An Empirical Study of w-Cutset Sampling for Bayesian Networks

Bozhena Bidyuk
Information and Computer Science
University Of California Irvine
Irvine, CA 92697-3425
bbidyuk@ics.uci.edu

Rina Dechter
Information and Computer Science
University Of California Irvine
Irvine, CA 92697-3425
dechter@ics.uci.edu

Abstract

The paper studies empirically the time-space trade-off between sampling and inference in the *cutset sampling* algorithm. The algorithm samples over a subset of nodes in a Bayesian network and applies exact inference over the rest. As the size of the sampling space decreases, requiring less samples for convergence, the time for generating each single sample increases. Algorithm w-cutset sampling selects a sampling set such that the induced-width of the network when the sampling set is observed is bounded by w, thus requiring inference whose complexity is exponentially bounded by w. In this paper, we investigate the performance of w-cutset sampling as a function of w. Our experiments over a range of randomly generated and real benchmarks, demonstrate the power of the cutset sampling idea and in particular show that an optimal balance between inference and sampling benefits substantially from restricting the cutset size, even at the cost of more complex inference.

1 Introduction

Sampling is a common method for approximate inference in Bayesian networks. It is often the only feasible approach (that has some guarantees) when exact inference is impractical due to prohibitive time and memory demands. A significant limitation of all existing sampling schemes, however, 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.

Two well-known variance reduction schemes for sampling methods are blocking([11]) and Rao-Blackwellisation ([6, 3]). Given two strongly correlated variables, we can either sample them simultaneously (blocking) or integrate one of them out (Rao-Blackwellisation). It can be shown that integration is preferred [16]. While Rao-Blackwellisation can be applied in the context of any sampling algorithm, we focus on Gibbs sampling, a member of MCMC sampling methods group, and its Rao-Blackwellised derivative, cutset sampling introduced recently [2].

Given a Bayesian network over the variables $X = \{X_1, ..., X_n\}$, and evidence e, Gibbs sampling [7, 8, 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. Then we can compute the quantities of interest, such as posterior probabilities, from the samples. Given enough samples, the estimated values are guaranteed to converge to the exact quantities.

The *cutset-sampling* scheme [2] allows sampling over any subset C of the variables X, from the distribution $P(C|e)$ at the cost of inference overhead (to compute $P(C|e)$) for each sample generation and for each posterior distribution estimate. By reducing the dimensionality of the sampled space from X to C we are guaranteed to converge over a smaller sample set, yet the cost of generating each sample may increase. The hope is that the increased convergence rate will compensate against the incurred inference overhead. Indeed, the inference overhead created (to compute $P(c^t, e)$) can sometimes be bounded if the structure of the Bayesian network is consulted when selecting the sampled set. This is because exact inference, that can be accomplished by variable elimination of join-tree algorithms (spigelhalter88,dechter99,jensen90), is controlled by the induced-width of the network whose evidence and sampling nodes are instantiated (the so-called *adjusted induced-width*).

Therefore, when a relatively small cutset induces an inference problem having a bounded adjusted induced-width, the resulting sampling+inference scheme may be more effective than either pure sampling or pure inference. This makes cutset-sampling a framework within which we can control the trade-off between sampling and inference and tune its balance to the given Bayesian network.

One point along this tradeoff line was already investigated in [2], where we demonstrated empirically that sampling over a cycle-cutset is cost-effective, yielding an order of magnitude improvement over Gibbs sampling.

The contribution of the current paper is in investigating the much broader trade-off range between sampling and inference as a function of w, thus fully evaluating the potential of this scheme. Specifically, we present an empirical study of the cutset sampling scheme over a variety of randomly generated networks, over grid structure networks as well as over known real-life benchmarks such as CPCS networks. We plot the effect of a gradual change in the cutset-size, as controlled by its adjusted induced width w, on the overall efficiency of w-cutset sampling and show that w-cutset sampling can be highly cost-effective over a range of w's (beyond the case of cycle-cutset). From these experiments we can conclude that cutset-sampling provides a framework within which the user can seek the optimal balance between inference and sampling and tune it to the given network instance.

In Section 2 we provide preliminaries and overview of Gibbs sampling and section 3 review the cutset-sampling scheme. Section 4 discusses statistical bounds of the errors derived from the sampling variance and section 5 introduce the empirical evaluation which is the primary contribution of this paper. Section 6 provides summary and conclusions.

2 Background

DEFINITION **2.1 (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.*

DEFINITION **2.2 (induced-width, cycle-cutset)** *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. A* cycle-cutset *of an undirected graph is a subset of nodes in the graph that, when removed, results in a graph without cycles. A cycle-cutset of a directed graph (also called loop-cutset) is a subset of nodes that when removed the resulting graph is a polytree.*

2.1 Gibbs sampling

Given a Bayesian network $\mathcal{B}$, Gibbs sampling generates a set of samples $x^t = \{x_1^t, x_2^t, ..., x_n^t\}$ where t denotes a sample and x_i^t is the value of X_i in sample t. Given a sample x^t, (evidence variables remain $fixed$), a new sample is generated by assigning a new value x_i^{t+1} to each variable X_i from its probability distribution conditioned on the values of the remaining variables:

$$x_i^{t+1} \leftarrow P(x_i|x^t \backslash x_i, e) \tag{1}$$

Here and elsewhere we will use notation $X \backslash X_i$ to describe a set of variables in X excluding X_i.

Once all the samples are generated, we can answer any query using the samples. In particular, posterior marginal belief $P(x_i|e)$ for each variable X_i can be estimated by averaging the conditional marginals:

$$\hat{P}(x_i|e) = \frac{1}{T} \sum_{t=1}^{T} P(x_i|x^t \backslash x_i) \tag{2}$$

As the number of samples increases, the probabilities $\hat{P}(x_i|e)$ converge to the exact ones [19, 17] under a few assumptions of the underlying Markov chain. The main requirement for convergence is that the network is ergodic.

In Bayesian networks, the value $P(x_i|x^t \backslash x_i, e)$ depends only on the values of the nodes in the Markov blanket of variable X_i consisting of the node's parents, children, and parents of its children. Therefore [19],

$$P(x_i|x^t \backslash x_i, e) = \alpha P(x_i|x^t_{pa(X_i)}) \prod_{\{j|X_j \in ch_i\}} P(x_j^t|x^t_{pa_j}) \tag{3}$$

3 Cutset-sampling

This section gives a brief overview of the cutset sampling method introduced in [2]. Given a subset of variables $C \subseteq X$, called *cutset* or *sampled* variables $C =$

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 $C_i \in C$.
2. **Generate samples:**
For t = 1 to T, generate a new sample c^t as follows:
For i = 1 to m, compute new value c_i^t for variable C_i as follows:
2.1 Using a **join-tree clustering** or variable elimination algorithm $JTC(C_i, (c^t_{(i)}, e))$, compute:

$$P(c_i) = P(c_i|c^t_{(i)}, e) \tag{4}$$

and Sample a new value c_i^t for C_i, from $P(c_i)$ End For i, End For t

Figure 1: *Cutset-sampling* Algorithm

$\{C_1, C_2, ..., C_k\}$, cutset-sampling generates samples c^t, t=1...T, over subspace C. Similar to Gibbs sampling, a new sample c^{t+1} is obtained by assigning a new value $c_i^{(t+1)}$ to each variable $C_i \in C$, sampled from the conditional probability distribution:

$$P(c_i|c^t \backslash c_i, e) \tag{5}$$

Given T samples, the posterior marginals for the sampled variables can be estimated as usual by:

$$P(c_i|e) = \frac{1}{T}\sum_t P(c_i|c^t \backslash c_i, e) \tag{6}$$

For variables that are not in the cutset we compute by exact inference the quantities $P(x_i|c^t, e)$ and then average:

$$P(x_i|e) = \frac{1}{T}\sum_t P(x_i|c^t, e) \tag{7}$$

The key idea of cutset sampling is that the relevant conditional distributions (eq. (4)) can be computed by exact inference algorithms whose complexity is tied to the network's structure and is improved by conditioning on the observed variables. 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, 4, 12]. The cutset-sampling algorithm is given in Figure 1. 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, namely, in the *adjusted induced width*. Therefore, given a parameter w, it is easy to describe and control the complexity of cutset-sampling using a notion of w-cutset.

DEFINITION **3.1 (w-cutset)** *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.*

It is easy to show that:

THEOREM **3.1 (Complexity of sample generation)** *If C is a w-cutset, the complexity of generating a single sample by cutset sampling is $O(|C| \cdot d \cdot n \cdot d^w)$ where d bounds the variables domain size, and n is the number of nodes.*

Computing $P(X_i|e)$ using equation (7) requires computing $P(x_i|c^t, e)$ for each variable which is also exponential in w, if C is a w-cutset.

THEOREM **3.2 (Complexity of posterior compuation)** *Given a w-cutset C, the complexity of computing the posterior of all the variables using cutset sampling over T samples is $O(T \cdot |C| \cdot d \cdot n \cdot d^w)$.*

Consequently, for the special case of cycle-cutset, both sampling and estimating the marginal posterior are linear in the size of the network multiplied by the cutset size and the number of samples.

Clearly, we should seek minimal w-cutset, those that do not include strict subsets that are also w-cutsets. However, determining if a w-cutset is minimal is costly, requiring to decide if a subgraph has an induced-width below w, which is time exponential in w. In our experiments we attempt to generate small w-cutset but will not insist on minimality. A cycle-cutset is a w-cutset when w is the family size. Yet, it is often not a minimal w-cutset.

Values $P(x_i|e)$ obtained by cutset sampling are (1) guaranteed to converge to the exact quantities ([2]) and (2) require fewer samples to converge than full sampling ([9, 3, 16]).

4 Computing an error bound

Gibbs sampling provides a simple sampling scheme for Bayesian networks that is guaranteed to converge to the correct posterior distribution in ergodic networks. The drawback of Gibbs sampling compared to many other sampling methods is that it is hard to estimate how many samples are needed to achieve a certain degree of convergence. It is possible to derive bounds on the absolute error based on sample variance for any sampling method if it generates independent samples, for example forward sampling and importance sampling. In Gibbs and other Monte Carlo methods, samples are dependent, and we cannot apply the confidence interval estimate directly.

We can create independent samles restarting the chain after every T samples. Let $\hat{P}_m(x|e)$ be an estimate derived from a single chain $m \in [1, ..., M]$ of length T (meaning containing T samples) as defined in equations (2)-(7). The estimates $\hat{P}_m(x|e)$ are independent random variables. Every time we restart the chain, we randomly assign new values to each sampling variable and this assignment is independent from the results generated in previous chains. If we generate a total of M such chains, the posterior marginals $\hat{P}(x|e)$ will be an average of the M results obtained from each chain:

$$\hat{P}(x|e) = \frac{1}{M}\sum_{m=1}^{M} \hat{P}_m(x|e) \tag{8}$$

Then, we can use the well-known sample variance estimate for random variables:

$$S^2 = \frac{1}{M-1}\sum_{m=1}^{M} (\hat{P}_m(x|e) - \hat{P}(x|e))^2$$

An equivalent representation for sampling variance is:

$$S^2 = \frac{\sum_{m=1}^{M} \hat{P}_m^2(x|e) - M\hat{P}^2(x|e)}{M-1} \tag{9}$$

where S^2 is easy to compute incrementally storing only the running sums of $\hat{P}_m(x|e)$ and $\hat{P}_m^2(x|e)$. By the Central

Limit Theorem, ergodic mean converges to Normal distribution $N(\mu, \sigma)$. Therefore, we can compute confidence interval in the $100(1-\alpha)$ percentile used for random variables with normal distribution for small sampling set sizes ([10]). Namely:

$$P\left[P(x|e) \in [\hat{P}(x|e) \pm t_{\frac{\alpha}{2},(M-1)}\frac{S}{\sqrt(M))}\right] = 1-\alpha \quad (10)$$

where $t_{\frac{\alpha}{2},(M-1)}$ is a table value from t distribution with $(M-1)$ degrees of freedom. In general, this method may yield confidence interval that is too large to be useful since the sample variance generally increases fast with size of the network in the standard Gibbs sampling. Cutset sampling allows us to sample a smaller variable set that results in a smaller sampling variance S and smaller error estimate. In the experimental section, we provide results showing 90% confidence interval computed for Gibbbs sampler and cutset sampling restarting Markov chain 20 times.

5 Selecting w-cutset

We compared full Gibbs sampling with cycle-cutset sampling and with w-cutset sampling for a range of w-values, $w = 2, 3, ...$ In all empirical studies, cycle-cutset of the network was found using the mga algorithm ([1]). We also devised a simple scheme for finding the smallest w-cutset for a given w. Our scheme starts with a set C that contains all nodes in X except evidence E: $C = X \backslash E$. Then, we first obtain 1-cutset by removing from C (in some order) all such nodes that the adjsuted induced width w of the min-fill ordering of nodes $X \backslash C, E$ is bounded by $w = 1$. The 1-cutset becomes a starting sampling set for selection of 2-cutset. We repeat this process selecting a (w+1)-cutset from w-cutset until maximum adjusted induced width w_{max} is reached. Following this scheme, nodes with smaller degrees will be removed from the sampling set first unless they are a part of a large family.

Proposed scheme is not optimal and other heuristics can be used instead. However, it guarantees, for the purpose of our empirical study, that (w+1)-cutset C_{w+1} is always a proper subset of the w-cutset C_w: $C_{w+1} \subset C_w$.

6 Experiments

6.1 Methodology

The primary goal of our empirical study was to investigate performance of w-cutset as a function of w. We compared the performance of Gibbs sampling, w-cutset sampling for different values of w and a special case of w-cutset sampling, cycle-cutset sampling. Our benchmarks are several CPCS networks, grid networks, 2-layer networks, and random networks. All the sampling algorithms were given a fixed time bound. In all networks, except cpcs54 and cpcs179 where exact inference is easy due to small network size and low induced width (w*=15 for cpcs54 and w*=8 for cpcs179), the sampling algorithm were allowed $\sim 60\%$ of the time necessary to generate the exact values.

In order to be able to generate confidence intervals of the absolute error, for all sampling algorithms, we ran M=20 independent sampling chains of size T where T is the maximum number of samples that an algorithm could generate within fixed period of time.The resulting chain length for each sampling algorithm is given in Figure 5(a). At the end of each chain m, we obtained an approximation $\hat{P}_m(x_i|e)$ for the posterior marginals over T samples as shown in eq.(8). We obtained a final estimate by averaging over $\hat{P}_m(x_i|e)$ values. For comparison, we also computed $\hat{P}(x_i|e)$ using a single chain with $M \cdot T$ samples in each.

We computed two error measures for each benchmark and each algorithm. We computed Mean Square Error (MSE) between the exact posterior marginals $P(x_i|e)$ and the approximate posterior marginals $\hat{P}(x_i|e)$:

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

and the absolute error:

$$\Delta = \frac{1}{N\sum_i |D(X_i)|}\sum_i \sum_{x_i \in D(X_i)} |\hat{P}(x_i|e) - P(x_i|e))$$

The exact posterior marginals were obtained via bucket-tree elimiation algorithm ([4]). All error measure were averaged over the number of instances tried.

Additionally, we evaluated the confidence interval estimate as described eariler since it provides a measure of the sampling algorithm performance where the comparison to the exact posterior marginals is not possible. We computed sampling variance S^2 from eq.(9) and 90% confidence interval (estimated absolute error) from eq:(10):

$$\Delta_{0.9}(x_i) = t_{0.05,19}\frac{S}{\sqrt{20}} \quad (11)$$

For each benchmark network, we computed average estimated absolute error $\Delta_{0.9}$:

$$\Delta_{0.9} = \frac{1}{N\sum_i |D(X_i)|}\sum_i \sum_{x_i \in D(X_i)} \Delta_{0.9}(x_i)$$

As noted earlier, estimated confidence interval can be too large to be practical. Thus, we compared $\Delta_{0.9}$ with exact average absolute error Δ and MSE to address two basic practical issues: first, whether $\Delta_{0.9}$ provides a feasible estimate of the absolute error, and second, whether the estimated absolute error $\Delta_{0.9}$ properly reflects the performance of the algorithm as compared to the average absolute error and average Mean Square Error.

For comparison, we also show the performance of Iterative Belief Propagation (IBP) algorithm on each benchmark after 25 iterations. IBP is an iterative message-passing algorithm that performs exact inference in Bayesian networks

without loops ([19]). Applied to Bayesian networks with loops. it computes approximate posterior marginals. The advantage of IBP as an approximate algorithm is that it requires linear space and usually converges very fast.

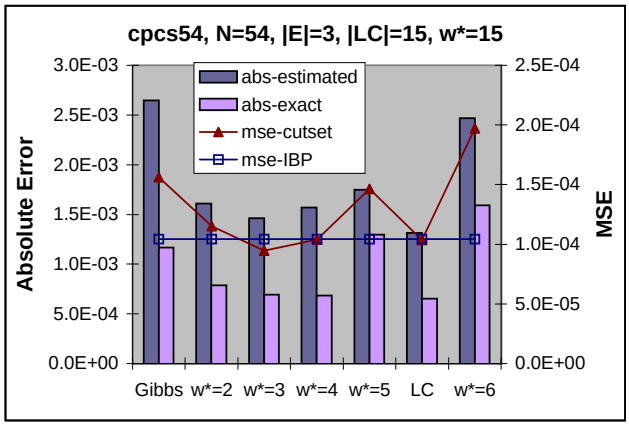
6.2 Benchmarks

CPCS networks. CPCS networks are derived from the Computer-based Patient Case Simulation system ([18, 20]). The nodes of CPCS networks correspond to diseases and findings and conditional probabilities describe their correlations.

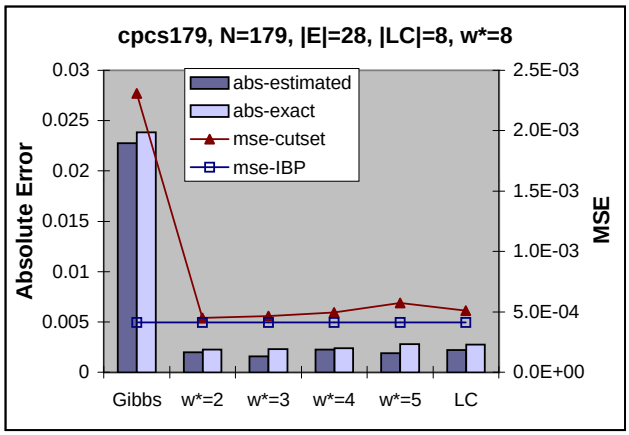
cpcs54 network consists of $N = 54$ nodes and has a relatively large cycle-cutset of size $|LC| = 15$ ($> 25\%$ of the nodes). Its induced width is 15. The performance of Gibbs sampling and cutset sampling is shown in Figure 2(a). The chart title contains the following notation: N - number of nodes in the network; $|E|$ - average number of evidence nodes; $|LC|$ - size of cycle-cutset; $w*$ - adjusted induced width of the network. The results are averaged over 10 instances with different evidence, $1-4$ observed nodes. The first graph, Figure 2(a), shows the means square error in the posterior marginals. The first point corresponds to Gibbs sampling ($\sim w* = 1$), the last point represents the cycle-cutset (or LC=loop-cutset), w*=15. In between, we plot the results for w-cutset sampling for a range of w* from 2 to 6. As we can see, w-cutset sampling improves over cycle-cutset sampling and IBP for w=2 and w=3 and then deteriorates for $w* \geq 4$. The same behavior is reflected in the absolute error measure shown in Figure 2(a).

cpcs179 network consists of N=179 nodes. It has a small cycle-cutset of size $|LC| = 8$ but with a relatively large corresponding adjusted induced width w*=8. w-cutsets with w=2 and w=3 are also small compared to the size of the network, but generate more samples per time period and achieve better accuracy than cycle-cutset sampling (Figure 2(b) and approach the accuracy of IBP (that does very well on those instances) with respect to both MSE and Absolute Error measure. All cutset sampling implementations are far superior to Gibbs sampling.

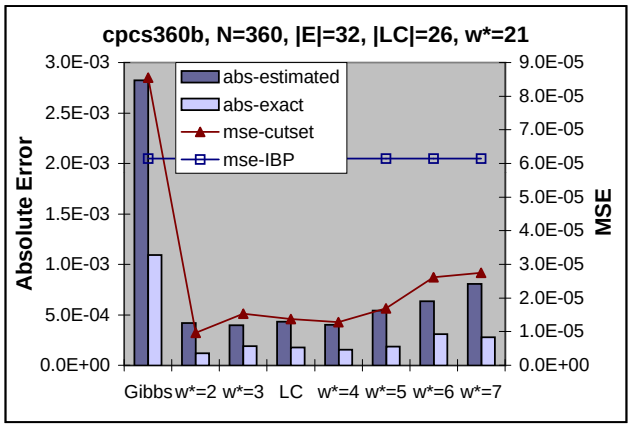
cpcs360b is a larger CPCS network with 360 nodes and adjusted induced width of 21 and cycle-cutset $|LC| = 26$. Exact inference on this network requires about 20 min with 2GHz Intel processor. We have allocated 12 min for each sampling algorithm. As we can see from Figure 2(c)), in terms of both MSE and Absolute Error, the cycle-cutset sampling yields comparable performance to w-cutset sampling for $w = 2-5$. The 2-cutset sampling appears to be slightly better than the rest, but not significantly given that sampling algorithm accuracy tends to fluctuate (even while converging). All cutset sampling implementations substantially outperform Gibbs sampling taking advantage of both sampling space reduction and greater efficiency in generating samples. cpcs360b is one of the networks where



(a) cpcs54, time bound=16 seconds.

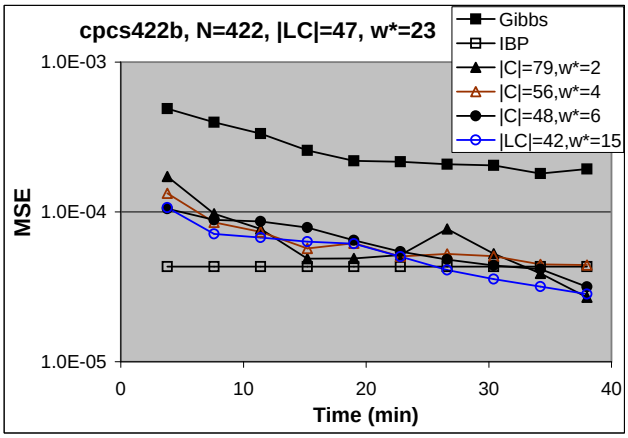


(b) cpcs179, time bound=12 seconds.



(c) cpcs360b, time bound=12 minutes.

Figure 2: CPCS networks, 10 instances each.



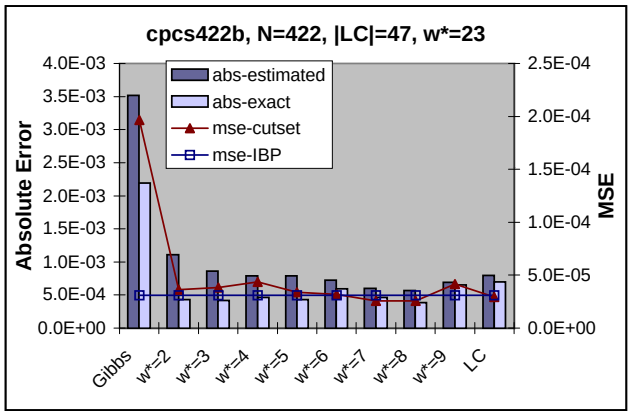


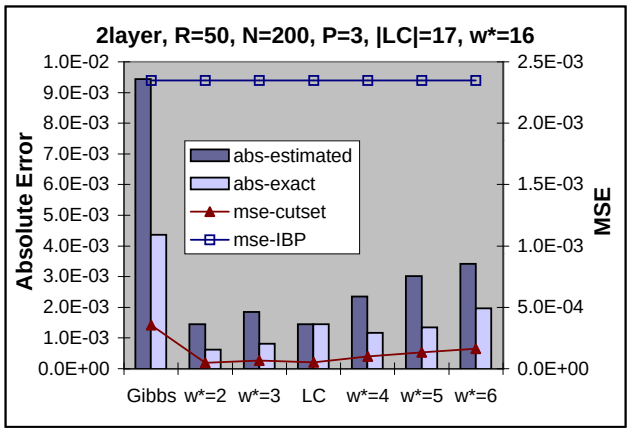
Figure 3: cpcs422b, 3 instances, time bound=40 minutes.

several instances of cutset sampling generate samples faster than Gibbs sampling. All cutset samling implementations outperform IBP as well although require more time.

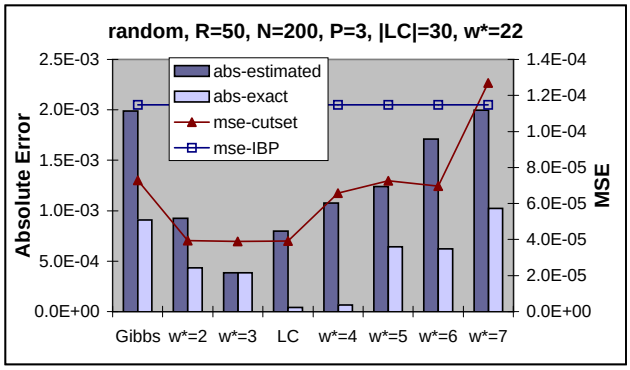
cpcs422b is the largest of the CPCS networks. It consisted of 422 nodes with cycle-cutset size $|LC| = 45$ and induced width $w* = 22$. The results are shown in Figure 6.2. The cpcs422b presents an example of a network where we increasing w can be done efficiently. The performance of w-cutset is good in a wide range of $w = 2-8$ and only begins to decline (slowly) at $w = 9$. It outperforms cycle-cutset sampling on $w = 7, 8$. Gibbs sampling was especially slow on cpcs422b and could not compete with any of the cutset sampling instances.

Grid networks. Grid networks with 450 nodes (15x30) were the only class of the networks where full Gibbs sampling was able to generate samples considerably faster than cutset sampling (nearly 10 times faster) and outperform cutset sampling. At the same time, we observed that loop-cutset with w*=3 did not perform best and we were able to slightly improve performance reducing the cutset size 3-cutset (w*=3) and 5-cutset (w*=5). The results are shown in Figure 4(c).

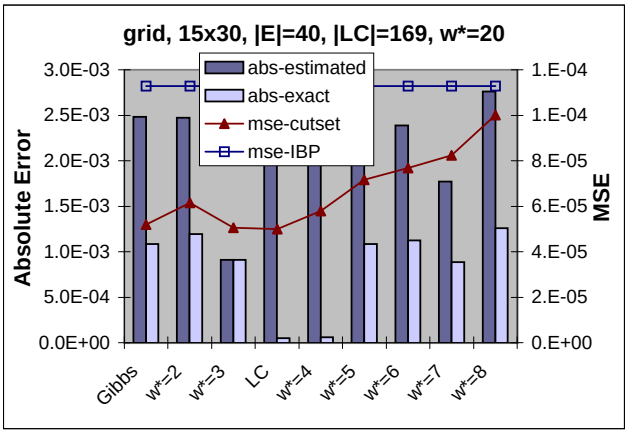
Random networks. We generated a set of random net-



(a) 2layer networks, time bound=25 sec.



(b) Random networks, time bound=60 seconds.



(c) Grid networks, 450 nodes (15x30), time bound=100 seconds.

Figure 4: Randomized networks , 10 instances each.

works with binary nodes. Each network contained total of 200 nodes. First 50 nodes, $\{X_1, ..., X_{50}\}$, were designated as root nodes. Each non-root node X_i was assigned 3 parents (selected randomly from $\{X_1, ..., X_{i-1}\}$). 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 4(b), bottom). For random networks, R is the number of root nodes, and P indicates number of parents for each non-root node. We can see that cycle-cutset (of average size of $|C| = 30$ nodes and w*=3) performs worse than any other cutset. Cutset sampling significantly improves by reducing the size of cutset while maintaining the same induced width $w* = 3$. The accuracy improves as well when increasing the size of the cutset to reduce induced width to 2. The second chart on Figure 4(b) show the 90 percent confidence estimate for each sampling set with 20 restarts and the exact average absolute error. As we can see, reducing sampling set size results in much lower absolute error and much better error bound estimate. Those results also indicate that it maybe reasonable to use the confidence interval estimate as the criteria for the quality of the answer.

2-Layer networks. We generated a set of random 2-layer networks with binary-valued nodes. Each network contained total of 200 nodes. First 40 nodes, $\{X_1, ..., X_{50}\}$, were designated as root nodes forming the top layer of the network. The remaining nodes were assigned 3 parents out of the root nodes. Those nodes formed the second layer of the network. 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 4(b)). 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).

6.3 Summary

Comparing performance of cutset sampling on the cutsets of different size with different corresponding induced width w* (induced width of the network with cutset instantiated), we observed two main results. First, there exists a range of w values where w-cutset sampling achieves an optimal performance. Increasing w, up to some threshold value, compensates for the incurred overhead in exact inference due to variance reduction (Rao-Blackwellisation) and, in some instances, also due to increased speed of generating samples. From the Figure 5(b), showing how cycle cutset size changes with w, we can see that the performance of w-cutset begins to deteriorate when increase in w results only in a small reduction of sampling set size. A good example is cpcs360b network. Starting with w=4, increasing w by 1 results in the reduction of samling set only by 1 node.

	#samples									
	Gibbs	LC	w*=2	w*=3	w*=4	w*=5	w*=6	w*=7	w*=8	w*=9
cpcs54	1000	700	700	450	300	200	100	-	-	-
cpcs179	130	100	175	120	40	10	-	-	-	-
cpcs360b	650	1000	1200	1000	800	450	300	190	-	-
cpcs422b	100	200	110	140	150	130	130	140	140	110
grid	2000	500	300	260	150	105	60	35	20	-
random	2000	1000	1400	700	450	300	140	75	-	-
2layer	200	700	900	320	150	75	40	-	-	-

(a) #samples in one markov chain.

	Sampling Set Size									
	Gibbs	LC	w*=2	w*=3	w*=4	w*=5	w*=6	w*=7	w*=8	w*=9
cpcs54	51	16	25	17	15	12	11	-	-	-
cpcs179	151	8	18	13	10	7	-	-	-	-
cpcs360b	328	26	28	21	17	15	14	13	-	-
cpcs422b	392	42	79	64	56	52	48	42	37	36
grid	410	169	163	119	95	75	60	50	13	-
random	190	30	61	26	25	24	18	17	-	-
2layer	185	17	22	15	13	12	11	-	-	-

(b) Average Sampling Set Size.

Figure 5: #samples and sampling set size for Gibbs sampling and cutset sampling as a function of w.

The table in Figure 5(a), shows how many samples each sampling algorithm generated within a fixed time period allocated for 1 chain. As expected, Gibbs sampling often generates the most samples. Yet, in some instances, where sampling set size is small enough to compensate for increase in w, cutset sampling generates more samples. Overall, the w-cutset sampling always outperformed Gibbs sampling and offered considerable improvement over IBP on several networks. The cycle-cutset sampling performed well, but in most cases yielded in performance to the w-cutset.

Our second set of observations are about the comparison of the three different error measures collected. We observed that average absolute error and MSE usually correlate well with each other and with the average estimated absolute error (90% confidence interval). Clearly, there are a few exceptions. In random networks on Figure 4(b) and grid networks on Figure 4(c), we observe spikes in estimated absolute error, in agreement with MSE, where the actual absolute error is the smallest. However, overall, confidence interval reflects accurately and with a reasonable error bound and could be used as an error estiamte where exact errors can not be computed.

Finally, we observed that the final approximation values $\hat{P}(x_i|e)$ were strongly dependent on the total number of samples $M \cdot T$, but not on the number of chains used (we do not present actual values here for lack of space).

7 Related Work and Conclusions

In this paper, we defined a notion of w-cutset and investigated the performance of w-cutset sampling, combining sampling and exact inference, as a function of the adjusted induce width parameter w that controls the complexity of the exact inference. The results suggest that there exists a range of w values for each network where w-cutset sampling performs best. Thus, user can find an optimal tradeoff between sampling and inference by examining the w-cutset sampling for different w values.

To reduce sampling variance, it is desirable to maximize w, up to some threshold value, and minimize sampling set size. Samling set size as a function of w can be used as an indicator of where the threshold is: increment w untilt he increase in w results only in minor reduction in sampling set size and requires substantially more time to compute new sample. Alternatively, the user can obtain results for several w-cutset and choose as a threshold the w where estimated confidence interval begins to increase.

We have investigated in this paper only the results for two extreme values of the number of independent markov chains M. We examined M=1 and M=20. We plan to continue this investigation and study the performance of the cutset sampling and confidence interval estimate as a function of M. We do not expect large variations in the performance of cutset sampling as our preliminary observations strongly indicate that accuracy of the sampling results is dependent on the total number of samples, not the number of chains. However, having fewer but longer chains may lead to a tighter absolute error bound.

Previously, sampling from a subset of variables was successfully applied to a particle factoring using importance sampling for Dynamic Bayesian networks (DBNs) [5]. In that study, the authors demonstrated that sampling from a subspace combined with exact inference yields a better approximation. However, in [5], the authors narrowly target the class of DBN networks and Particle Filtering scheme in particular where each time slice contains nodes independent from previous time slice and, therefore, are easy to sum over. The cutset sampling, first defined in [2], is a generalization of Rao-Blackwellisation scheme applicable to any Bayesian network, its performance defined in terms of adjusted induced width parameter that user can control to tune the algorithm to a particular network.

A different combination of sampling and exact inference for join trees was described in [14] and [13]. Each of those approaches proposes a scheme for sampling locally within a cluster and then combining those distributed results efficiently (introducing additional errors in the result). The cutset sampling is fundamentally different from the schemes above in that it does sampling on the full network and only takes advantage of the network structure to reduce the complexity of exact inference.

In [11], exact inference was used in combination with blocking Gibbs sampling. Again, the cutset sampling is different from the proposed approach in that it sums out variables instead of blocking.

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] B. Bidyuk and R. Dechter. Cycle-cutset sampling for bayesin networks. *Sixteenth Canadian Conference on Artificial Intelligence*, 2003.

[3] G. Casella and C. P. Robert. Rao-blackwellisation of sampling schemes. *Biometrika*, 83(1):81–94, 1996.

[4] R. Dechter. Bucket elimination: A unifying framework for reasoning. *Artificial Intelligence*, 113:41–85, 1999.

[5] 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.

[6] 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.

[7] 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.

[8] W. Gilks, S. Richardson, and D. Spiegelhalter. *Markov chain Monte Carlo in practice*. Chapman and Hall, 1996.

[9] M. H. De Groot. *Probability and Statistics, 2nd edition*. Addison-Wesley, 1986.

[10] R. V. Hogg and E. A. Tanis. *Probability and Statistical Inference*. Prentice Hall, 2001.

[11] 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.

[12] 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.

[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] R. Parker and R. Miller. Using causal knowledge to create simulated patient cases: the CPCS project as an extension of INTERNIST-1. In *Proc. 11th Symp. Comp. Appl. in Medical Care*, pages 473 – 480, 1987.

[19] J. Pearl. *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann, 1988.

[20] M. Pradhan, G. Provan, B. Middleton, and M. Henrion. Knowledge engineering for large belief networks. In *Proc. Tenth Conf. on Uncertainty in Artificial Intelligence*, 1994.