Cycle-Cutset sampling for Bayesian networks*

Bozhena Bidyuk and Rina Dechter

Information and Computer Science,
University of California - Irvine,
Irvine, CA 92697-3425, USA
{bbidyuk,dechter}@ics.uci.edu

Abstract. The paper presents a new sampling methodology for Bayesian networks called *cutset sampling* that samples only a subset of the variables and applies exact inference for the others. We show that this approach can be implemented efficiently when the sampled variables constitute a cycle-cutset for the Bayesian network and otherwise it is exponential in the induced-width of the network's graph, whose sampled variables are removed. Cutset sampling is an instance of the well known Rao-Blakwellisation technique for variance reduction investigated in [5, 2, 16]. Moreover, the proposed scheme extends standard sampling methods to non-ergodic networks with ergodic subspaces. Our empirical results confirm those expectations and show that cycle cutset sampling is superior to Gibbs sampling for a variety of benchmarks, yielding a simple, yet powerful sampling scheme.

1 Introduction

Sampling methods for Bayesian networks are commonly used approximation techniques, applied successfully where exact inference is not possible due to prohibitive time and memory demands. In this paper, we focus on Gibbs sampling, a member of the Markov Chain Monte Carlo sampling methods group for Bayesian networks [6, 7, 17]. Given a Bayesian network over the variables $X = \{X_1, ..., X_n\}$, and evidence e, Gibbs sampling [6, 7, 17] generates a set of samples $\{x^t\}$ from $P(X|e)$ where each sample $x^t = \{x_1^t, ..., x_n^t\}$ is an instantiation of all the variables in the network. It is well-known that a function $f(X)$ can be estimated using the generated samples by:

$$E[f(X)|e] \cong \frac{1}{T} \sum_t f(x^t) \tag{1}$$

where T is the number of samples. Namely, given enough samples, the estimate converges to the exact value. The central query of interest over Bayesian networks is computing the posterior marginals $P(x_i|e)$ for each value x_i of variable X_i. For this query, the above equation reduces to counting the fraction of occurrences of

* This work was supported in part by NSF grant IIS-0086529 and MURI ONR award N00014-00-1-0617.

$X_i = x_i$ in the samples. A significant limitation of all existing sampling schemes, including Gibbs sampler, is the increase in the statistical variance for high-dimensional spaces. In addition, standard sampling methods fail to converge to the target distribution when the network is not ergodic.

In this paper, we present a sampling scheme for Bayesian networks that addresses both of these limitations by sampling from a subset of the variables. It is rooted in the well established Rao-Blakwellisation methodology for sampling that was developed in the past years by various authors, most notably $[5, 2, 16]$. Based on the Rao-Blackwell theorem ([8]), it is easy to show that sampling from a subspace (if feasible computationally) can reduce the variance and therefore yield faster convergence to the target function.

The basic Rao-Blackwellisation scheme can be described as follows. Suppose we partition the space of variables X into two subsets C and Z. It can be shown that if we can efficiently compute $P(c|e)$ and $E[f(C, Z)|c, e]$ (by summing out Z in both cases), then we can perform sampling only on C generating $c^1, c^2, ..., c^T$ and approximate the quantity of interest by:

$$E[f(X)|e] \cong \frac{1}{T} \sum_t E[f(c^t, Z)|c, e] \qquad (2)$$

If function $f(X)$ is a posterior marginal of node X_i, then $f(X)|e = P(x_i|e)$ and $f(c^t, Z)|c, e = P(x_i|c^t, e)$, then Equation (2) instantiates to:

$$P(x_i|e) \cong \frac{1}{T} \sum_t P(x_i|c^t, e) \qquad (3)$$

In this paper, we propose to use the above scheme when the subspace C is such that conditioning on C yields a sparse Bayesian network where exact inference is polynomial, such as when C is a cycle-cutset. The proposed scheme is called *cutset sampling*. This yields a special application of Rao-Blackwellisation for sampling in Bayesian networks that offers two-fold benefits over regular sampling: 1. improved convergence and 2. convergence in non-ergodic networks.

Indeed, we show empirically that cycle-cutset sampling converges faster not only in terms of number of samples, as dictated by theory, but it is also time-wise cost-effective on all the benchmarks tried (CPCS networks, random networks, and coding networks). We also demonstrate the applicability of this scheme to non-ergodic networks such as Hailfinder network and coding networks.

The approach we propose is simple, however, to the best of our knowledge, it was not yet presented for general Bayesian networks except for the special case of Dynamic Bayesian networks [4]. In that paper, the authors apply Rao-Blackwellisation to particle filtering that iterates along the timeline, by selecting a specific sampling set C. Hence, the current paper extends the work of [4] to general Bayesian networks. Following background (Section 2), the paper presents cutset-sampling and analyzes its complexity (Section 4), provides empirical evaluation in Section 6 and concludes in Section 7.

2 Background

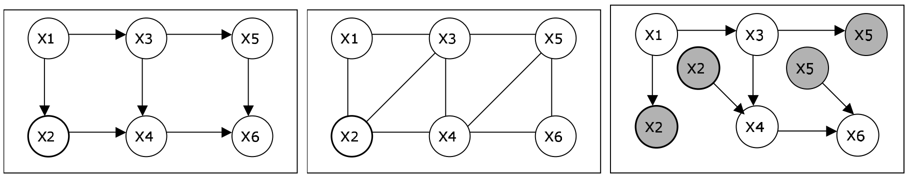

Fig. 1. Bayesian network (left), its moral graph(center), and conditioned polytree (right) (conditioned on $C = \{X_2, X_5\}$).

Definition 1 (belief networks). *Let* $X = \{X_1, ..., X_n\}$ *be a set of random variables over multi-valued domains* $D(X_1), ..., D(X_n)$. *A* belief network (BN) *is a pair* (G, P) *where* G *is a directed acyclic graph on* X *and* $P = \{P(X_i|pa_i)|i = 1, ..., n\}$ *is the set of conditional probability matrices associated with each* X_i. *A belief network is ergodic if any assignment* $x = \{x_1, ..., x_n\}$ *has non-zero probability, defined by* $P(x_1, ..., x_n) = \Pi_{i=1}^n P(x_i|x_{pa(X_i)})$. *An evidence* e *is an instantiated subset of variables* E. *The moral graph of a belief network is obtained by connecting the parents of the same child and eliminating the arrows. Figure 1 shows a belief network(left) and its moral graph(center).*

Definition 2 (induced-width). *The* width *of a node in an ordered undirected graph is the number of the node's neighbors that precede it in the ordering. The width of an ordering d, denoted w(d), is the width over all nodes. The induced width of an ordered graph, w*(d), is the width of the ordered graph obtained by processing the nodes from last to first. When node X is processed, all its preceding neighbors are connected. The resulting graph is called induced graph or triangulated graph.*

Definition 3 (induced-width, cycle-cutset). *A* cycle *in G is a path whose two end-points coincide. A* cycle-cutset *of undirected graph G is a set of vertices that contains at least one node in each cycle in G. A graph is* singly connected *(also called a* polytree*), if its underlying undirected graph has no cycles. Otherwise, it is called* multiply connected. *A* loop *in D is a subgraph of D whose underlying graph is a cycle. A vertex v is a* sink *with respect to loop* $\mathcal{L}$ *if the two edges adjacent to v in* $\mathcal{L}$ *are directed into v. A vertex that is not a sink with respect to a loop* $\mathcal{L}$ *is called an* allowed *vertex with respect to* $\mathcal{L}$. *A* cycle-cutset *of a directed graph D is a set of vertices that contains at least one allowed vertex with respect to each loop in D.*

2.1 Gibbs sampling

Gibbs sampling generates samples from $\hat{P}(X|e)$ which converges to $P(X|e)$ as the number of samples increases [18, 17] as long as the network is ergodic. Given a Bayesian network $\mathcal{B}$, Gibbs sampling generates a set of samples x^t where t denotes a sample and and x_i^t is the value of X_i in sample t. Given a sample $x^{t-1} = \{x_1^{t-1}, x_2^{t-1}, ..., x_n^{t-1}\}$ (evidence variables remain fixed), a new sample x^t is generated by assigning a new value x_i^t to each variable X_i in some order. Value x_i^t is computed by sampling from the conditional probability distribution: $P(x_i) = P(x_i|x_1^t, x_2^t, x_{i-1}^t ..., x_{i+1}^{t-1}, ..., x_n^{t-1}) = P(x_i|markov^t(x_i))$, where $markov^t(x_i)$ is the assignment in sample t to the Markov blanket of variable X_i which includes its parents, children, and parents of its children.

Once all the samples are generated, we can answer any query over the samples. In particular, computing a posterior marginal belief $P(x_i|e)$ for each variable X_i can be estimated by counting samples where $X_i = x_i$:

$$\hat{P}(x_i|e) = \frac{1}{T} \sum_{t=1}^{T} \delta_{x_i}(x^t) \tag{4}$$

(here $\delta_{x_i}(x^t) = 1$ if $x_i^t = x_i$ and equals 0 otherwise) or by averaging the conditional marginals (known as mixture estimator):

$$\hat{P}(x_i|e) = \frac{1}{T} \sum_{t=1}^{T} P(x_i|markov^t(x_i)) \tag{5}$$

This method is likely to converge faster than simple counting [18]. The Markov blanket of X_i ([18]) is given explicitly by:

$$P(x_i|markov^t(x_i)) = \alpha P(x_i|x^t_{pa(X_i)}) \prod_{\{j|X_j \in ch_j\}} P(x_j^t|x^t_{pa_j}) \tag{6}$$

Thus, generating a complete new sample requires $O(n \cdot r)$ multiplication steps where r is the maximum family size and n is the number of variables. Subsequently, computing the posterior marginals is linear in the number of samples.

3 Augmentation Schemes

Variable augmentation schemes exist that allow to improve the convergence properties of simple Gibbs sampler. The two main approaches are blocking (grouping variables together and sampling simultaneously) and Rao-Blackwellisation (integrating out some of the random variables). Given Bayesian network with three random variables: X, Y, and Z, we can schematically describe those three sampling schemes as follows:

1. Rao-Blackwellised: sample x from $P(x|y)$, sample y from $P(y|x)$ integrating out random variable z.
2. Blocking Gibbs: samples values from $P(x|y, z)$, $P(y, z|x)$
3. Standard Gibbs: samples values from $P(x|y, z)$, $P(y|x, z)$, $P(z|x, y)$

As shown in [16], the blocking Gibbs sampling scheme, where several variables are grouped together and sampled simultaneously, is expected to converge faster than standard Gibbs sampler. Variations to this scheme have been investigated in [10, 13]. Still, in a blocking Gibbs sampler a sample is an instantiation of all the variables in the network, same as standard Gibbs sampler. The Rao-Blackwellised sampling scheme actually allows to integrate some of the random variables out, thus reducing sampling space, and it is expected to converge the fastest [16]. Thus, of the two basic data augmentation scheme, namely Rao-Blackwellisation and Blocking, Rao-Blackwellisation is generally preferred.

The caveat in the utilization of the Rao-Blackwellised sampling scheme is that computation of the probabilities $P(x|y)$ and $P(y|x)$ must be efficient. In case of Bayesian networks, the task of integrating variables out equates to performing exact inference on the network where evidence nodes and sampling nodes are observed and its time complexity is exponential in the network size. Taken *a priori* that performance of the sampler will be severely impacted when many variables are integrated out, Rao-Blackwellisation has been applied only to a few special cases of Bayesian networks. In particular, it has been applied to the Particle Filtering (using importance sampling) method for Dynamic Bayesian networks [4] in cases where some of the variables can be integrated out easily either because they are conditionally independent given the sampled variables (plus evidence) or because their probability distribution permits tractable exact inference (for example, using Kalman filter).

In this paper, we define a general scheme for Rao-Blackwellised sampling for Bayesian networks (see Section 4) and show that Rao-Blackwellisation can be done efficiently when sampling set is a cycle-cutset of the Bayesian network. We demonstrate empirically for several networks that we can compute a new sample faster using cutset sampling scheme than standard Gibbs sampler. The gain is easily explained. In a Bayesian network of size $|X| = N$, Gibbs sampler maybe able to compute individual probabilities $P(x|markov^t(x))$ fast, but it has to repeat this computation N times. In Rao-Blackwellised scheme, where most variables are integrated out and sampling set $C \in X$ is of size $|C| = K$, $K < N$, it may take longer to compute $P(x|c, e)$, but we only have to repeat this computation K times (potentially, K can be much smaller than N). Most importantly, fewer samples are needed for convergence.

4 Cutset sampling

This section presents the cutset sampling method. As noted in the introduction, the basic scheme partitions variables X into two subsets C and Z. If we can efficiently compute $P(c|e)$ and $P(x_i|c^t, e)$, then we can sample only values of C efficiently and approximate the quantity of interest via equation (3).

4.1 Cutset sampling algorithm.

The cutset sampling algorithm is given in Figure 2. Given a subset of cutset variables C=$\{C_1, C_2, ..., C_m\}$, it generates samples c^t, t=1...T, over subspace

C. Here, c^t is an instantiation of the variables in C. Similarly to Gibbs sampling, we generate a new sample c^t by sampling a value c_i^t from the probability distribution $P(c_i|c_1^{t+1}, c_2^{t+1}, ...c_{i-1}^{t+1}, c_{i+1}^t, ..., c_m^t, e)$ for each C_i. We will denote $c_{(i)}^t = c_1^{t+1}, c_2^{t+1}, ...c_{i-1}^{t+1}, c_{i+1}^t, ..., c_m^t$ for conciseness.

The key idea is that the relevant conditional distributions (eq. (7)) can be computed by exact inference algorithms whose complexity is tied to the network's structure and is improved by conditioning. We use $JTC(X, e)$ as a generic name for a class of variable-elimination or join tree-clustering algorithms that compute the exact posterior beliefs for a variable X given evidence e [15, 3, 11]. It is known that the complexity of $JTC(X, e)$ is time and space exponential in the induced-width of the network's moral graph whose evidence variables E are removed.

Cutset Sampling
Input: A belief network ($\mathcal{B}$), cutset $C = \{C_1, ..., C_m\}$, evidence e.
Output: A set of samples c^t, $t = 1...T_c$.
1. **Initialize:** Assign random value c_i^0 to each $C_i \in C$ and assign e.
2. **Generate samples:**
   For t = 1 to T, generate a new sample c^t:
   For i = 1 to m, compute new value c_i^t for variable C_i as follows:
1. Using algorithm **join-tree clustering** $JTC(C_i, c_{(i)}^t, e)$, compute:

$$P(c_i) = P(c_i|c_{(i)}^t, e) \tag{7}$$

2. Sample a new value c_i^t for C_i, from (7).
End For i
End For t

Fig. 2. *Cutset sampling* Algorithm

4.2 Computing the posterior marginals.

Once the samples over the cutest C are available, we can compute the posterior beliefs of all variables as follows. For each cutset variable $C_i \in C$ (excluding evidence variables), the posterior marginals can be computed as in Gibbs sampling:

$$\hat{P}(c_i|e) = \frac{1}{T} \sum_t P(c_i|c_{(i)}^t, e) \tag{8}$$

If we record the distributions computed during sample generation (equation (7)), these quantities will be readily available for summation.

For each non-cutset variable $X_i \in X \backslash E, C$, and every sample c^t, $P(x_i|c^t, e)$ can be computed over the Bayesian network conditioned on c^t and e, by $JTC(X_i, c^t, e)$:

$$\hat{P}(c_i|e) = \frac{1}{T} \sum_t P(x_i|c^t, e) \tag{9}$$

Note that the probability distribution $P(x_i|c^t, e)$ can be computed as soon as sample c^t is generated. Namely, it is sufficient to keep a running sum (eq. 3) (relative to samples c^t) for each value x_i of each variable X_i.

We provide a proof of the convergence of this general scheme in Section 5. Namely, computing $P(x_i|e)$ by cutset sampling is (1) guaranteed to converge to the exact quantities. In general, cutset sampling requires fewer samples to converge than full sampling as a result of Rao-Blackwell theorem [2, 8, 16].

Example. Consider again a belief network shown in Figure 1. When sampling from set $C = \{X_2, X_5\}$ (although there is a better cutset $C = \{X_3\}$), we will have to compute for each sample t the probabilities $P(x_2|x_5^{t-1})$ and $P(x_5|x_2^t)$. These probabilities can be computed using belief propagation over the singly connected network (Figure 1, right) or bucket elimination in linear time. For each new value of variables X_2 and X_5, we profane the updated messages through the (singly-connected) network. The desired joint $P(x_2', x_5', e)$ can be computed at any variable and then normalized to yield the conditional distribution.

4.3 Complexity

Cutset sampling uses the *adjusted induced width* w, to control the size of the sampling set and thus can adjust the trade-off between sampling and inference. Given an undirected graph $G = (V, E)$, if C is a subset of V such that when removed from G, the induced width of the resulting graph is less or equal w, then C is called a *w-cutset* of G and the *adjusted induced width* of G relative to C is w. The cycle-cutset of a graph is a 1-cutset.

Clearly, computing a new sample c^t in cutset-sampling is more complex (step 1) than Gibbs sampling. However, it is still very efficient when the cutset C is a cycle-cutset of the Bayesian network (w=1). In this case, JTC reduces to belief propagation algorithm [18, 19] that can compute the joint probability $P(c_i, c_{(i)}^t, ..., c_m^t, e)$ in linear time and then normalize it relative to C_i yielding equation (7) (details are omitted). When C is a w-cutset, the complexity of JTC (equation 7) is exponential in w and will dominate the complexity of generating the next sample. Therefore:

Theorem 1 (Complexity of sample generation). *The complexity of generating a sample by cutset sampling with cutset C is $O(m \cdot d \cdot n \cdot d^w)$ where C is a w-custet of size m, d bounds the variables domain size, and n is the number of nodes.*

Corollary 1. *If C is a cycle-cutset, the complexity of generating a sample by cycle-cutset sampling is linear in the size of the network.*

Computing $P(X_i|e)$ using equation (3) requires computing $P(x_i|c^t, e)$ for each variable. The complexity of this computation by $JTC(X_i, c^t, e)$ is also exponential in w, the adjusted induced width relative to cutset:

Theorem 2. *Given a w-cutset C, the complexity of computing the posterior of all the variables using cutset sampling over T samples is $O(T \cdot n \cdot d^w)$.*

Corollary 2. *If C is a cycle-cutset, the complexity of computing the posterior of all the variables by cycle-cutset sampling is linear in the size of the network.*

In conclusion, when sampling over a cycle-cutset C, both sampling and estimating the marginal posterior are linear in the size of the network and the number of samples.

5 Convergence of cutset-sampling.

In this section we will show that $\hat{P}(c_i|e)$ and $\hat{P}(x_i|e)$ as defined in equations (8) and (9) converge to the correct probabilities $P(c_i|e)$ and $P(x_i|e)$ respectively.

Theorem 3 (cutset convergence). *Given a network $\mathcal{B}$ over X and a subset of evidence variables E, and given a cutset C, assuming $\hat{P}(c_i|e)$ and $\hat{P}(x_i|e)$ were computed by equations (8) and (9) over the cutset sample, then $\hat{P}(c_i|e) \rightarrow P(c_i|e)$ and $\hat{P}(x_i|e) \rightarrow P(x_i|e)$ as the number of samples T_c increses.*

While the result of theorem 3 is implied by the Rao-Blackwell theorem, the proof from first principles is simple enough.

Proof. Let $|C| = m$. Let $|X| = n$. The computation of $\hat{P}(c_i|e)$ is done exactly in the same way as in Gibbs sampling. There are several different ways to prove convergence of Gibbs sampling and we will not repeat them here. Therefore, based on the correct convergence of Gibbs sampling we can conclude that $\hat{P}(c_i|e) \rightarrow P(c_i|e)$ as the number of samples increases.

Consider now a variable X_i not in C and not in E. We could write the posterior distribution of variable X_i as follows: $P(x_i|e) = \sum_c P(x_i|c,e)P(c|e)$.

Assume that we have generated a collection of samples $c^1, c^2, ..., c^T$ from the correct distribution $P(C|e)$. Let $m(c)$ be the number of times c occurs in the samples. Then, for each tuple $C = c$:

$$P(c|e) = \frac{m(c)}{T} \tag{10}$$

After we substitute the right hand side of the equation 10 in the expression for $P(x_i|e)$:

$$P(x_i|e) = \sum_c P(x_i|c,e)\frac{m(c)}{T} \tag{11}$$

Factoring out $\frac{1}{T}$ we get:

$$P(x_i|e) = \frac{1}{T}\sum_c P(x_i|c,e)m(c). \tag{12}$$

Clearly, $\sum_c m(C) = T$. Therefore, we can sum over T instead of summing over instantiations of C, yielding:

$$\sum_c P(x_i|c,e)m(c) = \sum_{t=1}^{T} P(x_i|c,e) \tag{13}$$

After replacing the sum over C in (12) with the sum over T, we get:

$$P(x_i|e) = \frac{1}{T}\sum_{t=1}^{T} P(x_i|c,e) \qquad (14)$$

Therefore we obtained expression (14), assuming that $\frac{m(c)}{T}$ converges to the exact $P(C|e)$. Since $\hat{P}(c_i|e)$ converges to $P(c_i|e)$ in cutset-sampling, as we have already shown, then we can conclude that $\hat{P}(x_i|e) \to P(x_i|e)$. $\square$

6 Experiments

We compared cycle-cutset sampling with full Gibbs sampling on several CPCS networks, random networks, Hailfinder network, and coding networks. Generally, we are interested in how much accuracy we can achieve in a given period of time. Therefore, we provide here figures showing accuracy of the Gibbs and cycle-cutset sampling as a function of time. For comparison, we also show the accuracy of Iterative Belief Propagation algorithm (IBP) after 25 iterations. IBP is an iterative message-passing algorithm that performs exact inference in Bayesian networks without loops ([18]). It can also be applied to Bayesian networks with loops to compute approximate posterior marginals. The advantage of IBP as an approximate algorithm is that it is very efficient. It requires linear space and usually converges very fast. IBP was shown to perform well in practice ([9, 12]) and is considered the best algorithm for inference in coding networks where finding the most probable variable values equals the decoding process.

For each Bayesian network $\mathcal{B}$ with variables $X = \{X_1, ..., X_n\}$, we computed exact posterior marginals $P(x_i|e)$ using bucket-tree elimination and computed the mean square error (MSE) in the approximate posterior marginals $\hat{P}(x_i|e)$ for each approximation scheme:

$$MSE = \frac{1}{\sum_i |D(x_i)|} \sum_i \sum_{D(x_i)} (P(x_i|e) - \hat{P}(x_i|e))^2$$

and averaged MSE over the number of instances tried. In all networks, except for coding networks, evidence nodes were selected at random. The cutset was always selected so that evidence and sampling nodes together constitute a cycle-cutset of the network using the mga algorithm ([1]).

CPCS networks. We considered four CPCS networks derived from the Computer-based Patient Case Simulation system. The largest network, cpcs422b, consisted of 422 nodes with induced width w*=23. With evidence, its cycle-cutset size was 42. The results are shown in Figures 3-4. Each chart title specifies network name, number of nodes in the network N, the size of evidence set $|E|$, size of cycle-cutset (sampling set) $|C|$, and induced width w* of the network instance. For all four CPCS networks, we observed that the cutset sampling is far better than Gibbs sampling. In case of cpcs179 (Figure 6, middle), cpcs360b (Figure 6,

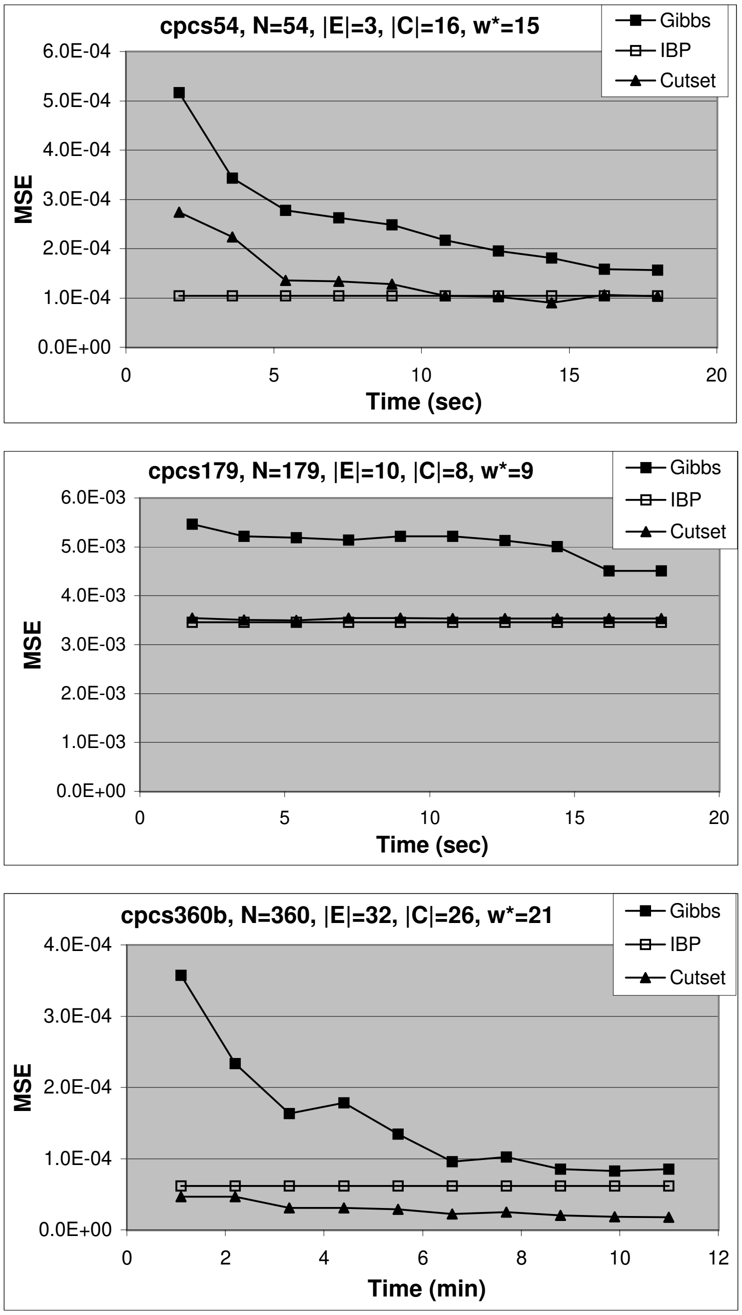

Fig. 3. Comparing cycle-cutset sampling, Gibbs sampling and IBP on CPCS networks averaged over 3 instances each. MSE as a function of time.

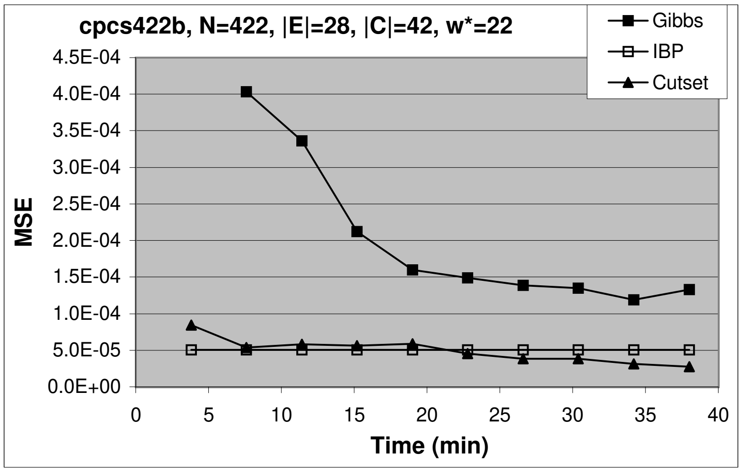

Fig. 4. Comparing cycle-cutset sampling, Gibbs sampling and IBP on cpcs422b network averaged over 2 instances. MSE as a function of time.

bottom), and cpcs422b (Figure 6) cutset sampling achieves even greater accuracy than IBP. Gibbs sampling does not converge on cpcs179 due to non-ergodic properties of the network. The cutset sampling overcomes this limitation because the cycle-cutset selected is an ergodic subspace.

Random networks. We generated a set of random networks with bi-valued nodes. Each network contained total of 200 nodes. The first 100 nodes, $\{X_1, ..., X_{100}\}$, were designated as root nodes. Each non-root node X_i was assigned 3 parents selected randomly from the list of predecessors $\{X_1, ..., X_{i-1}\}$. The conditional probability table values $P(X_i = 0|pa(X_i))$ were chosen randomly from a uniform distribution. We collected data for 10 instances (Figure 5, top). Cutset sampling always converged faster than Gibbs sampling.

2-Layer networks. We generated a set of random 2-layer networks with bi-valued nodes. Each network contained 50 root nodes (first layer) and a total of 200 nodes. Each non-root node (second layer) was assigned a maximum of 3 parents selected at random from the root nodes. The conditional probability table values $P(X_i = 0|pa(X_i))$ were chosen randomly from uniform distribution. We collected data for 10 instances (Figure 5, middle). On those types of networks, Iterative Belief Propagation often does not perform well. And, as our experiments show, cutset sampling outperfoms both Gibbs sampling and IBP (although it takes longer time to converge than IBP).

Coding Networks. We experimented with coding networks with 100 nodes (25 coding bits, 25 parity check bits). The results are shown in Figure 5, bottom. Those networks had cycle-cutset size between 12 and 14 and induced width between 13 and 16. The parity check matrix was randomized; each parity check bit had three parents. We computed MSE over all coding bits and averaged over 10 networks. Coding networks are not ergodic due to the deterministic parity

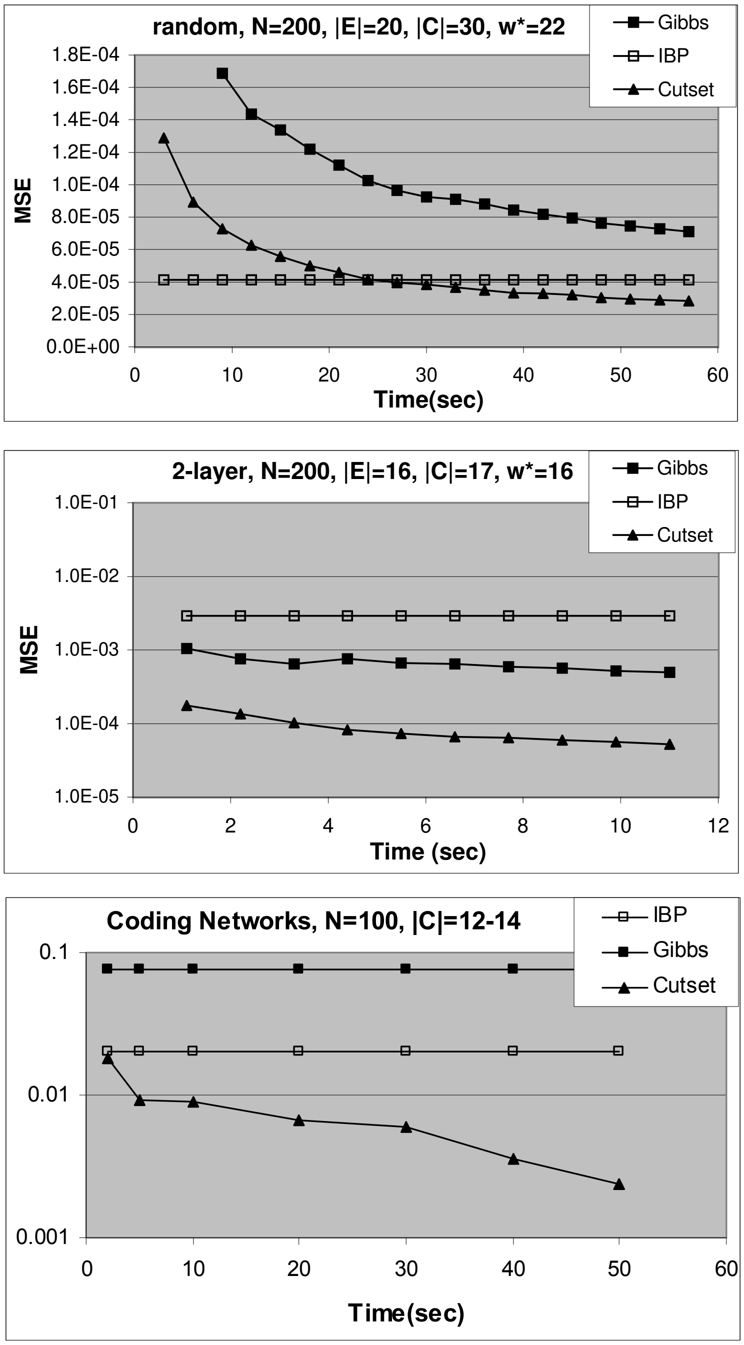

Fig. 5. Comparing cycle-cutset sampling, Gibbs sampling and IBP on random networks (top), 2-layer random networks (middle), and coding networks, σ=0.4 (bottom), averaged over 10 instances each. MSE as a function of time.

check function. As a result, Gibbs sampling does not converge. At the same time, the subspace of code bits only is ergodic and cutset sampling, that samples a subset of coding bits, converges and generates results comparable to those of IBP. In practice, IBP is certainly preferable for coding networks since the size of the cycle-cutset grows linearly with the number of code bits.

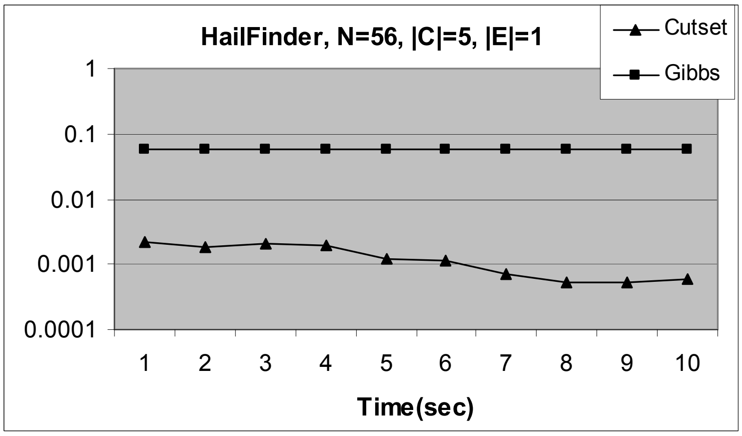

Fig. 6. Comparing cycle-cutset sampling and Gibbs sampling on Hailfinder network, 1 instance. MSE as a function of time.

Hailfinder network. Hailfinder is a non-ergodic network with many deterministic relationships. It has 56 nodes and cycle-cutset of size 5. Indeed, this network is easy to solve exactly. We used this network to compare the behavior cutset sampling and Gibbs sampling in non-ergodic networks. As expected, Gibbs sampling fails while cycle cutset sampling computes more accurate marginals and its accuracy continues to improve with time (Figure 6).

In summary, the empirical results demonstrate the cycle-cutset is cost-effective time-wise and is superior to Gibbs sampling. We measured the ratio $R = \frac{M_g}{M_c}$ of the number of samples generated by Gibbs M_g to the number of samples generated by cycle-cutset sampling M_c in the same time period. For cpcs54, cpcs179, cpcs360b, and cpcs422b the ratios were correspondingly 1.4, 1.7, 0.6, and 0.5. We also obtained R=2.0 for random networks and R=0.3 for random 2-layer networks. While cutset sampling algorithm often takes more time to generate a sample, it produced substantially better results overall due to its variance reduction. In some cases, cutset sampling could actually compute samples faster than Gibbs sampler. in which case the improvement in the accuracy was due to both large sample set and variance reduction. Cutset sampling also achieves better accuracy than IBP on some CPCS and random networks although takes more time to achieve same or better accuracy. In 2-layer networks and coding networks, cycle-cutset sampling achieves the IBP level of accuracy very quickly and is able to substantially improve with time.

7 Related Work and Conclusions

We presented a sampling scheme called cutset sampling for Bayesian networks that samples only a subset of variables in the network. The remaining nodes are marginalised out (by inference) which is an instance of a technique known as Rao-Blackwellisation. As we showed theoretically and empirically, cutset sampling: (1) improves convergence rate due to sampling from lower-dimensional space and (2) allows sampling from non-ergodic network that have ergodic subspace. The resulting scheme is a simple yet powerful extension of sampling in Bayesian networks that is likely to dominate regular sampling for any sampling method. While we focused on Gibbs sampling, other sampling techniques, with better convergence characteristics, can be implemented with cutset sampling as long as they permit to exploit Bayesian network structure in a similar manner.

Previously, sampling from a subset of variables was successfully applied to particle sampling for Dynamic Bayesian networks (DBNs) [4]. Indeed, the authors demonstrated that sampling from a subspace combined with exact inference yields a better approximation. Our scheme offers an elegant way of extending [4] and combining inference and sampling in Bayesian networks.

A different combination of sampling and exact inference for join trees was described in [14] and [13]. Both papers proposed to use sampling to estimate the probability distribution in each cluster from which they compute messages sent to the neighboring clusters. In this approximation scheme, sampling is always performed locally (within the cluster) and thus, the algorithm must rely on the approximated messages received from neighbors when generating new samples. In [14], the authors attempt to remedy this problem by iterative refinement. Our cutset-sampling algorithm does not encounter such problems since it takes into account the global state of the network when generating a new sample. Cutset sampling can also be seen as an approximation to cycle-cutset coditioning ([18]).

In [10], exact inference was used in combination with blocking Gibbs sampling. The major differences between our cutset sampling approach and one proposed in [10] are that first, in the proposed blocking Gibbs sampling, a sample consists of all the variables in the network (as usual) while cutset sampling never assigns values to those variables that are integrated out; second, in [10], exact inference is used to perform joint sampling step for a group of variables while cutset sampling uses exact inference to integrate variables out.

The direction of our future work is to investigate methods for finding a sampling set with good convergence properties. Some of the factors that strongly affect convergence of MCMC methods are the sampling set size, the complexity of sample generation, and the correlations between variables. Reducing sampling set size generally leads to a reduction in the sampling variance due to Rao-Blackwellisation, but it also results in the increased complexity of exact inference when generating a new sample. Another factor is strong correlations between sampled variables (deterministic probabilities, present in non-ergodic networks, are an extreme example of strong correlation). If two variables are strongly dependent, it is preferred to either integrate one of them out or group them together and sample jointly (as in blocking Gibbs sampler) (see [16]). Taking above into consideration, a good sampling set could be defined as a minimal w-cutset with a small w and with all strongly-correlated variables removed.

References

1. A. Becker, R. Bar-Yehuda, and D. Geiger. Random algorithms for the loop cutset problem. In *Uncertainty in AI (UAI'99)*, 1999.
2. G. Casella and C. P. Robert. Rao-blackwellisation of sampling schemes. *Biometrika*, 83(1):81–94, 1996.
3. R. Dechter. Bucket elimination: A unifying framework for reasoning. *Artificial Intelligence*, 113:41–85, 1999.
4. A. Doucet, N. de Freitas, K. Murphy, and S. Russell. Rao-blackwellised particle filtering for dynamic bayesian networks. In *Uncertainty in AI*, pages 176–183, 2000.
5. A. E. Gelfand and A. F. M. Smith. Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical Association*, 85:398–409, 1990.
6. S. Geman and D. Geman. Stochastic relaxations, gibbs distributions and the bayesian restoration of images. *IEEE Transaction on Pattern analysis and Machine Intelligence (PAMI-6)*, pages 721–42, 1984.
7. W. Gilks, S. Richardson, and D. Spiegelhalter. *Markov chain Monte Carlo in practice.* Chapman and Hall, 1996.
8. M. H. De Groot. *Probability and Statistics, 2nd edition.* Addison-Wesley, 1986.
9. K. Kask I. Rish and R. Dechter. Empirical evaluation of approximation algorithms for probabilistic decoding. In *Uncertainty in AI (UAI'98)*, 1998.
10. C. Jensen, A. Kong, and U. Kjaerulff. Blocking gibbs sampling in very large probabilistic expert systems. *International Journal of Human Computer Studies. Special Issue on Real-World Applications of Uncertain Reasoning.*, pages 647–666, 1995.
11. F.V. Jensen, S.L Lauritzen, and K.G. Olesen. Bayesian updating in causal probabilistic networks by local computation. *Computational Statistics Quarterly*, 4:269–282, 1990.
12. Y. Weiss K. P. Murphy and M. I. Jordan. Loopy belief propagation for approximate inference: An empirical study. In *Uncertainty in AI (UAI'99)*, 1999.
13. Uffe Kjærulff. Hugs: Combining exact inference and gibbs sampling in junction trees. In *Uncertainty in AI*, pages 368–375. Morgan Kaufmann, 1995.
14. D. Koller, U. Lerner, and D. Angelov. A general algorithm for approximate inference and its application to hybrid bayes nets. In *Uncertainty in AI*, pages 324–333, 1998.
15. S.L. Lauritzen and D.J. Spiegelhalter. Local computation with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society, Series B*, 50(2):157–224, 1988.
16. W.H. Wong Liu, J. and A. Kong. Covariance structure of the gibbs sampler with applications to the comparison of estimators and augmentation schemes. *Biometrika*, pages 27–40, 1994.
17. D.J.C MacKay. Introduction to monte carlo methods. In *Proceedings of NATO Advanced Study Institute on Learning in Graphical Models. Sept 27-Oct 7*, pages 175–204, 1996.
18. J. Pearl. *Probabilistic Reasoning in Intelligent Systems.* Morgan Kaufmann, 1988.
19. M. A. Peot and R. D. Shachter. Fusion and proagation with multiple observations in belief networks. *Artificial Intelligence*, pages 299–318, 1992.