for spatial models with intractable normalizing constants. *Journal of Statistical Computation and Simulation 80*(9), 1007–1022.

Mohammadi, A. and E. C. Wit (2015). Bayesian structure learning in sparse Gaussian graphical models. *Bayesian Analysis 10*(1), 109–138.

Mohammadi, R., H. Massam, and G. Letac (2021). Accelerating Bayesian structure learning in sparse Gaussian graphical models. *Journal of the American Statistical Association*, 1–14. Advance online publication.

Mohammadi, R. and E. C. Wit (2019). BDgraph: An R package for Bayesian structure learning in graphical models. *Journal of Statistical Software 89*(3), 1–30.

Murray, I., Z. Ghahramani, and D. J. C. MacKay (2006). MCMC for doubly-intractable distributions. In *Proceedings of the Twenty-Second Conference on Uncertainty in Artificial Intelligence*, UAI'06, Arlington, Virginia, USA, pp. 359–366. AUAI Press.

Roverato, A. (2002). Hyper inverse Wishart distribution for non-decomposable graphs and its application to Bayesian inference for Gaussian graphical models. *Scandinavian Journal of Statistics 29*(3), 391–411.

Tierney, L. (1994). Markov chains for exploring posterior distributions. *The Annals of Statistics 22*(4), 1701–1762.

Wang, H. and S. Z. Li (2012). Efficient Gaussian graphical model determination under G-Wishart prior distributions. *Electronic Journal of Statistics 6*, 168–198.