Data Min Knowl Disc (2008) 17:431–456
DOI 10.1007/s10618-008-0099-9

Parallell interacting MCMC for learning of topologies of graphical models

Jukka Corander · Magnus Ekdahl · Timo Koski

Received: 4 May 2007 / Accepted: 1 May 2008 / Published online: 16 May 2008

Abstract Automated statistical learning of graphical models from data has attained a considerable degree of interest in the machine learning and related literature. Many authors have discussed and/or demonstrated the need for consistent stochastic search methods that would not be as prone to yield locally optimal model structures as simple greedy methods. However, at the same time most of the stochastic search methods are based on a standard Metropolis–Hastings theory that necessitates the use of relatively simple random proposals and prevents the utilization of intelligent and efficient search operators. Here we derive an algorithm for learning topologies of graphical models from samples of a finite set of discrete variables by utilizing and further enhancing a recently introduced theory for non-reversible parallel interacting Markov chain Monte Carlo-style computation. In particular, we illustrate how the non-reversible approach allows for novel type of creativity in the design of search operators. Also, the parallel aspect of our method illustrates well the advantages of the adaptive nature of search operators to avoid trapping states in the vicinity of locally optimal network topologies.

Keywords MCMC · Equivalence search · Learning graphical models

Responsible editor: Charu Aggarwal.

J. Corander (✉)
Department of Mathematics, Åbo Akademi University, 20500 Abo, Finland
e-mail: jukka.corander@abo.fi

M. Ekdahl
Department of Mathematics, Linköping University, 581 83 Linkoping, Sweden

T. Koski
Department of Mathematics, Royal Institute of Technology, 100 44 Stockholm, Sweden

1 Introduction

Statistical learning of graphical models from databases has been extensively discussed both in the computer science and statistical literature (Chickering 2002a,b; Corander 2003; Cooper and Hershkovitz 1992; Jordan 1998; Janzura and Nielsen 2006; Lam and Bacchus 1994; Madigan et al. 1996; Poli and Roverato 1998; Madigan and Raftery 1994; Riggelsen 2005; Wong et al. 2003; Sanguesa and Cortes 1997; Suzuki 1996, 2006; Giudici and Castelo 2003; Dellaportas and Forster 1999; Koivisto and Sood 2004; Jones et al. 2005; Wedelin 1996).

Generally, the vast number of existing works agree on the main challenges related to such tasks, the first of which is the increase in the number of graphical model structures as a function of the number of nodes. The second main obstacle is considered to be the equivalence of the statistical models determined by different graphical models. The size of the space of graphical models poses difficulties both with respect to the computational complexity of the learning task, as well as the reliability to reach representative model structures.

Certain types of stochastic search methods, such as Markov chain Monte Carlo (MCMC) or simulated annealing can be proven to be consistent with respect to the identification of a structure maximizing posterior probability. However, a standard MCMC-based search method is built upon the theory of reversible Markov chains, which requires closed form expressions for the proposal probabilities between graphical structures. This often confines the search operators to be extremely simple and naive, and consequently, such operators may easily fail to traverse efficiently towards optimal structures.

In the current work we derive an algorithm for learning topologies of graphical models from samples of a finite set of discrete variables by utilizing and further enhancing a recently introduced theory for non-reversible parallel interacting Markov chain Monte Carlo-style computation (Corander et al. 2006). In particular, we illustrate how the non-reversible approach allows for novel type of creativity in the design of search operators, and show the advantages of the adaptive nature of search operators resulting from the parallel learning by analyzing several sets of data. The article is organized as follows. In the next section we provide the basic notation concerning graphical models and Bayesian learning, whereafter the stochastic search algorithm is derived, along with a statistical consistency result. Illustrations of the non-reversible parallel learning approach are given in Sect. 5. The final section contains a discussion on the possibilities of further developments and the technical details related to characteristics of graphical models, Bayesian model scoring and estimator consistency are provided in Appendix.

2 Graphical models and Bayesian learning

Let the set of nodes $V = \{1, \dots, d\}$ index the random variables $X_1, \dots, X_d$. Each variable X_i assume values in the discrete and finite alphabet $\mathcal{X}_i$. The set of all possible configurations of $X_1, \dots, X_d$ is denoted by $\mathcal{X} = \times_{i=1}^{d} \mathcal{X}_i$. Let the generic element of

$\mathcal{X}$ be denoted by $\boldsymbol{x}$. The symbol $\mathbf{X}$ designates a training set of data, or, samples of the variables, i.e., $\mathbf{X} = \left\{\boldsymbol{x}^{(l)}\right\}_{l=1}^{r}, \boldsymbol{x}^{(l)} \in \mathcal{X}$, where there are no missing values.

A *graph* $\boldsymbol{G}$ is a pair (V, E), where $E \subset V \times V \backslash \{(i, i) \in V \times V\}$ is the set of ordered pairs of nodes denoted as *edges*. If both $(i, j) \in E$ and $(j, i) \in E$ then there is an *undirected edge* between i and j written as $i - j$. In the case that $(i, j) \in E$ but $(j, i) \notin E$ there is a *directed edge* between i and j written as $i \to j$. Here a *path* is a sequence $v_0, \ldots, v_n$ of different nodes such that $v_i \neq v_j$ and $(v_{i-1}, v_i) \in E$ for all $i, j = 1, \ldots, n$. A *cycle* is a path with the only exception that $v_0 = v_n$. Let a *directed cycle* be a cycle which has at least one directed edge.

A *partially directed acyclic graph* (PDAG) is a graph $\boldsymbol{G}$ that does not contain any directed cycles. A *directed acyclic graph* (DAG) is a PDAG with only directed edges and an *undirected graph* (UG) is a graph with only undirected edges.

Since {DAG} $\subset$ {PDAG} and {UG} $\subset$ {PDAG} it is sufficient to state most results with respect to PDAGs. Here the probabilistic models that PDAGs (and DAGs as well as UGs) represent are characterized by a set of Markov properties called Lauritzen, Wermuth and Frydenberg (LWF) Markov properties. Further details of graphical models are provided in Appendix.

Bayesian learning of graphical models within any particular class $\mathcal{S}$ of interest is generally formulated in terms of the posterior distribution over the models. By letting $P(\mathbf{X} \mid \boldsymbol{G})$ denote the marginal likelihood of the data conditional on the graph $\boldsymbol{G}$, and $P(\boldsymbol{G})$ the prior probability of the graph, the corresponding posterior probability is obtained from the Bayes' theorem as

$$P(\boldsymbol{G}|\mathbf{X}) = \frac{P(\mathbf{X} \mid \boldsymbol{G})\, P(\boldsymbol{G})}{\sum_{\boldsymbol{G} \in \mathcal{S}} P(\mathbf{X} \mid \boldsymbol{G})\, P(\boldsymbol{G})}. \tag{1}$$

The exact derivation of the marginal likelihood under a range of further assumptions is shown in Appendix. A natural object of interest in Bayesian learning is the structure $\boldsymbol{G}^{\mathrm{opt}} \in \mathcal{S}$ associated with the highest posterior probability, i.e.,

$$\boldsymbol{G}^{\mathrm{opt}} \in \arg\max_{\boldsymbol{G} \in \mathcal{S}} P(\boldsymbol{G}|\mathbf{X}). \tag{2}$$

In the next section we show how a statistically consistent estimate of the posterior optimal structure may be obtained using stochastic search algorithms.

3 Bayesian learning with interacting stochastic search processes

In general, even for a moderately large number of nodes d, it is not possible in practice to make an exhaustive search through the space of considered models $\mathcal{S}$ to find the optimal model $\boldsymbol{G}^{\mathrm{opt}}$ and to characterize model uncertainty in terms of the posterior probabilities of the most likely topologies. Instead, a structure $\hat{\boldsymbol{G}}_t$ estimating $\boldsymbol{G}^{\mathrm{opt}}$ can be computed using the estimator

$$\hat{\boldsymbol{G}}_t \in \arg\max_{\boldsymbol{G} \in \mathcal{S}_t} P(\boldsymbol{G}|\mathbf{X}),$$

where $\mathcal{S}_t \subseteq \mathcal{S}$ is the part of the model space that has been explored by some (search) process, like MCMC, until time t. MCMC can be interpreted as a stochastic search within $\mathcal{S}$ that may in a relatively few steps (compared to the size of $\mathcal{S}$) tend to a structure resembling (one of) $\boldsymbol{G}^{\text{opt}}$. However, as we do not wish to restrict the learning only to approximating the posterior mode, the posterior distribution will also be approximated in the sequel (Theorem 1 (5)). Markov chain-based methods are generally described in (Robert and Casella 2004), as well as in (Isaacson and Madsen 1976).

An MCMC method known as Metropolis–Hastings (M-H) algorithm is compactly defined in the current context by its acceptance probability

$$\nu\left(\boldsymbol{G}^*|\boldsymbol{G}\right) = \min\left(1, \frac{P\left(\mathbf{X}|\boldsymbol{G}^*\right) P\left(\boldsymbol{G}^*\right) q(\boldsymbol{G}|\boldsymbol{G}^*)}{P\left(\mathbf{X}|\boldsymbol{G}\right) P\left(\boldsymbol{G}\right) q(\boldsymbol{G}^*|\boldsymbol{G})}\right), \tag{3}$$

where $q(\boldsymbol{G}^*|\boldsymbol{G})$ is a proposal distribution generating candidate topologies given any particular state of the search process. In other words, we draw a candidate $\boldsymbol{G}^*$ of a new model structure from this distribution conditionally on the current structure $\boldsymbol{G}$, and accept this as the next state of the process with probability $\nu\left(\boldsymbol{G}^*|\boldsymbol{G}\right)$. This algorithm can encounter practical problems in a finite time, such as:

1. $q(\boldsymbol{G}|\boldsymbol{G}^*)/q(\boldsymbol{G}^*|\boldsymbol{G})$ contributes to a large portion of Eq. 3 when the proposal distribution is not symmetric, and makes the transition or acceptance probability practically independent of $\mathbf{X}$.
2. The process is stuck in the vicinity of local maximum of the posterior.

These problems were addressed, e.g., in (Geyer and Thompson 1995; Corander et al. 2006), through the introduction of several parallel chains, that communicate at random times that are independent of the process in the search space.

To utilize this in the current context, we introduce first a sequence $\{\alpha_t\}$, the elements of which are the probabilities of success in a corresponding sequence of independent Bernoulli variables, the successes of which determine the times where the parallel chains communicate by drawing a new start value from a pooled posterior distribution.

Definition 1 $\{\alpha_t, t = 1, 2, \ldots\}$ is a strictly decreasing sequence of positive numbers such that $1 > \alpha_t > \alpha_{t+1}$ and $\lim_{t\to\infty} \alpha_t \to 0$.

Algorithm 1 introduces a Bernoulli process of independent random variables Z_t such that, if $Z_t = 1$ (which happens with probability $P(Z_t = 1) = \alpha_t$), then the chains interact by choosing a new structure as in Algorithm 2. We call $Z_t = 1$ a mixing event.

Intuitively it is not as likely that all parallel chains should get simultaneously stuck at domains of low posterior probability, and those who will, can make leaps to better areas of the model space through communicating with the other chains at mixing events.

Algorithm 1 ParalellInteractingProcess($\{\boldsymbol{G}_{0j}\}_{j=0}^{m}, n$)

```
for $t = 1$ to $n$ do
    $Z_t \sim Be(\alpha_t)$
    if $Z_t = 0$ then
        $\{\boldsymbol{G}_{(t+1)j}\}_{j=1}^{m} \leftarrow$ ParallelProcesses($\{\boldsymbol{G}_{tj}\}_{j=1}^{m}$)
    else
        $\{\boldsymbol{G}_{(t+1)j}\}_{j=1}^{m} \leftarrow$ InteractingProcesses($\{\boldsymbol{G}_{tj}\}_{j=1}^{m}$)
    end if
end for
```

Algorithm 2 InteractingProcess($\{\boldsymbol{G}_{tj}\}_{j=1}^{m}$)

```
for $j = 1$ to $m$ do
    $\boldsymbol{G}_{(t+1)j} \sim \frac{P(\boldsymbol{G}_{tj})P(\mathbf{X}|\boldsymbol{G}_{tj})}{\sum_{j=1}^{m} P(\boldsymbol{G}_{tj})P(\mathbf{X}|\boldsymbol{G}_{tj})}$
end for
```

Algorithm 3 ParallelProcess($\{\boldsymbol{G}_{tj}\}_{j=1}^{m}$)

```
for $j = 1$ to $m$ do
    $\boldsymbol{G}_{t+1}^{*} \sim Q|\boldsymbol{G}_{tj}$
    $p \sim U(0, 1)$
    if $p < \nu_{\text{par}}\left(\boldsymbol{G}_{t+1}^{*} \mid \boldsymbol{G}_{tj}\right)$ then
        $\boldsymbol{G}_{(t+1)j} \leftarrow \boldsymbol{G}_{t+1}^{*}$
    else
        $\boldsymbol{G}_{(t+1)j} \leftarrow \boldsymbol{G}_{tj}$
    end if
end for
```

The acceptance probabilities are now introduced like in simulated annealing algorithms (van Laarhoven and Aarts 1987),

$$\nu_{\text{par}}\left(\boldsymbol{G}^{*} \mid \boldsymbol{G}\right) = \min\left(1, \frac{P\left(\mathbf{X}|\boldsymbol{G}^{*}\right)}{P\left(\mathbf{X}|\boldsymbol{G}\right)}\right). \tag{4}$$

Here the prior is uniform, $P(\boldsymbol{G}) = 1/|\mathcal{S}|$, and hence, it cancels in the M-H ratio. We note that there is in general no exact expression for $|\mathcal{S}|$, when $\mathcal{S} = \{\text{LPDAG}\}$.

The removal of the explicit dependence on the proposal distribution $q(\cdot|\cdot)$ in (4) does not imply that a proposal mechanism need not be specified. The mechanism described in the Algorithm 4 is used here, and it is analogous to that used in a clustering context in (Corander et al. 2006), except for the last operator which utilizes the information in the data to propose intelligent splits in contrast to the random split operator in (Corander et al. 2006). If the nonempty sets $A, B, C \subset V$ form a partition of V, C separates A from B, and C is a complete subset of V, then (A, B, C) is a *proper decomposition* of V. An undirected graph G is *decomposable*, if it is complete or it possesses a proper decomposition (A, B, C), such that $G_{A\cup C}$ and $G_{B\cup C}$ are decomposable. It is worthwhile to notice, that the first two simple random search operators in the the Algorithm 4 could be computationally more efficient by the strategy exploited in (Giudici and Green 1999), which ensures that the proposal graphs are always decomposable.

Algorithm 4 Search operators

```
1: Delete a randomly chosen edge in the UG. If the operator leads to a non-decomposable UG, the graph
   is omitted and the operator is applied again to the original graph.
2: Add an edge between a randomly chosen pair of nodes lacking an edge in the UG. If the operator leads
   to a non-decomposable UG, the graph is omitted and the operator is applied again to the original graph.
3: Choose randomly two cliques of the UG and add edges between all pairs of nodes lacking edges.
4: 1. Choose a clique $c$ with $|c| > 1$ at random.
   2. Calculate the ML-estimate of the conditional KL-divergences from the joint distribution to the model
      arising under the conditional independence of $i$ and $j$ given $c\backslash\{i, j\}$ according to

      $$(KL)_{i,j} = n \sum_{x_c} p(x_c) \log \frac{p(x_c)}{\left[\frac{p(x_{c\backslash\{i\}})p(x_{c\backslash\{j\}})}{p(x_{c\backslash\{i,j\}})}\right]}$$

      for each pair $\{i, j\} \in \binom{c}{2}$. Denote by

      $$m \in Arg \max_{\{i,j\}} KL\{i, j\}$$

      a maximum KL-divergence. Define a dissimilarity matrix $D$ with elements $D_{i,j}$ for the $|c|$ nodes as

      $$m\mathbf{1}_{|c|\times|c|} - (KL)_{i,j}.$$

      (Thus, when two nodes are conditionally independent, their distance is maximum in $D$.) Let $u$ be
      uniformly distributed in the interval $[2, \ldots, |c|]$. A candidate for the split of $c$ into $u$ cliques with
      no edges between them is now obtained by cutting a complete linkage dendrogram $h$ based on $D$ at
      level $d$, such that the tree is split into $u$ separate components.
```

Computation of the KL-divergences exploited in the split proposal may be straightforwardly done using either maximum likelihood or maximum a posteriori estimates of the clique probabilities. With the total number of samples r, the maximum likelihood estimates for any subset a of nodes are defined as $\hat{p}(\boldsymbol{x}_a) = n_{a,\boldsymbol{x}_a}/r$, and the corresponding posterior estimates as $\hat{p}(\boldsymbol{x}_a) = (n_{a,\boldsymbol{x}_a} + \lambda_{a,\boldsymbol{x}_a})/(r + \sum_{\boldsymbol{x}_a \in \mathcal{X}_a} \lambda_{a,\boldsymbol{x}_a})$, where $\lambda_{a,\boldsymbol{x}_a}$ is a hyperparameter in a Dirichlet prior distribution (for details, see Appendix). A particular advantage of the Bayesian approach in this context is that it avoids the numerical instability of the KL-divergence occurring when the observed empirical count $n_{a,\boldsymbol{x}_a}$ is zero for some outcome $\boldsymbol{x}_a$. The Bayesian estimates can indeed be interpreted as a smoothed version of the empirical relative counts.

The illustration of the non-reversible M-H algorithm in (Corander et al. 2006) utilizes a search operator that splits randomly clusters into two parts, i.e., in the current context an analogous operator would remove all edges between two parts of the original clique. However, such an operator is extremely inefficient in finding sensible splits, when the clique size increases. Here we demonstrate in the context of graph learning how the non-reversible M-H algorithm enables one to invoke intelligent search operators without having to calculate the proposal probabilities.

The complete linkage tree attempts to allocate together nodes which have high values of dependence according to the KL-divergence. The search operator has some fixed proposal probability distribution, designated by Q in Algorithm 3, over all possible proposals in any given state. However, the explicit calculation of the proposal probabilities is in general complicated and would not lead to a practically implementable algorithm. It is worthwhile to notice also how a random join/split operator would behave in an ordinary M-H algorithm, c.f. (Corander et al. 2006). Also, this approach illustrates the possibility to embed rapid standard computational tools of data analysis into an MCMC-style computation.

Let $\boldsymbol{G}^* \cup \boldsymbol{G}$ denote the union of any two graphs on the same set of nodes, such that the resulting graph contains all the edges of both graphs. The following algorithm samples for each chain two graph structures from the pooled posterior distribution, merges them and replaces eventually the next chain state after the mixing event with the merged graph, if the latter is associated with a sufficiently high posterior probability compared to the original state.

Algorithm 5 UnionProcess($\{\boldsymbol{G}_{(t+1)j}\}_{j=1}^{m}$)

```
for $j = 1$ to $m$ do
    $\boldsymbol{G}_{1j} \sim \frac{P(\boldsymbol{G}_{(t+1)j})P(\mathbf{X}|\boldsymbol{G}_{(t+1)j})}{\sum_{j=1}^{m} P(\boldsymbol{G}_{(t+1)j})P(\mathbf{X}|\boldsymbol{G}_{(t+1)j})}$
    $\boldsymbol{G}_{2j} \sim \frac{P(\boldsymbol{G}_{(t+1)j})P(\mathbf{X}|\boldsymbol{G}_{(t+1)j})}{\sum_{j=1}^{m} P(\boldsymbol{G}_{(t+1)j})P(\mathbf{X}|\boldsymbol{G}_{(t+1)j})}$
    $p \sim U(0, 1)$
    $\boldsymbol{G}_{j*} \leftarrow \boldsymbol{G}_{1j} \cup \boldsymbol{G}_{2j}$
    if $p < \frac{P(\boldsymbol{G}_{j*})P(\mathbf{X}|\boldsymbol{G}_{j*})}{P(\boldsymbol{G}_{(t+1)j})P(\mathbf{X}|\boldsymbol{G}_{(t+1)j})+P(\boldsymbol{G}_{j*})P(\mathbf{X}|\boldsymbol{G}_{j*})}$ then
        $\boldsymbol{G}_{(t+1)j} \leftarrow \boldsymbol{G}_{j*}$
    else
        $\boldsymbol{G}_{(t+1)j} \leftarrow \boldsymbol{G}_{(t+1)j}$
    end if
end for
```

The above union operator, which is used at the mixing times in the algorithm, provides the possibility of merging plausible topologies together. Such a global operator can improve the convergence in particular for large graphs, where each of the distinct search processes may leave a distinct part of the graph structure unexplored. The union of the edges of the separate graphs can then provide a way of making larger leaps in the model space.

4 Consistent estimation of the maximum posterior graphs through parallel interacting chains

The construction introduced in the previous section turns out to guarantee that a set of communicating parallel processes moving with the acceptance probabilities in (4) eventually visits all states of $\mathcal{S}$, when the search operators results in an irreducible Markov chain with respect to Algorithm 3 (although in general the asymptotic distribution will *not* be $P\ (\boldsymbol{G}|\mathbf{X})$).

Theorem 1 *Let $\mathcal{S}_t \subseteq \mathcal{S}$ be the part of the space explored at time t by the search process defined by the acceptance probability in* (4)*, the Algorithms* 1–5*, and search operators that yield an irreducible Markov chain with respect to Algorithm* 3*.*

Let $\boldsymbol{G}$ be an arbitrary structure in $\mathcal{S}$. Let

$$\hat{P}_t\left(\boldsymbol{G}|\mathbf{X}\right) = \begin{cases} \frac{P(\mathbf{X}|\boldsymbol{G})}{\sum_{\boldsymbol{G}\in\mathcal{S}_t} P(\boldsymbol{G}|\mathbf{X})}, & \textit{if } \boldsymbol{G} \in \mathcal{S}_t, \\ 0 & \textit{elsewhere.} \end{cases} \tag{5}$$

Then

$$\hat{P}_t\left(\boldsymbol{G}|\mathbf{X}\right) \overset{a.s.}{\rightarrow} P\left(\boldsymbol{G}|\mathbf{X}\right), \quad \boldsymbol{G} \in \mathcal{S}$$

as $t \rightarrow \infty$.

The almost sure convergence ($\overset{a.s.}{\rightarrow}$) in the statement above is with respect to the probability measure on $\mathcal{S}$ (and some σ-field) determined by the search algorithm. The proof of this theorem will be given in Appendix section. The proof improves from (Corander et al. 2006) by removing the necessity of requiring aperiodicity. Theorem 1 (5) has the following straightforward corollary.

Corollary 1

$$\max_{\boldsymbol{G}\in\mathcal{S}_t} P\left(\boldsymbol{G}|\mathbf{X}\right) \overset{a.s.}{\rightarrow} \max_{\boldsymbol{G}\in\mathcal{S}} P\left(\boldsymbol{G}|\mathbf{X}\right) \tag{6}$$

as $t \rightarrow \infty$.

The above theorem ensures that computations in Eqs. 5 and 6 are consistent in the sense of asymptotically computing the exact posterior probability distribution $P\left(\boldsymbol{G}|\mathbf{X}\right)$ over $\mathcal{S}$, if based on the algorithms introduced in the previous section, which define an irreducible Markov chain. Consistency in this sense is different from the statistical consistency proved in (Suzuki 2006), where the 'true graph structure' (in the sense defined in Suzuki 2006) is found when the number of samples, r, grows to infinity.

5 Performance on real and simulated data sets for $\mathcal{S} = \{\text{UG}\}$

To illustrate our model search framework in the special case when $\mathcal{S} = \{\text{UG}\}$, we consider two real data sets investigated earlier in the graphical modeling literature, as well as two simulated data sets reflecting a more challenging graph topology. The first real data set comprises six binary risk factors for coronary heart disease, for which 1841 cases are presented in Table 1, and the corresponding variable labels are listed in Table 2. The second real data set (economic activity) contains 8 binary variables and 665 observations, presented in Tables 4 and 5, respectively.

Both real data sets are available in (Whittaker 1990), where the original data sources are cited as well. In particular, the first data set has been extensively investigated in the graphical modeling literature. Two types of stochastic searches were performed

Table 1 Prognostic factors in coronary heart disease

F	E	D	C	B	Yes		No	
				A	No	Yes	No	Yes
Neg	<3	<140	No		44	40	112	67
			Yes		129	145	12	23
		>140	No		35	12	80	33
			Yes		109	67	7	9
	>3	<140	No		23	32	70	66
			Yes		50	80	7	13
		>140	No		24	25	73	57
			Yes		51	63	7	16
Pos	<3	<140	No		5	7	21	9
			Yes		9	17	1	4
		>140	No		4	3	11	8
			Yes		14	17	5	2
	>3	<140	No		7	3	14	14
			Yes		9	16	2	3
		>140	No		4	0	13	11
			Yes		5	14	4	4

Table 2 Explanations of the labels in Table 1

Label	Meaning	Range
A	Smoking	No, yes
B	Strenuous mental work	No, yes
C	Strenuous physical work	No, yes
D	Systolic blood pressure	<140, >140
E	Ratio of β and α lipoproteins	<3, >3
F	Family anamnesis of coronary heart disease	No, yes

for both real data sets, one using the Algorithm 4 as such, and the other where a modification was made to keep the search processes independent from each other. For the simulated data sets we only considered the Algorithm 4, as the examples with real data already illustrate the poor performance of the independent search processes.

5.1 Coronary data set

We consider the search on the data set in Table 1 first. Figure 1a and b show the behavior of the logarithm of the marginal data distribution over the first 15 iterations of 1000 parallel search processes with and without mixing, respectively.

The difference between these two strategies is clearly visible in this early phase, as a mixing event takes place for the dependent processes at iteration 4, which absorbs the processes in the neighborhood of the global posterior mode. On the contrary, many of the independent processes continue outside the mode vicinity and are gradually absorbed towards this area in the equivalence class space. For this data even all independent processes are able to reach the mode neighborhood fairly rapidly. Both

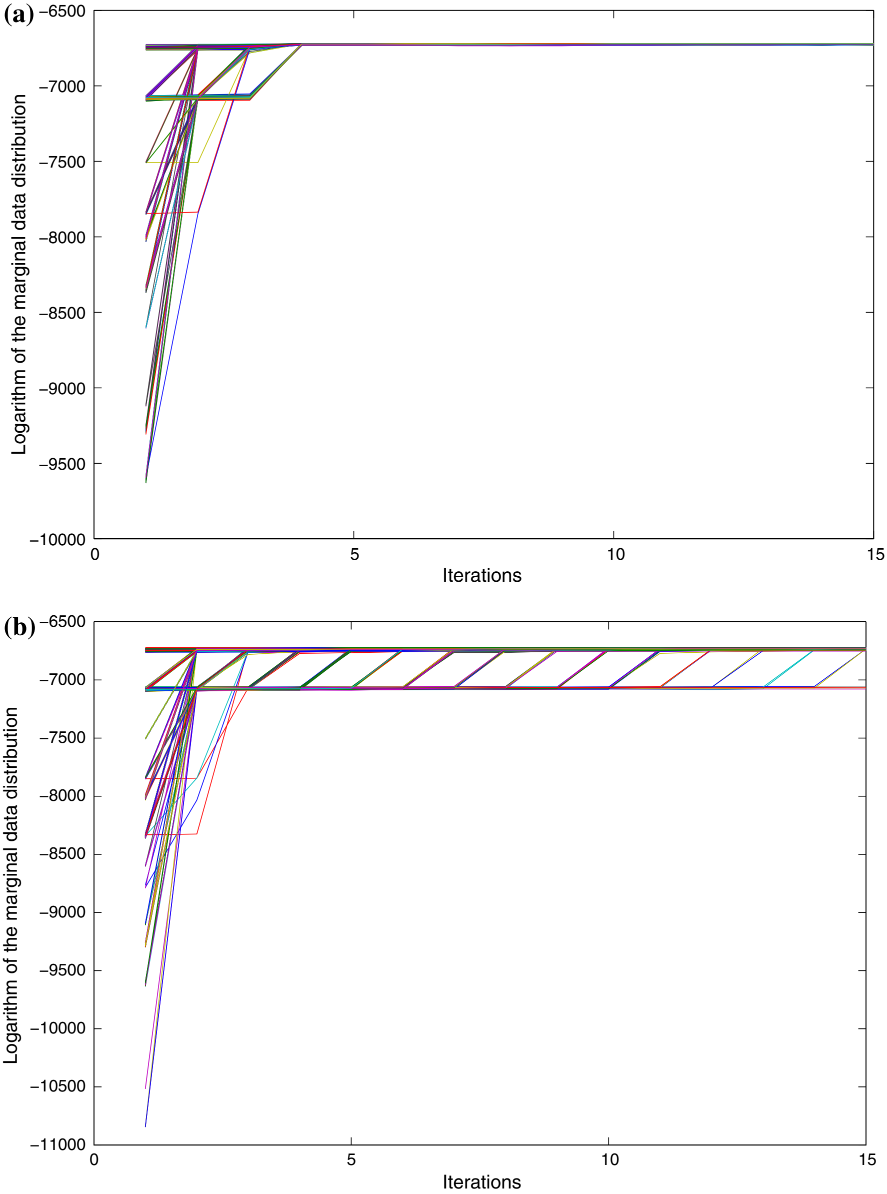

Fig. 1 **(a)** Behavior of log P ($\mathbf{X}|\boldsymbol{G}$) for 1000 dependent search processes for the coronary dataset. **(b)** Behavior of log P ($\mathbf{X}|\boldsymbol{G}$) for 1000 independent search processes for the coronary dataset

approaches yield the same optimal equivalence class, having the estimated posterior probability .32 and the cliques

$$\{F\}, \{B, C\}, \{A, C, E\}, \{A, D, E\}.$$

Our method enables also consistent estimation of the marginal posterior probabilities of any edges being present in the graphs, that is we estimate the marginal probabilities as

$$\hat{P}(e|\boldsymbol{X}) = \frac{\sum_{\{\mathbf{G}=(V,E)\in\{UG\}|\mathbf{G}\in\mathcal{S}_t, e\in E\}} P(\mathbf{G}|\boldsymbol{X})}{\sum_{\{\mathbf{G}\in\{UG\}|\mathbf{G}\in\mathcal{S}_t\}} P(\mathbf{G}|\boldsymbol{X})}.$$

These are illustrated in Table 3 where the probability of adjacency is given for all pairs of the considered variables. When the marginal posterior probabilities estimated by the two different searches were compared, the largest absolute difference was of magnitude ∼.0001.

5.2 Women's economic activity

As the coronary heart disease data contains only a very limited number of variables, even the independent search processes perform satisfactorily with respect to the convergence towards the optimum. However, the behavior of the search algorithms is already totally different for the second data set with the 8 variables described in Tables 4 and 5.

Figure 2a and b show the behavior of the logarithm of the marginal data distribution over the first 500 iterations of 100 parallel search processes with and without mixing, respectively.

A majority of the independent processes (Fig. 2b) has not reached the vicinity of the posterior mode by 500 iterations, whereas all the dependent processes have reached this by the mixing events around iterations 5, 60 and 120. Figure 3 shows the traces of the log $P(\mathbf{X}|\mathbf{G})$ for the dependent processes in a larger detail over the first 100 iterations.

Even if the both approaches to search reach the same optimum, this example illustrates well the benefits of allowing the processes to exchange information about their states at mixing events. This aspect becomes clearly increasingly important with an

Table 3 The estimated marginal posterior probabilities of edges for the coronary heart disease data

	A	B	C	D	E	F
A	–	0.1630	0.9999	0.9655	0.9997	0.0487
B	0.1630	–	1.0000	0.0002	0.1864	0.3003
C	0.9999	1.0000	–	0.0010	0.9313	0.0164
D	0.9655	0.0002	0.0010	–	0.9839	0.0553
E	0.9997	0.1864	0.9313	0.9839	–	0.1273
F	0.0487	0.3003	0.0164	0.0553	0.1273	–

Table 4 Women's economic activity: an eight-way table

5	0	2	1	5	1	0	0	4	1	0	0	6	0	2	0
8	0	11	0	13	0	1	0	3	0	1	0	26	0	1	0
5	0	2	0	0	0	0	0	0	0	0	0	0	0	1	0
4	0	8	2	6	0	1	0	1	0	1	0	0	0	1	0
17	10	1	1	16	7	0	0	0	2	0	0	10	6	0	0
1	0	2	0	0	0	0	0	1	0	0	0	0	0	0	0
4	7	3	1	1	1	2	0	1	0	0	0	1	0	0	0
0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0
18	3	2	0	23	4	0	0	22	2	0	0	57	3	0	0
5	1	0	0	11	0	1	0	11	0	0	0	29	2	1	1
3	0	0	0	4	0	0	0	1	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
41	25	0	1	37	26	0	0	15	10	0	0	43	22	0	0
0	0	0	0	2	0	0	0	0	0	0	0	3	0	0	0
2	4	0	0	2	1	0	0	0	1	0	0	2	1	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 5 Explanations of the labels in Table 4

Label	Meaning	Range
A	Wife economically active	No, yes
B	Age of wife > 38	No, yes
C	Husband unemployed	No, yes
D	Child $\leqslant 4$	No, yes
E	Wife's education, O level+	No, yes
F	Husband's education, O level+	No, yes
G	Asian origin	No, yes
H	Other household member working	No, yes

increasing number of nodes in the graph. The posterior mode is associated with the estimated probability .57 and the cliques of the corresponding graph are:

$$\{E, F\}, \{B, D, H\}, \{B, D, E\}, \{A, D, E\},$$
$$\{A, D, G\}, \{A, C, E\}, \{A, C, G\}, \{C, E, G\}.$$

As for the earlier data set, the marginal posterior probabilities of adjacencies are given in Table 6. The values given this table are estimated using the interacting processes, and when the estimates yielded by the two different searches were compared, the largest absolute difference was of magnitude .0028, indicating a slight increase from the difference obtained for the smaller variable set.

5.3 Simulated data sets

In order to investigate the performance of the parallel stochastic search method in a more challenging scenario, we simulated data from two Bayesian networks. First network consists of $d = 20$ binary nodes with the following dependence structure. We use $\sim$Unif(0, 1) in the sequel to denote that a random variable has the uniform distribution

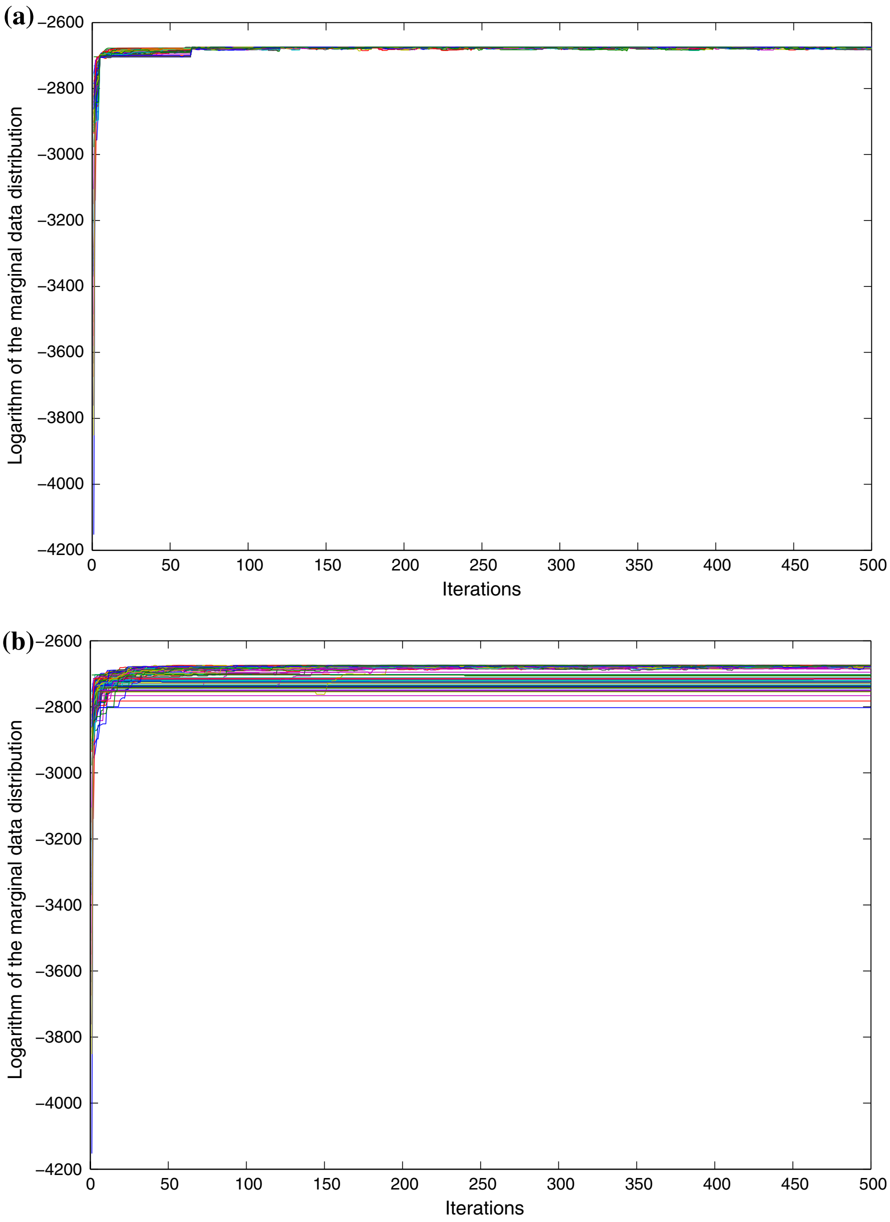

Fig. 2 **(a)** Behavior of log P ($\mathbf{X}|\boldsymbol{G}$) for 100 dependent search processes for the economic activity dataset. **(b)** Behavior of log P ($\mathbf{X}|\boldsymbol{G}$) for 100 independent search processes for the economic activity dataset

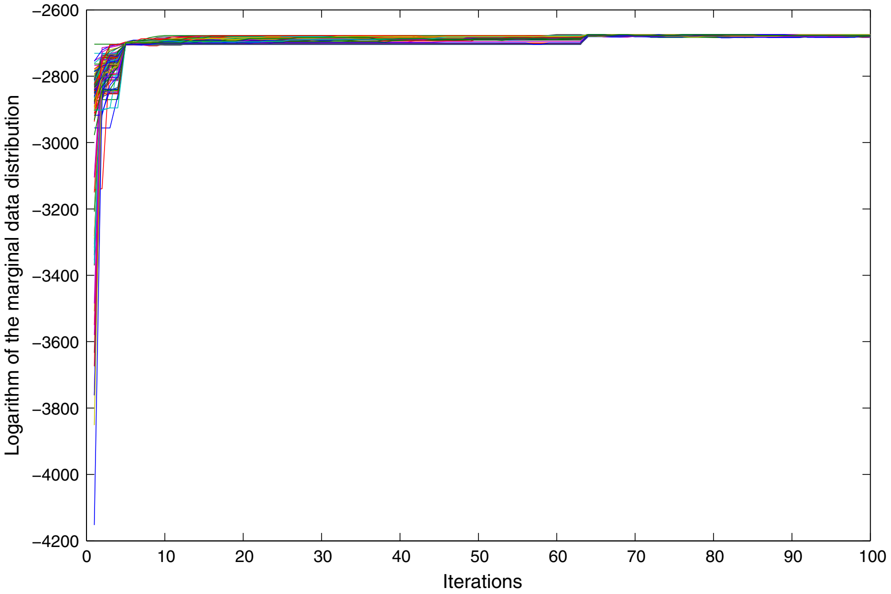

Fig. 3 Behavior of $\log P\,(\mathbf{X}|\boldsymbol{G})$ over first 100 iterations for 100 dependent search processes for the economic activity dataset

Table 6 The estimated marginal posterior probabilities of edges for the economic activity data

	A	B	C	D	E	F	G	H
A	–	0.1097	0.9999	1.0000	0.9432	0.0102	1.0000	0.0006
B	0.1097	–	0.0539	1.0000	0.9988	0.0094	0.1761	1.0000
C	0.9999	0.0539	–	0.0008	0.9847	0.0704	0.8731	0.0000
D	1.0000	1.0000	0.0008	–	0.8900	0.0258	0.9768	0.9157
E	0.9432	0.9988	0.9847	0.8900	–	1.0000	0.9907	0.0049
F	0.0102	0.0094	0.0704	0.0258	1.0000	–	0.9990	0.0001
G	1.0000	0.1761	0.8731	0.9768	0.9907	0.9990	–	0.0008
H	0.0006	1.0000	0.0000	0.9157	0.0049	0.0001	0.0008	–

on the interval (0, 1). Let the probability $p(x_1 = 1) \sim$Unif(0, 1) and the conditional probability $p(x_i = 1|x_{i-1}) \sim$Unif(0, 1), independently for $x_{i-1} = 0$ and $x_{i-1} = 1$, for $i = 2, \ldots, 5$. A realization of such a process creates a first-order Markov structure for a set of 5 nodes. Further, let the probabilities $p(x_i = 1)$, $p(x_{i+1} = 1|x_i)$, $p(x_{i+2} = 1|x_i, x_{i+1})$ all be independently distributed as Unif(0, 1), for $x_i, x_{i+1}, x_{i+2} \in \{0, 1\}$. A realization of this process corresponds to a complete graph for three nodes as an equivalence class. A network of 20 nodes is then comprised of three independent replicates of such three node systems, combined with two independent replications of the first-order Markov structure over 5 nodes, and an additional a single isolated node associated with the probability $p(x_i = 1) \sim$Unif(0, 1). For a realization of the network parameters, $r = 300$ observations were generated from the corresponding

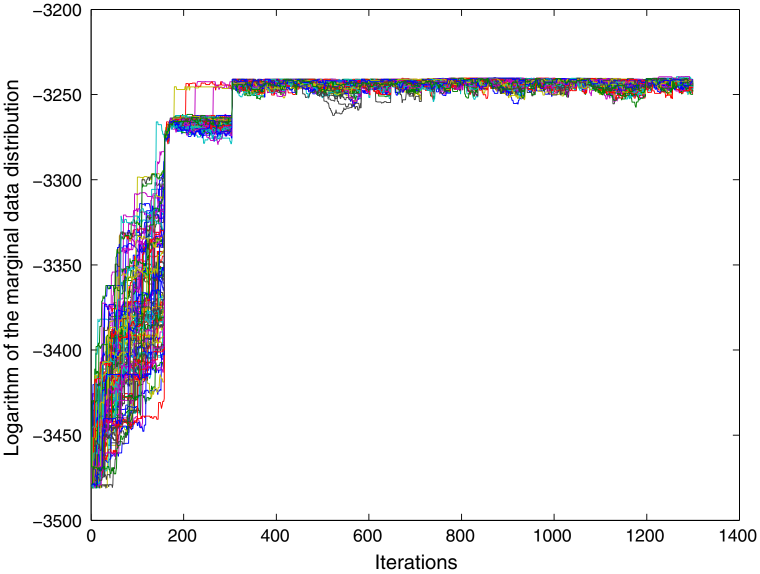

Fig. 4 Behavior of log $P\left(\mathbf{X}|\boldsymbol{G}\right)$ for 100 mixing search processes for the 20 node network learning

distribution and provided as the data $\mathbf{X} = \left\{\boldsymbol{x}^{(l)}\right\}_{l=1}^{r}$ for the stochastic learning algorithm.

We used 100 parallel search processes to learn the underlying graph topology given the data **X**. In Fig. 4, a realization of the search process is shown. The considerably higher statistical model uncertainty compared to the previous examples is immediately clear from behavior of the processes, as a large degree of variation is maintained in the process after it reaches a plateau. The usefulness of the process mixing is also well visible in the early phase of the search, where a majority of the processes are able to make huge leaps in the model space at the interaction event.

The relatively small size of the training data ($r = 300$) used here induces an uncertainty about the model structure, which should not be ignored by focusing solely on the estimated posterior optimal graph. The uncertainty about the graph topology can be represented by using the marginal posterior probabilities of the individual edges, which are consistently estimated with the aid of the results presented in Sect. 4. In the current example the generating graph topology is known, so it is instructive to connect the marginal posterior edge probabilities to the degrees of true and false findings. There are 17 edges in total in the generating model structure, out of which 13 (76%) are correctly identified when a threshold equal to 0.75 is used for a marginal edge probability to claim its presence in the graph. Correspondingly, out of the $\binom{20}{2} - 17$ absent edges, three were suggested to be included in the graph, when a threshold 0.10 was used for the marginal edge probabilities. Thus, a vast majority of the truly absent edges (91%) had very low marginal posterior probabilities. Finally, to investigate whether it is likely that the search processes have converged to a region relatively distant from the global optimum, we calculated the ratio of the posterior probabilities for the estimated optimal model $\hat{\boldsymbol{G}}_t$ and the generating model, respectively. This ratio equals approximately 117, which means that the randomness in the realized data renders the generating graph topology rather inferior compared to the estimated optimal structure. However, structural learning using the posterior nevertheless mimics the generating graph with relatively high fidelity, as can be seen from the reported results.

The second network consists of 50 binary nodes associated with a dependence structure extended from the first simulation scenario. In this network, six independent realizations of the first-order Markov structure over five nodes are combined. Additionally, we generated four similar replicated systems associated with a complete graph for three nodes. Finally, eight marginally independent nodes were added to the graph and the model probabilities were then generated analogously to the network of 20 nodes. Given these probabilities, 400 observations were simulated to be used as a training data.

Due to the increased statistical uncertainty in the learning of the network with 50 nodes, it is expected that posterior probabilities will eventually support a much sparser graph topology than the true generating structure. As in the previous example, we used 100 parallel search processes for the learning. In Fig. 5, a realization of the search process is shown. To investigate the accuracy of the topology learning, we use the same summaries as for the 20 node network. There are 36 edges in total in the generating model structure, out of which 15 (42%) are correctly identified when a threshold equal to 0.75 is used for a marginal edge probability to claim its presence in the graph. Correspondingly, out of the $\binom{50}{2} - 36$ absent edges, seven were suggested

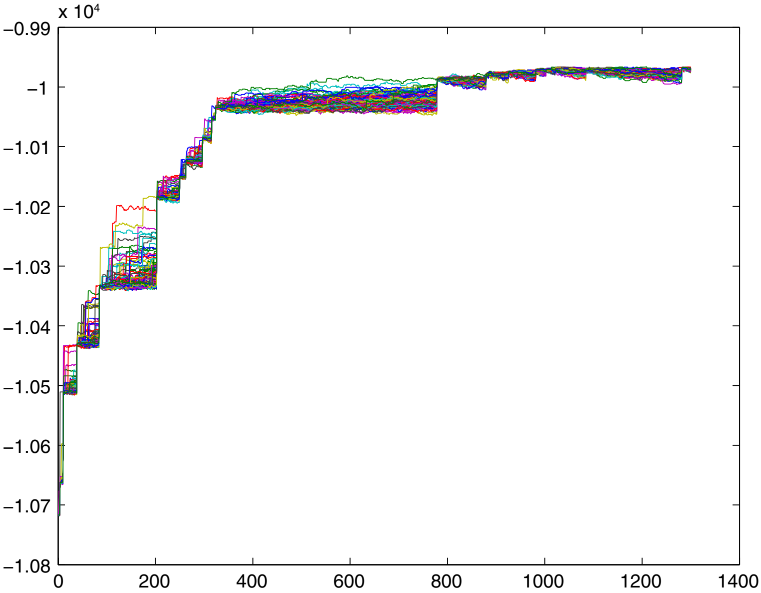

Fig. 5 Behavior of log $P(\mathbf{X}|\boldsymbol{G})$ for 100 mixing search processes for the 50 node network learning

to be included in the graph, when a threshold 0.10 was used for the marginal edge probabilities. Thus, 99.4% of the truly absent edges had very low marginal posterior probabilities. Finally, we calculated again the ratio of the posterior probabilities for the estimated optimal model $\hat{\boldsymbol{G}}_t$ and the generating model, respectively. This ratio equals approximately $4.3163 \cdot 10^{12}$, which, together with the proportion of false positive and true positive edges, illustrates the Occam's razor feature of the Bayesian topology learning, as the size of the training data is very small compared to the putative model complexity.

6 Discussion

MCMC has been widely considered as the tool of choice for consistent Bayesian model learning. However, albeit its satisfactory theoretical underpinnings, practical experience has clearly shown that some main challenges still remain to be solved for many problems. Two primary challenges are the convergence to local modes of the posterior and the limitations to relatively simple symmetric search operators for which the proposal probabilities needed in the Metropolis–Hastings algorithm are explicitly calculable.

Our present work illustrates in the context of graphical model learning how solutions to these two obstacles may be obtained using the non-reversible Markov chain theory in combination with a parallel search strategy. In particular, this offers considerable freedom in the design of the search operators, which was here concretized in terms of a rapid standard tool for data clustering (the complete linkage algorithm). However, in a general model learning context, many fast heuristic data analysis tools, such as principal and independent component analysis could be incorporated to guide the search operators in an intelligent manner. Obviously, the parallel architecture of the learning algorithm introduced here has a relatively high degree of computational complexity and is also memory intensive for large networks. An efficiently optimized implementation would still enable the use of such an approach in a single CPU environment for rather generic learning applications. We conclude by noting that assessing the convergence of the MCMC computation in practice is still an open issue as well. We aim to investigate this in the future to define systems that could assess the convergence semi-automatically.

Acknowledgements The work of M.E. and T.K. was partially supported by Swedish Research Council, Grants 4042/401 and 621-2004-4214, and EU 6th Programme Grant QLK3-CT-2002-02097 (BACDIVERS). The work of J.C. was supported by the Grant 121301 from Academy of Finland and the Swedish research council VR/NT 621-2004-4214.

Appendix

Characteristics of graphical models

Recall that the set of nodes $V = \{1, \ldots, d\}$ index the random variables $X_1, \ldots, X_d$. For a set $A \subset V$ let the *parents* of A be defined as $\mathrm{pa}(A) = \{i \in V | (i, j) \in E,$

$(j,i) \notin E, j \in A, i \notin A\}$. The *boundary* of a set $A \subset V$ is $\mathrm{bd}(A) = \mathrm{pa}(A) \cup \{i \in V|(i,j),(j,i) \in E, j \in A, i \notin A\}$. The *smallest ancestral set* is the smallest set $\mathrm{An}(A)$ such that $A \subset \mathrm{An}(A)$ and for all $i \in \mathrm{An}(A)$, $\mathrm{bd}(i) \subset \mathrm{An}(A)$. A *trail* is a sequence $v_0,\ldots,v_n$ of different nodes such that $v_i \neq v_j$, (v_{i-1},v_i) or/and $(v_i,v_{i-1}) \in E$ for all $i,j = 1,\ldots,n$. For $A,B,C \subset V$ the subset C separates A,B if for each $v_i \in A, v_j \in B$ any trail from v_i to v_j includes a node in C. Let a *subgraph* $\boldsymbol{G}_A$ of a graph $\boldsymbol{G}$ be the graph $(A, E \cap (A \times A))$ for $A \subset V$.

If there is a path from node v_0 to node v_n and a path from v_n to v_0 then v_0 and v_n *connect*. The *chain components* $\mathcal{T}(\boldsymbol{G})$ of a graph is the partition induced by the connected nodes in the graph $\hat{\boldsymbol{G}} = (V, \hat{E})$, where $\hat{E} \subset E$ is the set of undirected edges in E. Let $\mathrm{ch}(i) = \{j \in V|(i,j) \in E, (j,i) \notin E\}$ denote the *children* of a node i. We denote by $\boldsymbol{G}^m$ the *moral graph* obtained by adding undirected edges between all directed edges that have children in the same chain component and then undirecting the graph (making directed edges undirected). If for all $A \subset \mathcal{X}_i$ $P(X_i \in A|X_y, X_z) = P(X_i \in A|X_z)$, then X_i is *conditionally independent* of X_y given X_z, which will be denoted as $i \perp y|z$ and analogously for sets $B,C,S \subset V$ as $B \perp C|S$. The main properties of conditional independence are found in (Dawid 1979). The *descendants* of a node $i \in V$ is $\mathrm{de}(i) = \{j \in V|j \in V$, there is *a* path from i to j, there are no paths from j to $i\}$. The *non-descendants* are denoted by $\mathrm{nd}(i) = V \backslash (\mathrm{de}(i) \cup i)$.

Definition 2 LWF Markov properties P, L, G relative to a (DAG, UG, PDAG) are defined as

1. the pairwise Markov property **P**, if for any pair (i,j) of nodes $(i,j),(j,i) \notin E$ with $j \in \mathrm{nd}(i)$, $i \perp j|\mathrm{nd}(i)\backslash\{i,j\}$.
2. the local Markov property **L**, for any $i \in V$, $i \perp \mathrm{nd}(i)\backslash\mathrm{bd}(i)|\mathrm{bd}(i)$.
3. the global Markov property **G**, if for any triple of disjoint subsets $(A,B,S \subseteq V)$ such that S separates A from B in $(\boldsymbol{G}_{\mathrm{An}(A\cup B\cup S)})^m$, $A \perp B|S$.

There are alternative Markov properties (AMP) for PDAGs, neither AMP nor LWF is more expressive than the other (see Fig. 12 in Andersson et al. 2001, and Theorem 4.2 in Andersson et al. 1996). Variations include especially the representation of a unique graph in each equivalence class, an algorithm for finding such is presented in (Roverato and Studený 2006). One algorithmic problem with AMP is the lack of factorization beyond chain components, see (Andersson et al. 2001).

The next definition of a complex is due to (Studený 1998). This is the minimal complex in (Frydenberg 1990). A *complex* in a graph is a sequence of nodes $v_1,\ldots,v_k, k \geqslant 3$ such that $v_1 \to v_2, v_i - v_{i+1}$ for $i = 2,\ldots,k-2$, $v_{k-1} \leftarrow v_k$ and there are no other edges $(v_i,v_j) \in E$ for all $0 \leqslant i,j \leqslant k$.

The following theorem is proved in (Frydenberg 1990) and more specially in (Verma and Pearl 1990), see also (Andersson et al. 1997).

Theorem 2 *Two PDAGs have the same (LWF) Markov properties iff they have the same undirected graph and the same complexes.*

A graph $G = (V,E)$ is larger than the graph $\hat{G} = (V,\hat{E})$ if $\hat{E} \subset E$.

Theorem 3 (Frydenberg 1990) *For any PDAG* $\boldsymbol{G}$ *there exists a unique PDAG* $\tilde{\boldsymbol{G}} = (\tilde{V},\tilde{E})$ *with the same Markov properties as* $\boldsymbol{G}$ *such that* $\tilde{\boldsymbol{G}}$ *is larger than any other graph* $\hat{\boldsymbol{G}} = (\hat{V},\hat{E})$ *that has the same Markov properties as* $\boldsymbol{G}$.

Let an LPDAG be this largest PDAG. In (Volf and Studený 1999), which uses the chain graph notation instead of PDAG, an LPDAG is called a LCG. Those LPDAGs that represent the same Markov properties as an equivalence class of DAGs have been referred to as patterns (Spirtes et al. 1993), essential graphs (Andersson et al. 1997) and CPDAGs (Chickering 1995).

Let $|\cdot|$ denote the cardinality of a class of graphs. Then $|\{\text{CPDAG}\}| < |\{\text{DAG}\}| < |\{\text{PDAG}\}|$. Furthermore, $|\{\text{UG}\}| < |\{\text{CPDAG}\}|$ and we have that $|\{\text{UG}\}| < |\{\text{LPDAG}\}|$, making UG a good candidate for practical implementation. Note however that it is possible to represent a set of Markov properties in a PDAG that cannot be represented in a DAG or UG, such as $1 \rightarrow 2 - 3 \leftarrow 4$. For small d the known number of LPDAGs that cannot be represented by UGs is 0 for $d = 2, 3$ (Volf and Studený 1999). For $d = 4, 5$ the percentage of LPDAGs that cannot be represented as UGs is 6 and 22.

In (Chickering 2002b) a search space is said to have the following components:

1. a set of states,
2. a scheme of representation of states.

A straightforward example of a state is a directed acyclic graph (DAG). The generic state space is denoted as $\mathcal{S}$, $\boldsymbol{G} \in \mathcal{S}$.

The presence of equivalent DAGs suggests the use of search space $\mathcal{S}$, the states of which are equivalence classes of DAGs. That is, classes of DAGs that all have the same Markov properties according to Definition 2. Then the size of a search space, where the states are equivalence classes, is smaller than the space of all DAGs for $d \leqslant 10$ (Gillispie and Perlman 2001). An approximate investigation for $d \leqslant 20$ can be found in (Peña 2007).

Bayesian model scoring

Here we will introduce factorizations of priors and and the corresponding likelihoods under graphical models, to derive concrete forms for the marginal likelihood needed in Bayesian learning.

A graph $\boldsymbol{G} = (V, E)$ is *complete* if for all $i, j \in V$ either $(i, j) \in E$ or $(j, i) \in E$. The *closure* is $cl(A) = bd(A) \cup A$.

$P(\boldsymbol{x}|\boldsymbol{G})$ is the probability of $\boldsymbol{x}$ or observing $X_1, \ldots, X_d$ with the value $\boldsymbol{x}$ conditioned by the structure of the PDAG $\boldsymbol{G}$. When no risk of confusion is present, this will be written as $P(\boldsymbol{x})$. For a subset $A \subset V$, the corresponding random variable is $\boldsymbol{X}_A = (X_i)_{i \in A}$.

Theorem 4 (Frydenberg 1990) *A probability distribution on a discrete and finite sample space with strictly positive density P satisfies* $\mathtt{P}$ *(see Definition 2) over a PDAG $\boldsymbol{G}$ if and only if it factorizes as*

$$P(\boldsymbol{x}) = \prod_{\tau \in \mathcal{T}(\boldsymbol{G})} P\left(\boldsymbol{x}_\tau | \boldsymbol{x}_{bd(\tau)}\right), \tag{7}$$

where each factor $P\left(\boldsymbol{x}_\tau | \boldsymbol{x}_{bd(\tau)}\right)$ further factorizes along the undirected closure graph $\boldsymbol{G}^m_{cl(\tau)}$.

As noted by (Andersson et al. 1997), the factors in (7) $P\left(\boldsymbol{x}_\tau|\boldsymbol{x}_{bd(\tau)}\right)$ satisfy also **P** on $\boldsymbol{G}_\tau$, i.e., the graph induced by the chain component τ. Hence, as observed by (Andersson et al. 1997), the so-called *hyper-Dirichlet* distributions can be used as priors. The properties of hyper Markov priors on probability distributions satisfying the LWF Markov properties over graphs are given in (Dawid and Lauritzen 1993).

A *clique* is a complete subgraph $\boldsymbol{G}'_A$, such that there exists no (other) complete subgraph $\boldsymbol{G}'_{A'}$, $A \subset A' \subset V$ such that $\boldsymbol{G} \neq \boldsymbol{G}'$. A hyper-Dirichlet density is defined on cliques of an undirected graph, as pioneered by (Sundberg 1975).

Definition 3 For given real positive numbers $\lambda_c = \{\lambda_{c,\boldsymbol{x}_c}\}_{\boldsymbol{x}_c \in \mathcal{X}_c}$ let $\mathcal{D}(\lambda_c)$ denote the hyper-Dirichlet distribution for θ_c defined by the density

$$\pi\left(\theta_c|\lambda_c\right) = \frac{\Gamma\left(\lambda_0\right)}{\gamma_p} \prod_{\boldsymbol{x}_c \in \mathcal{X}_c} \theta_{c,\boldsymbol{x}_c}^{\lambda_{c,x_c}-1} \tag{8}$$

on the set $\left\{\theta_c | \sum_{\boldsymbol{x}_c \in \mathcal{X}_c} \theta_{c,\boldsymbol{x}_c} = 1, \theta_{c,\boldsymbol{x}_c} > 0\right\}$, where $\Gamma(x)$ is the Euler gamma function, and

$$\lambda_0 = \sum_{\boldsymbol{x}_c \in \mathcal{X}_c} \lambda_{c,\boldsymbol{x}_c},\ \gamma_p = \prod_{\boldsymbol{x}_c \in \mathcal{X}_c} \Gamma\left(\lambda_{c,\boldsymbol{x}_c}\right). \tag{9}$$

Let $\boldsymbol{G}$ be a PDAG, and let us set

$$P\left(\mathbf{X} \mid \theta, \boldsymbol{G}\right) = \prod_{\tau \in \mathcal{T}(\boldsymbol{G})} \prod_{l=1}^{r} P\left(\boldsymbol{x}_\tau^l | \theta_\tau, \boldsymbol{x}_{bd(\tau)}^l\right),$$

by (7), which is formally rewritten to include the parameters.

Next note that in (7) the probabilities $P\left(\boldsymbol{x}_\tau|\theta_\tau, \boldsymbol{x}_{bd(\tau)}\right)$ satisfy the LWF Markov property **P** on the undirected closure graph $\boldsymbol{G}^m_{cl(\tau)}$. Recall that a probability $P(\boldsymbol{x}|\theta) > 0$ satisfying the property **P** on an undirected graph factorizes as (see for example Cowell et al. 1999)

$$P(\boldsymbol{x}|\theta) = \frac{\prod_{c \in C} \theta_{c,\boldsymbol{x}_c}}{\prod_{s \in S} \theta_{s,\boldsymbol{x}_s}},$$

where $C \subset \{A|A \subset V\}$ is the set of cliques and $S \subset \{A|A \subset V\}$ is the multiset (including repetitions of elements) of separators (for two cliques C_i and C_j the *separator* $S = C_i \cap C_j$), respectively. Thus $\prod_{l=1}^r P\left(\boldsymbol{x}_\tau^l|\theta_\tau, \boldsymbol{x}_{bd(\tau)}^l\right)$ is a function of a product of expressions of the following form

$$\prod_{l=1}^{r} \theta(\boldsymbol{x}_c^l) = \prod_{\boldsymbol{x}_c \in \mathcal{X}_c} \theta_{c,\boldsymbol{x}_c}^{n_{c,\boldsymbol{x}_c}},$$

where $n_{c,\boldsymbol{x}_c}$ is the number of times the configuration $\boldsymbol{x}_c$ occurs in $\mathbf{X}_c = \left\{\boldsymbol{x}_c^{(l)}\right\}_{l=1}^r$.

On any clique c in $\boldsymbol{G}^m_{cl(\tau)}$ we introduce the integration with respect to $\mathcal{D}(\lambda_{c,i})$ as follows,

$$\int_{\left\{\theta_c|\sum_{\boldsymbol{x}_c\in\mathcal{X}_c}\theta_{c,\boldsymbol{x}_c}=1,\theta_{c,\boldsymbol{x}_c}>0\right\}}\prod_{\boldsymbol{x}_c\in\mathcal{X}_c}\theta_{c,\boldsymbol{x}_c}^{n_{c,x_c}}\pi\left(\theta_c|\lambda_c\right)d\theta_c.$$

By standard properties of the Dirichlet integral one gets from (8) to (9) that

$$P_c\left(\mathbf{X}_c\right)=\frac{\Gamma\left(\lambda_0\right)}{\Gamma(r+\lambda)}\prod_{\boldsymbol{x}_c\in\mathcal{X}_c}\frac{\Gamma\left(n_{c,\boldsymbol{x}_c}+\lambda_{c,\boldsymbol{x}_c}\right)}{\Gamma\left(\lambda_{c,\boldsymbol{x}_c}\right)},\tag{10}$$

where

$$r+\lambda=\sum_{\boldsymbol{x}_c\in\mathcal{X}_c}\left(n_{c,\boldsymbol{x}_c}+\lambda_{c,\boldsymbol{x}_c}\right).$$

The distribution in (10) is of the same form as the likelihood function for structures pioneered by (Cooper and Hershkovitz 1992).

Hence, if $C(\tau)$ and $S(\tau)$ are the set of cliques and separators in the undirected graph $\mathbf{G}_\tau$

$$\prod_{l=1}^{r}P\left(\boldsymbol{x}^l_\tau|\boldsymbol{x}^l_{bd(\tau)}\right)=\frac{\prod_{c\in C(\tau)}P_c\left(\mathbf{X}_c\right)}{\prod_{s\in S(\tau)}P_s\left(\mathbf{X}_c\right)}.\tag{11}$$

The probability $P_s\left(\mathbf{X}_c\right)$ can be constructed for a separator by marginalizing over a clique that includes S. For the validity of this argument and the properties of the marginal data distribution, see (Dawid and Lauritzen 1993). The expression (11) must be multiplied over all the chain components τ in order to yield the overall marginal data distribution

$$P\left(\mathbf{X}\mid\boldsymbol{G}\right)=\prod_{\tau\in\mathcal{T}(\boldsymbol{G})}\frac{\prod_{c\in C(\tau)}P_c\left(\mathbf{X}_c\right)}{\prod_{s\in S(\tau)}P_s\left(\mathbf{X}_c\right)}.\tag{12}$$

From (12) we define the final scoring function as the posterior probability

$$P\left(\boldsymbol{G}|\mathbf{X}\right)=\frac{P\left(\mathbf{X}\mid\boldsymbol{G}\right)P\left(\boldsymbol{G}\right)}{\sum_{\boldsymbol{G}\in\mathcal{S}}P\left(\mathbf{X}\mid\boldsymbol{G}\right)P\left(\boldsymbol{G}\right)}.\tag{13}$$

For a given set of nodes V let $\mathcal{S}=\{\text{LPDAG}\}$ be the search space. One particular goal for the topology learning can be stated as the identification of a structure $\boldsymbol{G}^{\text{opt}}\in\mathcal{S}$ having highest posterior probability (13), i.e.

$$\boldsymbol{G}^{\text{opt}}\in\arg\max_{\boldsymbol{G}\in\mathcal{S}}P\left(\boldsymbol{G}|\mathbf{X}\right).$$

Proof of consistency for the parallel learning procedure

Let $X = \{X_i\}_{i \geq 0}$ be a time homogeneous Markov chain with the state space $\mathcal{S}$. We set $P_x(X_i) = P(X_i | X_0 = x)$.

Definition 4 (*Durrett 1996*) Let $T_y^0 = 0$ and for $k \geqslant 1$ let

$$T_y^k = \inf\{n > T_y^{k-1} : X_n = y\} \tag{14}$$

Definition 5 (*Durrett 1996*) $\mathcal{S}$ is irreducible if $x, y \in \mathcal{S} \Rightarrow P_x(T_y^1 < \infty) > 0$.

Definition 6 (*Durrett 1996*) $\mathcal{S}$ is recurrent if $P_y(T_y^k < \infty) = 1$ for all k.

Theorem 5 (Durrett 1996) *If $\mathcal{S}$ is finite and closed, then every Markov chain that is irreducible on $\mathcal{S}$ is recurrent.*

Theorem 6 (Durrett 1996) *If $x \in \mathcal{S}$ is recurrent and $\mathcal{S}$ is irreducible then $y \in \mathcal{S}$ is recurrent and $P_x(T_y < \infty) = 1$ for all $x, y \in \mathcal{S}$.*

Let now $T_{\mathcal{S}}$ be defined as the first time, when the Markov chain X has visited every state of $\mathcal{S}$.

Theorem 7 (Durrett 1996) *When $\mathcal{S}$ is a closed, finite and irreducible set, then there exists a $M_1 > 0$ such that if $t > M_1$, then $P_x(T_{\mathcal{S}} < t) > 1 - \frac{\varepsilon}{2}$ for all $x \in \mathcal{S}$.*

Proof Let $x_1, \ldots, x_{|\mathcal{S}|}$ be any enumeration all $x \in \mathcal{S}$. Then by Theorem 5 and Theorem 6

$$P_x(T_{\mathcal{S}} < \infty) \geqslant P_x(T_{x_1} < \infty) P_{x_1}(T_{x_2} < \infty) \cdots P_{x_{|\mathcal{S}|-1}}(T_{x_{|\mathcal{S}|}} < \infty) = 1 \cdot 1 \cdots 1 = 1.$$

We get the limit by writing $P_x(T_{\mathcal{S}} < \infty)$ as a convergent sum

$$1 = P_x(T_{\mathcal{S}} < \infty) = \sum_{n=1}^{\infty} P_x(T_{\mathcal{S}} = n)$$

then for any $\frac{\varepsilon}{2} > 0$ there exists a M_1 such that

$$\left| 1 - \sum_{n=1}^{M_1} P_x(T_{\mathcal{S}} = n) \right| < \frac{\varepsilon}{2} \Leftrightarrow 1 - \frac{\varepsilon}{2} < \sum_{n=1}^{M_1} P_x(T_{\mathcal{S}} = n) = P_x\left(T_{\mathcal{S}} \leqslant M_1\right). \quad \square$$

Finally, to show consistency of the introduced posterior estimator, it suffices to consider one of the m parallel search processes defined in the Sect. 3, which is denoted by $\{\boldsymbol{G}_t\}_{t \geq 0}$ in the sequel. We recall that $Z = (Z_t)$, where Z_t $t \geqslant 0$, are independent Bernoulli variables such that $P(Z_t = 1) = \alpha_t$. We define first stopping times for enumerating the return times of Z to 1 or times of mixing events, where Z is the process defined in Algorithm 1.

$$T_Z^i := \inf\left\{t > T_Z^{i-1} | Z_t = 1\right\}$$

and the time between two return times

$$\tau_i := T_Z^{i+1} - T_Z^i - 1.$$

With the aid of these notions, the search process $\{\boldsymbol{G}_t\}_{t\geq 0}$ can be regarded as a sequence of Markov chains patched together at the times T_Z^i. Or, more precisely, for $T_Z^i \leq t < T_Z^{i+1}$, the search process $\boldsymbol{G}_t$ is a time homogeneous Markov chain $X^{(i)}$ with the state space $\mathcal{S}$ and a fixed transition kernel defined by (4), Algorithm 3. and an irreducible proposal mechanism. We write this as

$$\boldsymbol{G}_t = X_t^{(i)}, \quad T_Z^i \leq t < T_Z^{i+1}. \tag{15}$$

Next we need to show the irreducibility of the Markov chain $X^{(i)}$. For CPDAGs this will follow by the next result, which is Proposition 4.5 from (Andersson et al. 1997).

Theorem 8 *Consider two graphs* $\mathbf{G}$ *and* $\boldsymbol{H}$ *with same set of nodes in* $\mathcal{S} = \{LPDAG\}$. *Then there exists a finite sequence of LPDAGs* $\boldsymbol{G} \equiv \boldsymbol{G}_1, \ldots, \boldsymbol{G}_k \equiv \boldsymbol{H}$ *such that each consecutive pair* $\boldsymbol{G}_i, \boldsymbol{G}_{i+1}$ *differs by either (i) exactly one undirected edge, (ii) exactly by one directed edge, or (iii) by two edges that form an immorality.*

For the special case of decomposable UGs, (Frydenberg and Lauritzen 1989) shows that search operators adding and deleting of single edges in decomposable UGs is sufficient to guarantee irreducibility.

Corollary 2 *The Markov chain* $X^{(i)}$ *with transition mechanism defined by (4), Algorithm 3 and the search operator in Algorithm 4 is irreducible with respect to* $\mathcal{S}$ *= {UG}.*

The following lemma is an improved version of a similar lemma in (Corander et al. 2006). It and Theorem 6 are needed to establish that the search process, which is a Markov chain inside $\left[T_Z^i, T_Z^{i+1}\right)$, will eventually have enough time during this interval to visit all states of the space $\mathcal{S}$.

Lemma 1 *There exist* $M_2, M_3 > 0$ *such that if* $t > M_3$, *then*

$$P(\tau_i \geqslant M_2) \geqslant 1 - \frac{\varepsilon}{2}.$$

Proof

$$\begin{aligned}
P(\tau_i \geqslant M_2) &= 1 - P(\tau_i < M_2) = 1 - \sum_{t=0}^{M_2-1} P(\tau_i = t) \\
&= 1 - \sum_{t=0}^{M_2-1} \alpha_{t+T_Z^i+1} \prod_{s=T_Z^i+1}^{T_Z^i+t} (1-\alpha_s) \geqslant 1 - \sum_{t=0}^{M_2-1} \alpha_{t+T_Z^i+1} \\
&\geqslant 1 - M_2 \max\left\{\alpha_{t+T_Z^i+1} | t = 0, \ldots, M_2\right\} = 1 - M_2 \alpha_{T_Z^i+1}
\end{aligned}$$

Since $\lim_{t\to\infty} \alpha_t \to 0$ (Definition 1), there exists $M_3 > 0$ such that if $t > M_3$ then $\alpha_t < \frac{\varepsilon}{2M_2}$. □

Next we prove that the search process will eventually visit every state of the state space. We consider the following random time, as in (Corander et al. 2006).

$$T_{\mathcal{S}}^{\boldsymbol{G}} = \inf\{t|\mathcal{S}_t = \mathcal{S}\}, \tag{16}$$

where $\mathcal{S}_t \subseteq \mathcal{S}$ is the part of the model space that has been explored by $\boldsymbol{G}_t$. We define for any x in $\mathcal{S}$ and any $\mathcal{A} \subseteq \mathcal{S}$

$$P_x(\mathcal{S}_t = \mathcal{A}) = P\left(\mathcal{S}_t = \mathcal{A} \mid \boldsymbol{G}_0 = x\right).$$

Theorem 9 *Let $X^{(i)}$ be an irreducible Markov chain with respect to $\mathcal{S}$. Then for any x in $\mathcal{S}$*

$$\mathcal{S}_t \overset{a.s.}{\to} \mathcal{S} \tag{17}$$

Proof We consider the search process $\boldsymbol{G}_t$ and its arbitrary patch chain $X^{(i)}$, in other words $T_Z^i \leq t < T_Z^{i+1}$. As we are going to choose t larger than a certain constant, even i must implicitly be chosen large enough. We have

$$1 - P_x\left(\mathcal{S}_t = \mathcal{S}\right) \leqslant 1 - P_x\left(\mathcal{S}_t = \mathcal{S}, \tau_i > M_1\right).$$

Let now $T_{\mathcal{S}}^{X^{(i)}}$ be defined as the first time, when $X^{(i)}$ defined in (15) has visited every state of $\mathcal{S}$. $T_{\mathcal{S}}^{X^{(i)}}$ equals $+\infty$, if this event does not occur.

Clearly, if the chain $X^{(i)}$ has visited every state of $\mathcal{S}$, then a fortiori the search process $\boldsymbol{G}$ has also visited all of $\mathcal{S}$. Hence

$$P_x\left(\mathcal{S}_t = \mathcal{S} \mid \tau_i > M_1\right) \geqslant P_y\left(T_{\mathcal{S}}^{X^{(i)}} \leqslant M_1 \mid \tau_i > M_1\right)$$

for some $y \in \mathcal{S}$, where $X_{T_Z^i} = y$, where y is given by Algorithm 2. Hence

$$P\left(\tau_i > M_1\right) P_x\left(\mathcal{S}_t = \mathcal{S} | \tau_i > M_1\right) \geqslant P\left(\tau_i > M_1\right) P_y\left(T_{\mathcal{S}}^{X^{(i)}} \leqslant M_1 \mid \tau_i > M_1\right).$$

But the times τ_i are by construction independent of $X^{(i)}$. Thus, for $t > M_1$ we have $P_y\left(T_{\mathcal{S}}^{X^{(i)}} \leqslant M_1 \mid \tau_i > M_1\right) > 1 - \frac{\varepsilon}{2}$ by Theorem 7, since $X^{(i)}$ is irreducible ($\mathcal{S}$ is the whole state space and is clearly closed by irreducibility). Using the bounds above we get

$$\begin{aligned} &1 - P_x\left(\mathcal{S}_t = \mathcal{S}\right) \\ &\quad \leqslant 1 - P\left(\tau_i > M_1\right)\left(1 - \frac{\varepsilon}{2}\right) \end{aligned}$$

and by Lemma 1 there exists a $M_3(M_1)$, such that when $t > \max\{M_3(M_1), M_1\}$

$$\leqslant 1 - \left(1 - \frac{\varepsilon}{2}\right)^2.$$

Now Bernoulli's inequality entails

$$\leqslant 1 - \left(1 - 2\frac{\varepsilon}{2}\right) = \varepsilon,$$

which proves $\lim_{t\to\infty} P_x(\mathcal{S}_t = \mathcal{S}) = 1$. Finally, we can regard $\mathcal{S}_t$ as a random variable defined on a countable outcome space. In that case convergence in probability implies convergence almost surely (see for example formula 6.16 on page 55 in Durrett 1996). □

From this the proof of Theorem 1 is immediate, since the sums in $\hat{P}_t\,(\boldsymbol{G}|\mathbf{X})$ and in $P\,(\boldsymbol{G}|\mathbf{X})$ are finite.

References

Andersson SA, Madigan D, Perlman MD (1996) An alternative Markov property for chain graphs. In: Uncertainty in artificial intelligence: proceedings of the twelfth conference. Morgan Kaufmann, San Francisco, pp 40–48

Andersson SA, Madigan D, Perlman MD (1997) A characterization of Markov equivalence classes for acyclic digraphs. Ann Statist 25:505–541

Andersson SA, Madigan D, Perlman MD (2001) Alternative Markov properties for chain graphs. Scand J Stat 28:33–85

Chickering DM (1995) A transformational characterization of equivalent Bayesian network structures. In: Uncertainty in artificial intelligence: proceedings of the eleventh conference. Morgan Kaufmann, San Francisco, pp 87–98

Chickering DM (2002a) Learning equivalence classes of Bayesian network structures. J Mach Learn Res 2:445–498

Chickering DM (2002b) Optimal structure identification with greedy search. J Mach Learn Res 3:507–554

Cooper G, Hershkovitz E (1992) A bayesian method for the induction of probabilistic networks from data. Mach Learn 9:309–347

Corander J (2003) Bayesian graphical model determination using decision theory. J Multivariate Anal 85:253–266

Corander J, Gyllenberg M, Koski T (2006) Bayesian model learning based on parallel mcmc strategy. Stat Comput 16:355–362

Cowell RG, Dawid AP, Lauritzen SL, Spiegelhalter DJ (1999) Probabilistic networks and expert systems. Springer, New York

Dawid AP (1979) Conditional independence in statistical theory. J Roy Stat Soc B 41:1–31

Dawid AP, Lauritzen SL (1993) Hyper-Markov laws in the statistical analysis of decomposable graphical models. Ann Statist 21:1272–1317

Dellaportas P, Forster J (1999) Markov chain monte carlo model determination for hierarchical and graphical log-linear models. Biometrika 86:615–633

Durrett R (1996) Probability: theory and examples. Duxbury Press, CA

Frydenberg M (1990) The chain graph Markov property. Scand J Stat 17:333–353

Frydenberg M, Lauritzen SL (1989) Decomposition of maximum likelihood in mixed graphical interaction models. Biometrika 76:539–555

Geyer CJ, Thompson EA (1995) Annealing Markov chain Monte Carlo with applications to ancestral inference. J Am Stat Assoc 90:909–920

Gillispie SB, Perlman MD (2001) Enumerating Markov equivalence classes of acyclic digraph models. In: Uncertainty in artificial intelligence: proceedings of the seventeeth conference. Morgan Kaufmann, San Francisco, pp 171–177

Giudici P, Castelo R (2003) Improving Markov chain Monte Carlo model search for data mining. Mach Learn 50:127–158

Giudici P, Green PJ (1999) Decomposable graphical Gaussian model determination. Biometrika 86:785–801

Isaacson DL, Madsen RW (1976) Markov Chains: theory and applications. Wiley, New York

Janzura M, Nielsen J (2006) A simulated annealing-based method for learning Bayesian networks from statistical data. Int J Intell Syst 21:335–348

Jones B, Carvalho C, Dobra A et al (2005) Experiments in stochastic computation for high-dimensional graphical models. Stat Sci 20:388–400

Jordan MI (1998) Learning in graphical models. MIT Press, Cumberland

Koivisto M, Sood K (2004) Exact Bayesian structure discovery in Bayesian networks. J Mach Learn Res 5:549–573

Lam W, Bacchus F (1994) Learning Bayesian belief networks: An approach based on the MDL principle. Comput Intell 10:269–293

Madigan D, Andersson S, Perlman M, Volinsky C (1996) Bayesian model averaging and model selection for Markov equivalence classes of acyclic digraphs. Communtat Theor Meth 25:2493–2519

Madigan D, Raftery A (1994) Model selection and accounting for model uncertainly in graphicalmodels using Occam's window. J Am Stat Assoc 89:1535–1546

Peña JM (2007) Approximate counting of graphical models via MCMC. In: Proceedings of the 11th international conference on artificial intelligence, pp 352–359

Poli I, Roverato A (1998) A genetic algorithm for graphical model selection. J Italian Stat Soc 2:197–208

Riggelsen C (2005) MCMC learning of Bayesian network models by markov blanket decomposition. Springer, New York

Robert C, Casella G (2004) Monte Carlo statistical methods, 2nd edn. Springer, New York

Roverato A, Studený M (2006) A graphical representation of equivalence classes of AMP chain graphs. J Mach Learn Res 7:1045–1078

Sanguesa R, Cortes U (1997) Learning causal networks from data: a survey and a new algorithm to learn possibilistic causal networks from data.. AI Commun 4:1–31

Spirtes P, Glymour C, Scheines R (1993) Causation, prediction and search. Springer, New York

Studený M (1998) Bayesian networks from the point of view of chain graphs. Uncertainty in Artificial Intelligence: In: proceedings of the twelfth conference. Morgan Kaufmann, San Francisco, pp 496–503

Sundberg R (1975) Some results about decomposable (or markov-type) models for multidimensional contingency tables: distribution of marginals and partitioning of tests. Scand J Stat 2:771–779

Suzuki J (1996) Learning Bayesian belief networks based on the minimum description length principle. In: International Conference Machine on Learning, Morgan Kaufmann, San Francisco, pp 462–470

Suzuki J (2006) On strong consistency of model selection in classification. IEEE Trans Inform Theory 52:4767–4774

van Laarhoven PJM, Aarts EHJ (1987) Simulated annealing: theory and applications. Kluwer, Norwell

Verma E, Pearl J (1990) Equivalence and synthesis of causal models. In: Uncertainty in artificial intelligence: proceedings of the sixth conference. Elsevier, New York, pp 220–227

Volf M, Studený M (1999) A graphical characterization of the largest chain graphs. Int J Approx Reason 20:209–236

Wedelin D (1996) Efficient estimation and model selection in large graphical models. Stat Comput 6:313–323

Whittaker J (1990) Graphical models in applied multivariate statistics. Wiley, Chichester

Wong F, Carter C, Kohn R (2003) Efficient estimation of covariance selection models. Biometrika 90:809–830