Proceedings of the Twenty-Third International Florida Artificial Intelligence Research Society Conference (FLAIRS 2010)

Local Importance Sampling in Multiply Sectioned Bayesian Networks

Karen H. Jin
School of Computer Science
University of Windsor
Windsor, Ontario
Canada N9B 3P4

Dan Wu
School of Computer Science
University of Windsor
Windsor, Ontario
Canada N9B 3P4

Abstract

The multiply sectioned Bayesian network (MSBN) is a well-studied model for probability reasoning in a multi-agent setting. Exact inference, however, becomes difficult as the problem domain grows larger and more complex. We address this issue by integrating approximation techniques with the MSBN Linked Junction Tree Forest (LJF) framework. In particular, we investigate the application of importance sampling in an LJF local junction tree. We propose an LJF local adaptive importance sampler (LLAIS) with improved sampling convergence and effective inter-agent message calculation. Our preliminary experiments confirm that the LLAIS sampler delivers a good approximation of MSBN local posterior beliefs as well as the message calculation over LJF linkage trees.

1. Overview

Multiply Sectioned Bayesian Network (MSBN) provides an exact model for cooperative agents to reason about the states of a distributed uncertain domain. Such a problem domain can be decomposed into subdomains, each individually represented and managed by a relatively lightweight single agent. Typically, inference in MSBN is carried out in a secondary structure called *linked junction tree forest* (LJF). Agents communicate through messages passed over LJF linkage trees, and belief updates in each LJF local junction tree (JT) are performed upon the arrival of a new inter-agent message. The LJF provides a coherent framework for exact inference with MSBNs, and is known to support consistent local inference in the absence of MSBN system-wide message passing (Xiang 2003). However, the computational costs may render such exact calculation impractical for larger and more complex problem domains. For example, a network may contain subnets that are too large to admit exact local representation. It is natural to consider the possibility of trading off exact inference against the calculation speed and communication cost with approximate approaches.

Although approximate techniques have been well developed in traditional BNs, the extension of these solutions to MSBNs has been very limited. The methods of two stochastic sampling techniques, forward sampling and Markov sampling, have been extended and compared with the LJF-based exact inference algorithms (Xiang 2003). Both proposed algorithms forgo the LJF structure and sample an MSBN directly in the global context. It has been shown that such approximation indeed requires more inter-agent messages passing, and at the cost of revealing more private knowledge of each local subnet. Furthermore, MSBN global sampling schema tend to explore only a small part of the entire multi-agent domain space.

We thus aim to maintain the LJF framework and explore localized approximation. In this paper, we focus on the technique of stochastic sampling. Local sampling in MSBNs would be straightforward if the subnets were valid BNs. Unfortunately, we have either an original subnet of a DAG structure with no marginal representation guaranteed, or an LJF local JT that is calibrated, but in the form of a JT (Jin and Wu 2008). In the case of the former, local sampling is not feasible due to the lack of prior marginal information. We can only resort to the local sampling of calibrated local JTs. Although there has been work done on stochastic sampling of BN JTs, we need an LJF local JT-based sampler that operates in a multi-agent context and can be integrated with the existing inter-agent communication schema. In particular, such a sampler should support efficient message calculation over the LJF linkage tree structure.

As we study the extension of BN importance sampling techniques to JTs, we present an LJF-based local adaptive importance sampler(LLAIS), which is viewed as the main contribution of this paper. We design our importance function as tables of posterior probabilities over the clusters of an LJF local JT. We adopt the adaptive importance sampling (Cheng and Druzdzel 2000), such that the importance functions are learned sequentially to approach the optimal distribution. One innovative feature of the LLAIS is that it facilitates inter-agent message calculation. We obtain an approximation of messages over linkage trees from the learned importance function. Although our experiments are preliminary, the results have shown that the LLAIS algorithm converges much faster compared to the other two local JT-based importance samplers we implemented. Also, with the LLAIS, a good approximation of inter-agent messages is available before the local sampling is completed.

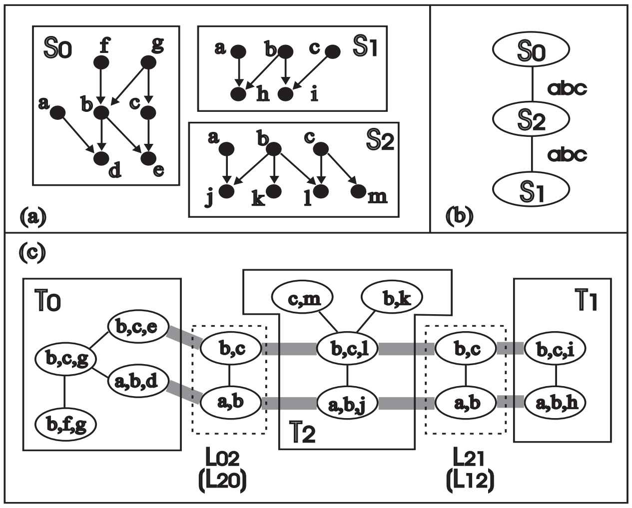

Figure 1: A simple MSBN. (a) three subnets; (b) the hypertree; (c) the corresponding LJF.

2. Background

2.1 Multiply Sectioned Bayesian Networks (MSBNs)

In this paper, we assume that the reader is familiar with Bayesian networks (BNs) and basic probability theory (Koller and Friedman 2009). Extended from the traditional BN model, MSBNs (Xiang 2002) provide a coherent framework for distributed probabilistic reasoning in the multi-agent paradigm. An MSBN is composed of a set of BN subnets each representing a partial view of a larger problem domain. The union of all subnet DAGs must also be a DAG, denoted $\mathcal{G}$. An MSBN is populated such that each BN subnet is maintained by an individual agent. These subnets are organized into a tree structure called a *hypertree* (Xiang 2002), denoted $\mathcal{H}$. Each hypertree node, known as *hypernode*, corresponds to a subnet; each hypertree link, known as *hyperlink*, corresponds to a *d-sepset*, which is the set of shared variables between adjacent subnets. A hypertree $\mathcal{H}$ is purposely structured so that (1) for any variable x contained in more than one subnet with its parents $\pi(x)$ in $\mathcal{G}$, there must exist a subnet containing $\pi(x)$; (2) the shared variables between any two subnets N_i and N_j are contained in each subnet on the path between N_i and N_j in $\mathcal{H}$. A hyperlink renders two sides of the network conditionally independent, similar to a separator in a junction tree (JT).

Fig. 1 (a) shows a simple MSBN with 3 subnets and Fig. 1 (b) shows the corresponding hypertree. Note that exactly one of all occurrences of a variable x (in a subnet containing $\{x\} \cup \pi(x)$) is assigned the CPD $p(x|\pi(x))$. For example, the CPD of $P(a)$ is assigned to only one of the three subnets. Hence, the MSBN local subnets are partially quantified DAGs, not valid BNs (Jin and Wu 2008).

A derived dependence structure called a *linked junction tree forest* (LJF) is typically used for distributed inference in MSBNs. An LJF is constructed through a process of cooperative and distributed compilation. Each hypernode in the hypertree $\mathcal{H}$ is transformed into a local JT, and each hyperlink is transformed into a *linkage tree*, which is a JT constructed from the d-sepset. Each cluster of a linkage tree is called a *linkage*, and each separator, a *linkage separator*. The cluster in a local JT that contains a linkage is called a *linkage host*. Although two adjacent subnets may each maintain a different linkage tree over the same d-sepset, it has been proven that the communication between the two subnets is not affected (Xiang 2002).

Fig. 1 (c) shows an LJF constructed from the MSBN in Fig. 1 (a) and (b). Local JTs, T_0, T_1 and T_2, constructed from BN subnets G_0, G_1 and G_2 respectively, are enclosed by boxes with solid edges. The linkage trees, $L_{01}(L_{10})$ and $L_{12}(L_{21})$, are enclosed by boxes with dotted edges. The linkage tree L_{02} contains two linkages $\{b, c\}$ and $\{a, b\}$, with linkage separator b (not shown in the figure). The linkage hosts of T_0 for L_{02} are clusters $\{b, c, e\}$ and $\{a, b, d\}$.

2.2 LJF Inter-agent Communication

Existing MSBN LJF inter-agent communication schema typically involve two rounds of LJF global messages passing (Xiang 2002; Xiang, Jensen, and Chen 2006). Inter-agent messages are passed recursively, inward and outward, relative to a randomly selected LJF root node. A message, delivered from an LJF subnet T_i to its adjacent subnet T_j, consists of *extended linkage potential*s over T_i's linkage tree L_{ij} to T_j (Xiang 2002). The calculation starts with the linkage potential $\Phi(Q_i) = \sum_{C_i \setminus Q_i} \Phi(C_i)$, where C_i is the linkage host of a linkage Q_i in T_i, but redundant information over the linkage tree separators presents in the set of linkage potentials. For example, in Fig. 1 (c), the set of linkage potentials from T_0 to T_2 consists of potentials for linkages $\{b, c\}$ and $\{a, b\}$, both carrying information over b. To solve this problem, each linkage separator R_i is associated with one neighboring linkage Q_i as Q_i's *peer separator*. Therefore, by defining *extended linkage potential* as $\Phi^*(Q) = \Phi(Q)/\Phi(R)$, we have the redundancy removed by division (Xiang 2002).

Exact inference in MSBN LJFs could come to a halt due to communication bottlenecks caused by certain large nodes. Since an agent's belief must be updated exactly and repeatedly whenever a new message arrives, a large amount of local calculation is required. It is thus natural for us to attempt localized approximation, as we trade off certain levels of accuracy to improve the local computation and overall performance of LJF global inference.

2.3 Importance Sampling for BNs

Importance sampling is a class of Monte Carlo algorithms for approximate reasoning in BNs. As a commonly used simulation technique, importance sampling samples a modified distribution, known as the *importance function*, to estimate a hard-to-sample distribution. In order to evaluate a sum $I = \sum_{x \in X} g(x)$ for some real function g, samples are generated from an importance function f such that $g(x) \neq 0 \Longrightarrow f(x) \neq 0$. We then estimate I as follows:

$$I = \sum_{x \in X} g(x) = \sum_{x \in X} \frac{g(x)}{f(x)} f(x) = E_f[\frac{g(x)}{f(x)}]$$

and

$$\hat{I} = \frac{1}{N}\sum_{i=1}^{N} w(x^i),$$

where $w(x^i) = \frac{g(x^i)}{f(x^i)}$ is called the *sample weight*.

In order to compute the probability of evidence $P(\mathbf{E} = \mathbf{e})$ from a JPD $P(\mathbf{X}) = \prod_{i=1}^{n} P(X_i|Pa(X_i))$ of a BN model, we need to sum over all non-evidence nodes:

$$P(\mathbf{E} = \mathbf{e}) = \sum_{P(\mathbf{X}\backslash\mathbf{E},\mathbf{E}=\mathbf{e})} \prod_{i=1}^{n} P(X_i|Pa(X_i))|_{\mathbf{E}=\mathbf{e}}$$

and we can apply the principle of importance sampling. We can obtain the posterior probability distribution $P(\mathbf{X}|\mathbf{E})$ by separately computing the two terms $P(\mathbf{X}, \mathbf{E})$ and $P(\mathbf{E})$, and then combining them by the definition of conditional probability.

Essentially, importance sampling algorithms for BNs only differ in the way they obtain the importance function which represents the sampling distribution. Several choices are available ranging from the prior distribution as in the likelihood weighing algorithm (Koller and Friedman 2009), to more sophisticated alternatives. The latter includes algorithms that update the importance function through a learning process (Cheng and Druzdzel 2000), or calculate the importance function directly with loopy belief propagation (Yuan and Druzdzel 2003). These methods try to gradually approach the optimal importance function, which is usually a function proportional to the posterior distribution, and preferably with a thick tail (Yuan and Druzdzel 2003).

3. Basic Importance Sampling for LJF local JT

Earlier research has suggested difficulties in applying stochastic sampling to MSBNs at a global level (Xiang 2003). Direct local sampling in an MSBN subnet is also not feasible due to the absence of a valid BN structure. However, an LJF local JT, the secondary structure of a subnet, can be calibrated with a marginal over all the local variables (Jin and Wu 2008), making local sampling possible. Algorithms have been proposed to combine sampling with JT belief propagation (Kjaerulff 1995; Paskin 2004; Koller and Friedman 2009), mainly for hybrid BNs. Although generally applicable to a calibrated LJF local JT, these algorithms do not support efficient inter-agent message calculation in the context of MSBNs.

We now introduce a JT-based importance sampler, which will be extended in the next sections. Importance sampling in JTs was previously studied (Luis D. Hernndez and Salmern 1998), such that the importance function was composed of some factors of JT clusters. In our case, however, an explicit form of the importance function is necessary as it facilitates the learning of the optimal sampling distribution, as well as the efficient calculation of inter-agent messages.

The JPD over all the variables in a calibrated local JT can be recovered as a decomposable model similar to the BN DAG factorization. Let $C_1, ..., C_m$ be the m JT clusters given in an ordering that satisfies the running intersection property. The *separator* is $S_i = \emptyset$ for $i = 1$ and $S_i = C_i \cap (C_1 \cup ... \cup C_{i-1})$ for $i = 2, ..., m$. Since $S_i \subset C_i$, we have the *residual* defined as $R_i = C_i \backslash S_i$. The JT running intersection property guarantees that the separator S_i separates the residual R_i from the set $(C_1 \cup ... \cup C_{i-1})\backslash S_i$ in the JT. Thus we apply the chain rule to partition residuals given by the separators and have the JPD expressed as $P(C_1, ..., C_m) = \prod_{i=1}^{m} P(R_i|S_i)$. Essentially, we select a root from the JT clusters and direct all links(separators) away from the root to form a directed *sampling JT*. This directed tree is analogous to a BN due to their similar forms of recursive factorization.

Given a JPD factorization of an LJF local JT, we define the importance function P' in our basic sampler as

$$P'(\mathbf{X}\backslash\mathbf{E}) = \prod_{i=1}^{m} P(R_i\backslash\mathbf{E}|S_i)|_{\mathbf{E}=\mathbf{e}} \tag{1}$$

This importance function is factored into a set of local components, each corresponding to a JT cluster. Given the calibrated potential on each JT cluster C_i, we can calculate $P(R_i|S_i)$ for each cluster directly. For the root cluster, that is $P(R_i|S_i) = P(R_i) = P(C_i)$.

We traverse a sampling JT and sample variables of the residue set in each cluster corresponding to the local conditional distribution. This is done similarly to the sampling of BNs, except that we now sample a group of nodes in a cluster instead of an individual node. If we encounter a cluster that contains a node in the evidence set $\mathbf{E}$, we simply assign to the node the value given by the evidence assignment. A complete sample consists of assignments to all non-evidence nodes according to the local JT's prior distribution. The score of each sample s_i is calculated as

$$Score_i = \frac{P(s_i, \mathbf{E})}{P'(s_i)}. \tag{2}$$

Unfortunately, using the "prior" as the sampling distribution, our basic sampler may perform poorly if the posterior distribution of the network bears little resemblance to the prior. It is proven that the optimal importance function for BN importance sampling is the posterior distribution $P(\mathbf{X}|\mathbf{E} = \mathbf{e})$ (Cheng and Druzdzel 2000). Applying this result to JTs, we can define the corresponding optimal importance function as

$$\rho(\mathbf{X}\backslash\mathbf{E}) = \prod_{i=1}^{m} P(R_i\backslash\mathbf{E}|\mathbf{E} = \mathbf{e}). \tag{3}$$

Eq. (3) takes into account the influences of all evidence from all clusters in the sample of the current cluster, whereas Eq. (1) only counts the influence from the precedent cluster, thus causing poor sampling results. Moreover, as there are potentially large differences between the two distributions, we can not exploit the form of the importance function Eq. (1) in our basic sampler for inter-agent message estimation.

4. LJF-based Local Adaptive Importance Sampler (LLAIS)

Our objectives in designing an LJF local JT importance sampler are: 1) search for a good importance function for the best approximation, and 2) facilitate inter-agent message calculation over LJF linkage trees. In this section, we introduce the LLAIS sampler which incorporates a learning process to update the importance function Eq. (1) of our basic sampler. We also show that inter-agent messages can be composed directly from the learned importance function.

4.1 Updating the Sampling Distribution

Although we know that the posterior distribution is the optimal sampling distribution, it is usually difficult to compute the optimal importance function in Eq. (3) directly. We can, however, parameterize the sampling distribution to be as close as possible to the posterior distribution. We choose a sub-optimal importance function

$$\rho(\mathbf{X}\backslash\mathbf{E}) = \prod_{i=1}^{m} P(R_i\backslash\mathbf{E}|S_i, \mathbf{E} = \mathbf{e}) \quad (4)$$

and represent it as a set of local tables which is learned to approach the optimal sampling distribution. These tables are called the *Clustered Importance Conditional Probability Table*(CICPT).

The CICPT tables, one for each local JT cluster, are tables of probabilities indexed by the separator to the precedent cluster (based on the cluster ordering in the sampling tree) and conditioned by the evidence. They have a similar structure to the factored importance function in our basic importance sampling algorithm. However, the CICPT tables are updated periodically by the scores of samples generated from the previous tables. A CICPT table is analogous to an ICPT table of BN adaptive importance sampling (Cheng and Druzdzel 2000), but applied in the context of LJF local JTs.

A simple learning strategy is to re-calculate the CICPT table based on the most recent batch of samples, so we count the influence of all evidence through the current sample set. But such a learning process could oscillate as we completely ignore the previous CICPT tables at the calculation of new ones. Therefore, we adopt a smooth learning function and our algorithm takes the form:

Algorithm LLAIS
Step 1. *Specify the total sample number M, total updates K and update interval L. Initialize CICPT tables as in Eq. (1).*
Step 2. *Generate L samples with scores according to the current CICPT table. Estimate $P'(R_i|S_i, \mathbf{e})$ by normalizing the score for each residue set given the states of the separator set.*
Step 3. *Update the CICPT tables based on the following learning function (Cheng and Druzdzel 2000):*
$$P^{k+1}(R_i|S_i, \mathbf{e}) = (1 - \eta(k))P^k(R_i|S_i, \mathbf{e}) + \eta(k)P'(R_i|S_i, \mathbf{e}),$$
where $\eta(k)$ is the learning function.
Step 4. *Modify the importance function if necessary, with the heuristic of ϵ-cutoff. For the next update, go to step 2.*
Step 5. *Generate the M samples from the learned importance function and calculate scores as in Eq. (2).*
Step 6. *Output posterior distribution for each node.*

In LLAIS, the importance function is dynamically tuned from the initial prior distribution. New samples are obtained from the current importance function and then used to gradually refine the distribution. The learning overhead is expected to be compatible with that of the BN adaptive importance sampling (Cheng and Druzdzel 2000). Given that thick tails are desirable for importance sampling in BNs (Yuan and Druzdzel 2003), we also adopt the heuristic of ϵ-cutoff (Cheng and Druzdzel 2000). If less than a threshold ϵ, the small probabilities will be replaced by ϵ, and the change will be compensated by subtracting the difference from the largest probability entry.

4.2 Handling Evidence

In BN JTs, if an observed node is contained in more than one cluster, the evidence is typically inserted randomly into any of the clusters. With our LLAIS sampler, however, we enter the observation into a local JT cluster which contains the evidence node and is also the nearest to the local JT's root cluster. This simple rule is based on the following theorem.

Theorem 1. Suppose a sampling tree T. $Anc(\mathbf{E})$ is the ancestor cluster(s) to the clusters that contain evidence $\mathbf{E}$. Then, $C_i \notin Anc(\mathbf{E}) \Longrightarrow P(R_i|S_i, \mathbf{E}) = P(R_i|S_i)$

Proof. Suppose for cluster C_i, the values of its corresponding S_i of R_i are set. Then R_i is dependent on evidence $\mathbf{E}$ given S_i only when R_i is d-connecting with $\mathbf{E}$ given S_i. Since T is a directed tree, this happens only when there exists a cluster of C_i's descendants that belongs to the clusters containing evidence $\mathbf{E}$. That is, $C_i \notin Anc(\mathbf{E})$. □

Based on Theorem 1, if a cluster is not the ancestor of clusters with evidence entered, its CICPT table remains unchanged. That is, after the CICPT tables are initialized in Step 1 of our algorithm, we simply need to update the tables for clusters that are the ancestors of the evidence. By entering new evidence into a cluster nearest to the root, we maximize the number of CICPT tables that require no updates with regard to the evidence node. This will result in considerable savings in the learning process of the importance function.

4.3 Calculating Inter-agent Message over Linkage Tree

In an MSBN LJF, agents propagate the impact of their local observation through inter-agent messages passing. Originated from one LJF local JT to one of its adjacent local JTs, an inter-agent message consists of extended linkage potentials over their corresponding linkage tree. With the basic importance sampler, we can only estimate these potentials from the complete sample set. By exploiting the adaptive feature of our LLAIS sampler, however, we are able to obtain an approximation of the extended linkage potentials directly from the learned importance function.

Suppose we have a linkage tree L that spans over a set of linkage hosts including the root cluster of a local JT T. We can prove that, for each linkage Q in L, there exists at least one linkage host C_Q with a CICPT table $P(R_Q|S_Q, \mathbf{E})$,

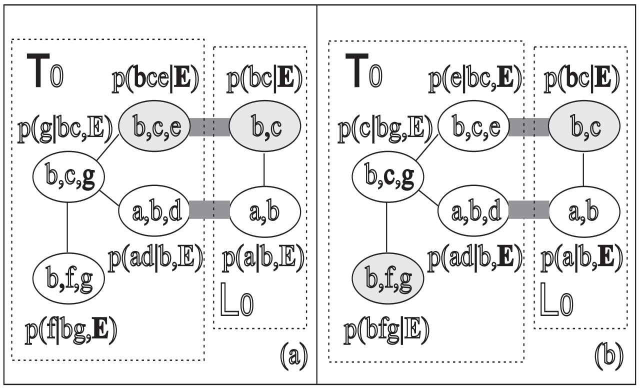

Figure 2: Estimation of extended linkage potentials.

such that the extended linkage potential of Q can be estimated as

$$\Phi^*(Q) \approx \sum_{N_i \notin Q} P(R_Q|S_Q, \mathbf{E}) \tag{5}$$

That is, for each linkage Q, we always have a linkage host C_Q from which an approximation of the extended linkage potential of Q can be obtained by summing out all irrelevant variables from C_Q's CICPT table. Complete proof is omitted due to space limitation.

Consider the example shown in Fig. 2(a). T_0 is the local JT and L_0 is the linkage tree. The root cluster of T_0 is $\{b, c, e\}$ (marked as shaded). L_0 spans over two linkage hosts $\{b, c, e\}$ and $\{a, b, d\}$ of T_0. By selecting the linkage $\{b, c\}$ as the root linkage of L_0, we can estimate the extended linkage potential of both linkages $\{b, c\}$ and $\{a, b\}$ by marginalization from each corresponding linkage host's CICPT table.

If a linkage tree does not host in the local JT's root cluster, we can still apply the same method of estimation for all linkages except the root linkage. As shown in Fig. 2(b), the local JT T_0 is now rooted at cluster $\{b, f, g\}$, instead of $\{b, c, e\}$ from Fig. 2(a). This change of rooting in local JT will affect the calculation of L_0's root linkage $\{b, c\}$. While we can still obtain the extended linkage potential of L_0's all non-root linkages directly with Eq. (5), the root linkage $\{b, c\}$'s extended linkage potential $P(bc|\mathbf{E})$ cannot be marginalized directly from $P(e|bc, \mathbf{E})$, which is the CICPT table of $\{b, c\}$'s linkage host $\{b, c, e\}$. However, several solutions are available to solve this special case. One option is to obtain the estimation from the most recent update of the linkage host's CICPT table.

The main advantage of our message estimation schema is that we can estimate inter-agent messages before the complete set of samples is available. The approximation error decreases as the importance function approaches the optimal distribution. Essentially, the closer the CICPT table is to the true posterior distribution, the less error there is in our message estimation. Overall, the compromised accuracy can be properly compensated by the increased efficiency of LJF global communication.

5. Experiments

We conducted our preliminary experiments by comparing the LLAIS algorithm with two other variations of LJF local JT importance samplers, which are the basic importance sampler described in section 3 and the adaptive importance sampler described in section 4.1. We have not located in literature any previous application of importance sampling to MSBNs LJFs, or JT-based importance sampling with explicit forms of importance function. We performed initial tests on a sampling JT constructed from the Alarm network (total 37 nodes). We evaluated the approximation accuracy in terms of the Mean Square Error(MSE)

$$MSE = \sqrt{\frac{1}{\sum_{X_i \in X \setminus E} N_i} \sum_{x_i \in x \setminus E} \sum_{j=1}^{n} (p'(x_{ij}) - p(x_{ij}))^2}$$

where N is the set of all nodes, E is the set of evidence and N_i is the number of outcomes of node i. $P'(X_{ij})$ and $P(X_{ij})$ are the sampled and exact marginal probability of the state j of a node i. We obtained the gold standard potential using the standard JT propagation.

We generated a total of 30 test cases which include three sequences of 10 test cases each. The three sequences had 9, 11 and 13 evidence nodes respectively. Most evidence nodes were in the leaf clusters of the sampling JT. Each algorithm was evaluated with $M = 5000$ samples. With LLAIS, we used the learning function (Cheng and Druzdzel 2000) $\eta(k) = a(\frac{b}{a})^{k/k_{max}}$ and set $a = 0.4, b = 0.14$ and the total updates $K = 5$. In each updating step, $L = 2000$. We also separately ran the basic local JT importance sampler without evidence in the same network with 5000 samples for 10 times, resulting in an average MSE of 0.006. This result reflected the optimal accuracy since the results of probabilistic logic sampling without evidence approach the limit of how well stochastic sampling can perform.

Fig. 3 shows the results for all test cases of our first experiment. Each test case was run 10 times and the average MSE was recorded as a function of the probability of the evidence. As far as the magnitude of difference was concerned, the LLAIS performed much better and with significantly better stability than the other two importance samplers, named as LLS1 and LLS2 respectively. In particular, the performance of LLAIS does not degenerate with less likely evidence, which is consistent with the results reported with the BN adaptive importance sampling. The minimum MSE of 0.0056 is within the range of the optimal result. The average MSE of LLAIS is 0.0106 with a medium of 0.0093, which is much smaller than the average MSE of 0.1376 and 0.0551 with the other two samplers. Although the average result for the LLAIS was larger than the optimal accuracy, it was understandable since we updated our importance function for only 5 times, and used a small set of 2000 samples. This short process imposed a small learning overhead, but might not have included enough iterations required for converging to the optimal distribution.

We also performed simulations to evaluate the accuracy of inter-agent message estimation. We randomly selected a subtree from the sampling JT and treated it as a linkage tree. We assumed each linkage host contained the same nodes as the corresponding linkage, and the JT root cluster was in-

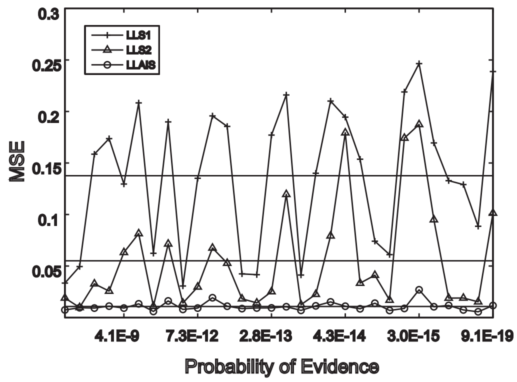

Figure 3: Performance of LLAIS, compared with two variations of LJF local importance samplers: MSE for each of 30 test cases plotted against the probability of evidence.

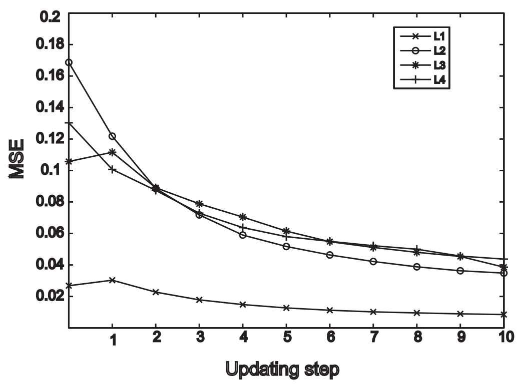

Figure 4: Convergence of extended linkage potentials of 4 linkages simulated in a test run of LLAIS.

cluded as a linkage host. We used a test case from the previous experiment, which contained 11 evidence nodes. We compared the estimates with the exact results of extended linkage potentials for a total of 4 linkages.

Fig. 4 shows the convergence of the extended linkage potentials with K=10 and L=2000. At each update, the average MSE of 10 runs was recorded for each linkage. It showed that although minor conciliation occurred at the early stage, all 4 linkage potentials had converged to an average of 0.0314. The error for non-root linkages L_2, L_3 and L_4, however, was larger than we had seen in the first experiment. We believe it was due to the simple method we used to calculate $P(R_i|S_i, E = e)$. We estimated $P(C_i, E)$ and $P(S_i, E)$ separately from the same set of samples, and combine them by conditional probability. This method may introduce a large variance if the numeric value of $P(R_i|S_i, E = e)$ is extreme. Nevertheless, our message approximation scheme enables the propagation of local beliefs at a much earlier stage of the whole sampling process, which promotes efficient inter-agent communication at the LJF global level.

6. Conclusion

In this paper, we have studied the application of importance sampling in MSBN subnets. We have presented the LLAIS sampler, which integrates local importance sampling with the existing LJF framework. The LLAIS sampler adopts the adaptive importance sampling technique for improved sampling accuracy. The dynamic tuning of our importance function also facilitates inter-agent message calculation over LJF linkage trees. In our preliminary experiments, the LLAIS sampler demonstrated promising results for the estimations of both local posterior belief and linkage tree messages. We believe our algorithm represents an important step in solving MSBN communication bottlenecks and realizing practical inference for larger scale multi-agent probabilistic systems.

One direction of our future work will be improving the LLAIS algorithm, which includes methods to estimate posterior distribution with better accuracy, and to improve the learning process for importance functions. Moreover, an important question that remains unanswered is how local accuracy will affect the overall performance of the entire network. As currently we have only simulated LJF local JTs from BN JTs, further experiments are necessary in full scale MSBNs.

References

Cheng, J., and Druzdzel, M. J. 2000. BN-AIS: An adaptive importance sampling algorithm for evidential reasoning in large bayesian networks. *Artificial Intelligence Research* 13:155–188.

Jin, K. H., and Wu, D. 2008. Marginal calibration in multi-agent probabilistic systems. In *Proceedings of the 20th IEEE Int'l Conference on Tools with Artificial Intelligence*.

Kjaerulff, U. 1995. Hugs: Combining exact inference and gibbs sampling in junction trees. In *Eleventh Conference on Uncertainty in Artificial Intelligence*, 368–375.

Koller, D., and Friedman, N. 2009. *Probabilistic Graphical Models: Principles and Techniques*. MIT Press.

Luis D. Hernndez, S. M., and Salmern, A. 1998. A monte carlo algorithm for probabilistic propagation in belief networks based on importance sampling and stratified simulation techniques. *International Journal of Approximate Reasoning* 18:53–91.

Paskin, M. A. 2004. Sample propagation. In Thrun, S.; Saul, L.; and B.Scholkoph., eds., *Advances in Neural Information Processing Systems 16*. Cambridge, MA: MIT Press.

Xiang, Y.; Jensen, F. V.; and Chen, X. 2006. Inference in multiply sectioned bayesian networks: Methods and performance comparison. In *IEEE Transaction on Systems, Man, and Cybernetics*, volume 36, 546–558.

Xiang, Y. 2002. *Probabilistic Reasoning in Multuagent Systems: A Graphical Models Approach*. Cambridge.

Xiang, Y. 2003. Comparison of multiagent inference methods in multiply sectioned bayesian networks. *Approximate Reasoning* 33:235–254.

Yuan, C., and Druzdzel, M. J. 2003. An importance sampling algorithm based on evidence pre-propagation. In *19th Conference on Uncertainty in Artificial Intelligence*, 624–631.