Finding Minimal Separators in LWF Chain Graphs

Mohammad Ali Javidian JAVIDIAN@EMAIL.SC.EDU
Marco Valtorta MGV@CSE.SC.EDU
Department of Computer Science & Engineering, University of South Carolina, Columbia, SC, 29201, USA.

Abstract

We address the problem of finding a minimal separator in a LWF chain graph, namely, finding a set Z of nodes that separates a given non-adjacent pair of nodes such that no proper subset of Z separates that pair. We analyze several versions of this problem and offer polynomial time algorithms for each. These include finding a minimal separator from a restricted set of nodes, finding a minimal separator for two given disjoint sets, and testing whether a given separator is minimal.

Keywords: chain graph; Markov property; conditional independence; minimal separator.

1. Introduction

Probabilistic graphical Models (PGM) use graphs, either undirected, directed, or mixed, to represent possible dependences among the variables of a multivariate probability distribution. Two types of graphical representations of distributions are commonly used, namely, Bayesian networks and Markov random fields (Markov networks). Both families encompass the properties of factorization and independences, but they differ in the set of independences they can encode and the factorization of the distribution that they induce.

Chain graphs, which admit both directed and undirected edges, are a type of graphical model in which there are no partially directed cycles. LWF Chain graphs were introduced by Lauritzen, Wermuth and Frydenberg (Frydenberg, 1990; Lauritzen and Wermuth, 1989) as a generalization of graphical models based on undirected graphs and directed acyclic graphs (DAGs) and widely studied e.g. in (Lauritzen, 1996; Lauritzen and Richardson, 2002; Cowell et al., 1999; Drton, 2009; Ma et al., 2008; Peña, 2015; Peña et al., 2014; Richardson, 1998; Sonntag, 2014; Studený, 1997).

In this paper we propose and solve an optimization problem related to separation in LWF chain graphs. The basic problem may be formulated as follows: given a pair of non-adjacent nodes, x and y, in a LWF chain graph, G, find a minimal set of nodes that separates x and y. We analyze several versions of this problem and offer polynomial time algorithms for each. These include the following problems:

Problem 1 (test for minimal separation) Given two non-adjacent nodes X and Y in a LWF chain graph G and a set Z that separates X from Y , test if Z is minimal i.e., no proper subset of Z separates X from Y.

Problem 2 (minimal separation) Given two non-adjacent nodes X and Y in a LWF chain graph G, find a minimal separating set between X and Y, namely, find a set Z such that Z, and no proper subset of Z, separates X from Y.

Problem 3 (restricted separation) Given two non-adjacent nodes X and Y in a LWF chain graph G and a set S of nodes not containing X and Y, find a subset Z of S that separates X from Y.

Problem 4 (restricted minimal separation) Given two non-adjacent nodes X and Y in a LWF chain graph G and a set S of nodes not containing X and Y, find a subset Z of S which is minimal and separates X from Y.
Problem 5 (minimal separation of two disjoint non-adjacent sets) Given two disjoint non-adjacent sets X and Y in a LWF chain graph G, find a minimal separating set between X and Y, namely, find a set Z such that Z, and no proper subset of Z, separates X from Y.
Problem 6 (enumeration of all minimal separators) Given two non-adjacent nodes (or disjoint subsets) X and Y in a LWF chain graph G, enumerate all minimal separating sets between X and Y.

The rest of the paper is organized as follows: in section 2, we briefly describe several concepts which are basic for subsequent development, such as the separation criterion. In section 3, we prove that it is possible to transform our problem into a separation problem, where the undirected graph in which we have to look for the minimal set separating X from Y depends only on X and Y. In Section 4, we propose and analyze an algorithm for each above mentioned problem that, taking into account the previous results, solves it.

2. Definitions and Concepts

In this Section, we describe the notation and some basic concepts used throughout the paper.

Definition 1 *If $A \subseteq V$ is a subset of the vertex set in a graph $G = (V, E)$, it induces a subgraph $G_A = (A, E_A)$, where the edge set $E_A = E \cap (A \times A)$ is obtained from G by keeping edges with both endpoints in A.*

Definition 2 *If there is an arrow from a pointing towards b, a is said to be a parent of b. The set of parents of b is denoted as $pa(b)$. If there is an undirected edge between a and b, a and b are said to be adjacent or neighbors. The set of neighbors of a vertex a is denoted as $ne(a)$. The expressions $pa(A)$ and $ne(A)$ denote the collection of parents and neighbors of vertices in A that are not themselves elements of A. The boundary $bd(A)$ of a subset A of vertices is the set of vertices in $V \setminus A$ that are parents or neighbors to vertices in A. The closure of A is $cl(A) = bd(A) \cup A$.*

Definition 3 *A path of length n from a to b is a sequence $a = a_0, \ldots, a_n = b$ of distinct vertices such that $(a_i, a_{i+1}) \in E$, for all $i = 1, \ldots, n$. If there is a path from a to b we say that a leads to b and write $a \mapsto b$. The vertices a such that $a \mapsto b$ and $b \not\mapsto a$ are the ancestors $an(b)$ of b. If $bd(a) \subseteq A$, for all $a \in A$ we say that A is an ancestral set. The smallest ancestral set containing A is denoted by $An(A)$. A chain of length n from a to b is a sequence $a = a_0, \ldots, a_n = b$ of distinct vertices such that $(a_i, a_{i+1}) \in E$, or $(a_{i+1}, a_i) \in E$, or $\{a_i, a_{i+1}\} \in E$, for all $i = 1, \ldots, n$.*

Definition 4 *Given an undirected graph G. Two vertices are said to be adjacent if they are connected by an edge. A subset $S \subseteq V$ that does not contain a or b is said to be an (a, b)-separator if all paths from a to b intersect S. A set S of nodes that separates a given pair of nodes such that no proper subset of S separates that pair is called a minimal separator.*

Note that removing an (a, b)-separator disconnects a graph into two connected components, one containing a, and another containing b. Conversely, if a set S disconnects a graph into a connected component including a and another connected component including b, then S is an (a, b)-separator.

Definition 5 *Two disjoint vertex subsets A and B of V are adjacent if there is at least one pair of adjacent vertices $u \in A$ and $v \in B$. Let A and B be two disjoint non-adjacent subsets of V. Similarly, we define an (A, B)-separator to be any subset of $V \setminus (A \cup B)$ whose removal separates A and B in distinct connected components. A minimal (A, B)-separator does not contain any other (A, B)-separator.*

Definition 6 (Global Markov property for LWF chain graphs) *For any triple (A, B, S) of disjoint subsets of V such that S separates A from B in $(G_{An(A\cup B\cup S)})^m$, in the moral graph (Lauritzen, 1996, page 7) of the smallest ancestral set containing $A \cup B \cup S$, we have $A \perp\!\!\!\perp B|S$ (or $\langle A, B|S\rangle$) i.e., A is independent of B given S.*

3. Main Theorem

In this section we prove that it is possible to transform our problem into a separation problem, where the undirected graph in which we have to look for the minimal set separating X from Y depends only on X and Y. Later, in the next sections, we shall apply this result to developing an efficient algorithm that solves our problems.

The next proposition shows that if we want to test a separation relationship between two disjoint sets of nodes X and Y in a LWF chain graph, where the separating set is included in the smallest ancestral set of $X \cup Y$, then we can test this relationship in a smaller chain graph, whose set of nodes is formed only by the ancestors of X and Y.

Proposition 7 *Given a LWF chain graph $G = (V, E)$. Consider that X, Y, and Z are three disjoint subsets of V, and $Z \subseteq An(X \cup Y)$. Let $H = G_{An(X\cup Y)}$ be the subgraph of G induced by $An(X \cup Y)$. Then $\langle X, Y|Z\rangle_G \Leftrightarrow \langle X, Y|Z\rangle_H$.*

Proof ($\Rightarrow$) The necessary condition is obvious, because a separator in a graph is also a separator in all of its subgraphs.

($\Leftarrow$) Let $\langle X, Y|Z\rangle_H$ and $Z \subseteq An(X \cup Y)$, then $An(X \cup Y \cup Z) = An(X \cup Y)$. Consider that $\langle X, Y \not| Z\rangle_G$. This means that X is not separated from Y given Z in $(G_{An(X\cup Y\cup Z)})^m \equiv (G_{An(X\cup Y)})^m$. In other words, there is a chain C between X and Y in $H^m = (G_{An(X\cup Y)})^m$ that bypasses Z. Once again using $Z \subseteq An(X \cup Y)$, we obtain that X and Y are not separated by Z in H, in contradiction to the assumption $\langle X, Y|Z\rangle_H$. Therefore, it has to be $\langle X, Y|Z\rangle_G$. ■

The following proposition establishes the basic result necessary to solve our optimization problems.

Proposition 8 *Given a LWF chain graph $G = (V, E)$. Consider that X, Y, and Z are three disjoint subsets of V such that $\langle X, Y|Z\rangle$ and $\langle X, Y \not| Z'\rangle, \forall Z' \subsetneq Z$. Then $Z \subseteq An(X \cup Y)$.*

Proof Suppose that $Z \not\subseteq An(X \cup Y)$. Define $Z' = Z \cap An(X \cup Y)$. Then, by assumption we have $\langle X, Y \not| Z'\rangle$. Since $Z' \subseteq An(X \cup Y)$, it is obvious that $An(X \cup Y \cup Z') = An(X \cup Y)$. So, X and Y are not separated by Z' in $(G_{An(X\cup Y)})^m$, hence there is a chain C between X and Y in $(G_{An(X\cup Y)})^m$ that bypasses Z' i.e., the chain C is formed from nodes in $An(X\cup Y)$ that are outside of Z. Since $An(X \cup Y) \subseteq An(X \cup Y \cup Z)$, then $(G_{An(X\cup Y)})^m$ is a subgraph of $(G_{An(X\cup Y\cup Z)})^m$. Then, the previously found chain C is also a chain in $(G_{An(X\cup Y\cup Z)})^m$ that bypasses Z, which means that X and Y are not separated by Z in $(G_{An(X\cup Y\cup Z)})^m$, in contradiction to the assumption $\langle X, Y|Z\rangle$. Therefore, it has to be $Z \subseteq An(X \cup Y)$. ■

The next proposition shows that, by combining the results in propositions 7 and 8, we can reduce our problems to a simpler one, which involves a smaller graph.

Proposition 9 *Let $G = (V, E)$ be a LWF chain graph, and $X, Y \subseteq V$ are two disjoint subsets. Then the problem of finding a minimal separating set for X and Y in G is equivalent to the problem of finding a minimal separating set for$X$ and Y in the induced subgraph $G_{An(X\cup Y)}$.*

Proof The proof is very similar to the proof of Proposition 3 in (Acid and Campos, 1996). Let $H = G_{An(X\cup Y)}$, and let us to define sets $S_G = \{Z \subseteq V | \langle X, Y|Z\rangle_G\}$ and $S_H = \{Z \subseteq An(X \cup Y)|\langle X, Y|Z\rangle_H\}$. Then we have to prove that $\min_{Z\in S_G} |Z| = \min_{Z\in S_H} |Z|$, and therefore, by proposition 8, the sets of minimal separators are the same. From proposition 7, we deduce that $S_H \subseteq S_G$, and therefore $\min_{Z\in S_H} |Z| \geq \min_{Z\in S_G} |Z|$.
($\Rightarrow$) Let $T = \min(Z \in S_G)$. Then $\forall T' \subsetneq T$ we have $T' \notin S_G$, and from proposition 8 we obtain $T \subseteq An(X\cup Y)$, and now using proposition 7 we get $T \in S_H$. So, we have $|T| = \min_{Z\in S_H} |Z| \geq \min_{Z\in S_G} |Z| = |T|$, hence $|T| = \min_{Z\in S_H} |Z|$.
($\Leftarrow$) Let $T = \min(Z \in S_H)$. If, $|T| = \min_{Z\in S_H} |Z| > \min_{Z\in S_G} |Z| = |Z_0|$, we have $\forall Z' \subsetneq Z_0, Z' \notin S_G$, and therefore, once again using proposition 8 and 7, we get $Z_0 \in S_H$, so that $|Z_0| \geq \min_{Z\in S_H} |Z| = |T|$, which is a contradiction. Thus, $|T| = \min_{Z\in S_G} |Z|$. ■

Theorem 10 *The problem of finding a minimal separating set for X and Y in a LWF chain graph G is equivalent to the problem of finding a minimal separating set for$X$ and Y in the undirected graph $(G_{An(X\cup Y)})^m$.*

Proof The proof is very similar to the proof of Theorem 1 in (Acid and Campos, 1996). Using the same notation from proposition 9, let H^m be the moral graph of $H = G_{An(X\cup Y)}$, and $S_H^m = \{Z \subseteq An(X \cup Y)|\langle X, Y|Z\rangle_{H^m}\}$. Let Z be any subset of $An(X \cup Y)$. Then taking into account the characteristics of ancestral sets, it is clear that $H_{An(X\cup Y\cup Z)} = H$. Then, we have

$$Z \in S_H \Leftrightarrow \langle X, Y|Z\rangle_H \Leftrightarrow \langle X, Y|Z\rangle_{(H_{An(X\cup Y\cup Z)})^m} \Leftrightarrow \langle X, Y|Z\rangle_{H^m} \Leftrightarrow Z \in S_H^m.$$

Hence, $S_H = S_H^m$. Now, using proposition 9, we obtain $|T| = \min_{Z\in S_G} |Z| \Leftrightarrow |T| = \min_{Z\in S_H^m} |Z|$.
■

4. Algorithms for Finding Minimal Separators

In undirected graphs we have efficient methods of testing whether a separation set is minimal, which are based on the following criterion.

Theorem 11 *Given two nodes X and Y in an undirected graph, a separating set Z between X and Y is minimal if and only if for each node u in Z, there is a path from X to Y which passes through u and does not pass through any other nodes in Z.*

Proof See the proof of Theorem 5 in (Tian et al., 1998). ■

This theorem leads to the following algorithm for Problem 1. The idea is that if Z is minimal then all nodes in Z can be reached using Breadth First Search (BFS) that starts from both X and Y without passing any other nodes in Z.

Algorithm 1 (test for minimal separation) (Tian et al., 1998, Algorithm 2)

1. If Z contains any node which is not in $An(X \cup Y)$, then Z is not minimal, stop.
2. Construct $G_{An(X \cup Y)}$.
3. Construct $(G_{An(X \cup Y)})^m$.
4. Starting from X, run BFS. Whenever a node in Z is met, mark it if it is not already marked, and do not continue along that path. When BFS stops, if not all nodes in Z are marked, Z is not minimal, stop. Remove all markings.
5. Starting from Y, run BFS. Whenever a node in Z is met, mark it if it is not already marked, and do not continue along that path. When BFS stops, if not all nodes in Z are marked, Z is not minimal. If all nodes in Z are marked, Z is minimal.

Analysis (Tian et al., 1998): Let $|E^m_{An}|$ stands for the number of edges in $(G_{An(X \cup Y)})^m$. Step 3-5 each requires $O(|E^m_{An}|)$ time. Thus, the complexity of Algorithm 1 is $O(|E^m_{An}|)$.

A variant of Algorithm 1 solves the Problem 2.

Algorithm 2 (minimal separation)

1. Construct $G_{An(X \cup Y)}$.
2. Construct $(G_{An(X \cup Y)})^m$.
3. Set Z' to be $ne(X)$ (or $ne(Y)$) in $(G_{An(X \cup Y)})^m$. (Z' is a separator because, according to the local Markov property of an undirected graph, a vertex is conditionally independent of all other vertices in the graph, given its neighbors (Lauritzen, 1996)).
4. Starting from X, run BFS. Whenever a node in Z' is met, mark it if it is not already marked, and do not continue along that path. When BFS stops, let Z'' be the set of nodes which are marked. Remove all markings.
5. Starting from Y, run BFS. Whenever a node in Z'' is met, mark it if it is not already marked, and do not continue along that path. When BFS stops, let Z be the set of nodes which are marked.
6. Return (Z).

Analysis: Step 2-5 each requires $O(|E^m_{An}|)$ time. Thus, the overall complexity of Algorithm 2 is $O(|E^m_{An}|)$.

Theorem 12 *Given two nodes X and Y in a LWF chain graph G and a set S of nodes not containing X and Y, there exists some subset of S which separates X and Y if only if the set $S' = S \cap An(X \cup Y)$ separates X and Y.*

Proof ($\Rightarrow$) Proof by contradiction. Let $S' = S \cap An(X \cup Y)$ and $\langle X, Y \not| S' \rangle$. Since $S' \subseteq An(X \cup Y)$, it is obvious that $An(X \cup Y \cup S') = An(X \cup Y)$. So, X and Y are not separated by S' in $(G_{An(X \cup Y)})^m$, hence there is a chain C between X and Y in $(G_{An(X \cup Y)})^m$ that bypasses S' i.e., the chain C is formed from nodes in $An(X \cup Y)$ that are outside of S. Since $An(X \cup Y) \subseteq An(X \cup Y \cup S''), \forall S'' \subseteq S$, then $(G_{An(X \cup Y)})^m$ is a subgraph of $(G_{An(X \cup Y \cup S)})^m$. Then, the previously found chain C is also a chain in $(G_{An(X \cup Y \cup S'')})^m$ that bypasses S'', which means that X and Y are not separated by any $S'' \subseteq S$ in $(G_{An(X \cup Y \cup S'')})^m$, which is a contradiction.
($\Leftarrow$) It is obvious. ■

Therefore, Problem 3 is solved by testing if $S' = S \cap An(X \cup Y)$ separates X and Y.

Algorithm 3 (restricted separation)

1. Construct $G_{An(X \cup Y)}$.
2. Construct $(G_{An(X \cup Y)})^m$.
3. Set $S' = S \cap An(X \cup Y)$.
4. Remove S' from $(G_{An(X \cup Y)})^m$.
5. Starting from X, run BFS. When Y is met, do not continue and Return False. Otherwise, when BFS stops, Return S'.

Analysis: This requires $O(|E^m_{An}|)$ time.

According to theorem 12, Problem 4 is solved using Algorithm 3 and then, if False not returned, Algorithm 2 with $Z' = S \cap An(X \cup Y)$. The time complexity of this algorithm is also $O(|E^m_{An}|)$.

In order to solve Problem 5, i.e., to find the minimal set separating two disjoint non-adjacent subsets of nodes X and Y (instead of two single nodes) in a LWF chain graph G, first we build the undirected graph $(G_{An(X \cup Y)})^m$. Next, starting out from this graph, we construct a new undirected graph $Aug[G : \alpha_X, \alpha_Y]$ by adding two artificial (dummy) nodes α_X, α_Y, and connect them to those nodes that are adjacent to some node in X and Y, respectively. So, the separation of X and Y in $(G_{An(X \cup Y)})^m$ is equivalent to the separation of α_X and α_Y in $Aug[G : \alpha_X, \alpha_Y]$. Moreover, the minimal separating set for α_X and α_Y in $Aug[G : \alpha_X, \alpha_Y]$ cannot contain nodes from $(X \cup Y)$. Therefore, in order to find the minimal separating set for X and Y in G, it is suffice to find the minimal separating set for α_X and α_Y in $Aug[G : \alpha_X, \alpha_Y]$. So, we have reduced this problem to one of separation for single nodes, which can be solved using the Algorithm 2.

Shen and Liang in (Shen and Liang, 1997) presents an efficient algorithm for enumerating all minimal (X, Y)-separators, separating given non-adjacent vertices X and Y in an undirected connected simple graph $G = (V, E)$. This algorithm requires $O(n^3 R_{XY})$ time, where $|V| = n$ and R_{XY} is the number of minimal (X, Y)-separators. The algorithm can be generalized for enumerating all minimal (X, Y)-separators that separate non-adjacent vertex sets $X, Y \subseteq V$, and it requires $O(n^2(n - n_X - n_Y)R_{XY})$ time. In this case, $|X| = n_X$, $|Y| = n_Y$, and R_{XY} is the number of all minimal (X, Y)-separators. According to theorem 10, using this algorithm for $(G_{An(X \cup Y)})^m$ solves Problem 6.

Remark 13 *Since DAGs (directed acyclic graphs) are subclass of chain graphs, one can use the same technique to enumerate all minimal separators in DAGs.*

Conclusion and Summary

We have studied and solved the problem of finding minimal separating sets for pairs of variables in LWF chain graphs. We have also studied some extensions of the basic problem include finding a minimal separator from a restricted set of nodes, finding a minimal separator for two given disjoint sets, testing whether a given separator is minimal, and listing all minimal separators, given two non-adjacent nodes (or disjoint subsets) X and Y in a LWF chain graph G. Potential applications of this research include learning chain graphs from data and problems related to the selection of the variables to be instantiated when using chain graphs for inference tasks.

Acknowledgements

This work has been partially supported by Office of Naval Research grant ONR N00014-17-1-2842. This research is based upon work supported in part by the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), award/contract number 2017-16112300009. The views and conclusions contained therein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes, notwithstanding annotation therein. We thank the three anonymous reviewers for their useful comments.

References

S. Acid and L. M. D. Campos. An algorithm for finding minimum d-separating sets in belief networks. *Proceedings of the Twelfth international conference on Uncertainty in artificial intelligence*, pages 3–10, 1996.

R. Cowell, A. P. Dawid, S. Lauritzen, and D. J. Spiegelhalter. *Probabilistic networks and expert systems. Statistics for Engineering and Information Science*. Springer-Verlag, 1999.

M. Drton. Discrete chain graph models. *Bernoulli*, 15(3):736–753, 2009.

M. Frydenberg. The chain graph Markov property. *Scandinavian Journal of Statistics*, 17(4):333–353, 1990.

S. Lauritzen. *Graphical Models*. Oxford Science Publications, 1996.

S. Lauritzen and T. Richardson. Chain graph models and their causal interpretations. *Journal of the Royal Statistical Society. Series B, Statistical Methodology*, 64(3):321–348, 2002.

S. Lauritzen and N. Wermuth. Graphical models for associations between variables, some of which are qualitative and some quantitative. *The Annals of Statistics*, 17(1):31–57, 1989.

Z. Ma, X. Xie, and Z. Geng. Structural learning of chain graphs via decomposition. *Journal of Machine Learning Research*, 9:2847–2880, 2008.

J. M. Peña. Every LWF and AMP chain graph originates from a set of causal models. *Symbolic and quantitative approaches to reasoning with uncertainty, Lecture Notes in Comput. Sci., 9161, Lecture Notes in Artificial Intelligence, Springer, Cham*, pages 325–334, 2015.

J. M. Peña, D. Sonntag, and J. Nielsen. An inclusion optimal algorithm for chain graph structure learning. *In Proceedings of the 17th International Conference on Artificial Intelligence and Statistics*, pages 778–786, 2014.

T. S. Richardson. Chain graphs and symmetric associations. *In: Jordan M.I. (eds) Learning in Graphical Models. NATO ASI Series (Series D: Behavioural and Social Sciences), vol 89*, pages 229–259, 1998.

H. Shen and W. Liang. Efficient enumeration of all minimal separators in a graph. *Theoretical Computer Science*, 180:169–180, 1997.

D. Sonntag. *A Study of Chain Graph Interpretations (Licentiate dissertation) [https://doi.org/10.3384/lic.diva-105024]*. Linköping University, 2014.

M. Studený. A recovery algorithm for chain graphs. *International Journal of Approximate Reasoning*, 17:265–293, 1997.

J. Tian, A. Paz, and J. Pearl. *Finding minimal d-separators*. Technical report, R-254, 1998.