Electronic Journal of Statistics
Vol. 13 (2019) 2865–2897
ISSN: 1935-7524
https://doi.org/10.1214/19-EJS1585

Bayesian learning of weakly structural Markov graph laws using sequential Monte Carlo methods

Jimmy Olsson[†], **Tatjana Pavlenko**[*] **and Felix L. Rios**

Department of Mathematics
The Royal Institute of Technology, Stockholm, Sweden
e-mail: jimmyol@math.kth.se*;* pavlenko@math.kth.se*;* flrios@math.kth.se

Abstract: We present a sequential sampling methodology for weakly structural Markov laws, arising naturally in a Bayesian structure learning context for decomposable graphical models. As a key component of our suggested approach, we show that the problem of graph estimation, which in general lacks natural sequential interpretation, can be recast into a sequential setting by proposing a recursive Feynman-Kac model that generates a flow of junction tree distributions over a space of increasing dimensions. We focus on particle McMC methods to provide samples on this space, in particular on particle Gibbs (PG), as it allows for generating McMC chains with global moves on an underlying space of decomposable graphs. To further improve the PG mixing properties, we incorporate a systematic refreshment step implemented through direct sampling from a backward kernel. The theoretical properties of the algorithm are investigated, showing that the proposed refreshment step improves the performance in terms of asymptotic variance of the estimated distribution. The suggested sampling methodology is illustrated through a collection of numerical examples demonstrating high accuracy in Bayesian graph structure learning in both discrete and continuous graphical models.

MSC 2010 subject classifications: Primary 62L20, 62L20; secondary 62-09.
Keywords and phrases: Structure learning, sequential sampling, decomposable graphical models, particle Gibbs.

Received June 2018.

Contents

[*] TP gratefully acknowledges support by the Swedish Research Council, Grant C0595201.
[†] JO gratefully acknowledges support by the Swedish Research Council, Grant 2018-05230.

1. Introduction

Understanding the underlying dependence structure of a multivariate distribution is becoming increasingly important in modern applications when analysing complex data. These dependencies are conveniently represented by a *graphical model* (GM) in which the set of nodes represents feature variables in the model and the set of edges encodes the dependence structure. A specific family of undirected graphical models extensively studied in the literature are those which are Markov with respect to decomposable graphs, usually referred to as *decomposable graphical models* (DGMs), to which we restrict our attention in the present paper. For these models, joint densities factorise into products of densities over certain subsets of nodes described by *cliques* and *separators*. This makes such models attractive from a computation point of view, since key statistical quantities – such as likelihood ratios and prior distributions – can be calculated or specified locally and graphs can be build up sequentially; see e.g. Lauritzen (1996).

Recently, the family of *weakly structural Markov* (WSM) probabilistic laws for decomposable graphs was introduced by Green and Thomas (2017), providing an analogous clique-separator factorisation for the graph law as for the data distribution. In this paper, we focus on a fully Bayesian and computational approach for inferring posterior graph laws given observed data, a process usually called *structure learning*. Specifically, we consider *strong hyper-* and weakly structural Markov prior laws for the model parameters and graphs respectively, so that the resulting graph posterior also factorises over the set of cliques and separators; see e.g. Dawid and Lauritzen (1993).

The common strategy of Bayesian structure learning i based on the class of Markov chain Monte Carlo (McMC) methods such as e.g. the Metropolis-Hastings sampling scheme. These methods generate, by performing local perturbations on the edge set, Markov chains by either operating directly on the space of decomposable graph or their corresponding junction trees; see for example

Giudici and Green (1999); Dellaportas and Forster (1999); Jones et al. (2005); Green and Thomas (2013). Further pertinent approaches include e.g. Stingo and Marchetti (2015) who focus on Gaussian DGMs and propose edge moves by dynamically updating the perfect sequence of the cliques in the graph. A completely different strategy is presented in Elmasri (2017a,b) where a node-driven McMC sampler operates on tree-dependent bipartite graphs.

The main issue for the above-mentioned samplers as well as other McMC strategies based on local moves is the limited mobility of their corresponding Markov chains, since at each step, only a small part of the edge set is altered. To tackle this issue, we present a procedure for recasting the problem of structure learning in WSM laws, which in general lacks natural sequential interpretation, into a sequential setting by an auxiliary construction that we refer to as a *temporal embedding*, relying partly on the methodology of *sequential Monte Carlo (SMC) samplers*; see Del Moral, Doucet and Jasra (2006). Specifically, we propose a recursive Feynman-Kac model which generates a flow of junction tree distributions over a space of increasing dimensions and develop an efficient SMC sampler on this space. The SMC algorithm is then incorporated as an inner loop of a particle Gibbs (PG) sampler (Andrieu, Doucet and Holenstein, 2010), providing global moves on the underlying graph space. In order to reduce the variance and improve the mobility of the standard PG sampler, we further introduce a step of systematic refreshment by means of backwards sampling.

Our suggested temporal embedding of WSM laws is constructed by a four step *temoralisation* procedure which can be summarised as follows. The procedure is initiated by defining a family of laws on decomposable graph spaces defined on all subsets of the node set. In the context of Bayesian structure learning, these laws will correspond to graph posteriors defined over the corresponding subsets of random variables. The second step of the temporalisation is to extend each graph law to the space of junction tree representations. Following Green and Thomas (2013), this is carried through by rescaling of the underlying graph probabilities by the number of equivalent junction tree representations. In this construction, the marginal law of an underlying graph will be preserved from the first step. Now, in the context of sequential Bayesian structure learning the user may, by always processing the nodes in some given order, run the risk of overlooking dependence relations running counter to this specific order. It is hence desirable to allow the node processing order to be randomised. For this purpose, the third step of the temporalisation procedure augments the junction tree distributions to mixtures of junction tree distributions over different subsets of underlying graph nodes. Finally, in the last step of the temporalisation process, we introduce a sequence of Markov kernels allowing the distributions formed in the third step to be embedded into a recursive Feynman-Kac-distribution flow. The distributions of the resulting Feynman-Kac flow, with ”time parameter” given by the number of nodes of the underlying graph, can be sampled efficiently using sequential Monte Carlo methods.

A central part in the construction of any SMC algorithm is the design of a proposal distribution, which should both dominate the target of interest and preferably be computationally efficient. In our case the junction tree representation introduced in the second step of the temporalisation is of key importance, since it enables us to fulfill these requirements through the so-called Christmas tree algorithm (CTA), presented in the companion paper Olsson, Pavlenko and Rios (2018). The CTA by construction defines a Markov kernel, with closed-form transition probabilities, that dominates the temporalised version of the graph law. Up to our knowledge, the last property seems much harder to obtain by, e.g., operating directly on a path space of decomposable graphs; see e.g. Markenzon, Vernet and Araujo (2008).

The rest of the paper is structured as follows. In Section 2 we introduce some notation and present standard theoretical results for decomposable graphs and the junction tree representation. Section 3 presents the four stage temporalisation strategy procedure. The SMC sampler is designed in Section 4 along with the standard PG and its systematic refreshment extension. In Section 5 we present two motivating examples showing how the WSM laws arise in a Bayesian inference context. In Section 6 we investigate numerically the performance of the suggested PG sampler for three examples of Bayesian structure learning in DGMs. Appendix A contains some graph theoretical notations, proofs and a lemma.

2. Preliminaries

Notational convention

We will always assume that all random variables are well defined on a common probability space $(\Omega, \mathcal{F}, \mathbb{P})$. We denote by $\mathbb{N}$ the positive natural numbers and for any $(m, n) \in \mathbb{N}^2$ we use $[\![m, n]\!]$ to denote the unordered set $\{m, \dots, n\}$. By $\mathbb{R}_+$ and $\mathbb{R}_+^*$ we denote the non-negative and positive real numbers respectively.

Measurable spaces

Given some measurable space $(\mathsf{X}, \mathcal{X})$, we denote by $\mathbb{M}(\mathcal{X})$ and $\mathbb{M}_1(\mathcal{X})$ the sets of measures and probability measures on $(\mathsf{X}, \mathcal{X})$, respectively. In the case where X is a finite set, $\mathcal{X}$ is always assumed to be the power set $\wp(\mathsf{X})$ of X, and we simply write $\mathbb{M}(\mathsf{X})$ and $\mathbb{M}_1(\mathsf{X})$ instead of $\mathbb{M}(\wp(\mathsf{X}))$ and $\mathbb{M}_1(\wp(\mathsf{X}))$, respectively. In the finite case, counting measures will be denoted by $|dx|$. We let $\mathbb{F}(\mathcal{X})$ be the set of measurable functions on $(\mathsf{X}, \mathcal{X})$.

Kernel notation

Let μ be a measure on some measurable space $(\mathsf{X}, \mathcal{X})$. Then for any μ-integrable function h, we use the standard notation

$$\mu h := \int h(x) \, \mu(dx)$$

to denote the Lebesgue integral of h w.r.t. μ.

In addition, let $(\mathsf{Y}, \mathcal{Y})$ be some other measurable space and $\mathbf{K}$ some possibly unnormalised transition kernel $\mathbf{K} : \mathsf{X} \times \mathcal{Y} \to \mathbb{R}_+$. The kernel $\mathbf{K}$ induces two integral operators, one acting on functions and the other on measures. More specifically, given a measure ν on $(\mathsf{X}, \mathcal{X})$ and a measurable function h on $(\mathsf{Y}, \mathcal{Y})$, we define the measure

$$\nu\mathbf{K} : \mathcal{Y} \ni A \mapsto \int \mathbf{K}(x, A)\, \nu(dx)$$

and the function

$$\mathbf{K}h : \mathsf{X} \ni x \mapsto \int h(y)\, \mathbf{K}(x, dy)$$

whenever these quantities are well-defined.

Finally, given a third measurable space $(\mathsf{Z}, \mathcal{Z})$ and a second kernel $\mathbf{L} : \mathsf{Y} \times \mathcal{Z} \to \mathbb{R}_+$ we define, with $\mathbf{K}$ as above, the *product kernel*

$$\mathbf{KL} : \mathsf{X} \times \mathcal{Z} \ni (x, B) \mapsto \int \mathbf{L}(y, B)\, \mathbf{K}(x, dy),$$

whenever this is well-defined.

Decomposable graphs and junction trees

The notion of decomposable graphs and junction trees are introduced below. For general graph theoretical concepts and notations the reader is referred to A.1. A graph G is called decomposable if and only if its cliques can be be arranged in a so-called *junction tree*, i.e. a tree whose nodes are the cliques in G, and where for any pair of cliques Q and Q' in G, the intersection $Q \cap Q'$ is contained in each of the cliques on the unique path $Q \sim Q'$. Decomposable graphs are sometimes alternatively termed *chordal* or *triangulated*, as an equivalent requirement is that every cycle of length 4 or more is chorded, see e.g Diestel (2005). Each edge (Q, Q') in a junction tree is associated with the intersection $S = Q \cap Q'$, which is referred to as a *separator*. Since all junction tree representations of a specific decomposable graph G have the same separators, it makes sense to speak about “the separators of a decomposable graph”. We denote by $\mathcal{S}(G)$ the *multiset* of separators formed by a graph G, where each separator has a multiplicity. The set of equivalent junction tree representations of a decomposable graph G is denoted by $\mathcal{T}(G)$, and $\mu(G) := |\mathcal{T}(G)|$ denotes the number of such representations. The unique graph underlying a specific junction tree T is denoted by $g(T)$.

3. Temporal embedding of weakly structural Markov laws

From now on, let V be a fixed set of $p \in \mathbb{N}$ distinct nodes. Without loss of generality, we let $V = [\![1, p]\!]$. For $U \subseteq V$, we denote by $\mathcal{G}_U$ the space of decomposable graphs with nodes U, i.e., $\mathcal{G}_U := \{(U, E) : E \subseteq U \times U\}$. In particular, set $\mathcal{G} := \mathcal{G}_V$. In addition, let $\bar{\mathcal{G}} := \cup_{U \subseteq V} \mathcal{G}_U$ be the space of all decomposable graphs with nodes given by V or some subset of the same.

Definition 1. *A positive function* γ *on* $\bar{\mathcal{G}}$ *is said to satisfy the* clique-separator factorisation *(CSF) if for all* $G \in \bar{\mathcal{G}}$*,*

$$\gamma(G) = \frac{\prod_{Q \in \mathcal{Q}(G)} \gamma(Q)}{\prod_{S \in \mathcal{S}(G)} \gamma(S)}.$$

For some given function γ satisfying the CSF, the aim of this paper is to develop a strategy for sampling from the family of so-called *weakly structural Markov* laws on $\mathbb{M}_1(\mathcal{G})$ (Green and Thomas, 2017), which assuming full support on $\mathcal{G}$, are characterised as

$$\eta^\star(dG) = \frac{\gamma^\star(dG)}{\gamma^\star \mathbb{1}_{\mathcal{G}}}, \tag{3.1}$$

where

$$\gamma^\star(dG) := \gamma|_{\mathcal{G}}(G)\,|dG|,$$

with $\gamma|_{\mathcal{G}}$ denoting the restriction of γ to $\mathcal{G}$ and $|dG|$ the counting measure on $\mathcal{G}$. The normalising constant $\gamma^\star \mathbb{1}_{\mathcal{G}} = \sum_{G \in \mathcal{G}} \gamma^\star(G)$ will be considered as intractable, as computing the same requires the summation of over the whole space $\mathcal{G}$, which is impractical as the cardinality of $\mathcal{G}$ is immense already for moderate p.

Our goal is now to develop an efficient strategy for sampling from distributions of form (3.1). As mentioned in the introduction, particle McMC methods are appealing as these allow McMC chains with "global" moves to be defined also on large spaces. However, unlike our setting, SMC methods sample from *sequences* of distributions, and a key ingredient of our developments is hence to provide an auxiliary, sequential reformulation of the sampling problem under consideration. This construction, which we will refer to as *temporalisation*, comprise four steps described in the following.

Step I

Using the function γ inducing the target (3.1) of interest, define, for each $U \subseteq V$, the measure

$$\eta^\star\langle U \rangle(dG) = \frac{\gamma^\star\langle U \rangle(dG)}{\gamma^\star\langle U \rangle \mathbb{1}_{\mathcal{G}_U}}$$

in $\mathbb{M}_1(\mathcal{G}_U)$, where

$$\gamma^\star\langle U \rangle(dG) := \gamma|_{\mathcal{G}_U}(G)\,|dG|,$$

with $\gamma|_{\mathcal{G}_U}$ denoting the restriction of γ to $\mathcal{G}_U$ and $|dG|$ the counting measure on $\mathcal{G}_U$. Note that $\eta^\star\langle V \rangle$ coincides with $\eta^\star$, the target of interest. As usual, we will let the same symbols $\gamma^\star\langle U \rangle$ and $\eta^\star\langle U \rangle$ denote the probability functions of these measures.

Step II

Extend each distribution $\eta^\star\langle U\rangle$ to a distribution $\eta^*\langle U\rangle$ on $\mathcal{T}_U := \cup_{G\in\mathcal{G}_U}\mathcal{T}(G)$, the space of junction tree representations of graphs in $\mathcal{G}_U$. Following Green and Thomas (2013), one way of carrying through this extension is to define, for each $U \subseteq V$, the measure

$$\eta^*\langle U\rangle(dT) := \frac{\gamma^*\langle U\rangle(dT)}{\gamma^*\langle U\rangle \mathbb{1}_{\mathcal{T}_U}} \tag{3.2}$$

in $\mathbb{M}_1(\mathcal{T}_U)$, where

$$\gamma^*\langle U\rangle(dT) := \frac{\gamma^\star\langle U\rangle \circ g(T)}{\mu \circ g(T)}\,|dT|,$$

with $|dT|$ denoting the counting measure on $\mathcal{T}_U$. In particular, we set $\gamma^* = \gamma^*\langle V\rangle$ and $\eta^* = \eta^*\langle V\rangle$.

Step III

Let, for all $m \in [\![1,p]\!]$, S_m be the space of all m-combinations of elements in $[\![1,p]\!]$. An element $S_m \in \mathsf{S}_m$ is of form $S_m = (S_{1|m},\dots,S_{m|m})$ where $\{S_{\ell|m}\}_{\ell=1}^m \subseteq [\![1,p]\!]$ are distinct. In particular, $\mathsf{S}_p = \{(1,\dots,p)\}$. For $(\ell,\ell') \in [\![1,m]\!]^2$ such that $\ell \le \ell'$, we denote $S_{\ell:\ell'|m} := (S_{\ell|m},\dots,S_{\ell'|m})$. In addition, we define, for all $m \in [\![1,p]\!]$, the extended state spaces

$$\mathcal{X}_m := \bigcup_{S_m\in\mathsf{S}_m} \left(\{S_m\}\times\mathcal{T}_{S_m}\right),$$

and, for some given discrete probability distribution σ_m on S_m, extended target distributions

$$\eta_m(dx_m) = \frac{\gamma_m(dx_m)}{\gamma_m\mathbb{1}_{\mathcal{X}_m}},$$

in $\mathbb{M}_1(\mathcal{X}_m)$, where

$$\gamma_m(dx_m) = \gamma_m(dS_m, dT_m) := \gamma^*\langle S_m\rangle(dT_m)\,\sigma_m(dS_m).$$

Here we have chosen to write T_m instead of T_{S_m} in order to avoid double subscript notation. The measures $\{\sigma_m\}_{m=1}^p$ are supposed to satisfy the recursion

$$\sigma_{m+1} = \sigma_m\bar{\mathbf{\Sigma}}_m,$$

where

$$\bar{\mathbf{\Sigma}}_m(S_m, dS_{m+1}) := \delta_{S_m}(dS_{1:m|m+1})\,\mathbf{\Sigma}_m(S_{1:m|m+1}, dS_{m+1|m+1}), \tag{3.3}$$

with $\mathbf{\Sigma}_m$ being a Markov transition kernel from S_m to $[\![1,p]\!]$ such that $\mathbf{\Sigma}_m(S_m,j) = 0$ for all $j \in S_m$.

Step IV

Let $\{\mathbf{R}_m\}_{m=1}^{p-1}$ be a sequence of Markov transition kernels acting in the *reversed* direction, i.e., for each m, $\mathbf{R}_m : \mathcal{X}_{m+1} \times \wp(\mathcal{X}_m) \to [0,1]$, and define, following Del Moral, Doucet and Jasra (2006), for all $m \in [\![1,p]\!]$,

$$\bar{\gamma}_m(dx_{1:m}) := \gamma_m(dx_m) \prod_{\ell=1}^{m-1} \mathbf{R}_\ell(x_{\ell+1}, dx_\ell) \tag{3.4}$$

and

$$\bar{\eta}_m(dx_{1:m}) := \frac{\bar{\gamma}_m(dx_{1:m})}{\bar{\gamma}_m \mathbb{1}_{\mathcal{X}_{1:m}}} = \frac{\bar{\gamma}_m(dx_{1:m})}{\gamma_m \mathbb{1}_{\mathcal{X}_m}}, \tag{3.5}$$

where $\mathcal{X}_{1:m} := \prod_{\ell=1}^m \mathcal{X}_\ell$.[1]

Trivially, $\bar{\eta}_m$ allows η_m as a marginal distribution with respect to the last component x_m, therefore we regard (3.5) as a *temporal embedding* of (3.1). We conclude this section by some remarks and comments on the steps of the above described procedure. We first note that, in step II, by Lemma 3 (see A.2), for $G \in \mathcal{G}_U$,

$$\mathbb{P}_{\eta^*\langle U\rangle}\left(\tau = T \mid g(\tau) = G\right) = \frac{\mathbb{P}_{\eta^*\langle U\rangle}\left(\tau = T, g(\tau) = G\right)}{\mathbb{P}_{\eta^*\langle U\rangle}\left(g(\tau) = G\right)} = \frac{\eta^*\langle U\rangle(T)}{\eta^\star\langle U\rangle(G)} \mathbb{1}_{\{G=g(T)\}}.$$

Moreover, using (A.1), the right hand side can be expressed as

$$\frac{\eta^*\langle U\rangle(T)}{\eta^\star\langle U\rangle(G)} \mathbb{1}_{\{G=g(T)\}} = \frac{\eta^\star\langle U\rangle \circ g(T)}{\eta^\star\langle U\rangle(G)\mu \circ g(T) \mathbb{1}_{\{G=g(T)\}}} = \frac{1}{\mu(G)} \mathbb{1}_{\mathcal{T}(G)}(T),$$

i.e., under $\eta^*\langle U\rangle$, conditionally on the event $\{g(\tau) = G\}$, the tree τ is *uniformly* distributed over the set $\mathcal{T}(G)$ (recall that $\mu(G)$ is the cardinality of $\mathcal{T}(G)$). In other words, a draw from $\eta^*\langle U\rangle$ can be generated by drawing a graph according to $\eta^\star\langle U\rangle$ and then drawing a tree uniformly over all junction tree representations of that graph.

In step III, each $\gamma_m(dx_m)$ has a density $\gamma_m(x_m) = \gamma^*\langle S_m\rangle(T_m)\sigma_m(S_m)$ (by abuse of notation, we reuse the same symbol) w.r.t. $|dx_m|$, the counting measure on $\mathcal{X}_m$. Moreover, since $\sigma_p = \delta_{[\![1,p]\!]}$, η^* is the marginal of η_p with respect to the T_p component. Further we note that $\bar{\mathbf{\Sigma}}_m$ is a Markov transition kernel from S_m to S_{m+1}. In other words, $\bar{\mathbf{\Sigma}}_m$ transforms a given m-combination S_m into an $(m+1)$-combination S_{m+1} by selecting randomly an element s^* from the (non-empty) set $[\![1,p]\!] \setminus S_m$ according to $\mathbf{\Sigma}_m(S_m, \cdot)$ and adding the same to S_m. When selecting s^*, several approaches are possible; s^* can, e.g., be selected randomly from the set $\{s \in [\![1,p]\!] : \min_{s' \in S_m} |s - s'| \leq \delta\}$ for some prespecified distance $\delta \in [\![1,p]\!]$. Also the initial distribution σ_1 can be designed freely, e.g., as the uniform distribution over $[\![1,p]\!]$.

[1] Here and in the following, we put a bar on top of a measure, kernel, function, etc., in order to indicate that the quantity is defined on a path space.

In step IV, as the reversed kernels are assumed to be Markovian and known to the user, each extended target distribution $\bar{\eta}_m$ is known up to the same normalising constant $\gamma_m \mathbb{1}_{\mathcal{X}_m}$ as its marginal η_m. The algorithm that we propose is based on the observation that the distribution flow $\{\eta_m\}_{m=1}^{p}$ satisfies the recursive *Feynman-Kac model*

$$\eta_{m+1}(dx_{m+1}) = \frac{\eta_m \mathbf{Q}_m(dx_{m+1})}{\eta_m \mathbf{Q}_m \mathbb{1}_{\mathcal{X}_{m+1}}} \quad (m \in [\![1, p-1]\!]), \tag{3.6}$$

where we have defined the un-normalised transition kernel

$$\mathbf{Q}_m(x_m, dx_{m+1}) := \begin{cases} \dfrac{\gamma_{m+1}(dx_{m+1})\, \mathbf{R}_m(x_{m+1}, x_m)}{\gamma_m(x_m)}, & x_m \in \operatorname{Supp}(\gamma_m), \\ 0, & \text{otherwise.} \end{cases}$$

In the SMC sampler framework of Del Moral, Doucet and Jasra (2006), focus is set on sampling from a *sequence* of probability densities known up to normalising constants and defined on the *same* state space. In this context, the authors propose to transform the given distribution sequence into a sequence of distributions over state spaces of increasing dimension (given by powers of the original space) by means an auxiliary Markovian transition kernel. In this construction, each extended distribution is of form (3.4), with $\mathcal{X}_m \equiv \mathcal{X}$ for all m, and allows the original density of interest as a marginal with respect to the last component x_m. Having access to such a flow of distributions over spaces of increasing dimensions, standard SMC methods provide numerically stable online approximation of the marginals, the latter satisfying a Feynman-Kac recursion of form (3.6).

In our case, we arrive at the recursion (3.6) from an entirely different direction, i.e., by starting off with a *single* distribution defined on a possibly high-dimensional space and constructing an auxiliary sequence of increasingly complex distributions used for directing an SMC particle sample towards the distribution of interest (see the next section).

4. Particle approximation of temporalised weakly structural Markov laws

In the following we discuss how to obtain a particle interpretation of the recursion (3.6). Assume for the moment that we have at hand a sequence $\{\mathbf{K}_m\}_{m=1}^{p-1}$ of proposal kernels such that $\mathbf{Q}_m(x_m, \cdot) \ll \mathbf{K}_m(x_m, \cdot)$ for all $m \in [\![1, p-1]\!]$ and all $x_m \in \mathcal{X}_m$. In our applications, we will let these proposal kernels correspond to the so-called *Christmas tree algorithm* (CTA) proposed in the companion paper Olsson, Pavlenko and Rios (2018) and overviewed in Section 4.1.2.

4.1. Sequential Monte Carlo approximation

We proceed recursively and assume that we are given a sample $\{(\xi_m^i, \omega_m^i)\}_{i=1}^{N}$ of particles, each particle $\xi_m^i = (\varsigma_m^i, \tau_m^i)$ being a random draw in $\mathcal{X}_m$ (more specifically, ς_m^i is a random m-combination in $[\![1, p]\!]$ and τ_m^i a random draw in $\mathsf{Z}_{\varsigma_m^i}$), with associated importance weights (the ω_m^i's) approximating η_m in the sense that for all $h \in \mathbb{F}(\mathcal{X}_m)$,

$$\eta_m^N h \simeq \eta_m h \quad \text{as } N \to \infty,$$

where

$$\eta_m^N(dx_m) := \sum_{i=1}^N \frac{\omega_m^i}{\Omega_m^N} \delta_{\xi_m^i}(dx_m),$$

with $\Omega_m^N := \sum_{i=1}^N \omega_m^i$, denotes the weighted empirical measure associated with the particle sample.

In order to produce an updated particle sample $\{(\xi_{m+1}^i, \omega_{m+1}^i)\}_{i=1}^N$ approximating η_{m+1}^N, we plug η_m^N into the recursion (3.6) and sample from the resulting distribution

$$\frac{\eta_m^N \mathbf{Q}_m(dx_{m+1})}{\eta_m^N \mathbf{Q}_m \mathbb{1}_{\mathcal{X}_{m+1}}} = \sum_{i=1}^N \frac{\omega_m^i \mathbf{Q}_m(\xi_m^i, dx_{m+1})}{\sum_{\ell=1}^N \omega_m^\ell \mathbf{Q}_m \mathbb{1}_{\mathcal{X}_{m+1}}(\xi_m^\ell)}$$

by means of importance sampling. For this purpose we first extend the previous measure to the index component, yielding the mixture

$$\check{\eta}_{m+1}^N(di, dx_{m+1}) := \frac{\omega_m^i \mathbf{Q}_m(\xi_m^i, dx_{m+1})}{\sum_{\ell=1}^N \omega_m^\ell \mathbf{Q}_m \mathbb{1}_{\mathcal{X}_{m+1}}(\xi_m^\ell)} \, |di|$$

on the product space $[\![1, N]\!] \times \mathcal{X}_{m+1}$, and sample from the latter by drawing i.i.d. samples $\{(I_{m+1}^i, \xi_{m+1}^i)\}_{i=1}^N$ from the proposal distribution

$$\rho_{m+1}^N(di, dx_{m+1}) := \frac{\omega_m^i}{\Omega_m^N} \mathbf{K}_m(\xi_m^i, dx_{m+1}) \, |di|.$$

Each draw (I_{m+1}^i, ξ_{m+1}^i) is assigned an importance weight

$$\omega_{m+1}^i := w_m(\xi_m^{I_{m+1}^i}, \xi_{m+1}^i) \propto \frac{d\check{\eta}_{m+1}^N}{d\rho_{m+1}^N}(I_{m+1}^i, \xi_{m+1}^i),$$

where we have defined the importance weight function

$$w_m(x_m, x_{m+1}) := \frac{d\mathbf{Q}_m(x_m, \cdot)}{d\mathbf{K}_m(x_m, \cdot)}(x_{m+1}) = \frac{\gamma_{m+1}(x_{m+1}) \mathbf{R}_m(x_{m+1}, x_m)}{\gamma_m(x_m) \mathbf{K}_m(x_m, x_{m+1})}. \tag{4.1}$$

Finally, the weighted empirical measure

$$\eta_{m+1}^N(dx_{m+1}) := \sum_{i=1}^N \frac{\omega_{m+1}^i}{\Omega_{m+1}^N} \delta_{\xi_{m+1}^i}(dx_{m+1})$$

is returned as an approximation of η_{m+1}.

We will always assume that the proposal kernel $\mathbf{K}_m$ is of form

$$\mathbf{K}_m(x_m, dx_{m+1}) = \bar{\mathbf{\Sigma}}_m(S_m, dS_{m+1})\, \mathbf{K}^*_m\langle S_m, S_{m+1}\rangle(T_m, dT_{m+1}), \tag{4.2}$$

where $\bar{\mathbf{\Sigma}}_m$ is defined in (3.3) and for all $(S_m, S_{m+1}) \in \mathsf{S}_m \times \mathsf{S}_{m+1}$, $\mathbf{K}^*_m\langle S_m, S_{m+1}\rangle$ is a Markov transition kernel from $\mathcal{X}_{S_m}$ to $\mathcal{X}_{S_{m+1}}$. Each law $\mathbf{K}^*_m\langle S_m, S_{m+1}\rangle(T_m, \cdot)$, $T_m \in \mathcal{T}_{S_m}$, has a probability function, which we denote by the same symbol. Note that the assumption (4.2) implies that for all $i \in [\![1, N]\!]$,

$$\varsigma^i_{1:m|m+1} = \varsigma_m^{I^i_{m+1}},$$

and, consequently, by (3.3),

$$\sigma_{m+1}(\varsigma^i_{m+1}) = \sigma_m \bar{\mathbf{\Sigma}}_m(\varsigma^i_{m+1}) = \sigma_m(\varsigma_m^{I^i_{m+1}}) \bar{\mathbf{\Sigma}}_m(\varsigma_m^{I^i_{m+1}}, \varsigma^i_{m+1}).$$

Thus, the importance weight (4.1) simplifies according to

$$\begin{aligned} &w_m(\xi_m^{I^i_{m+1}}, \xi^i_{m+1}) \\ &\quad = \frac{\gamma^*\langle \varsigma^i_{m+1}\rangle(\tau^i_{m+1})\sigma_{m+1}(\varsigma^i_{m+1})\mathbf{R}_m(\xi_m^{I^i_{m+1}}, \xi^i_{m+1})}{\gamma^*\langle \varsigma_m^{I^i_{m+1}}\rangle(\tau_m^{I^i_{m+1}})\sigma_m(\varsigma_m^{I^i_{m+1}})\bar{\mathbf{\Sigma}}_m(\varsigma_m^{I^i_{m+1}}, \varsigma^i_{m+1})\mathbf{K}^*_m\langle \varsigma_m^{I^i_{m+1}}, \varsigma^i_{m+1}\rangle(\tau_m^{I^i_{m+1}}, \tau^i_{m+1})} \\ &\quad = \frac{\gamma^*\langle \varsigma^i_{m+1}\rangle(\tau^i_{m+1})\mathbf{R}_m(\xi_m^{I^i_{m+1}}, \xi^i_{m+1})}{\gamma^*\langle \varsigma_m^{I^i_{m+1}}\rangle(\tau_m^{I^i_{m+1}})\mathbf{K}^*_m\langle \varsigma_m^{I^i_{m+1}}, \varsigma^i_{m+1}\rangle(\tau_m^{I^i_{m+1}}, \tau^i_{m+1})}. \end{aligned}$$

Further, we have the identity

$$\begin{aligned} \frac{\gamma^*\langle \varsigma^i_{m+1}\rangle(\tau^i_{m+1})}{\gamma^*\langle \varsigma_m^{I^i_{m+1}}\rangle(\tau_m^{I^i_{m+1}})} &= \frac{\mu \circ g(\tau_m^{I^i_{m+1}})}{\mu \circ g(\tau^i_{m+1})} \times \frac{\prod_{Q \in \mathcal{Q}(g(\tau^i_{m+1}))} \gamma(Q) \prod_{Q \in \mathcal{Q}(g(\tau_m^{I^i_{m+1}}))} \gamma(Q)^{-1}}{\prod_{S \in \mathcal{S}(g(\tau^i_{m+1}))} \gamma(S) \prod_{S \in \mathcal{S}(g(\tau_m^{I^i_{m+1}}))} \gamma(S)^{-1}} \\ &= \frac{\mu \circ g(\tau_m^{I^i_{m+1}})}{\mu \circ g(\tau^i_{m+1})} \times \frac{\prod_{Q \in \mathcal{Q}(g(\tau^i_{m+1})) \triangle \mathcal{Q}(g(\tau_m^{I^i_{m+1}}))} \gamma(Q)^{\mathbf{1}\langle \tau^i_{m+1}\rangle(Q)}}{\prod_{S \in \mathcal{S}(g(\tau^i_{m+1})) \triangle \mathcal{S}(g(\tau_m^{I^i_{m+1}}))} \gamma(S)^{\mathbf{1}\langle \tau^i_{m+1}\rangle(S)}}, \end{aligned} \tag{4.3}$$

where $\triangle$ denotes symmetric difference and

$$\mathbf{1}\langle \tau^i_{m+1}\rangle(Q) := 2\mathbb{1}_{\mathcal{Q}(g(\tau^i_{m+1}))}(Q) - 1$$

($\mathbf{1}\langle \tau^i_{m+1}\rangle(S)$ is defined similarly).

The computational burden involved in computing the first factor of (4.3) can be substantially reduced by exploiting the factorisation presented in Olsson, Pavlenko and Rios (2018, Theorem 7) and restated below. Let $G_{m+1} \in \mathcal{G}_{m+1}$ be a graph expanded from a graph $G_m \in \mathcal{G}_m$ in the sense that $G_{m+1}[\{1, \ldots, m\}] =$

G_m, then we can define the set $\mathcal{S}^\star \subset \mathcal{S}(G_{m+1})$ consisting of the separators created by the expansion. The factorisation is then given as

$$\frac{\mu(G_m)}{\mu(G_{m+1})} = \frac{\prod_{s \in \mathcal{U}_1} \nu_G(s)}{\prod_{s \in \mathcal{U}_2} \nu_{G_{m+1}}(s)},$$

where $\mathcal{U}_1 = \{s \in \mathcal{S}(G_m) : \exists s' \in \mathcal{S}^\star, \text{ such that } s \subset s'\}$ and $\mathcal{U}_2 = \{s \in \mathcal{S}(G_{m+1}) : \exists s' \in \mathcal{S}^\star, \text{ such that } s \subset s'\}$ are the set of separators in G_m and G_{m+1} respectively, contained in some separator in $\mathcal{S}^\star$. The function $\nu_G(s)$ denotes the number of equivalent junction trees that can be obtained by randomizing a junction tree for the graph G at the separator s. For a more detailed presentation see Olsson, Pavlenko and Rios (2018). The sets $\mathcal{Q}(g(\tau^i_{m+1})) \triangle \mathcal{Q}(g(\tau_m^{I^i_{m+1}}))$ and $\mathcal{S}(g(\tau^i_{m+1})) \triangle \mathcal{S}(g(\tau_m^{I^i_{m+1}}))$ in the second factor might be composed by only a few cliques and separators, respectively, and computing the products in the numerator and denominator of (4.3) will in that case be an easy operation. This is the case for the CTA described in Section 4.1.2 below.

In summary the identity (4.3) suggests that the first part of the importance weights may, in principle, be computed with a complexity that does not increase with the iteration index m as long as the proposal kernel $\mathbf{K}^*_m$ only modifies and extends *locally* the junction tree (and, consequently, the underlying graph).

The SMC update described above is summarised in Algorithm 1. Here and in the following, we let $\mathsf{Pr}(\{a_\ell\}_{\ell=1}^N)$ denote the categorical probability distribution induced by a set $\{a_\ell\}_{\ell=1}^N$ of positive (possibly unnormalised) numbers; thus, writing $W \sim \mathsf{Pr}(\{a_\ell\}_{\ell=1}^N)$ means that the variable W takes the value $\ell \in [\![1, N]\!]$ with probability $a_\ell / \sum_{\ell'=1}^N a_{\ell'}$.

```
Data: $\{(\xi^i_m, \omega^i_m)\}_{i=1}^N$
Result: $\{(\xi^i_{m+1}, \omega^i_{m+1})\}_{i=1}^N$
for $i \leftarrow 1, \ldots, N$ do
    draw $I^i_{m+1} \sim \mathsf{Pr}(\{\omega^\ell_m\}_{\ell=1}^N)$;
    draw $\varsigma^i_{m+1} \sim \bar{\mathbf{\Sigma}}_m(\varsigma_m^{I^i_{m+1}}, dS_{m+1})$;
    draw $\tau^i_{m+1} \sim \mathbf{K}^*_m\langle \varsigma_m^{I^i_{m+1}}, \varsigma^i_{m+1}\rangle(\tau_m^{I^i_{m+1}}, dT_{m+1})$;
    set $\xi^i_{m+1} \leftarrow (\varsigma^i_{m+1}, \tau^i_{m+1})$;
    set $\omega^i_{m+1} \leftarrow \dfrac{\gamma^*\langle \varsigma^i_{m+1}\rangle(\tau^i_{m+1}) \mathbf{R}_m(\xi_m^{I^i_{m+1}}, \xi^i_{m+1})}{\gamma^*\langle \varsigma_m^{I^i_{m+1}}\rangle(\tau_m^{I^i_{m+1}}) \mathbf{K}^*_m\langle \varsigma_m^{I^i_{m+1}}, \varsigma^i_{m+1}\rangle(\tau_m^{I^i_{m+1}}, \tau^i_{m+1})}$;
```

Algorithm 1: SMC update

Naturally, the SMC algorithm is initialised by drawing i.i.d. draws $(\xi^i_1)_{i=1}^N$ from some initial distribution $\kappa \in \mathbb{M}_1(\mathcal{X}_1)$ and letting $\omega^i_1 = \gamma_1(\xi^i_1)/\kappa(\xi^i_1)$ for all i, where the density (with respect to dx_1) of κ is denoted by the same symbol.

In addition, letting κ be of form

$$\kappa(dx_1) = \sigma_1(dS_1)\kappa^*\langle S_1\rangle(dT_1)$$

yields the weights $\omega_1^i = \gamma^*\langle\varsigma_1^i\rangle(\tau_1^i)/\kappa^*\langle\varsigma_1^i\rangle(\tau_1^i)$.

As a by-product, Algorithm 1 provides, for all $m \in [\![1, p]\!]$ and $h \in \mathbb{F}(\mathcal{X}_p)$, unbiased estimators

$$\gamma_m^N h := \frac{1}{N^m}\left(\prod_{\ell=1}^{m-1}\Omega_\ell^N\right)\sum_{i=1}^{N}\omega_m^i h(\xi_m^i)$$

of $\gamma_m h$. In particular,

$$\gamma_p^N \mathbb{1}_{\mathcal{X}_p} = \frac{1}{N^p}\prod_{\ell=1}^{p}\Omega_\ell^N$$

is an unbiased estimator of the normalising constant $\gamma_p\mathbb{1}_{\mathcal{X}_p} = \gamma^*\mathbb{1}_{\mathcal{T}_p}$ of the distribution of interest.

4.1.1. Design of retrospective dynamics

As we will see, the reversed kernels $\{\mathbf{R}_m\}_{m=1}^{p-1}$ will typically be designed on the basis of the forward proposal kernels $\{\mathbf{K}_m\}_{m=1}^{p-1}$. It is clear that for all $m \in [\![1, p-1]\!]$, the constraint that $\mathbf{Q}_m(x_m, \cdot) \ll \mathbf{K}_m(x_m, \cdot)$ for all $x_m \in \mathcal{X}_m$ is satisfied as soon as the retrospective kernel $\mathbf{R}_m$ is such that $\mathrm{Supp}(\mathbf{R}_m(x_{m+1}, \cdot)) \subseteq \mathrm{Supp}(\mathbf{K}_m(\cdot, x_{m+1}))$ for all $x_{m+1} \in \mathrm{Supp}(\gamma_{m+1})$. Consequently, if for all $m \in [\![1, p-1]\!]$,

$$\mathrm{Supp}(\eta_1\mathbf{K}_1\cdots\mathbf{K}_m) = \mathrm{Supp}(\gamma_{m+1}), \tag{4.4}$$

one may, e.g., construct each retrospective kernel $\mathbf{R}_m$ by identifying, for all $x_{m+1} \in \mathrm{Supp}(\gamma_{m+1})$, a nonempty set

$$\mathsf{S}_m(x_{m+1}) \subseteq \mathrm{Supp}(\mathbf{K}_m(\cdot, x_{m+1})) \cap \mathrm{Supp}(\gamma_m),$$

and letting

$$\mathbf{R}_m(x_{m+1}, x_m) := |\mathsf{S}_m(x_{m+1})|^{-1}\mathbb{1}_{\mathsf{S}_m(x_{m+1})}(x_m) \quad (x_{m+1} \in \mathrm{Supp}(\gamma_{m+1})), \tag{4.5}$$

i.e., $\mathbf{R}_m(x_{m+1}, dx_m)$ is the uniform distribution over $\mathsf{S}_m(x_{m+1})$. The existence of such a nonempty set is guaranteed by (4.4). Indeed, let $x_{m+1} \in \mathrm{Supp}(\gamma_{m+1})$; then, by (4.4),

$$\sum_{x_m \in \mathcal{X}_m} \eta_1\mathbf{K}_1\cdots\mathbf{K}_m(x_m)\mathbf{K}_m(x_m, x_{m+1}) > 0,$$

i.e., there exists at least one $x_m^* \in \mathcal{X}_m$ such that $\eta_1\mathbf{K}_1\cdots\mathbf{K}_m(x_m^*) > 0$ and $\mathbf{K}_m(x_m^*, x_{m+1}) > 0$. Thus, again by (4.4), $x_m^* \in \mathrm{Supp}(\mathbf{K}_m(\cdot, x_{m+1}))\cap\mathrm{Supp}(\gamma_m)$, which is hence nonempty. For $x_{m+1} \notin \mathrm{Supp}(\gamma_{m+1})$, $\mathbf{R}_m(x_{m+1}, dx_m)$ may be defined arbitrarily. As we will see next, the property (4.4) is satisfied by the junction tree expanders used by us.

4.1.2. The Christmas tree algorithm

Following the presentation of Olsson, Pavlenko and Rios (2018) we disregard, without loss of generality, the permutations of the nodes for the underlying graphs specified by $\mathcal{X}_m$. This implies that we consider a fixed set of ordered nodes $S_m = (1, \ldots, m) \in \mathsf{S}_m$ and by $\mathcal{T}_m$ we mean $\mathcal{T}_{S_m}$.

As previously mentioned $\{\mathbf{K}_m\}_{m=1}^{p-1}$ and $\{\mathbf{R}_m\}_{m=1}^{p-1}$ will here correspond to the kernels induced by the CTA and its reversed version, respectively. The CTA kernel takes as input a junction tree $T_m \in \mathcal{T}_m$ and expands it into a new junction tree $T_{m+1} \in \mathcal{T}_{m+1}$ according to $\mathbf{K}_m(T_m, dT_{m+1})$ by adding the internal node $m+1$ to the underlying graph $g(T_m)$ in such a way that $g(T_{m+1})[\{1, \ldots, m\}] = g(T_m)$. It requires two input parameters $(\alpha, \beta) \in (0, 1)^2$ jointly controlling the sparsity of the produced underlying graph. Specifically, at the initial step of the algorithm, a Bernoulli trial with parameter β is performed in order to determine whether or not the internal node $m+1$ is being isolated in the underlying graph of the produced tree. If $m+1$ is not isolated, a high value of the parameter α controls the number of cliques in T_{m+1} that will contain $m+1$. In this sense, $\mathbf{K}_m(T_m, dT_{m+1})$ is a mixture distribution with weight parameter β.

4.2. Particle Gibbs sampling

In the following, we discuss how to sample from the extended target $\eta_{1:p}$, having the distribution η^* of interest as a marginal distribution, using *Markov chain Monte Carlo* (McMC) methods. A *particle Gibbs* (PG) *sampler* constructs, using SMC, a Markov kernel $\mathbf{P}_p^N$ leaving $\eta_{1:p}$ invariant. Algorithmically, the more or less only difference between the PG kernel and the standard SMC algorithm is that the PG kernel, which is described in detail in Algorithm 2, evolves the particle cloud *conditionally* on a fixed reference trajectory specified *a priori*; this *conditional SMC* algorithm is constituted by Lines 1–16 in Algorithm 2. After having evolved, for p time steps, the particles of the conditional SMC algorithm, the PG kernel draws randomly a particle from the last generation (Lines 17–19), traces the genealogical history of the selected particle back to the first generation (Lines 20–22), and returns the traced path (Line 23).

As as established in (Chopin and Singh, 2015, Proposition 8), $\mathbf{P}_p^N$ is $\eta_{1:p}$-reversible and thus leaves $\eta_{1:p}$ invariant. Interestingly, reversibility holds true for any particle sample size $N \in \mathbb{N} \setminus \{1\}$. Thus, on the basis of $\mathbf{P}_p^N$, the PG sampler generates (after possible burn-in) a Markov chain $\{X_{1:p}^\ell\}_{\ell \in \mathbb{N}}$ according to

$$X_{1:p}^1 \xrightarrow{\mathbf{P}_p^N} X_{1:p}^2 \xrightarrow{\mathbf{P}_p^N} X_{1:p}^3 \xrightarrow{\mathbf{P}_p^N} X_{1:p}^4 \rightarrow \cdots$$

and returns $\sum_{\ell=1}^M h(X_{1:p}^\ell)/M$ as an estimate of $\eta_{1:p}h$ for any $\eta_{1:p}$-integrable objective function $h \in \mathbb{F}(\mathcal{X}_{1:p})$. Here $M \in \mathbb{N}$ denotes the McMC sample size. In particular, in the case where the objective function h depends on the argument

T_p only, we obtain the estimator

$$\sum_{\ell=1}^{M} h(Z_p^\ell)/M \tag{4.6}$$

of $\eta^* h$, where each Z_p^ℓ variable is extracted, on Line 18, at iteration $\ell - 1$. of Algorithm 2.

```
   Data: a reference trajectory $x_{1:p} \in \mathcal{X}_{1:p}$
   Result: a draw $X_{1:p}$ from $\mathbf{P}_p^N(x_{1:p}, dx'_{1:p})$
1  for $i \leftarrow 1, \dots, N-1$ do
2      draw $\varsigma_1^i \sim \sigma_1(dS_1)$;
3      draw $\tau_1^i \sim \kappa^*\langle\varsigma_1^i\rangle(dT_1)$;
4      set $\xi_1^i \leftarrow (\varsigma_1^i, \tau_1^i)$;
5  set $\xi_1^N \leftarrow x_1$;
6  for $i \leftarrow 1, \dots, N$ do
7      set $\omega_1^i \leftarrow \gamma^*\langle\varsigma_1^i\rangle(\tau_1^i)/\kappa^*\langle\varsigma_1^i\rangle(\tau_1^i)$;
8  for $m \leftarrow 1, \dots, p-1$ do
9      for $i \leftarrow 1, \dots, N-1$ do
10         draw $I_{m+1}^i \sim \mathsf{Pr}(\{\omega_m^\ell\}_{\ell=1}^N)$;
11         draw $\varsigma_{m+1}^i \sim \bar{\mathbf{\Sigma}}_m(\varsigma_m^{I_{m+1}^i}, dS_{m+1})$;
12         draw $\tau_{m+1}^i \sim \mathbf{K}_m^*\langle\varsigma_m^{I_{m+1}^i}, \varsigma_{m+1}^i\rangle(\tau_m^{I_{m+1}^i}, dT_{m+1})$;
13         set $\xi_{m+1}^i \leftarrow (\varsigma_{m+1}^i, \tau_{m+1}^i)$;
14     set $\xi_{m+1}^N \leftarrow x_{m+1}$;
15     for $i \leftarrow 1, \dots, N$ do
16         set $\omega_{m+1}^i \leftarrow \dfrac{\gamma^*\langle\varsigma_{m+1}^i\rangle(\tau_{m+1}^i)\mathbf{R}_m(\xi_m^{I_{m+1}^i}, \xi_{m+1}^i)}{\gamma^*\langle\varsigma_m^{I_{m+1}^i}\rangle(\tau_m^{I_{m+1}^i})\mathbf{K}_m^*\langle\varsigma_m^{I_{m+1}^i}, \varsigma_{m+1}^i\rangle(\tau_m^{I_{m+1}^i}, \tau_{m+1}^i)}$;
17 draw $J_p \sim \mathsf{Pr}(\{\omega_p^\ell\}_{\ell=1}^N)$;
18 set $Z_p \leftarrow \tau_p^{J_p}$;
19 set $X_p \leftarrow (\llbracket 1, p \rrbracket, Z_p)$;
20 for $m \leftarrow p-1, \dots, 1$ do
21     set $J_m \leftarrow I_m^{J_{m+1}}$;
22     set $X_m \leftarrow \xi_m^{J_{m+1}}$;
23 set $X_{1:p} \leftarrow (X_1, \dots, X_p)$;
24 return $X_{1:p}$
```

Algorithm 2: One transition of PG.

4.3. Particle Gibbs with systematic refreshment

For the graph-oriented applications of interest in the present paper, the naive implementation of the PG sampler will suffer from bad mixing, even though the distribution of interest, η^*, is defined only on the marginal space $\mathcal{T}_p$. Thus, we will modify slightly the standard PG sampler by inserting an intermediate *refreshment step* in between the PG iterations. More specifically, define

$$\mathbf{G}_p(x_{1:p}, dx'_{1:p}) = \delta_{x_p}(dx'_p) \prod_{m=1}^{p-1} \mathbf{R}_m(x'_{m+1}, dx'_m) \quad (x_{1:p} \in \mathcal{X}_{1:p}).$$

Given $x_{1:p}$, drawing $X_{1:p} \sim \mathbf{G}_p(x_{1:p}, dx'_{1:p})$ amounts to setting, deterministically, $X_p = x_p$ and simulating $X_{1:p-1}$ according to the Markovian retrospective dynamics induced by the kernels $\{\mathbf{R}_m\}_{m=1}^{p-1}$. Note that each distribution $\mathbf{G}_p(x_{1:p}, dx'_{1:p})$ depends exclusively on x_p. Describing a standard Gibbs substep for sampling from $\eta_{1:p}$, $\mathbf{G}_p$ is $\eta_{1:p}$-reversible; see, e.g., (Cappé, Moulines and Rydén, 2005, Proposition 6.2.14). Consequently, also the product kernel $\mathbf{P}_p^N \mathbf{G}_p$ is $\eta_{1:p}$-invariant. Unlike standard PG, the McMC sampling scheme that we propose, which is summarised in Algorithm 3, generates (after possible burn-in) a Markov chain $\{X_{1:p}^\ell\}_{\ell \in \mathbb{N}}$ according to

$$X_{1:p}^1 \overset{\mathbf{P}_p^N \mathbf{G}_p}{\longrightarrow} X_{1:p}^2 \overset{\mathbf{P}_p^N \mathbf{G}_p}{\longrightarrow} X_{1:p}^3 \overset{\mathbf{P}_p^N \mathbf{G}_p}{\longrightarrow} X_{1:p}^4 \to \cdots$$

and returns

$$\eta_{1:p}^{N,M} h := \frac{1}{M} \sum_{\ell=1}^{M} h(X_{1:p}^\ell)$$

as an estimator of $\eta_{1:p} h$ for any $\eta_{1:p}$-integrable function h. In addition, as previously, in the case where the objective functions h depends on the argument T_p only, we obtain the estimator

$$\eta^{*N,M} h := \frac{1}{M} \sum_{\ell=1}^{M} h(Z_p^\ell) \tag{4.7}$$

of $\eta^* h$, where each Z_p^ℓ variable is extracted, on Line 2, at iteration $\ell - 1$ of Algorithm 3.

```
Data: a reference trajectory $x_{1:p} \in \mathcal{X}_{1:p}$
Result: a draw $X_{1:p}$ from $\mathbf{P}_p^N \mathbf{G}_p(x_{1:p}, dx'_{1:p})$
1 draw, using Algorithm 2, $X'_{1:p} \sim \mathbf{P}_p^N(x_{1:p}, dx'_{1:p})$;
2 set $X_p = ([\![1, p]\!], Z_p) \leftarrow X'_p$;
3 for $m \leftarrow p-1, \ldots, 1$ do
4     draw $X_m \sim \mathbf{R}_m(X_{m+1}, dx_m)$;
5 set $X_{1:p} \leftarrow (X_1, \ldots, X_p)$;
6 return $X_{1:p}$
```

Algorithm 3: One transition of PG with systematic refreshment.

4.3.1. Particle Gibbs with systematic refreshment vs. standard particle Gibbs

As established by the following theorem, the systematic refreshment step improves indeed the mixing of the algorithm. For any functions g and h in $\mathsf{L}_2(\eta_{1:p})$ we define the *scalar product* $\langle g, h\rangle := \eta_{1:p}(gh)$. Moreover, for all $\eta_{1:p}$-invariant Markov kernels $\mathbf{M}$ on $(\mathcal{X}_{1:p}, \mathcal{X}_{1:p})$ and functions $h \in \mathsf{L}_2(\eta_{1:p})$ such that

$$\sum_{\ell=1}^{\infty} |\langle h, \mathbf{M}^{\ell} h\rangle| < \infty, \tag{4.8}$$

we define the asymptotic variance

$$v(h, \mathbf{M}) := \lim_{M\to\infty} \frac{1}{M} \operatorname{Var}\left(\sum_{\ell=1}^{M} h(X_{1:p}^{\ell})\right), \tag{4.9}$$

where $\{X_{1:p}^{\ell}\}_{\ell=1}^{\infty}$ is a Markov chain with initial distribution $\eta_{1:p}$ and transition kernel $\mathbf{M}$. (The assumption (4.8) can be shown to imply the existence of the limit (4.9)). In the case where the latter Markov chain satisfies a central limit theorem for the objective function h, the corresponding asymptotic variance is given by (4.9). As established by the following result, whose proof relies on asymptotic theory for inhomogeneous Markov chains developed in Maire, Douc and Olsson (2014), the improved mixing implied by systematic refreshment of the trajectories implies a decrease of asymptotic variance w.r.t. standard PG.

Theorem 2. *For all $N \in \mathbb{N}$ and all functions $h^* \in \mathsf{L}_2(\eta^*)$ such that both $\mathbf{P}_p^N \mathbf{G}_p$ and $\mathbf{P}_p^N$ satisfy the summation condition (4.8) with $h := \mathbb{1}_{\mathcal{X}_{1:p-1}} \otimes h^*$, it holds that*

$$v(h, \mathbf{P}_p^N \mathbf{G}_p) \leq v(h, \mathbf{P}_p^N).$$

Proof. See A.2. □

5. Application to decomposable graphical models

In this section we show in more detail how distributions of form (3.1) appear in Bayesian analysis of *graphical models*. To rigorously describe the setting, we shall need some further notations. Let $\{(\mathsf{Y}_m, \mathcal{Y}_m)\}_{m=1}^{p}$, $p \in \mathbb{N}$, be a sequence of measurable spaces and define $\mathsf{Y} := \prod_{m=1}^{p} \mathsf{Y}_m$ and $\mathcal{Y} := \bigotimes_{m=1}^{p} \mathcal{Y}_m$. Let $Y = (Y_1, \ldots, Y_p) : \Omega \to \mathsf{Y}$ be a random element. We consider a fully dominated model where the distribution of Y has a density f on Y with respect to some reference measure $\nu := \bigotimes_{m=1}^{p} \nu_m$ on $(\mathsf{Y}, \mathcal{Y})$, where each ν_m belongs to $\mathbb{M}(\mathcal{Y}_m)$. For some subset $\{a_1, \ldots, a_m\} \subseteq [\![1, p]\!]$ with $a_1 \leq \ldots \leq a_m$, we let $Y_A := (Y_{a_1}, \ldots, Y_{a_m})$ and define $\mathsf{Y}_A = \prod_{\ell=1}^{m} \mathsf{Y}_{a_\ell}$ and $\mathcal{Y}_A = \bigotimes_{\ell=1}^{m} \mathcal{Y}_{a_\ell}$. By slight abuse of notation, we denote by $f(y_A)$ the marginal density of Y_A with respect to $\nu_A := \bigotimes_{\ell=1}^{m} \nu_{a_\ell}$. For disjoint subsets A, B, and S of $[\![1, p]\!]$, we say, following Lauritzen (1996), that Y_A and Y_B are *conditionally independent given* Y_S, denoted $Y_A \perp Y_B \,|\, Y_S$, if it holds that

$$f(y_{A\cup B} \,|\, y_S) = f(y_A \,|\, y_S) f(y_B \,|\, y_S), \quad \text{for all } y_A \in \mathsf{Y}_A,\ y_B \in \mathsf{Y}_B,\ y_S \in \mathsf{Y}_S, \tag{5.1}$$

where the conditional densities are defined as $f(y_A \,|\, y_S) := f(y_{A\cup S})/f(y_S)$. The distribution of Y is said to be *globally Markov* w.r.t. the undirected graph $G = (V, E)$, with $V = [\![1, p]\!]$ and $E \subseteq V \times V$, if for disjoint subsets A, B, and S of V it holds that

$$A \perp_G B \,|\, S \Rightarrow Y_A \perp Y_B \mid Y_S.$$

We call the distribution governed by f a *decomposable model* if it is globally Markov w.r.t. a decomposable graph. Then, by repeated use of (5.1), it is easily shown that the density of a decomposable model satisfies the CSF-type identity

$$f(y) = \frac{\prod_{Q\in\mathcal{Q}(G)} f(y_Q)}{\prod_{S\in\mathcal{S}(G)} f(y_S)}, \tag{5.2}$$

where we, in order to justify the notation y_Q and y_S, by slight abuse of notation identify the cliques Q and the separators S (which are complete graphs) with the corresponding subsets of V.

In the following we consider the dependence structure G as *unknown*, and take a Bayesian approach to the estimation of the same on the basis of a given, fixed data record $y \in \mathsf{Y}$. For this purpose, we assign a prior distribution

$$\pi(dG) := \frac{\varpi^\star(dG)}{\varpi^\star \mathbb{1}_{\mathcal{G}}} \tag{5.3}$$

in $\mathbb{M}_1(\mathcal{G})$ to G, where

$$\varpi^\star(dG) := \varpi|_{\mathcal{G}}(G)\, |dG|$$

and $\varpi : \bar{\mathcal{G}} \to \mathbb{R}_+^*$ is a function satisfying the CSF in Definition 1. For instance, in the completely uninformative case, $\varpi \equiv 1$; in the presence of prior information concerning the maximal clique size of the underlying graph, one may let $\varpi(G) = \mathbb{1}\{\vee_{Q\in\mathcal{Q}(G)} |Q| \le M\}$ for some $M \in \mathbb{N}$ controlling the sizes of the cliques. In both cases, the CSF is immediately checked, see e.g Bornn et al. (2011). We let the same symbol π denote also the corresponding probability function.

In this Bayesian setting, focus is set on the *posterior* distribution $\eta^\star$ of the graph G given the available data y, which is, by Bayes' formula, obtained via (3.1) with $\gamma^\star$ induced by

$$\gamma(G) = \frac{\prod_{Q\in\mathcal{Q}(G)} f(y_Q)\varpi(G_Q)}{\prod_{S\in\mathcal{S}(G)} f(y_S)\varpi(G_S)}, \quad G \in \bar{\mathcal{G}}.$$

The problem of computing the posterior may consequently be perfectly cast into the setting of Section 3.

The model will in general comprise additional unknown parameters collected in a vector θ, which is assumed to belong to some measurable parameter space

$(\Theta_G, \mathcal{P}_G)$ depending on the graph G. We add θ and G to the notation of the likelihood, which is assumed to be of form

$$f(y \,|\, \theta, G) = \frac{\prod_{Q \in \mathcal{Q}(G)} f(y_Q \,|\, \theta_Q)}{\prod_{S \in \mathcal{S}(G)} f(y_S \,|\, \theta_S)}. \tag{5.4}$$

Our Bayesian approach calls for a prior also on $\theta = \{\theta_Q, \theta_S : Q \in \mathcal{Q}(G), S \in \mathcal{S}(G)\}$, and we will always assume that this is *hyper Markov* w.r.t. the underlying graph G. More specifically, we assume that the conditional distribution of θ given G has a density w.r.t. some reference measure, denoted $d\theta$ for simplicity, on $(\Theta, \mathcal{P})$. This density is assumed to be of form

$$\pi(\theta \,|\, G; \vartheta) = \frac{\prod_{Q \in \mathcal{Q}(G)} \pi(\theta_Q; \vartheta_Q)}{\prod_{S \in \mathcal{S}(G)} \pi(\vartheta_S; \vartheta_S)}, \tag{5.5}$$

where $\vartheta = \{\vartheta_Q, \vartheta_S : Q \in \mathcal{Q}(G), S \in \mathcal{S}(G)\}$ is a set of hyperparameters and each factor $\pi(\theta_Q; \vartheta_Q)$ (and $\pi(\vartheta_S; \vartheta_S)$) is a probability density $\pi(\theta_Q; \vartheta_Q) = z(\theta_Q; \vartheta_Q)/I(\vartheta_Q)$ with $I(\vartheta_Q) = \int z(\theta_Q; \vartheta_Q)\, d\theta_Q$ being a normalising constant.

In the case where each $\pi(\theta_Q; \vartheta_Q)$ is a *conjugate prior* for the corresponding likelihood factor $f(y_Q \,|\, \theta_Q)$ it holds that

$$f(y_Q \,|\, \theta_Q) z(\theta_Q; \vartheta_Q) = c^{|Q|} z(\theta_Q; \vartheta'_Q(y_Q)), \tag{5.6}$$

for some updated hyperparameter $\vartheta'_Q(y_Q)$ and some constant $c > 0$. If the normalising constants $I(\vartheta_Q)$ are tractable, we may marginalise out the parameter and consider directly the posterior of G given data y. Indeed, since for all hyperparameters,

$$\int \frac{\prod_{Q \in \mathcal{Q}(G)} z(\theta_Q; \vartheta_Q)}{\prod_{S \in \mathcal{S}(G)} z(\vartheta_S; \vartheta_S)}\, d\theta = \frac{\prod_{Q \in \mathcal{Q}(G)} I(\vartheta_Q)}{\prod_{S \in \mathcal{S}(G)} I(\vartheta_S)},$$

the marginalised likelihood is obtained as

$$\begin{aligned} f(y \,|\, G) &= \int \frac{\prod_{Q \in \mathcal{Q}(G)} f(y_Q \,|\, \theta_Q) \pi(\theta_Q; \vartheta_Q)}{\prod_{S \in \mathcal{S}(G)} f(y_S \,|\, \theta_S) \pi(\theta_S; \vartheta_S)}\, d\theta \\ &= c^p \int \frac{\prod_{Q \in \mathcal{Q}(G)} z(\theta_Q; \vartheta'_Q(y_Q))/I(\vartheta_Q)}{\prod_{S \in \mathcal{S}(G)} z(\theta_S; \vartheta'_S(y_S))/I(\vartheta_S)}\, d\theta \\ &= c^p \frac{\prod_{Q \in \mathcal{Q}(G)} I(\vartheta'_Q(y_Q))/I(\vartheta_Q)}{\prod_{S \in \mathcal{S}(G)} I(\vartheta'_S(y_S))/I(\vartheta_S)}, \end{aligned}$$

Thus, by Bayes' formula, the marginal posterior $\eta^\star$ of G given the available data y can be expressed by (3.1) with $\gamma^\star$ induced by

$$\gamma(G) = \frac{\prod_{Q \in \mathcal{Q}(G)} \varpi(G_Q) I(\vartheta'_Q(y_Q))/I(\vartheta_Q)}{\prod_{S \in \mathcal{S}(G)} \varpi(G_S) I(\vartheta'_S(y_S))/I(\vartheta_S)}, \quad G \in \bar{\mathcal{G}}. \tag{5.7}$$

Example 1 (discrete log-linear models)**.** *Let V be a set of p criteria defining a contingency table. Without loss of generality, we let $V = [\![1, p]\!]$ and denote the table by $\mathsf{I} = \mathsf{I}_1 \times \cdots \times \mathsf{I}_p$, where each I_m is a finite set. An element $i \in \mathsf{I}$ is referred to as a* cell. *In this setting, $I = (I_1, \ldots, I_p)$ is a discrete-valued random vector whose distribution θ is assumed to be globally Markov w.r.t. some decomposable graph $G = (V, E)$ with $E \subseteq V \times V$, i.e.,*

$$\theta(i) = \mathbb{P}(I = i) = \frac{\prod_{Q \in \mathcal{Q}(G)} \theta(i_Q)}{\prod_{S \in \mathcal{S}(G)} \theta(i_S)}, \quad i \in \mathsf{I}. \tag{5.8}$$

The vector I may, e.g., characterise a randomly selected individual w.r.t. the table I. Given G, the parameter space of the model is determined by the clique and separator marginal probability tables $\theta(i_Q)$ and $\theta(i_S)$; more specifically,

$$\Theta_G = \left\{ \theta(i_Q) \in (0,1), \theta(i_S) \in (0,1) : i \in I, Q \in \mathcal{Q}(G), S \in \mathcal{S}(G), \sum_{i \in \mathsf{I}} \theta(i) = 1 \right\}.$$

Let Y be a collection of $n \in \mathbb{N}$ i.i.d. observations from the model; e.g., Y is an $n \times p$ matrix where each row corresponds to an observation of I. Then also Y forms a DGM with state space $\mathsf{Y} = \mathsf{I}_1^n \times \cdots \times \mathsf{I}_p^n$ and probability function $f(y \,|\, \theta, G)$ given by (5.4) with

$$f(y_Q \,|\, \theta_Q) = \prod_{i_Q \in \mathsf{I}_Q} \theta(i_Q)^{n(i_Q)}$$

(and similarly for $f(y_S \,|\, \theta_S)$), where $\mathsf{I}_Q = \prod_{m \in Q} \mathsf{I}_m$, $\theta_Q := \{\theta(i_Q)\}_{i_Q \in \mathsf{I}_Q}$, and $n(i_Q)$ counts the number of elements of y_Q belonging to the marginal cell i_Q.

The problem of estimating the dependence structure G is complicated further by the fact that also the probabilities θ are unknown in general. When assigning a prior $\pi(\theta \,|\, G; \vartheta)$ to the latter conditionally on the former, we follow Dawid and Lauritzen (1993) and let the prior $\pi(\theta_Q; \vartheta_Q)$ of each θ_Q be a standard Dirichlet distribution, $\operatorname{Dir}(\vartheta_Q)$, *where $\vartheta_Q = \{\vartheta_Q(i_Q)\}_{i_Q \in \mathsf{I}_Q}$ are hyper parameters often referred to as* pseudo counts. *Under the assumption that the collection $\{\pi(\theta_Q; \vartheta_Q)\}_{Q \in \mathcal{Q}(G)}$ is pairwise* hyper consistent *in the sense that for all $(Q, Q') \in \mathcal{Q}(G)^2$ such that $Q \cap Q' \neq \varnothing$, $\pi(\theta_Q; \vartheta_Q)$ and $\pi(\theta_{Q'}; \vartheta_{Q'})$ induce the same law on $\theta_{Q \cap Q'}$, which in this case is implied by the condition*

$$\begin{aligned}\vartheta_Q(i_{Q \cap Q'}) &:= \sum_{j_Q \in \mathsf{I}_Q : j_{Q \cap Q'} = i_{Q \cap Q'}} \vartheta_Q(j_Q) \\ &= \sum_{j_{Q'} \in \mathsf{I}_{Q'} : j_{Q \cap Q'} = i_{Q \cap Q'}} \vartheta_{Q'}(j_{Q'}) = \vartheta_{Q'}(i_{Q \cap Q'}),\end{aligned}$$

(Dawid and Lauritzen, 1993, Theorem 3.9) implies the existence of a unique hyper Dirichlet *law of the form (5.5). Thus, $z(\theta_Q; \vartheta_Q) = \prod_{i_Q \in \mathsf{I}_Q} \theta(i_Q)^{\vartheta_Q(i_Q)}$, $I(\vartheta_Q) = B(\vartheta_Q) := \prod_{i_Q \in \mathsf{I}_Q} \Gamma(\vartheta_Q(i_Q)) / \Gamma(\sum_{i_Q \in \mathsf{I}_Q} \vartheta_Q(i_Q))$ (the beta function), and the conjugacy (5.6) holds with $c = 1$ and $\vartheta'_Q(y_Q) = \{\vartheta'_Q(i_Q)(y_Q)\}_{i_Q \in \mathsf{I}_Q}$,*

where $\vartheta'_Q(i_Q)(y_Q) = \vartheta_Q(i_Q) + n(i_Q)$*. Then, putting a prior of form* (5.3) *on the graph,* (5.7) *implies that the marginal posterior of* G *given data* y *is obtained through* (3.1) *with* $\gamma^\star$ *induced by*

$$\gamma(G) = \frac{\prod_{Q \in \mathcal{Q}(G)} \varpi(Q) B(\vartheta'_Q(y_Q))/B(\vartheta_Q)}{\prod_{S \in \mathcal{S}(G)} \varpi(S) B(\vartheta'_S(y_S))/B(\vartheta_S)}, \quad G \in \bar{\mathcal{G}}.$$

Example 2 (Gaussian graphical models)**.** *A* p*-dimensional Gaussian random vector forms a* Gaussian graphical model *if it is globally Markov w.r.t. some graph* $G = (V, E)$ *with* $V = [\![1, p]\!]$ *and* $E \subseteq V \times V$*. In the following we assume that the model has zero mean (for simplicity) and is, given* G*, parameterised by its* precision *(inverse covariance)* matrix *belonging to the set*

$$\Theta_G = \{\theta \in \mathsf{M}_p^+ : \theta_{ij} = 0 \textit{ for all } (i,j) \notin E\},$$

where M_p^+ *denotes the space of* $p \times p$ *positive definite matrices. It is well known that in this model, a zero in the precision matrix,* $\theta_{ij} = 0$*, is equivalent to conditional independence of the* i*th and* j*th variables given the rest of the variables, see Speed and Kiiveri (1986). In addition, when* G *is decomposable, a model with* $\theta \in \Theta_G$ *is globally Markov w.r.t.* G*. In the following, for any matrix* $p \times p$ *matrix* M *and* $A \subseteq [\![1, p]\!]$*, denote by* M_A *the* $|A| \times |A|$ *matrix obtained by extracting the elements* $(M_{ij})_{(i,j) \in A^2}$ *from* M*. Suppose that* G *is decomposable and that are we have access to* n *independent observations from the model. The observations are stored in an* $n \times p$ *data matrix* Y*, whose likelihood* $f(y \,|\, \theta, G)$ *is, as a consequence of the global Markov property, given by* (5.4) *with*

$$f(y_Q \,|\, \theta_Q) = \frac{1}{(2\pi)^{|Q|}} |\theta_Q|^{n/2} \exp\left(-\mathrm{tr}(\theta_Q s_Q)/2\right),$$

(and similarly for $f(y_S \,|\, \theta_S)$*) where* $s = y^\intercal y$*,* $|Q|$ *is the cardinality of* Q*, and* $|\theta_Q|$ *is the determinant of* θ_Q*.*

For Bayesian inference on θ*, we follow Dawid and Lauritzen (1993) and furnish, given* G*,* θ *with a* hyper Wishart *prior* $\pi(\theta \,|\, G)$ *of form* (5.5)*, with each* $\pi(\theta_Q; \vartheta_Q)$ *being proportional to*

$$z(\theta_Q; \vartheta_Q) = |\theta_Q|^{\beta_Q} \exp\left(-\mathrm{tr}(\theta_Q v_Q)/2\right),$$

where $\beta_Q := (\delta + |Q| - 1)/2$*,* $\vartheta_Q = (\delta, v_Q)$ *with* $v \in \mathsf{M}_p^+$ *being a scale matrix and* $\delta > \vee_{Q \in \mathcal{Q}(G)} |Q| - 1$ *the number of degrees of freedom, and normalising constant*

$$I(\vartheta_Q) = 2^{\delta|Q|/2} \frac{\Gamma_{|Q|}(\beta_Q)}{|v_Q|^{\beta_Q}},$$

where Γ_p *denotes the multivariate gamma function. Since all hyperparameters* ϑ_Q *are extracted from the* same *scale matrix* v*, the collection of priors* $\{\pi(\theta_Q; \vartheta_Q)\}_{Q \in \mathcal{Q}(G)}$ *is automatically pairwise hyper consistent, and the existence of the (unique) hyper Wishart prior is guaranteed by Dawid and Lauritzen (1993, Theorem 3.9). As*

$$f(y_Q \,|\, \theta_Q) z(\theta_Q; \vartheta_Q) = \frac{1}{(2\pi)^{|Q|}} |\theta_Q|^{\alpha_Q} \exp\left(-\mathrm{tr}\{\theta_Q(s_Q + v_Q)\}/2\right),$$

where $\alpha_Q := (\delta + n + |Q| - 1)/2$, we conclude that the conjugacy condition (5.6) holds for $c = 1/(2\pi)$ and $\vartheta'_Q(y_Q) = (\delta'_Q, v'_Q)$ with $\delta'_Q = \delta + n$ and $v'_Q = s_Q + v_Q$ (and similarly for factors corresponding to separators). Consequently, assigning also a prior of form (5.3) to the graph, (5.7) implies that the marginal posterior of G given data y is, in this case, obtained through (3.1) with $\gamma^\star$ induced by

$$\gamma(G) = \frac{\prod_{Q \in \mathcal{Q}(G)} \varpi(Q)\rho(Q)}{\prod_{S \in \mathcal{S}(G)} \varpi(S)\rho(S)}, \quad G \in \bar{\mathcal{G}},$$

with

$$\rho(Q) := \frac{|v_Q|^{\alpha_Q}}{|v_Q + s_Q|^{\beta_Q}} \frac{\Gamma_{|Q|}(\alpha_Q)}{\Gamma_{|Q|}(\beta_Q)},$$

and $\rho(S)$ defined analogously.

6. Numerical study

In this section we investigate numerically the performance of the suggested PG algorithm for three example datasets. The first example treats the classical Czech autoworkers dataset found in e.g. Edwards and Havránek (1985). The second one considers simulated data generated from the discrete $p = 15$ nodes structure, introduced in Jones et al. (2005). The third example investigates a continuous dataset simulated from a Gaussian DGM of dimensionality $p = 50$ with a time-varying dependence structure.

The proposal and backward kernels $\{\mathbf{K}_m\}_{m=1}^{p-1}$ and $\{\mathbf{R}_m\}_{m=1}^{p-1}$ are given by the CTA and its reversed version, respectively, provided in Section 3 and 4 of the companion paper Olsson, Pavlenko and Rios (2018). The transition kernels $\{\mathbf{\Sigma}_m\}_{m=1}^{p-1}$ introduced in (3.3) are defined by selecting s^* uniformly at random from the set $\{s \in [\![1, p]\!] : \min_{s' \in S_m} |s - s'| \le \delta\}$ as suggested in Section 3.

We assign the uniform prior for the graph structure in each of the examples. The estimated graph posteriors are summarized in terms of marginal edge distributions presented as heatmaps, where the probability of an edge (a, b) is estimated according to (4.7) by letting $h(Z) = \mathbb{1}_{(a,b) \in E}$, where E here denotes the edge set for $g(Z)$.

We study the number of edges in the graph (*graph size*) in order to evaluate the mixing properties. Since in many practical situations the aim is to select one particular model that best represents the underlying dependence structure, we also present the maximum aposteriori (MAP) graph for each of the examples.

All the experiments were performed on the Tegnér cluster at PDC having one intel 2×12 core Intel E5-2690v3 Haswell processor per node. The Python program used to generate the examples is part of the *trilearn* library available at https://github.com/felixleopoldo/trilearn.

6.1. *Czech autoworkers data*

This dataset, previously analyzed many times in the literature comprises 1841 men cross-classified with respect to six potential risk factors for coronary thrombosis randomly selected from a population of Czech autoworkers: (Y_1) smoking, (Y_2) strenuous mental work, (Y_3) strenuous physical work, (Y_4) systolic blood pressure, (Y_5) ratio of beta and alpha lipoproteins and (Y_6) family anamnesis of coronary heart disease. In absence of any prior information we assume that the data are generated from the discrete log-linear DGM presented in Section 5. Each of the 64 cells in the contingency table is assigned a pseudo count of 1/64, which in turn induces hyper parameters in the conjugate prior Hyp-Dir(ϑ) defined by $\{\pi(\theta_Q;\vartheta_Q)\}_{Q\in\mathcal{Q}(G)}$ where $\vartheta_Q = \{\vartheta_{Q(i_Q)}\}_{i\in\mathsf{I}_Q}$ and $\vartheta_{Q(i_Q)} = |\mathsf{I}_{V\setminus Q}|/64$. This type of low dimensional model is suitable for evaluation purposes since it is possible to exactly compute the posterior distribution. Specifically, the total number of decomposable graphs with six nodes is equal to 18154, allowing for full computation of the posterior distribution.

All the estimators are based on $N = 100$ particles and averaging is performed over $M = 10000$ PG-runs according to equation (4.7). Due to the absence of a time-dependent dynamic, we set the bandwidth δ to p.

The heatmaps for the exact and estimated posterior distributions are displayed in Figure 1, along with the estimated auto-correlation. A visual inspection of the marginal edge probabilities in the heatmaps indicates a good agreements between the distributions. From the fast decay from one to zero in the auto-correlation plot we deduce that the PG-sampler exhibit very good mixing properties for this problem.

Table 1 summarizes the edge sets for the top five graphs on both the exact and estimated posterior distribution along with their corresponding probabilities. It is important to note that the top five graphs are exactly the same for these two distributions and the estimated probabilities are in a good agreement with

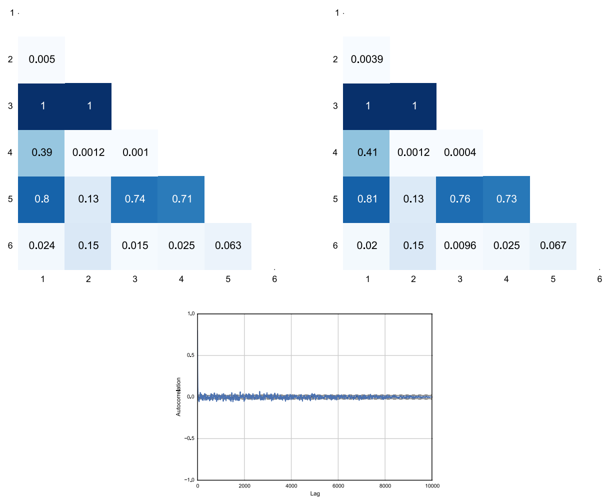

Fig 1. *True edge heatmap (left), estimated heatmap (middle) and auto-correlation of the number of edges in the trajectory of graph (right).*

the exact ones. Our findings are also consistent with the results obtained by Massam, Liu and Dobra (2009) and Madigan and Raftery (1994). Specifically, our top highest posterior probability graphs are the same as those identified by Massam, Liu and Dobra (2009), see Table 2, case $\alpha = 1.0$ in that paper.

TABLE 1
The estimated graph probabilities are compared to the true posterior probabilities for the five graphs with the highest posterior probabilities.

Edge set	Exact	Estimated
$(1,3),(1,5),(2,3),(3,5),(4,5)$	0.248	0.263
$(1,3),(1,4),(1,5),(2,3),(3,5),(4,5)$	0.104	0.115
$(1,3),(1,4),(1,5),(2,3),(3,5)$	0.101	0.103
$(1,3),(2,3),(2,5),(4,5)$	0.059	0.062
$(1,3),(1,5),(2,3),(2,6),(3,5),(4,5)$	0.051	0.051

Finally, after evaluating a range of different combinations, we conclude that our results obtained in this example appear to be insensitive to the choice of CTA parameters α and β for this small scale problem.

6.2. Discrete data with $p = 15$

In this example we study a discrete log-linear DGM with $p = 15$ nodes and the dependence structure displayed in Figure 2, presented in (Jones et al., 2005, Figure 1). The parameters were selected to satisfy (5.8) thereby ensuring that the distribution Θ_G specified in Example 1 will be Markov with respect to G. Analogously to the previous example, the we use the Hyp-Dir(ϑ) prior and assign to each cells in the contingency table a pseudo count of $1/2^{15}$. Due to the absence of any time interpretation of the model, the bandwidth parameter δ is selected as p. We used the CTA parameters $\alpha = 0.2$ and $\beta = 0.8$, obtained as the parameter setting giving best mixing properties within all the possible combination on the grid $\alpha, \beta = 0.2, 0.5, 0.8$.

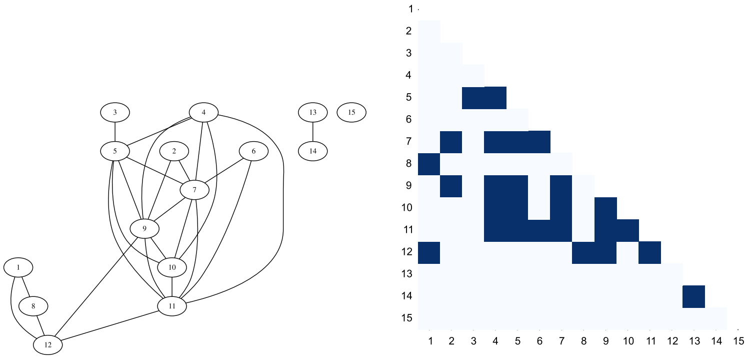

FIG 2. *The true underlying decomposable graph on $p = 15$ nodes along with its adjacency matrix.*

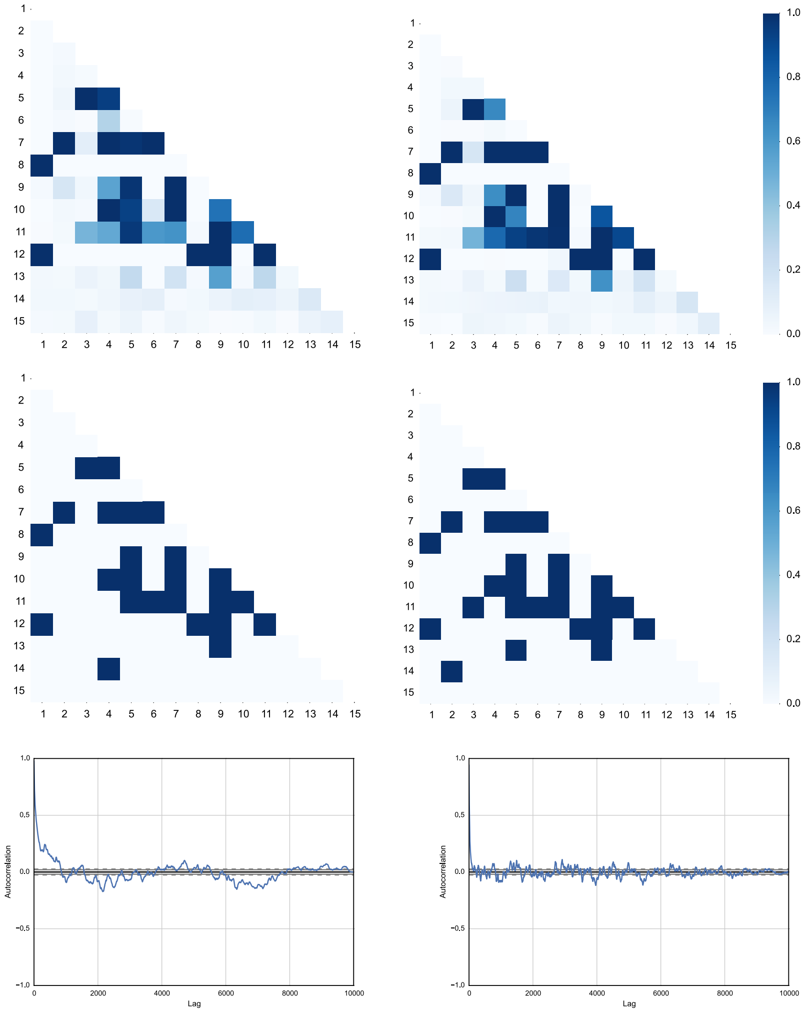

FIG 3. *Estimation of the graph posterior for the log-linear model with* $p = 15$ *and* $n = 100$. *The CTA parameters are* $\alpha = 0.2$, $\beta = 0.8$ *and* $\delta = 15$. *The number of McMC sweeps* M *is set to* 10000. *The left and right panel correspond to* $N = 20$ *and* $N = 100$, *respectively. For both panels from top to bottom, the first figure presents the estimated edge heatmap, the estimated MAP graph and estimated auto-correlation of the number of edges in the graph.*

To evaluate how the estimation accuracy is affected by the number of particles, we sampled $n = 1000$ data vectors and estimated both the graph posterior and the auto-correlation function with $N = 20$ and $N = 100$. By comparing the true underlying graph in Figure 2 with the heatmaps and the MAP in Figure 3, we observe that increasing N from 20 to 100 gives a slightly better agreement with the true adjacency matrix. This effect can be further explained by the behavior of the estimated auto-correlation function; by increasing N a clear reduction of the auto-correlation can be noted. Qualitatively we conclude that the mobility of the PG-sampler is improved when increasing N.

6.3. Continuous data with temporal dependence

In this example, we study a Gaussian DGM where a temporal interpretation of the underlying dependence structure is suitable. The graph structure along with its adjacency matrix are displayed in Figure 4 and can be interpreted as an AR-process with lag varying between 1 and 5. The model to be considered is represented by a Gaussian distribution with zero mean and covariance matrix θ^{-1} defined as

$$(\theta^{-1})_{ij} = \begin{cases} \sigma^2, & \text{if } i = j \\ \rho\sigma^2, & \text{if } (i,j) \in G \end{cases}$$

and $\theta_{ij} = 0$ if $(i,j) \notin G$. This is a modification of the second order intra-class structure considered in Green and Thomas (2013), where the bandwidth is varying. We have sampled $n = 100$ data vectors from this model where the variance σ^2, and correlation coefficient ρ, were set to 1.0 and 0.9, respectively.

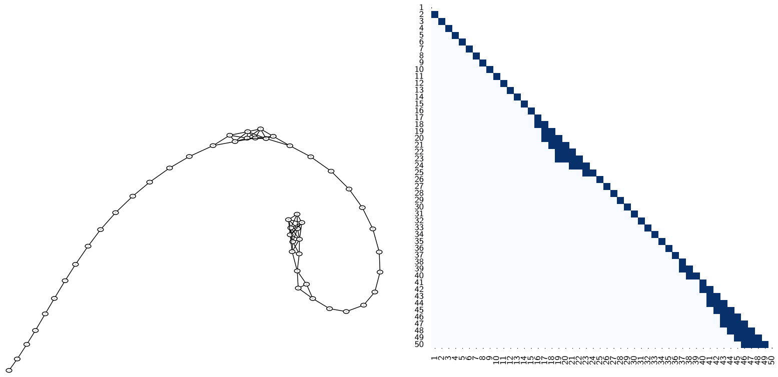

Fig 4. *The true underlying decomposable graph on $p = 50$ nodes along with its adjacency matrix.*

Following Example 2, θ is assigned a hyper Wishart prior, where for each clique Q the degrees of freedom is set to be equal to $p = 50$, and the scale matrix is set to be the identity matrix of dimension $|Q|$.

In this example, 10 PG trajectories of length $M = 10000$ were sampled, of which the first 3000 samples were removed as burn-in period. The temporal interpretation of the structure of this graph is particularly suited for investigating the role of δ as a tuning parameter. The heatmaps and MAP graph estimates most similar to the true graph for the configurations $\delta = 5$ and $\delta = 50$ are

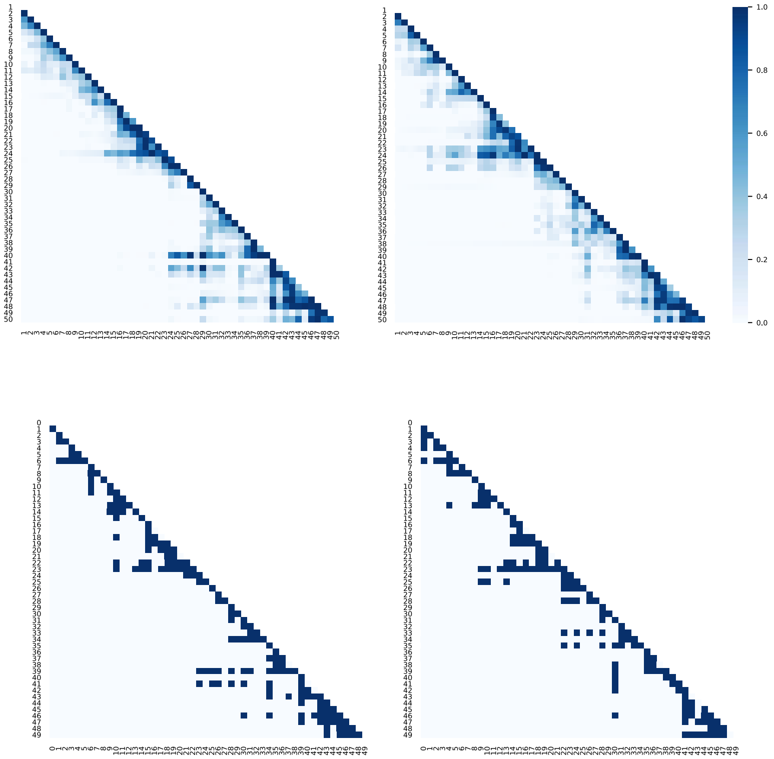

Fig 5. *Heatmaps (top row) and MAP graph estimates (bottom row) for the PG-sampler with $\delta = 5$ (left panel) and $\delta = 50$ (right panel).*

diplayed in Figure 5. By comparing the two heatmaps one can notice that the dependence structure can be better captured by selecting a value of δ which corresponds to the maximal bandwidth size of 5 for the true graph. In addition, by letting $\alpha = 0.5$ and $\beta = 0.8$ we are able to express a priority for connected graphs, we obtain a heatmap pattern which better mimics the true one. This effect is also reflected by the log-likelihood trajectories in the bottom row of Figure 6. The graph size auto-correlation (after burn-in) shown in the middle row of Figure 6, decays slightly faster by a smaller δ and the mobility of the size trajectory, top row of Figure 6 is improved in this example.

6.4. Comparison to the Metropolis-Hastings algorithm

We have compared the PG-sampler with the Metropolis-Hastings (MH) algorithm proposed in Green and Thomas (2013) for the Gaussian example in Section 6.3. Following the suggestions from that paper we randomise the junction tree every λ iteration, we executed their algorithm for both $\lambda = 100$ and $\lambda = 1000$. We can confirm the claim that a more frequent junction tree randomi-

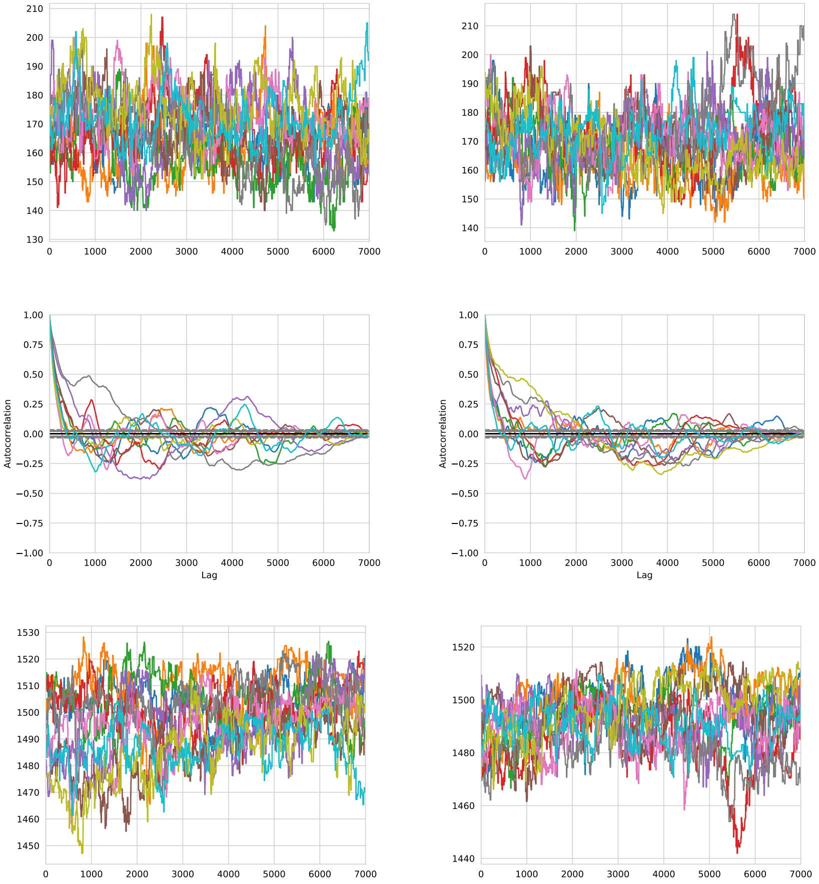

Fig 6. *Ten size trajectories (top row), estimated size auto-correlations (middle row) and the graph log-likelihoods (bottom row) for the PG-sampler with $\delta = 5$ (left panel) and $\delta = 50$ (right panel).*

sation has an improving effect on recovering the underlying model. Therefore, results only for $\lambda = 100$ are demonstrated here.

The main advantage of the PG-sampler as compared to the MH-sampler is its mixing properties. Figure 8 shows the 10 trajectories of the MH-sampler after 350000 iterations (in total 500000 graphs were sampled), out of which 4 trajectories seem to have reached stationarity; see the size- and log-likelihood trajectories displayed in green, orange, gray and brown. In the middle panel of Figure 6 and Figure 8, it is seen that the estimated auto-correlation of the MH-sampler is substantially stronger than that for the PG-sampler for both choices of δ, being on average about 20000 for the MH-sampler as compared to about 500 for the PG-samples with $\delta = 5$. Figure 7 shows the heatmap and MAP estimate corresponding to the green trajectories in Figure 8.

On the other hand, the MH-sampler is superior in speed, which is at cost of

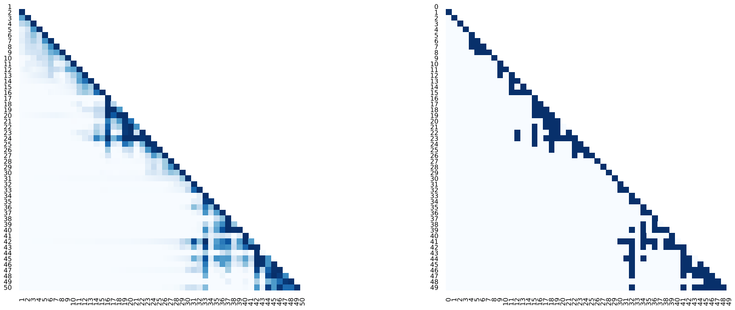

FIG 7. *Heatmaps (left panel) and MAP graph estimates (right panel) for the MH-sampler with junction tree randomization at every $\lambda = 100$ iteration.*

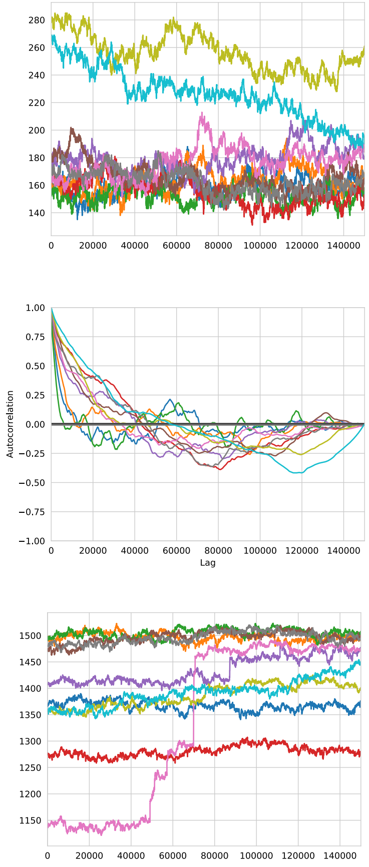

FIG 8. *Ten size trajectories (top panel), estimated size auto-correlations (middle panel) and the graph log-likelihoods (bottom panel) for the MH-sampler with junction tree randomization at each $\lambda = 100$ iteration.*

the slower mixing seemingly inherited by the local moves. Using the Java implementation from Green and Thomas (2013), the MH-sampler with randomising interval $\lambda = 100$ were able to sample about 20000 samples per second, while each sample took about 3 seconds for the PG-sampler. Note that the implementation by Green and Thomas (2013) considers the intra-class model introduced Section 6.3 so that θ is defined by the two parameters σ^2 and ρ, and two independent priors are assigned for these instead of the hyper Wishart distribution as presented in Example 2. However, a gain in sample time for the MH-sampler is expected since, in each PG iteration, the conditional SMC procedure generates $p(N-1)$ junction trees with $\mathcal{O}(p^2)$ internal nodes per PG sample. Also, when a new junction tree is proposed in the SMC algorithm, the previous junction tree, which it stems from is copied since it could potentially be an ancestor for other trees as well due to the re-sampling step. As scope for future research, investigating new data structures for junction trees which are tailored to sequential sampling is of great interest. We also expect that the speed of the PG-sampler could be improved by, for example parallelizing the SMC-updates and by improved caching strategies.

Appendix A

A.1. Graph theory

Given a set $V = \{a_1, \dots, a_p\}$ of $p \in \mathbb{N}$ distinct elements, an *undirected graph* G with *nodes* V is specified by a set of *edges* $E \subseteq V \times V$, and we write $G = (V, E)$. In addition, we say that $G' = (V', E')$ is a *subgraph* of G if $V' \subseteq V$ and $E' \subset E$. For any pair (a, b) of nodes in G, a *path* from a to b, denoted by $a \sim b$, is a sequence $\{a_{n_k}\}_{k=1}^{\ell+1}$, with $\ell \in [\![1, p-1]\!]$, of distinct nodes such that $a_{n_1} = a$, $a_{n_{\ell+1}} = b$, and $(a_{n_k}, a_{n_{k+1}}) \in E$ for all $k \in [\![1, \ell]\!]$. Here ℓ is called the *length* of the path. A graph is called a *tree* if there is a unique path between any pair of nodes. A graph is *connected* when there is a path between every pair of nodes, and a *subtree* is a connected subgraph of a tree. Let $G = (V, E)$ be a graph and A, B, and S subsets of V; then S is said to *separate* A from B if for all $a \in A$ and $b \in B$, every path $a \sim b$ intersects S. This is denoted by $A \perp_G B \mid S$. A graph is *complete* if $E = V \times V$. Let $V' \subseteq V$; then the *induced subgraph* $G[V'] = (V', E')$ is the subgraph of G with nodes V' and edges E' given by the subset of edges in E having both endpoints in V'. We write $G' = (V', E') \leq G = (V, E)$ to indicate that $G' = G[V']$. A subset $W \subseteq V$ is a *complete set* if it induces a complete subgraph. A complete subgraph is called a *clique* if it is not an induced subgraph of any other complete subgraph. We denote by $\mathcal{Q}(G)$ the family of cliques formed by a graph G.[2] A triple (A, B, S) of disjoint subsets of V is a *decomposition* of $G = (V, E)$ if $A \cup B \cup S = V$, $A \neq \varnothing$, $B \neq \varnothing$, S is complete, and it holds that $A \perp_G B \mid S$.

[2] We use calligraphy uppercase to denote families of graphs, or, more generally, families of sets (as a graph is, given the nodes, specified through the edge set). Consequently, calligraphy uppercase will also used to denote σ-fields.

A.2. Proofs and lemmas

The following lemma, proved in a slightly different form in Thomas and Green (2009), establishes that each extension (3.2) has the correct marginal w.r.t. the graph, i.e., that $\eta^\star\langle U\rangle$ is the distribution of $g(\tau)$ when $\tau \sim \eta^*\langle U\rangle$.

Lemma 3. *For all $U \subseteq V$ and $h \in \mathbb{F}(\mathcal{G})$,*

$$\mathbb{E}_{\eta^*\langle U\rangle}\left[h \circ g(\tau)\right] = \eta^\star\langle U\rangle h,$$

where $\eta^*\langle U\rangle$ is defined in (3.2)*.*

Proof. Since

$$\begin{aligned}\gamma^*\langle U\rangle \mathbb{1}_{\mathcal{T}_U} = \sum_{T \in \mathcal{T}_U} \gamma^*\langle U\rangle(T) &= \sum_{G \in \mathcal{G}_U} \sum_{T \in \mathcal{T}(G)} \frac{\gamma^\star\langle U\rangle \circ g(T)}{\mu \circ g(T)} \\ &= \sum_{G \in \mathcal{G}_U} \mu(G) \frac{\gamma^\star\langle U\rangle(G)}{\mu(G)} = \gamma^\star\langle U\rangle \mathbb{1}_{\mathcal{G}_U},\end{aligned}$$

it holds that

$$\eta^*\langle U\rangle(dT) = \frac{\eta^\star\langle U\rangle \circ g(T)}{\mu \circ g(T)}\, |dT|. \tag{A.1}$$

Now, let $h \in \mathbb{F}(\mathcal{G}_U)$; then by a similar computation,

$$\begin{aligned}\mathbb{E}_{\eta^*\langle U\rangle}\left[h \circ g(T)\right] &= \sum_{G \in \mathcal{G}_U} \sum_{T \in \mathcal{T}(G)} h \circ g(T) \frac{\eta^\star\langle U\rangle \circ g(T)}{\mu \circ g(T)} \\ &= \sum_{G \in \mathcal{G}_U} \mu(G) h(G) \frac{\eta^\star\langle U\rangle(G)}{\mu(G)} = \eta^\star\langle U\rangle h,\end{aligned}$$

which completes the proof. □

Proof of Theorem 2. First, as established in (Chopin and Singh, 2015, Proposition 8), the standard PG kernel $\mathbf{P}_p^N$ is $\eta_{1:p}$-reversible. As mentioned above, the kernel $\mathbf{G}_p$ is straightforwardly $\eta_{1:p}$-reversible as a standard Gibbs substep. Moreover, for all $x_{1:p} \in \mathcal{X}_{1:p}$, $\mathbf{G}_p(x_{1:p}, \mathcal{X}_{1:p-1} \times \{x_p\}) = 1$ and $\mathbf{G}_p$ dominates trivially the Dirac mass on the *off-diagonal*, in the sense that for all $A \in \mathcal{X}_{1:p}$ and $x_{1:p} \in \mathcal{X}_{1:p}$, $\mathbf{G}_p(x_{1:p}, A \setminus x_{1:p}) \geq \delta_{x_{1:p}}(A \setminus x_{1:p}) = 0$. The assumptions of (Maire, Douc and Olsson, 2014, Lemma 18) are thus fulfilled, and the proof is concluded through application of the latter. □

Acknowledgements

We are grateful to the editor and two anonymous referees for the very helpful and constructive comments, which led to substantial improvement of the paper. We should also like to thank the PDC Center for High Performance Computing at the KTH Royal Institute of Technology, for providing computing resources for our numerical study. We are also thankful to Jim Holmström for sharing his Python knowledge with us.

References

Andrieu, C., Doucet, A. and Holenstein, R. (2010). Particle Markov chain Monte Carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **72** 269–342. MR2758115

Bornn, L., Caron, F. et al. (2011). Bayesian clustering in decomposable graphs. *Bayesian Analysis* **6** 829–846. MR2869966

Cappé, O., Moulines, E. and Rydén, T. (2005). *Inference in hidden Markov models*. Springer New York. MR2159833

Chopin, N. and Singh, S. S. (2015). On particle Gibbs sampling. *Bernoulli* **21** 1855–1883. MR3352064

Dawid, A. P. and Lauritzen, S. L. (1993). Hyper Markov laws in the statistical analysis of decomposable graphical models. *The Annals of Statistics* **21** 1272–1317. MR1241267

Del Moral, P., Doucet, A. and Jasra, A. (2006). Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)* **68** 411–436. MR2278333

Dellaportas, P. and Forster, J. J. (1999). Markov chain Monte Carlo model determination for hierarchical and graphical log-linear models. *Biometrika* **86** 615–633. MR1723782

Diestel, R. (2005). *Graph theory (Graduate texts in mathematics)* **173**. Springer Heidelberg. MR2159259

Edwards, D. and Havránek, T. (1985). A fast procedure for model search in multidimensional contingency tables. *Biometrika* **72** 339–351. MR0801773

Elmasri, M. (2017a). On decomposable random graphs. *ArXiv e-prints*.

Elmasri, M. (2017b). Sub-clustering in decomposable graphs and size-varying junction trees. *ArXiv e-prints*.

Giudici, P. and Green, P. J. (1999). Decomposable graphical Gaussian model determination. *Biometrika* **86** 785–801. MR1741977

Green, P. J. and Thomas, A. (2013). Sampling decomposable graphs using a Markov chain on junction trees. *Biometrika* **100** 91–110. MR3034326

Green, P. J. and Thomas, A. (2017). A structural Markov property for decomposable graph laws that allows control of clique intersections. *Biometrika* **105** 19–29. MR3768862

Jones, B., Carvalho, C., Dobra, A., Hans, C., Carter, C. and West, M. (2005). Experiments in stochastic computation for high-dimensional graphical models. *Statistical Science* **20** 388–400. MR2210226

Lauritzen, S. L. (1996). *Graphical Models*. Oxford University Press. MR1419991

Madigan, D. and Raftery, A. E. (1994). Model selection and accounting for model uncertainty in graphical models using Occam's window. *Journal of the American Statistical Association* **89** 1535–1546.

Maire, F., Douc, R. and Olsson, J. (2014). Comparison of asymptotic variances of inhomogeneous Markov chains with application to Markov chain Monte Carlo methods. *The Annals of Statistics* **42** 1483–1510. MR3262458

Markenzon, L., Vernet, O. and Araujo, L. (2008). Two methods for the generation of chordal graphs. *Annals of Operations Research* **157** 47–60. MR2360270

Massam, H., Liu, J. and Dobra, A. (2009). A conjugate prior for discrete hierarchical log-linear models. *The Annals of Statistics* **37** 3431–3467. MR2549565

Olsson, J., Pavlenko, T. and Rios, F. L. (2018). Sequential sampling of junction trees for decomposable graphs. *ArXiv e-prints*.

Speed, T. P. and Kiiveri, H. T. (1986). Gaussian Markov distributions over finite graphs. *The Annals of Statistics* **14** 138–150. MR0829559

Stingo, F. and Marchetti, G. M. (2015). Efficient local updates for undirected graphical models. *Statistics and Computing* **25** 159–171. MR3304918

Thomas, A. and Green, P. J. (2009). Enumerating the junction trees of a decomposable graph. *Journal of Computational and Graphical Statistics* **18** 930–940. MR2598034