AND/OR Importance Sampling

Vibhav Gogate and Rina Dechter
Department of Information and Computer Science,
University of California, Irvine, CA 92697,
{vgogate,dechter}@ics.uci.edu

Abstract

The paper introduces an AND/OR importance sampling scheme for probabilistic graphical models. In contrast to conventional importance sampling, AND/OR importance sampling caches samples in the AND/OR space and then extracts a new sample mean from the stored samples. We prove that the AND/OR sample mean may have lower variance than conventional importance sampling; thereby providing a theoretical justification for preferring it over conventional importance sampling. Our empirical evaluation confirms that AND/OR importance sampling is far more accurate than conventional importance sampling in many cases.

1 Introduction

Many problems in probabilistic graphical models such as computing probability of evidence in Bayesian networks, solution counting in constraint networks and computing the partition function in Markov random fields are *summation problems*, defined as a sum of a function over a domain. Because these problems are NP-hard in general, sampling based techniques are often used to approximately compute the sum. The focus of the current paper is on a specific sampling technique called *importance sampling*.

The main idea in importance sampling [Geweke, 1989, Rubinstein, 1981] is to transform the summation problem to that of computing a weighted average over the domain by using a special distribution called the proposal (or importance) distribution. Importance sampling then generates samples from the proposal distribution and approximates the true average by a weighted average over the samples. The sample average is simply a ratio of the sum of sample weights and the number of samples, and it can be computed in a *memory-less* fashion since it is required to keep only these two quantities in memory.

The main idea in this paper is to extend importance sampling with memoization or caching in order to exploit conditional independencies that exist in the graphical model while computing the estimator. Specifically we organize the generated samples in an AND/OR search tree or a graph which respects the graphical model and then compute a new weighted average over that AND/OR structure, yielding, as we show, an unbiased estimator that has a smaller variance than the conventional importance sampling estimator. Our scheme builds upon the framework of AND/OR search spaces for graphical models that was introduced to exploit problem decomposition during search [Dechter and Mateescu, 2007] and we hence call it *AND/OR importance sampling*. Similar to AND/OR search [Dechter and Mateescu, 2007], AND/OR importance sampling recursively combines samples that are cached in independent components yielding an increase in the effective sample size which is part of the reason that its estimates have lower variance.

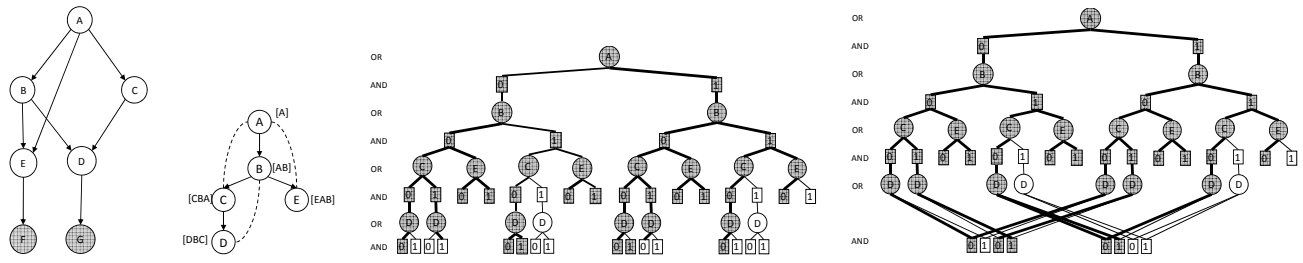
We present a detailed experimental evaluation comparing importance sampling with AND/OR importance sampling on various benchmark Bayesian networks. We observe that the latter outperforms the former on most benchmarks and in some cases quite significantly.

The rest of the paper is organized as follows. In the next section, we describe preliminaries on graphical models, importance sampling and AND/OR search spaces. In section 3,4 and 5 we formally describe AND/OR importance sampling and prove that its sample mean has lower variance than conventional importance sampling. Experimental results are described in section 6 and we conclude with a discussion of related work and summary.

2 Preliminaries

We represent sets by bold capital letters and members of a set by capital letters. An assignment of a value to a variable is denoted by a small letter while bold small letters indicate an assignment to a set of variables.

DEFINITION **2.1** (**belief networks).** *A* belief network (BN) *is a graphical model* $\mathscr{R} = (\mathbf{X}, \mathbf{D}, \mathbf{P})$*, where* $\mathbf{X} = \{X_1, \ldots, X_n\}$ *is a set of random variables over multi-valued domains* $\mathbf{D} = \{\mathbf{D_1}, \ldots, \mathbf{D_n}\}$*. Given a directed acyclic graph*



(a) (b) (c) (d)

Figure 1: (a) Bayesian Network, (b) Pseudo-tree (c) OR-space (d) AND/OR tree (e) AND/OR search graph

G over $\mathbf{X}$, $\mathbf{P} = \{P_i\}$, *where* $P_i = P(X_i|\mathbf{pa}(X_i))$ *are conditional probability tables (CPTs) associated with each* X_i. $\mathbf{pa}(X_i)$ *is the set of parents of the variable* X_i *in G. A belief network represents a probability distribution over* $\mathbf{X}$, $P(\mathbf{X}) = \prod_{i=1}^{n} P(X_i|\mathbf{pa}(X_i))$. *An* evidence set $\mathbf{E} = \mathbf{e}$ *is an instantiated subset of variables.The moral graph (or primal graph) of a belief network is the undirected graph obtained by connecting the parent nodes and removing direction.*

DEFINITION **2.2** (**Probability of Evidence).** *Given a belief network* $\mathscr{R}$ *and evidence* $\mathbf{E} = \mathbf{e}$*, the probability of evidence* $P(\mathbf{E} = \mathbf{e})$ *is defined as:*

$$P(\mathbf{e}) = \sum_{\mathbf{X}\setminus\mathbf{E}} \prod_{j=1}^{n} P(X_j|\mathbf{pa}(X_j))_{|\mathbf{E}=\mathbf{e}} \tag{1}$$

The notation $h(\mathbf{X})_{|\mathbf{E}=\mathbf{e}}$ stands for a function h over $\mathbf{X} \setminus \mathbf{E}$ with the assignment $\mathbf{E} = \mathbf{e}$.

2.1 AND/OR search spaces

We can compute probability of evidence by search, by accumulating probabilities over the search space of instantiated variables. In the simplest case, this process defines an OR search tree, whose nodes represent partial variable assignments. This search space does not capture the structure of the underlying graphical model. To remedy this problem, [Dechter and Mateescu, 2007] introduced the notion of AND/OR search space. Given a bayesian network $\mathscr{R} = (\mathbf{X}, \mathbf{D}, \mathbf{P})$, its AND/OR search space is driven by a pseudo tree defined below.

DEFINITION **2.3** (**Pseudo Tree).** *Given an undirected graph* $G = (V, E)$*, a directed rooted tree* $T = (V, E)$ *defined on all its nodes is called pseudo tree if any arc of G which is not included in E is a back-arc, namely it connects a node to an ancestor in* T.

DEFINITION **2.4** (**Labeled AND/OR tree).** *Given a graphical model* $\mathscr{R} = \langle \mathbf{X}, \mathbf{D}, \mathbf{P} \rangle$*, its primal graph G and a backbone pseudo tree T of G, the associated AND/OR search tree, has alternating levels of AND and OR nodes. The OR nodes are labeled* X_i *and correspond to the variables. The AND nodes are labeled* $\langle X_i, x_i \rangle$ *and correspond to the value assignments in the domains of the variables. The structure of the AND/OR search tree is based on the underlying backbone tree* T*. The root of the AND/OR search tree is an OR node labeled by the root of* T.

Each OR arc, emanating from an OR node to an AND node is associated with a **labeling function** *which can be derived from the CPTs of the bayesian network [Dechter and Mateescu, 2007]. Each OR node and AND node is also associated with a* **value** *that is recursively used for computing the quantity of interest.*

Semantically, the OR states represent alternative assignments, whereas the AND states represent problem decomposition into independent subproblems, all of which need be solved. When the pseudo-tree is a chain, the AND/OR search tree coincides with the regular OR search tree. The probability of evidence can be computed from a AND/OR tree by labeling it appropriately with weights which are derived from the CPTs and then recursively computing the value of all nodes from leaves to the root. For more information see [Dechter and Mateescu, 2007].

Example 2.1. *Figure 1(a) shows a bayesian network over seven variables with domains of* $\{0,1\}$*. F and G are evidence nodes. Figure 1(c) shows the AND/OR-search tree for the bayesian network based on the Pseudo-tree in Figure 1(b). Note that because F and G are instantiated, the search space has only 5 variables.*

2.2 Computing Probability of Evidence Using Importance Sampling

Importance sampling [Rubinstein, 1981] is a simulation technique commonly used to evaluate the following sum: $M = \sum_{\mathbf{x}\in\mathbf{X}} f(\mathbf{x})$ for some real function f. The idea is to generate samples $\mathbf{x^1}, \ldots, \mathbf{x^N}$ from a proposal distribution Q (satisfying $f(\mathbf{x}) > 0 \Rightarrow Q(\mathbf{x}) > 0$) and then estimate M as follows:

$$M = \sum_{\mathbf{x}\in\mathbf{X}} f(\mathbf{x}) = \sum_{\mathbf{x}\in\mathbf{X}} \frac{f(\mathbf{x})}{Q(\mathbf{x})} Q(\mathbf{x}) = \mathbb{E}_Q[\frac{f(\mathbf{x})}{Q(\mathbf{x})}] \tag{2}$$

$$\widehat{M} = \frac{1}{N} \sum_{i=1}^{N} w(\mathbf{x}^i) \ , \ where \ \ w(\mathbf{x}^i) = \frac{f(\mathbf{x}^i)}{Q(\mathbf{x}^i)} \tag{3}$$

w is often referred to as the sample weight. It is known that the expected value $\mathbb{E}(\widehat{M}) = M$ [Rubinstein, 1981].

To compute the probability of evidence by importance sampling, we use the substitution:

$$f(\mathbf{x}) = \prod_{j=1}^{n} P(X_j|\mathbf{pa}(X_j))_{|\mathbf{E}=\mathbf{e}} \tag{4}$$

Several choices are available for the proposal distribution $Q(\mathbf{x})$ ranging from the prior distribution as in likelihood weighting to more sophisticated alternatives such as IJGP-Sampling [Gogate and Dechter, 2005] and EPIS-BN [Yuan and Druzdzel, 2006] where the output of belief propagation is used to compute the proposal distribution.

As in prior work [Cheng and Druzdzel, 2000], we assume that the *proposal distribution is expressed in a factored product form*: $Q(\mathbf{X}) = \prod_{i=1}^{n} Q_i(X_i|X_1,\ldots,X_{i-1}) = \prod_{i=1}^{n} Q_i(X_i|\mathbf{Y_i})$, where $\mathbf{Y_i} \subseteq \{X_1,\ldots,X_{i-1}\}$, $Q_i(X_i|\mathbf{Y_i}) = Q(X_i|X_1,\ldots,X_{i-1})$ and $|\mathbf{Y_i}| < c$ for some constant c. We can generate a full sample from Q as follows. For i = 1 to n, sample $X_i = x_i$ from the conditional distribution $Q(X_i|X_1 = x_1,\ldots,X_{i-1} = x_{i-1})$ and set $X_i = x_i$.

3 Computing Sample Mean in AND/OR-space

The main idea in our new scheme is to store all samples generated from the proposal distribution Q on an AND/OR-tree and then compute a new weighted average (mean) over the samples arranged in an AND/OR tree. The intuition is that using the AND/OR independencies, we derive a large set of virtual samples from the input, and this may yield a better estimator. In this section, we describe the technical details involved.

Note that because our aim is to find a new sample mean on a AND/OR-tree given a set of samples, all nodes and edges which are not sampled are clearly irrelevant. We therefore define the notion of a AND/OR sample tree which is restricted to the generated samples:

DEFINITION **3.1** (**AND/OR Sample Tree).** *Given a sequence of assignments (samples)* $\mathbf{S}$ *and a arc-labeled AND/OR tree S_{AOT} (whose labels are set according to Definition 3.2), a AND/OR sample tree is constructed from S_{AOT} by removing all edges and corresponding nodes whose frequency is zero (i.e. they are not sampled).*

DEFINITION **3.2** (**Arc Labels).** *Given a a graphical model $\mathscr{R} = \langle \mathbf{X}, \mathbf{D}, \mathbf{P} \rangle$, a pseudo-tree $T(V,E)$, a proposal distribution $Q = \prod_{i=1}^{n} Q(X_i|\mathbf{Anc}(X_i))$ such that $\mathbf{Anc}(\mathbf{X_i})$ is a subset of all ancestors of X_i in T, a sequence of samples (assignments) to all non-evidence variables and a AND/OR sample tree S_{A0T} (see Definition 3.1), the Arc-label for an OR node X_i to an AND node $X_i = x_i$ in S_{AOT} is a pair $\langle w, \# \rangle$ where:*

- $w = \frac{P(X_i=x_i,\mathbf{anc}(x_i))}{Q(X_i=x_i|\mathbf{anc}(x_i))}$ *is called the weight of the arc.* $\mathbf{anc}(x_i)$ *is the assignment of values to all variables from the node X_i to the root node of S_{AO} and $P(X_i = x_i, \mathbf{anc}(x_i))$ is the product of all functions in $\mathscr{R}$ that mention X_i but do not mention any variable ordered below it in T given $(X_i = x_i, \mathbf{anc}(x_i))$.*
- # *called the frequency of the arc is the number of times the assignment $(X_i = x_i, \mathbf{anc}(X_i))$ is sampled.*

Example 3.1. *Consider the example bayesian network given in Figure 2(a). The proposal distribution $Q(XYZ)$ is uniform. Figure 2(b) shows four random samples drawn from Q. Figure 2(c) shows the AND/OR sample tree over to the four samples. Each arc from an OR node to an AND node in the AND/OR sample tree is labeled with appropriate frequencies and weights by using Definition 3.2. Figure 2(c) shows derivation of arc-weights for two arcs.*

The main virtue of arranging the samples on an AND/OR sample tree is that we can exploit the independencies to estimate a new sample mean. Next, we formally define this new mean called as AND/OR sample mean.

DEFINITION **3.3** (**AND/OR sample mean).** *Given a AND/OR sample tree with arcs labeled according to Definition 3.2, the AND/OR sample mean is defined as the value of the root node where the value of a node is defined as follows. The value of leaf AND nodes is "1" and the value of leaf OR nodes is "0". Let $\mathbf{C}(\mathbf{n})$ denote the child nodes and $v(n)$ denotes the value of node n. If n is a AND node then: $v(n) = \prod_{n' \in \mathbf{C}(\mathbf{n})} v(n')$ and if n is a OR node then*

$$v(n) = \frac{\sum_{n' \in \mathbf{C}(\mathbf{n})} (\#(n,n') w(n,n') v(n'))}{\sum_{n' \in \mathbf{C}(\mathbf{n})} \#(n,n')}$$

We now justify how the AND/OR sample mean can be derived on our example bayesian network of Figure 2(a). Let $Q(ZXY) = Q(Z)Q(X|Z)Q(Y|Z)$ be the proposal distribution. For simplicity, let us assume that $f(Z) = P(Z)$, $f(XZ) = P(Z|X)P(A = a|X)$ and $f(YZ) = P(Z|Y)P(B = b|Y)$.

We can express probability of evidence $P(a,b)$ as:

$$\begin{aligned} P(a,b) &= \sum_{XYZ} \frac{f(Z)f(XZ)f(YZ)}{Q(Z)Q(X|Z)Q(Y|Z)} Q(Z)Q(X|Z)Q(Y|Z) \\ &= \mathbb{E}\left[\frac{f(Z)f(XZ)f(YZ)}{Q(Z)Q(X|Z)Q(Y|Z)}\right] \end{aligned} \quad (5)$$

(Note that all expectations are taken w.r.t. Q). We can now decompose the expectation in Equation 5 into smaller components.

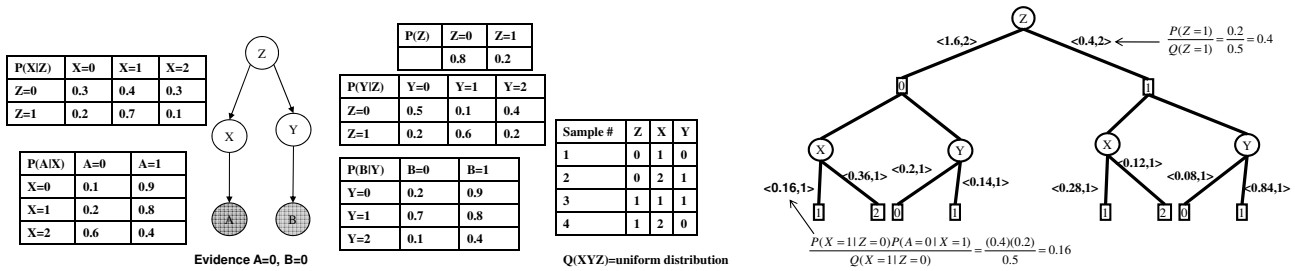
$$P(a,b) = \sum_{Z} \frac{f(Z)}{Q(Z)} Q(Z) \left(\sum_{X} \frac{f(XZ)Q(X|Z)}{Q(X|Z)}\right)\left(\sum_{Y} \frac{f(YZ)Q(Y|Z)}{Q(Y|Z)}\right) \quad (6)$$

Notice that the quantities in the two brackets in Equation 6 are, by definition, conditional expectations of a function over X and Y respectively given Z. Therefore, Equation 6 can be written as:

$$P(a,b) = \sum_{Z} \frac{f(Z)}{Q(Z)} \mathbb{E}\left[\frac{f(XZ)}{Q(X|Z)}|Z\right] \mathbb{E}\left[\frac{f(YZ)}{Q(Y|Z)}|Z\right] Q(Z) \quad (7)$$

Let $g_X(Z) = \mathbb{E}\left[\frac{f(XZ)}{Q(X|Z)}|Z\right]$ and let $g_Y(Z) = \mathbb{E}\left[\frac{f(YZ)}{Q(Y|Z)}|Z\right]$ be the functions corresponding to the conditional expectations taken over X and Y respectively. We can now rewrite Equation 7 as:

$$P(a,b) = \sum_{Z} \frac{f(Z)g_X(Z)g_Y(Z)}{Q(Z)} Q(Z) \quad (8)$$



(a) (b) (c)

Figure 2: (a) Bayesian Network, its CPTs, (b) Proposal Distribution and Samples (c) AND/OR sample tree

By definition, Equation 8 can be written as:

$$P(a,b) = \mathbb{E}\left[\frac{f(Z)g_X(Z)g_Y(Z)}{Q(Z)}\right] \tag{9}$$

Let us assume that we are given samples $(z^1,x^1,y^1),\ldots,(z^N,x^N,y^N)$ generated from Q. For simplicity, let $\{0,1\}$ be the domain of Z and let $Z=0$ and $Z=1$ be sampled N_0 and N_1 times respectively. From Equations 7-9, we can derive the following unbiased estimator for $P(a,b)$:

$$\widehat{P(a,b)} = \frac{1}{N}\sum_{j=0}^{1}\frac{N_j f(Z=j)\widehat{g_X(Z=j)}\widehat{g_Y(Z=j)}}{Q(Z=j)}$$

$$\text{where } \widehat{g_X(Z=j)} = \frac{1}{N_j}\sum_{i=1}^{N}\frac{f(x^i,Z=j)I(x^i,Z=j)}{Q(x^i,Z=j)}$$

$$\text{and } \widehat{g_Y(Z=j)} = \frac{1}{N_j}\sum_{i=1}^{N}\frac{f(y^i,Z=j)I(y^i,Z=j)}{Q(y^i,Z=j)} \tag{10}$$

where $I(x^i,Z=j)$ (or $I(y^i,Z=j)$) is an indicator function which is 1 iff the tuple $(x^i,Z=j)$ (or $(y^i,Z=j)$) is generated in any of the N samples and 0 otherwise.

Conventional importance sampling on the other hand would estimate $P(a,b)$ as follows:

$$\widetilde{P(a,b)} = \frac{1}{N}\sum_{j=0}^{1}N_j\frac{f(Z=j)}{Q(Z=j)} * \frac{1}{N_j}\sum_{i=1}^{N}\frac{f(x^i,Z=j)f(y^i,Z=j)}{Q(x^i|Z=j)Q(Y^i|Z=j)}I(x^i,y^i,Z=j) \tag{11}$$

where $I(x^i,y^i,Z=j)$ is an indicator function which is 1 iff the tuple $(x^i,y^i,Z=j)$ is generated in any of the N samples and 0 otherwise.

We can see that in Equation 10, we derive the sample average at X given Z (denoted by $\widehat{g_X(Z)}$ in Equation 10) separately from the sample average at Y given Z (denoted by $\widehat{g_Y(Z)}$ in Equation 10). On the other hand, conventional importance sampling (Equation 11) computes the sample average over the joint random variable XY given Z. The estimates given by Equation 10 are actually equal to the AND/OR sample mean over a AND/OR-sample tree as illustrated by the following example:

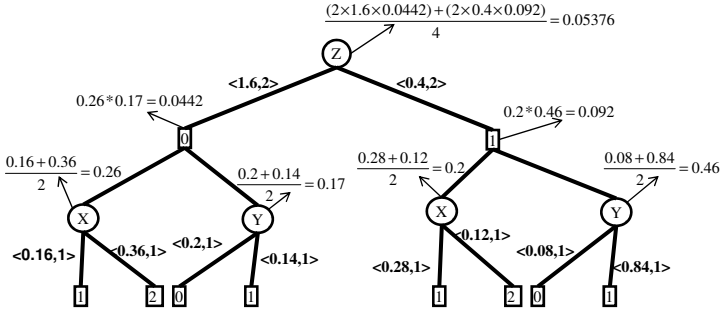


Figure 3: Computation of Values of OR and AND nodes in a AND/OR sample tree. The value of root node is equal to the AND/OR sample mean

Example 3.2. *The calculations involved in computing the sample mean on the AND/OR sample tree using Equation 10 are shown in Figure 3. Each AND node and OR node in Figure 3 is marked with a value that is computed recursively using definition 3.3. The value of OR nodes X and Y given* $Z=j\in\{0,1\}$ *is equal to* $\widehat{g_X(Z=j)}$ *and* $\widehat{g_Y(Z=j)}$ *respectively. The value of the root node is equal to the AND/OR sample mean which is equal to the sample mean given by Equation 10.*

Algorithm 1 AND/OR Importance Sampling

Input: an ordering $O=(X_1,\ldots,X_n)$,a Bayesian network BN and a proposal distribution Q
Output: Estimate of Probability of Evidence

1: Generate samples $\mathbf{x}^1,\ldots,\mathbf{x}^N$ from Q along O.
2: Build a AND/OR sample tree S_{AOT} for the samples $\mathbf{x}^1,\ldots,\mathbf{x}^N$ along the ordering O.
3: Initialize all labeling functions $\langle w,\# \rangle$ on each arc from an Or-node n to an And-node n' using Definition 3.2.
4: **FOR** all leaf nodes i of S_{AOT} do
5: **IF And-node** v(i)= 1 **ELSE** v(i)=0
6: **For** every node n from leaves to the root do
7: Let $C(n)$ denote the child nodes of node n
8: **IF** $n=\langle X,x\rangle$ is a AND node, then $v(n)=\prod_{n'\in C(n)}v(n')$
9: **ELSE** if $n=X$ is a OR node then

$$v(n)=\frac{\sum_{n'\in C(n)}(\#(n,n')w(n,n')v(n'))}{\sum_{n'\in C(n)}\#(n,n')}.$$

10: Return v(root node)

We now extend the computations described Example 3.2 to a general AND/OR sample tree yielding Algorithm AND/OR importance sampling (see Algorithm 1). In Steps 1-3, the algorithm generates samples from Q and stores them on an AND/OR sample tree. The algorithm then computes the AND/OR sample mean over the AND/OR sample tree recursively from leaves to the root in Steps $4-9$.

Note that the value $v(n)$ of a node in the AND/OR sample tree stores the sample average of the subproblem rooted at n, subject to the current variable instantiation along the path from the root to n. If n is the root, then $v(n)$ is the AND/OR sample mean which is our AND/OR estimator of probability of evidence.

Based on properties of expectation and conditional expectation, it is easy to show that:

THEOREM **3.3.** *Algorithm AND/OR importance sampling returns an unbiased estimate of probability of evidence.*

Finally, we summarize the complexity of computing AND/OR sample mean in the following proposition:

THEOREM **3.4** (Complexity of AND/OR importance sampling)**.** *Given N samples and n variables (with constant domain size), the time complexity of computing AND/OR sample mean is $O(nN)$ (same as conventional importance sampling) and its space complexity is $O(nN)$ (the space complexity of conventional importance sampling is constant).*

4 Variance Reduction

In this section, we prove that the AND/OR sample mean may have lower variance than the sample mean computed using conventional importance sampling (Equation 3).

THEOREM **4.1** (Variance Reduction)**.** *Variance of AND/OR sample mean is less than or equal to the variance of conventional importance sampling sample mean.*

Proof. The details of the proof are quite complicated and therefore we only provide the intuitions involved in the proof. As noted earlier the guiding principle of AND/OR sample mean is to take advantage of conditional independence in the graphical model. Let us assume that we have three random variables **X**, **Y** and **Z** with the following relationship: **X** and **Y** are independent of each other given **Z** (similar to our example bayesian network). The expression for variance derived here can be used in an induction step (induction is carried on the nodes in the AND/OR search tree) to prove the theorem.

In this case, conventional importance sampling generates samples $((\mathbf{x}^1,\mathbf{y}^1,\mathbf{z}^1),\ldots,(\mathbf{x}^N,\mathbf{y}^N,\mathbf{z}^N))$ along the order $\langle \mathbf{Z},\mathbf{X},\mathbf{Y}\rangle$ and estimates the mean as follows:

$$\mu^{IS}(\mathbf{XYZ}) = \frac{\sum_{i=1}^{N} \mathbf{x}^i\mathbf{y}^i\mathbf{z}^i}{N} \tag{12}$$

Without loss of generality, let $\{\mathbf{z_1},\mathbf{z_2}\}$ be the domain of $\mathbf{Z}$ and let these values be sampled N_1 and N_2 times respectively. We can rewrite 12 as follows:

$$\mu^{IS}(\mathbf{XYZ}) = \frac{1}{N}\sum_{j=1}^{2} N_j\mathbf{z_j}\frac{\sum_{i=1}^{N}\mathbf{x}^i\mathbf{y}^i I(\mathbf{z}_j,\mathbf{x}^i,\mathbf{y}^i)}{N_j} \tag{13}$$

where $I(\mathbf{z}_j,\mathbf{x}^i,\mathbf{y}^i)$ is an indicator function which is 1 iff the partial assignment $(\mathbf{z}_j,\mathbf{x}^i,\mathbf{y}^i)$ is generated in any of the N samples and 0 otherwise.

AND/OR sample mean is defined as:

$$\mu^{AO}(\mathbf{XYZ}) = \frac{1}{N}\sum_{j=1}^{2} N_j\mathbf{z}_j\left(\frac{\sum_{i=1}^{N}\mathbf{x}^i I(\mathbf{z}_j,\mathbf{x}^i)}{N_j}\right)\left(\frac{\sum_{i=1}^{N}\mathbf{y}^i I(\mathbf{z}_j,\mathbf{y}^i)}{N_j}\right) \tag{14}$$

where $I(\mathbf{x}^j,\mathbf{z}_i)$ (and similarly $I(\mathbf{y}^j,\mathbf{z}_i)$) is an indicator function which equals 1 when one of the N samples contains the tuple $(\mathbf{x}^j,\mathbf{z}_i)$ (and similarly $(\mathbf{y}^j,\mathbf{z}_i)$) and is 0 otherwise.

By simple algebraic manipulations, we can prove that the variance of estimator $\mu^{IS}(\mathbf{XYZ})$ is given by:

$$Var(\mu^{IS}(\mathbf{XYZ})) = \left(\sum_{j=1}^{2}\mathbf{z}_j^2 Q(\mathbf{z_j})\Big(\mu(\mathbf{X}|\mathbf{z}_j)^2 V(\mathbf{Y}|\mathbf{z}_j) + \mu(\mathbf{Y}|\mathbf{z}_j)^2 V(\mathbf{X}|\mathbf{z}_j) + V(\mathbf{X}|\mathbf{z}_j)V(\mathbf{Y}|\mathbf{z_j})\Big)\right)/N - \mu^2_{\mathbf{XYZ}}/N \tag{15}$$

Similarly, the variance of AND/OR sample mean is given by:

$$Var(\mu^{AO}(\mathbf{XYZ})) = \left(\sum_{j=1}^{2}\mathbf{z}_j^2 Q(\mathbf{z_j})\Big(\mu(\mathbf{X}|\mathbf{z}_j)^2 V(\mathbf{Y}|\mathbf{z}_j) + \mu(\mathbf{Y}|\mathbf{z}_j)^2 V(\mathbf{X}|\mathbf{z}_j) + \frac{V(\mathbf{X}|\mathbf{z}_j)V(\mathbf{Y}|\mathbf{z_j})}{N_j}\Big)\right)/N - \mu^2_{\mathbf{XYZ}}/N \tag{16}$$

where $\mu(\mathbf{X}|\mathbf{z}_j)$ and $V(\mathbf{X}|\mathbf{z}_j)$ are the conditional mean and variance respectively of $\mathbf{X}$ given $\mathbf{Z}=\mathbf{z}_j$. Similarly, $\mu(\mathbf{Y}|\mathbf{z}_j)$ and $V(\mathbf{Y}|\mathbf{z}_j)$ are the conditional mean and variance respectively of $\mathbf{Y}$ given $\mathbf{Z}=\mathbf{z}_j$.

From Equations 15 and 16, if $N_j = 1$ for all j, then we can see that the $Var(\mu^{AO}(\mathbf{XYZ})) = Var(\mu^{IS}(\mathbf{XYZ}))$. However if $N_j > 1, Var(\mu^{AO}(\mathbf{XYZ})) < Var(\mu^{IS}(\mathbf{XYZ}))$. This proves that the variance of AND/OR sample mean is less than or equal to the variance of conventional sample mean on this special case. As noted earlier using this case in induction over the nodes of a general pseudo-tree completes the proof.

□

5 Estimation in AND/OR graphs

Next, we describe a more powerful algorithm for estimating mean in AND/OR-space by moving from AND/OR-trees to AND/OR graphs as presented in [Dechter and Mateescu, 2007]. An AND/OR-tree may contain nodes that root identical subtrees (i.e. their root nodes values are identical); called as unifiable nodes. When unifiable nodes are merged, the tree becomes a graph and its size becomes smaller. Some unifiable nodes can be identified using contexts defined below.

DEFINITION **5.1** (Context)**.** *Given a belief network and the corresponding AND/OR search tree S_{AOT} relative to a pseudo-tree T, the context of any AND node $\langle X_i, x_i \rangle \in S_{AOT}$, denoted by context(X_i), is defined as the set of ancestors of X_i in T, that are connected to X_i and descendants of X_i.*

The context minimal AND/OR graph is obtained by merging all the context unifiable AND nodes. The size of the largest context is bounded by the tree width w^* of the pseudo-tree [Dechter and Mateescu, 2007]. Therefore, the time and space complexity of a search algorithm traversing the context-minimal AND/OR graph is $O(exp(w^*))$.

Example 5.1. *For illustration, consider the context-minimal graph in Figure 1(e) of the pseudo-tree from Figure 1(c). Its size is far smaller that that of the AND/OR tree from Figure 2(c) (30 nodes vs. 38 nodes). The contexts of the nodes can be read from the pseudo-tree in Figure 1(b) as follows: context(A) = $\{A\}$, context(B) = $\{B,A\}$, context(C) = $\{C,B,A\}$, context(D) = $\{D,C,B\}$ and context(E) = $\{E,A,B\}$.*

The main idea in AND/OR-graph estimation is to store all samples on a AND/OR-graph instead of a AND/OR-tree. Similar to an AND/OR sample tree, we can define an identical notion of an AND/OR sample graph.

DEFINITION **5.2** (AND/OR sample graph)**.** *Given an AND/OR sample tree S_T, an AND/OR sample graph S_G is obtained by merging all context unifiable AND nodes in S_T.*

Example 5.2. *The bold edges and nodes in Figure 1(c) define a AND/OR sample tree. The bold edges and nodes in Figure 1(d) define a AND/OR sample graph constructed from AND/OR sample tree of Figure 1(c) by merging all context unifiable nodes.*

The algorithm for computing the sample mean on AND/OR sample graphs is identical to the algorithm for AND/OR-tree (Steps 4-10 of Algorithm 1). The main reason in moving from trees to graphs is that the variance of the sample mean computed on a AND/OR sample graph can be even smaller than that computed on a AND/OR sample tree. More formally,

THEOREM **5.3.** *Let $V(\mu_{AOG})$, $V(\mu_{AOT})$ and $V(\mu_{IS})$ be the variance of AND/OR sample mean on an AND/OR sample graph, variance of AND/OR sample mean on an AND/OR sample tree and variance of sample mean of conventional importance sampling respectively. Then given the same set of input samples:*

$$V(\mu_{AOG}) \leq V(\mu_{AOT}) \leq V(\mu_{IS})$$

We omit the proof due to lack of space.

THEOREM **5.4** (Complexity of computing AND/OR graph sample mean)**.** *Given a graphical model with n variables, a psuedo-tree with treewidth w^* and N samples, the time complexity of AND/OR graph sampling is $O(nNw^*)$ while its space complexity is $O(nN)$.*

6 Experimental Evaluation

6.1 Competing Algorithms

The performance of importance sampling based algorithms is highly dependent on the proposal distribution [Cheng and Druzdzel, 2000]. It was shown that computing the proposal distribution from the output of a Generalized Belief Propagation scheme of Iterative Join Graph Propagation (IJGP) yields better empirical performance than other available choices [Gogate and Dechter, 2005]. Therefore, we use the output of IJGP to compute the proposal distribution Q. The complexity of IJGP is time and space exponential in its i-bound, a parameter that bounds cluster sizes. We use a i-bound of 5 in all our experiments.

We experimented with three sampling algorithms for benchmarks which do not have determinism: (a) (pure) IJGP-sampling, (b) AND/OR-tree IJGP-sampling and (c) AND/OR-graph IJGP-sampling. Note that the underlying scheme for generating the samples is identical in all the methods. What changes is the method of accumulating the samples and deriving the estimates. On benchmarks which have zero probabilities or determinism, we use the SampleSearch scheme introduced by [Gogate and Dechter, 2007] to overcome the rejection problem. We experiment with the following versions of SampleSearch on deterministic networks: (a) pure SampleSearch, (b) AND/OR-tree SampleSearch and (c) AND/OR-graph SampleSearch.

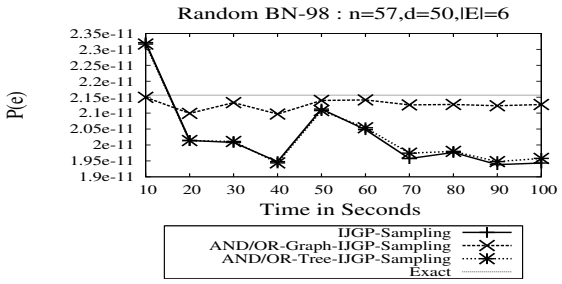
6.1.1 Results

We experimented with three sets of benchmark belief networks (a) Random networks, (b) Linkage networks and (c) Grid networks. Note that only linkage and grid networks have zero probabilities on which we use SampleSearch.The exact $P(\mathbf{e})$ for most instances is available from the UAI 2006 competition web-site.

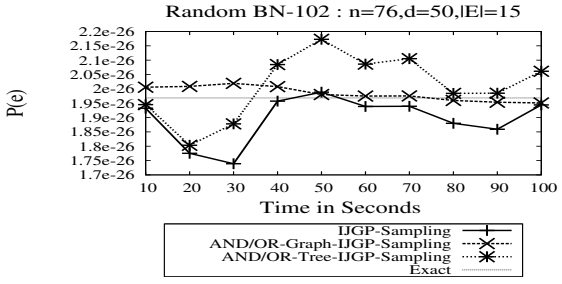
Our results are presented in Figures 4-6. Each Figure shows approximate probability of evidence as a function of time. The bold line in each Figure indicates the exact probability of evidence. The reader can visualize the error from the distance between the approximate curves and the exact line. For lack of space, we show only part of our results. Each Figure shows the number of variables n, the maximum-domain size d and the number of evidence nodes $|E|$ for the respective benchmark.

Random Networks From Figures 4(a) and 4(b), we see that AND/OR-graph sampling is better than AND/OR-tree sampling which in turn is better than pure IJGP-sampling. However there is not much difference in the error because the proposal distribution seems to be a very good approximation of the posterior.

Grid Networks All Grid instances have 1444 binary nodes and between 5-10 evidence nodes. From Figures 5(a) and 5(b), we can see that AND/OR-graph SampleSearch and AND/OR-tree SampleSearch are substantially better than

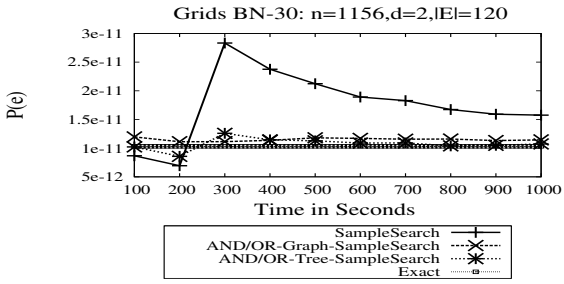


(a)

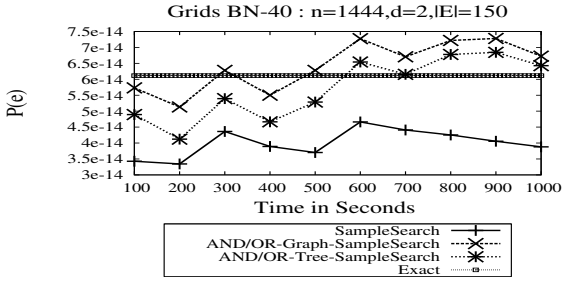


(b)

Figure 4: Random Networks



(a)



(b)

Figure 5: Grid Networks

pure SampleSearch.

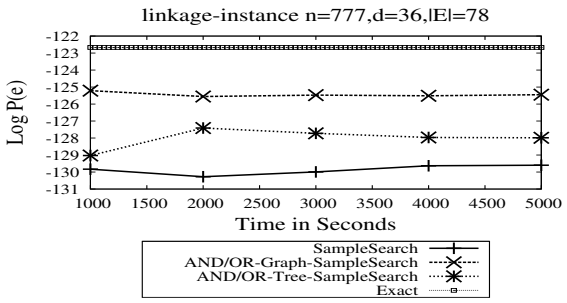
Linkage Networks The linkage instances are generated by converting a Pedigree to a Bayesian network [Fishelson and Geiger, 2003]. These networks have between 777-2315 nodes with a maximum domain size of 36. Note that it is hard to compute exact probability of evidence in these networks [Fishelson and Geiger, 2003]. We observe from Figures 6(a),(b) (c) and (d) that AND/OR-graph SampleSearch is substantially more accurate than AND/OR-tree SampleSearch which in turn is substantially more accurate than pure SampleSearch. Notice the log-scale in Figures 6 (a)-(d) which means that there is an order of magnitude difference between the errors. Our results suggest that AND/OR-graph and tree estimators yield far better performance than conventional estimators especially on problems in which the proposal distribution is a bad approximation of the posterior distribution.

7 Discussion

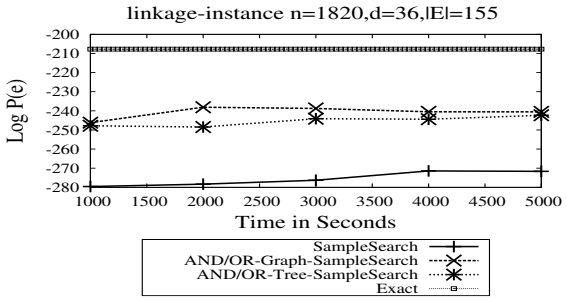
The work presented here is related to the work by [Hernndez and Moral, 1995, Kjærulff, 1995, Dawid et al., 1994] who perform sampling based inference on a junction tree. The main idea in these papers is to perform message passing on a junction tree by substituting messages which are too hard to compute exactly by their sampling-based approximations. [Kjærulff, 1995, Dawid et al., 1994] use Gibbs sampling while [Hernndez and Moral, 1995] use importance sampling to approximate the messages. Similar to some recent works on Rao-Blackwellised sampling such as [Bidyuk and Dechter, 2003, Paskin, 2004, Gogate and Dechter, 2005], variance reduction is achieved in these junction tree based sampling schemes because of some exact computations; as dictated by the Rao-Blackwell theorem. AND/OR estimation, however, does not require exact computations to achieve variance reduction. In fact, variance reduction due to Rao-Blackwellisation is orthogonal to the variance reduction achieved by AND/OR estimation and therefore the two could be combined to achieve more variance reduction. Also, unlike our work which focuses on probability of evidence, the focus of these aforementioned papers was on the belief updating task.

AND/OR-estimates are also closely related to *cross match estimates* [Kong, Augustine et al., 1997] which are based on Hoeffding's U-statistics. To derive cross-match estimates, the original function over a set of variables is divided into several marginal functions which are defined only on a subset of variables. Then, each marginal function is sampled independently and the cross-match sample mean is derived by considering all possible combinations of the samples. For example, if there are k marginal functions and m samples are taken over each function, the cross match sample mean is computed over m^k combinations. It was shown in [Kong, Augustine et al., 1997] that the cross match sample mean has lower variance than conventional sample mean; similar to our work.

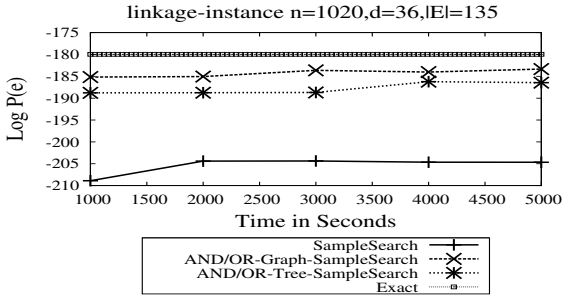
Given that the space complexity of our schemes is $O(nN)$, the reader may think that as more samples are drawn our algorithms would run out of memory. One can, however, perform multi-stage (adaptive) sampling to circumvent this problem. Here, at each stage we stop storing samples when



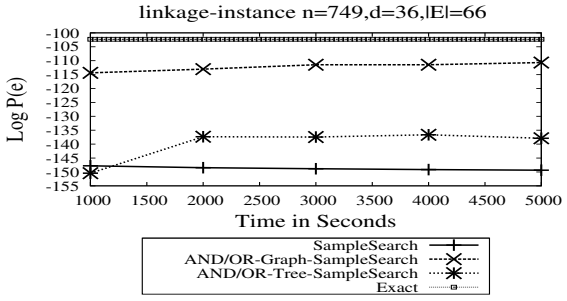
(a)



(b)



(c)



(d)

Figure 6: Linkage Bayesian Networks

a pre-specified memory limit is reached. Then AND/OR sample mean is computed from the stored samples and the samples are thrown away, repeating the process until a time bound expires or enough samples are drawn. The final sample mean is then simply the average of sample means computed at each stage. By linearity of expectation, the final sample mean way would be unbiased and obviously would have lower variance than conventional sample mean.

AND/OR sampling is based on a simple viewpoint: "make the most out of the generated samples". This is especially useful when samples are very expensive to obtain as in the case of bayesian networks with zero probabilities. Here, the only known practical scheme *SampleSearch* [Gogate and Dechter, 2007] generates a sample by solving a constraint satisfaction problem (CSP). Because solving a CSP is NP-complete, very few samples may be generated.

8 Summary

The paper introduces a new sampling based estimation technique called AND/OR importance sampling. The main idea of our new scheme is to generate samples in the usual way from a proposal distribution and then to derive statistics on the generated samples by using a AND/OR-tree or a AND/OR graph that takes advantage of the independencies present in the graphical model. We proved that the sample mean computed on a AND/OR-tree or a AND/OR graph may have smaller variance than the sample mean computed using the conventional approach. Our experimental evaluation is preliminary but quite promising showing that on most instances AND/OR sample mean has lower error than conventional importance sampling and sometimes by significant margins.

References

[Bidyuk and Dechter, 2003] Bidyuk, B. and Dechter, R. (2003). An empirical study of w-cutset sampling for bayesian networks. In *Proceedings of the 19th Annual Conference on Uncertainty in Artificial Intelligence (UAI-03)*.

[Cheng and Druzdzel, 2000] Cheng, J. and Druzdzel, M. J. (2000). Ais-bn: An adaptive importance sampling algorithm for evidential reasoning in large bayesian networks. *J. Artif. Intell. Res. (JAIR)*, 13:155–188.

[Dawid et al., 1994] Dawid, A. P., Kjaerulff, U., and Lauritzen, S. L. (1994). Hybrid propagation in junction trees. In *IPMU'94*, pages 87–97, London, UK. Springer-Verlag.

[Dechter and Mateescu, 2007] Dechter, R. and Mateescu, R. (2007). AND/OR search spaces for graphical models. *Artificial Intelligence*, 171(2-3):73–106.

[Fishelson and Geiger, 2003] Fishelson, M. and Geiger, D. (2003). Optimizing exact genetic linkage computations. In *RECOMB 2003*.

[Geweke, 1989] Geweke, J. (1989). Bayesian inference in econometric models using monte carlo integration. *Econometrica*, 57(6):1317–39.

[Gogate and Dechter, 2005] Gogate, V. and Dechter, R. (2005). Approximate inference algorithms for hybrid bayesian networks with discrete constraints. *UAI-2005*.

[Gogate and Dechter, 2007] Gogate, V. and Dechter, R. (2007). Samplesearch: A scheme that searches for consistent samples. *AISTATS 2007*.

[Hernndez and Moral, 1995] Hernndez, L. D. and Moral, S. (1995). Mixing exact and importance sampling propagation algorithms in dependence graphs. *International Journal of Approximate Reasoning*, 12(8):553–576.

[Kjærulff, 1995] Kjærulff, U. (1995). Hugs: Combining exact inference and gibbs sampling in junction trees. In *UAI*, pages 368–375.

[Kong, Augustine et al., 1997] Kong, Augustine, Liu, Jun S., and Wong, Wing Hung (1997). The properties of the cross-match estimate and split sampling. *The Annals of Statistics*, 25(6):2410–2432.

[Paskin, 2004] Paskin, M. A. (2004). Sample propagation. In Thrun, S., Saul, L., and Schölkopf, B., editors, *Advances in Neural Information Processing Systems 16*. MIT Press, Cambridge, MA.

[Rubinstein, 1981] Rubinstein, R. Y. (1981). *Simulation and the Monte Carlo Method*. John Wiley & Sons, Inc., New York, NY, USA.

[Yuan and Druzdzel, 2006] Yuan, C. and Druzdzel, M. J. (2006). Importance sampling algorithms for Bayesian networks: Principles and performance. *Mathematical and Computer Modelling*.