An Exact Algorithm for Solving Most Relevant Explanation in Bayesian Networks

Xiaoyuan Zhu and Changhe Yuan

Queens College
City University of New York
65-30 Kissena Blvd., Queens, NY 11367
{xiaoyuan.zhu, changhe.yuan}@qc.cuny.edu

Abstract

Most Relevant Explanation (MRE) is a new inference task in Bayesian networks that finds the most relevant partial instantiation of target variables as an explanation for given evidence by maximizing the Generalized Bayes Factor (GBF). No exact algorithms have been developed for solving MRE previously. This paper fills the void and introduces a breadth-first branch-and-bound (BFBnB) algorithm. The key idea of the proposed method is to decompose the whole Bayesian network into a set of overlapping subnetworks. Each subnetwork is characterized by a subset of evidence variables and its Markov blanket. An upper bound for the GBF of an MRE solution is derived by solving independent optimization problems in the subnetworks. To tighten the upper bound, we also introduce a greedy algorithm for merging the Markov blankets that share target variables. The upper bound is used in the BFBnB algorithm for pruning the search space. We evaluate the proposed algorithm in a set of benchmark diagnostic Bayesian networks. The experimental results show that BFBnB algorithm makes exact MRE inference tractable in Bayesian networks that could not be solved previously.

Introduction

Bayesian networks are probabilistic models that represent the conditional dependences between random variables as directed acyclic graphs, and provide principled approaches to explaining evidence, e.g., Most Probable Explanation (MPE), Maximum a Posteriori (MAP), and Most Relevant Explanation (MRE). Both MPE and MAP are solved by optimizing the joint posterior probability of the target variables given the evidence. However, in explanation problems (Lacave and Diez 2002; Nielsen, Pellet, and Elisseeff 2008), MPE and MAP often produce overspecified explanations (Pacer et al. 2013; Yuan et al. 2009), i.e., irrelevant target variables may be included. This is because maximizing the joint probability alone cannot exclude irrelevant target variables directly. To address the limitation of MPE and MAP, MRE (Yuan, Lim, and Lu 2011; Yuan and Lu 2008) was proposed to find a partial instantiation of the target variables that maximizes the Generalized Bayes Factor (GBF) (Fitelson 2007; Good 1985) as an explanation for the evidence. GBF is a rational function of probabilities that is suitable for comparing explanations with different cardinalities. Theoretically, MRE is shown able to prune away independent and less relevant variables from the final explanation (Yuan, Lim, and Lu 2011): a recent human study also provided empirical evidence to support the claim (Pacer et al. 2013).

Due to the difficulty of finding a meaningful bound for GBF, no exact algorithms have been developed to solve MRE. Only local search and Markov chain Monte Carlo methods have been proposed previously (Yuan, Lim, and Littman 2011). In this paper, we introduce the first non-trivial exact MRE algorithm, a Breadth-First Branch-and-Bound (BFBnB) algorithm. The key idea of the proposed method is to decompose the whole Bayesian network into a set of overlapping subnetworks. Each subnetwork is characterized by a subset of evidence variables and its Markov blanket. An upper bound for the GBF of an MRE solution is derived by solving independent optimization problems in the subnetworks. To tighten the upper bound, we also introduce a greedy algorithm for merging the Markov blankets that share target variables. The upper bound is used in the BFBnB algorithm for pruning the search space. We evaluate the proposed algorithm in a set of benchmark diagnostic Bayesian networks. The experimental results show that BFBnB algorithm makes exact MRE inference tractable in Bayesian networks that could not be solved previously.

Background

A Bayesian network (Pearl 1988; Koller and Friedman 2009) is represented as a Directed Acyclic Graph (DAG). The nodes in the DAG represent random variables. The lack of arcs in the DAG define conditional independence relations among the nodes. If there is an arc from node $Y$ to $X$, we say that $Y$ is a parent of $X$, and $X$ is a child of $Y$ (We use upper-case letters to denote variables $X$ or variable sets $X$, and lower-case letters for values of scalars $x$ or vectors $x$.). A node $Y$ is an ancestor of a node $X$, if there is a directed path from $Y$ to $X$. Let $\text{an}(X)$ denote all the ancestors of $X$, then the smallest ancestral set $\text{an}(X)$ of node set $X$ is defined as $\text{an}(X) = X \cup (\bigcup_{X_i \in X} \text{an}(X_i))$.

$D$-separation describes the conditional independence relation between two set of nodes $X$ and $Y$, given a third set of nodes $Z$, i.e., $p(X|Z,Y) = p(X|Z)$. The Markov blanket of $X$, denoted by $MB(X)$, is the smallest node set which d-separates $X$ from the remaining nodes in the network. The network as a whole represents the joint probability distribution of $\prod_X p(X|PA(X))$, where $PA(X)$ is the set of all the parents of $X$.

The moral graph $\bar{G}$ of a DAG $G$ is an undirected graph with the same set of nodes. There is an edge between $X$ and
Y in \( G^m \) if and only if there is an edge between them in \( G \) or if they are parents of the same node in \( G \). In an undirected graph, \( Z \) separates \( X \) and \( Y \), if \( Z \) intercepts all paths between \( X \) and \( Y \). Moral graph is a powerful construction to explain d-separation. Lemma 1 (Lauritzen et al. 1990) links d-separation in DAG to separation in undirected graphs.

\textbf{Lemma 1.} Let \( X, Y, \) and \( Z \) be disjoint subsets of nodes in a DAG \( G \). Then \( Z \) d-separates \( X \) from \( Y \) if and only if \( Z \) separates \( X \) from \( Y \) in \( (G_{An}(X \cup Y \cup Z))^m \), where \( (G_{An}(X \cup Y \cup Z))^m \) is the moral graph of the subgraph of \( G \) with node set \( An(X \cup Y \cup Z) \).

In an inference problem in a Bayesian network, the nodes are often classified into three categories: target, evidence, and auxiliary. The target set \( M \) represents variables of inference interest. The evidence set \( E \) represents observed information. The auxiliary set represents the variables that are not of interest in the inference. MRE (Yuan, Lim, and Lu 2011) finds a partial instantiation of \( M \) as an explanation for given evidence \( e \) in a Bayesian network. Here, explanation refers to the explanation of evidence, whose goal is to explain why some observed variables are in their particular states using the target variables in the domain. The search space of MRE is large and the complexity of MRE is conjectured to \( NP \) complete (Yuan, Lim, and Littman 2011). Formally, the MRE problem is formulated as follows:

\textbf{Definition 1.} Let \( M \) be a set of targets, and \( e \) be the given evidence in a Bayesian network. Most Relevant Explanation is the problem of finding a partial instantiation \( \bar{x} \) of \( M \) that has the maximum generalized Bayes factor score \( GBF(\bar{x}; e) \) as the explanation for \( e \), i.e.,

\[
MRE(M; e) = \arg \max_{\bar{x}, \emptyset \subset \bar{X} \subseteq M} GBF(\bar{x}; e),
\]

where \( GBF \) is defined as

\[
GBF(\bar{x}; e) = \frac{p(\bar{x}|e)}{p(e|\bar{x})}.
\]

Here, \( \bar{x} \) is the instantiation of \( X \), and \( \bar{x} \) represents all of the alternative explanations of \( x \).

Different from MPE and MAP which maximize a joint posterior probability function, MRE maximizes the rational function of probabilities in Equation 2. This makes it possible for MRE to compare explanations with different cardinalities in a principled way.

Belief update ratio is a useful concept. The belief update ratio of \( X \) given \( e \) is defined as follows:

\[
r_1(\bar{x}; \bar{e}) = \frac{p(\bar{x}|\bar{e})}{p(\bar{x})}.
\]

GBF can be calculated from the belief update ratio as follows:

\[
GBF(\bar{x}; e) = \frac{p(\bar{x}|e)(1 - p(\bar{x}))}{p(\bar{x})(1 - p(\bar{x}|e))} = \frac{r(\bar{x}; e) - p(\bar{x}|e)}{1 - p(\bar{x}|e)}
= 1 + \frac{r(\bar{x}; e) - 1}{1 - p(\bar{x}|e)}.
\]

Many local search methods have been applied to solve both MAP (MPE) (Park and Darwiche 2001) and MRE (Yuan, Lim, and Littman 2011), such as tabu search (Glover 1990). Tabu search starts at an empty solution set. At each step, it generates the neighbors of the current solution by adding, changing, or deleting one target variable. Then tabu search selects the best neighbor which has the highest GBF score and has not been visited before. In tabu search, the best neighbor can be worse than the current solution. To stop tabu search properly, upper bounds are set on both the total number of search steps \( M \) and the number of search steps since the last improvement \( L \) as the stopping criteria. The local search methods can only provide approximate solutions which are not guaranteed to be optimal. Furthermore, the accuracy and efficiency of these methods are typically highly sensitive to tunable parameters.

Branch-and-bound algorithms have been developed for solving MAP and MPE by using upper bounds derived based on the property of optimization criterion or the structure of Bayesian networks. A mini-bucket upper bound is proposed in (Dechter and Rish 2003) and applied to And/Or tree search for solving MPE (Marinescu and Dechter 2009). The work in (Choi, Chavira, and Darwiche 2007) showed that the mini-bucket upper bound can be derived from a node splitting scheme. To solve MAP exactly, an upper bound is proposed in (Park and Darwiche 2003) by commuting the order of max and sum operations in the MAP calculation. In (Huang, Chavira, and Darwiche 2006), an exact algorithm is proposed for solving MAP by computing upper bounds on an arithmetic circuit compiled from a Bayesian network. No upper bound or exact algorithm has been proposed for solving MRE, however.

\textbf{An Exact MRE Algorithm Based on A Novel Upper Bound}

It is difficult to solve MRE problems exactly because of both an exponential search space and the need for probabilistic inference at each search step. A naive brute-force search method can scale to Bayesian networks with at most 15 targets. In this work, we propose a breadth-first branch-and-bound algorithm that uses a new upper bound based on Markov blanket decomposition to prune the search space. The algorithm makes it possible to solve MRE problems with more targets exactly.

\textbf{Search space formulation}

Assuming there are \( n \) targets, and each target has \( d \) states, the search space of MRE contains \( (d + 1)^n - 1 \) possible joint states (or solutions). Each state contains values of a subset of the targets. We organize the search space as a search tree by instantiating the targets according to a total order \( \pi \) of the targets. The search tree has the empty state as the root. For a state \( y \) in the tree, \( Y \) is defined as the expanded set of targets, and \( X = \{X|X \in M; \forall Y_i \in Y, Y_i \sqsubset_{\pi} X\} \) is the unexpanded set. That is, the unexpanded set only contains targets that trail all of the expanded targets in \( \pi \). Each child state of \( Y \) instantiates one more unexpanded target. Figure 1 shows an example with two targets \( A = \{a, a\} \) and \( B = \{b, b\} \) in this particular order. Assuming the current state is \( \{\} \), \( \{\} \) is the expanded set of target(s), and \( \{\} \) is the unexpanded
set. So \{b\} has no child state. As a result, different branches of the search tree may have different numbers of layers, but all of the states with the same cardinality appear in the same layer.

It is possible to use dynamic ordering to expand the targets that can most improve the GBF score first. However, it was shown in (Yuan and Hansen 2009) that a static ordering can actually make computing upper bounds and ultimately the search more efficient. We therefore simply ordered the targets according to their indices in this work.

An upper bound based on Markov blanket decomposition

For MRE inference, an upper bound of a state \(x\) should be greater than the GBF score of not only \(x\) but also all descendant states of \(x\). We can prune the whole subtree if the upper bound is less than the GBF of the current best solution.

We introduce the following novel upper bound for MRE inference. We first divide the evidences into a set of exclusive and exhaustive subsets, i.e., \(E = \cup_i E_i\), by using depth-first search discussed later. These evidence subsets naturally decompose the whole network into overlapping subnetworks, each of which contains an evidence subset \(E_i\) and its Markov blanket \(MB(E_i)\). An upper bound on GBF is derived by multiplying the upper bounds on the belief update ratios calculated on the subnetworks.

We first derive an upper bound on the belief update ratio in Theorem 1.

**Theorem 1.** Let \(M = \{X_1, X_2, \ldots, X_n\}\) be a set of targets, \(e\) be the evidence, and \(MB(E_i)\) be the Markov blanket of the \(i\)th subset of evidences \(E_i\). Then, for any subset \(\emptyset \subset X \subseteq M\), the belief update ratio \(r(x; e)\) is upper bounded as follows.

\[
\max_{x \in \emptyset \subset X \subseteq M} r(x; e) \leq \left( \prod_{i} \max_{x = MB(E_i)} r(x; e_i) \right) \cdot C, \quad (5)
\]

where \(C = \prod_i p(e_i)/p(e)\).

**Proof.** From the formulation of \(r(M; e)\), we have

\[
r(M; e) = \frac{p(M; e)/p(M)}{p(e)/p(e)} = \prod_i \frac{p(e_i|MB(E_i))/p(e)}{p(e_i)/p(e)} = \prod_i \frac{p(MB(E_i)|e_i)p(e_i)}{p(MB(E_i))}/p(e).
\]

Thus, the third equality is based on the property of Markov blankets.

From Equation 7, we have the following upper bound on \(r(m; e)\),

\[
\max_{m} r(m; e) \leq \left( \prod_{i} \max_{x = MB(E_i)} r(x; e_i) \right) \cdot C. \quad (6)
\]

For any \(\emptyset \subset X \subseteq M, M = X \cup \bar{X}\), let \(S_i = MB(E_i) \cap X, T_i = MB(E_i) \cap \bar{X}, U = \{i : T_i \neq \emptyset\}, \) and \(V = \{i : T_i = \emptyset\}\), we have:

\[
p(e|X) = \sum_X p(e|M)p(\bar{X}|X) = \sum_X \left( \prod_{i \in U} p(e_i|MB(E_i)) \right) p(\bar{X}|X) = \prod_{i \in V} \max_{x = MB(E_i)} p(e_i|MB(E_i)) = \left( \prod_{i \in U} \max_{z = T_i} p(e_i|S_i \cup z) \right) p(\bar{X}|X) \cdot C. \quad (9)
\]

Since \(p(e|X) = r(X; e)p(e)\), we have the following upper bound on \(r(x; e)\):

\[
\max_{x \in \emptyset \subset X \subseteq M} r(x; e) \leq \left( \prod_{i} \max_{x = MB(E_i)} r(x; e_i) \right) \cdot C. \quad (10)
\]

Thus, for any \(\emptyset \subset X \subseteq M\), we obtain the final upper bound by combining Equations 8 and 10:

\[
\max_{x \in \emptyset \subset X \subseteq M} r(x; e) \leq \left( \prod_{i} \max_{x = MB(E_i)} r(x; e_i) \right) \cdot C. \quad (11)
\]

In MRE inference, the evidence \(e\) is given, thus \(C\) is a constant. Theorem 1 assumes that the expanded target set \(Y = \emptyset\), which is true at the beginning of search. During the search when \(Y \neq \emptyset\), we have the following corollary to derive the upper bound on the belief update ratio:

**Corollary 2.** Let \(M = \{X_1, X_2, \ldots, X_n\}\) be a set of targets, \(e\) be the evidence, \(MB(E_i)\) be the Markov blanket of the \(i\)th subset of evidences \(E_i\). Let \(X\) and \(Y\) be the unexpanded and expanded target sets. Let \(T_i = MB(E_i) \cap Y\), and \(MB(E_i) = T_i \cup \bar{T}_i\). Then, for any subset \(\emptyset \subset Z \subseteq X\), the belief update ratio \(r(z \cup y; e)\) is upper bounded as follows.

\[
\max_{z, \emptyset \subset Z \subseteq \bar{X}} r(z \cup y; e) \leq \left( \prod_{i} \max_{z = T_i} r(z \cup t_i; e_i) \right) \cdot C, \quad (11)
\]

where \(C = \prod_i p(e_i)/p(e)\).

Based on Corollary 2, we can derive the upper bound on GBF in Theorem 3:

**Theorem 3.** Let \(M = \{X_1, X_2, \ldots, X_n\}\) be a set of targets, \(e\) be the evidence, \(MB(E_i)\) is the Markov blanket of the \(i\)th subset of evidences \(E_i\). Let \(X\) and \(Y\) be the unexpanded and expanded target sets. Let \(T_i = MB(E_i) \cap Y\), and \(MB(E_i) = T_i \cup \bar{T}_i\). Then, for any subset \(\emptyset \subset Z \subseteq X\),
The above theorems are based on factoring the conditional joint distribution \( p(e|M) \) into the product of a set of conditional distributions \( \prod_i p(e_i|MB(E_i)) \). Thus partitioning the whole evidence set \( E \) into the subsets \( \cup_i E_i \) and compiling their Markov blankets is a key part of the proposed method. The Markov blanket of a single node includes its parents, children, and children’s other parents. In MRE problems, some of these nodes may be auxiliary nodes. To derive the upper bound, however, we need to find the Markov blankets containing only targets. Thus the standard definition of Markov blanket cannot be directly used in our problem.

In the proposed method, we first generate the smallest ancestral set containing the target set \( M \) and the evidence set \( E \), i.e., \( An(M \cup E) \). Then we compile a moral graph \( (G_{An(M \cup E)})^m \). Figure 2 illustrates an example moral graph with two evidence nodes. Using Lemma 1, for each evidence subset \( E_i \), we need to find a set of targets which separates \( E_i \) from other evidence nodes and targets in \((G_{An(M \cup E)})^m\). This is achieved by doing a depth first graph traversal starting from each evidence node if it has not been visited before. There are three scenarios when a node is being visited.

- **Case 1:** When an evidence node is visited, we add the evidence to the current evidence subset \( E_i \), mark it as visited, and continue to visit its unmarked neighbors.
- **Case 2:** When a target is visited, we add the target to \( MB(E_i) \) and mark it as visited.
- **Case 3:** When an auxiliary node is visited, we mark it as visited and continue to visit its unmarked neighbors.

When restarting the search on a new evidence node, we unmark all targets and auxiliary nodes, because the same targets may occur in different Markov blankets as shown in Figure 2. The algorithm stops when all evidence nodes have been visited. The depth first search algorithm automatically partitions the whole evidence set into subsets \( \cup_i E_i \) and finds the Markov blanket \( MB(E_i) \) of each subset.

### Merging Markov blankets

When computing the upper bound, we maximize each belief update ratio \( r(MB(E_i); e_i) \) according to \( MB(E_i) \), independently. Thus the common targets of two different Markov blankets \( MB(E_i) \) and \( MB(E_j) \) may be set to inconsistent values based on the maximization. Too much inconsistency may result in a loose bound. We can tighten the upper bound by merging Markov blankets that share targets. On the other hand, if the number of targets in an individual Markov blanket is too large, it will make calculating the upper bound inefficient. We propose to merge Markov blankets to reduce
the number of duplicates of a target under the constraint that the number of targets in a Markov blanket cannot exceed $K$.

We can use an undirected graph to represent the problem of merging Markov blankets. The nodes in the graph denote Markov blankets. If two Markov blankets share targets, there is an edge between the two corresponding nodes. The weight of edge is the number of targets shared by the two Markov blankets. This formulation translates the problem of merging Markov blankets into a graph partition problem. More specifically, the merging problem can be addressed by recursively solving the minimum bisection problem (Feige and Krauthgamer 2002), which partitions the vertices into two equal halves so as to minimize the sum of weights of the edges between the two partitions, on the undirected graph.

Minimum bisection problem is an NP-hard problem, however. We cannot afford to spend too much on computing the upper bound. We therefore use a hierarchical clustering-like greedy algorithm for merging the Markov blankets. We first merge any two Markov blankets if one of them covers the other. Then for all the remaining Markov blankets, we each time merge two Markov blankets that share the most number of targets as long as the number of targets in the resulting Markov blanket does not exceed $K$. The algorithm finishes when no Markov blankets can be merged.

**Breadth-first branch-and-bound algorithm**

In MRE inference, all of the search nodes are potential solutions except the root node. We can choose from a variety of search methods to explore the search tree, e.g., depth-first search, best-first search, and breadth-first search. We chose breadth-first search in MRE inference for two reasons. First, it is convenient to identify the convergence of MRE by monitoring the number of states in the search queue. Usually, the size of the search queue first increases and then decreases in the BFBnB algorithm. Second, since MRE prunes away independent and less relevant targets, usually the number of targets in the optimal solution is not large. Thus breadth-first search may reach the optimal solutions faster than other search strategies. The breadth-first search costs more memory to store the unexpanded states in a layer, however.

Before the breadth-first search, we calculate the belief update ratios of all configurations of $MB(E_i)$ in a belief table. Then the breadth-first search algorithm explores the search tree layer by layer while keeping track of the highest-scoring state. For each state $S$, we search for a configuration of each Markov blanket that is consistent with the expanded targets $t_i$ and has the highest belief update ratio. We then calculate the upper bound of $S$ using Theorem 3. We prune $S$ if the upper bound is less than the current best GBF.

**Experiments**

We tested the proposed algorithm on six benchmark diagnostic Bayesian networks listed in Table 1, i.e., Alarm (Ala), Carpo (Car), Hepar (Hep), Insurance (Ins), Emdec6h (Emd), and Tcc4g (Tcc) (Beinlich et al. 1989; Binder et al. 1997; Onisko 2003). Among them, Alarm, Carpo, Hepar, and Insurance are small networks with fewer than 100 nodes. Emdec6h and Tcc4g are larger networks with more than 100 nodes. We also listed the number of leaf nodes (Leaves), the average number of node states (States), and the number of arcs (Arcs) of the Bayesian networks in Table 1. The experiments were performed on a 2.67GHz Intel Xeon CPU E7 with 512G RAM running a 3.7.10 Linux kernel.

**Experimental design**

Since the proposed method is the first nontrivial exact MRE algorithm, we had to use a naive Breadth-First Brute-Force search algorithm (BFBF) as the baseline; basically BFBF is BFBnB with the bound set to be infinity. We also included the results of tabu search to indicate the difficulty of MRE problems. In BFBnB, we set the maximum number of targets in a Markov blanket $K$ to be 18. In tabu search, we set the number of search steps since the last improvement $L$ and the number of search steps $M$ according to different network settings. For the networks with 12 targets, we set $L$ to be 20 and $M$ to be $\{400, 800, 1600, 3200, 6400\}$. For the networks with about 20 targets, we set $L$ to be 80 and $M$ to be $\{12800, 25600, 51200\}$. To evaluate search performance, we compared the solutions of tabu search and BFBnB, and counted the number of test cases on which tabu search achieved optimal solutions.

BFBF search can only solve test cases with fewer than 15 targets. To compare to BFBF, we randomly generated five test settings of each network, each setting consisting of all leaf nodes as evidence, 12 of the remaining nodes as targets, and others as auxiliary nodes. Then for each setting, we randomly generated 20 configurations of evidence (test cases) by sampling from the prior distributions of the networks.

We also evaluated BFBnB on test cases with about 20 targets. For each network, we generated five test settings and 20 test cases for each setting. In the four smaller networks, we set all the leaf nodes as evidence, 80% of the remaining nodes as targets, and others as auxiliary nodes. In the two larger networks, we set all the leaf nodes as evidence, 20 of the remaining nodes as targets, and others as auxiliary nodes.

**Evaluation of BFBnB on networks with 12 targets**

In Table 2, we compared the proposed BFBnB algorithm with BFBF and tabu search on the test cases with 12 targets. For tabu search, we listed the accuracy (top) and running time (bottom) at different numbers of search steps. Both BFBnB and BFBF were able to solve the test cases exactly. BFBnB is shown to be significantly faster than BFBF because of the pruning by the upper bound. T400 is the fastest
Table 2: Comparison of BFBnB, BFBF, and tabu on running time and accuracy on Bayesian networks with 12 targets.

<table>
<thead>
<tr>
<th></th>
<th>Ala</th>
<th>Car</th>
<th>Hep</th>
<th>Ins</th>
<th>Emd</th>
<th>Tcc</th>
</tr>
</thead>
<tbody>
<tr>
<td>BFBF</td>
<td>1.7e3</td>
<td>66.1</td>
<td>270</td>
<td>1.6e5</td>
<td>212</td>
<td>162</td>
</tr>
<tr>
<td>BFBnB</td>
<td>17.0</td>
<td>3.5</td>
<td>14.6</td>
<td>1.6e3</td>
<td>15.1</td>
<td>5.6</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>T6400</th>
<th>T3200</th>
<th>T1600</th>
<th>T800</th>
<th>T400</th>
</tr>
</thead>
<tbody>
<tr>
<td>sec</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BFBF</td>
<td>92</td>
<td>85</td>
<td>79</td>
<td>76</td>
<td>72</td>
</tr>
<tr>
<td>BFBnB</td>
<td>17.3</td>
<td>13.2</td>
<td>4.7</td>
<td>6.9</td>
<td>1.8</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>T51200</th>
<th>T25600</th>
<th>T12800</th>
</tr>
</thead>
<tbody>
<tr>
<td>min</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BFBnB</td>
<td>173</td>
<td>60</td>
<td>60</td>
</tr>
<tr>
<td>BFBF</td>
<td>0.98</td>
<td>5.55</td>
<td>1.50</td>
</tr>
</tbody>
</table>

Figure 3: Distributions of logarithm running time pairs of BFBnB and BFBF on Bayesian networks with 12 targets.

Evaluation of BFBnB on networks with about 20 targets

We also evaluated BFBnB on the test cases with about 20 targets. We listed the number targets of each network in the first row of Table 3. It is not surprising that BFBnB took longer to finish comparing with the results in Table 2, but the pruning of the upper bound slowed down the exponential growth of the running time significantly. Also from the results, we can see that the running time of BFBnB depends on not only the number of targets, but also the tightness of upper bound and the Bayesian network structures, which control the number of pruned states and the size of belief table in each Markov blanket $MB(E_i)$, respectively. Increasing $M$ from 12800 to 51200 is not helpful in preventing tabu search from getting stuck in the local optima. Moreover, the performance of tabu search varies greatly on different test networks.

Discussions and Conclusions

The main contribution of this paper is a BFBnB algorithm for solving MRE exactly using an upper bound based on Markov blanket decomposition. This is the first exact algorithm for solving MRE in Bayesian networks. The proposed upper bound can be calculated efficiently for two reasons. First, each Markov blanket $MB(E_i)$ is usually much smaller than the whole target set $M$. Second, in the original MRE problem, we need to search all the subsets of target set $M$ to find the best solution. However, to calculate upper bound, we only need to search on a fixed target set $MB(E_i)$ of each subnetwork. The search space has been dramatically reduced. The experimental results show that BFBnB is significantly faster than the brute-force algorithm. BFBnB can solve MRE inference exactly in Bayesian networks which could not be solved previously.

Different from the brute-force algorithm, the search time of BFBnB is no longer mainly dependent on the size of search space (i.e., the number of targets and the number of states of each target), but also on the tightness of the upper bound and the structure of Bayesian networks. For Bayesian networks with a large number of targets, an upper bound can be efficiently generated as long as the the number of targets in each Markov blanket is small. For the Markov blanket with a large number of targets, we can decompose it into subnetworks using the methods such as node splitting (Choi, Chavira, and Darwiche 2007). This is also one of our future research directions.

Acknowledgements This work was supported by NSF grants IIS-0953723, IIS-1219114, and a PSC-CUNY enhancement award.
References


