Lifted Aggregation in Directed First-order Probabilistic Models

Jacek Kisyński KISYNSKI@CS.UBC.CA
David Poole POOLE@CS.UBC.CA
Department of Computer Science, University of British Columbia, 2366 Main Mall, Vancouver, BC V6T 1Z4 Canada

Keywords: first-order probabilistic models, lifted inference, aggregation

Abstract

There is an ongoing effort to design efficient lifted inference algorithms for first-order probabilistic models. This paper focuses on directed first-order models that require an aggregation operator when a parent random variable is parameterized by logical variables that are not present in a child random variable. We describe our work on extending Milch et al.'s C-FOVE lifted inference algorithm with efficient lifted aggregation procedures.

1. Introduction

Representations that mix graphical models and first-order logic—called either first-order or relational probabilistic models—are becoming increasingly popular (Getoor & Taskar, 2007; De Raedt et al., 2008). In these models, random variables are parameterized by individuals belonging to a population. Even for very simple first-order models, inference at the propositional level—that is, inference that explicitly considers every individual—is intractable. The idea behind *lifted inference* is to carry out as much inference as possible without propositionalizing. An exact lifted inference procedure for first-order probabilistic directed models was originally proposed by Poole (2003). It was later extended to a broader range of problems by de Salvo Braz et al. (2007). Further work by Milch et al. (2008) expanded the scope of lifted inference and resulted in the C-FOVE algorithm, which is currently the state of the art in exact lifted inference.

While early work on lifted probabilistic inference by Poole (2003) considered directed models, the later work by de Salvo Braz et al. (2007) and Milch et al. (2008) focused on undirected models. Although their results can also be used for directed models, one aspect that arises exclusively in directed models is the need for aggregation which occurs when a parent random variable is parameterized by logical variables that are not present in a child random variable. In this paper we give an overview of our work (Kisyński & Poole, 2009) on incorporating efficient lifted aggregation procedures into the C-FOVE algorithm.

Presented at *ILP-MLG-SRL*, Leuven, Belgium, 2009.

2. Aggregation in first-order probabilistic models

A *population* is a set of *individuals*. A population corresponds to a domain in logic. A *parameter* corresponds to a logical variable and is typed with a population. A *substitution* is of the form $\{X_1/t_1 \ldots, X_k/t_k\}$, where the X_i are distinct parameters, and each *term* t_i is a parameter typed with a population or a constant denoting an individual from a population. A *ground substitution* is a substitution, where each t_i is a constant. A *parameterized random variable* is of the form $f(t_1, \ldots, t_k)$, where f is a functor (either a function symbol or a predicate symbol) and t_i are terms. Each functor has a set of values called the *range* of the functor. A parameterized random variable $f(t_1, \ldots, t_k)$ represents a set of random variables, one for each possible ground substitution to all of its parameters. The range of the functor of the parameterized random variable is the domain of random variables represented by the parameterized random variable. The idea of parameterized belief networks is similar to the notion of plates (Buntine, 1994); we use plates notation in our figures.

In a directed first-order probabilistic model, when a child parameterized random variable has a parent parameterized random variable with extra parameters, in the corresponding propositional model the child variable has an unbounded number of parents (see Figure 1). We need some aggregation operator to describe how the child variable depends on the parent variable.

We want to satisfy two conditions: (1) the length of a representation of the aggregation in the model should be inde-

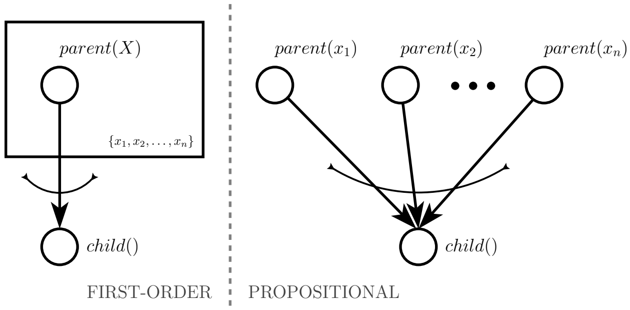

Figure 1. Directed first-order probabilistic model and its equivalent belief network.

pendent of the sizes of involved populations; (2) the time complexity of aggregation during lifted inference should be at most linear, and preferably logarithmic, in the size of the population of the extra parameter in the parent variable.

The first condition assures that one will be able to fully specify a model, that is, its structure and the accompanying probability distributions, before knowing the individuals in the modeled domain. This means in particular that, even though we might not know the sizes of the populations, we still should be able to specify the model.

Following (Zhang & Poole, 1996), we assume that the range of the parent variable is a subset of the range of the child variable, and choose to use a commutative and associative deterministic binary operator over the range of the child variable as an aggregation operator $\otimes$. Given probabilistic input to the parent variable, we can construct any causal independence model covered by the definition of causal independence from (Zhang & Poole, 1996), which in turn covers common causal independence models such as noisy-OR (Pearl, 1986) and noisy-MAX (Díez, 1993) as special cases. In other words, this allows any causal independence model to act as underlying mechanism for aggregation in directed first-order probabilistic models.

3. Aggregation during lifted inference

The C-FOVE algorithm operates on data structures called *parfactors* (Poole, 2003). Parfactors are used to store sets of initial conditional probability distributions as well as intermediate results of the inference. In (Kisyński & Poole, 2009) we added data structures called *aggregation parfactors* together with a set of inference procedures operating on aggregation parfactors to C-FOVE. Aggregation parfactors use deterministic binary operators (see Section 2) to represent intentionally aggregation relationships between parameterized random variables. The size of the aggregation parfactors is independent of the populations sizes. They can be used during the modeling phase and then, during inference with C-FOVE algorithm, once populations are known, they can be translated to parfactors on *counting formulas* (Milch et al., 2008). Such a solution allows us to take advantage of the modeling properties of aggregation parfactors and C-FOVE inference capabilities. A lifted version of factorization of Díez and Galán (2003) can translate aggregation parfactors based on MIN or MAX operators to parfactors that do not contain counting formulas.

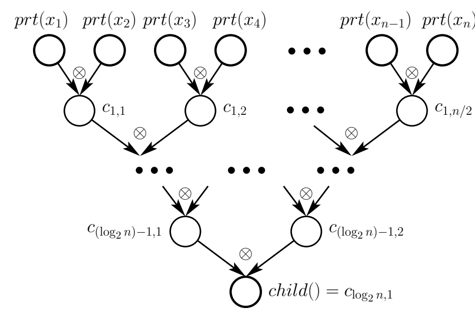

Figure 2. Decomposition of the aggregation.

In many cases, inference can also be preformed directly with the use of aggregation parfactors. This reduces the polynomial time complexity (in the size of the population of the extra parameter) of previously available lifted inference methods to logarithmic complexity. In particular, efficiency gains are achieved during elimination of parent parameterized random variable. Aggregation can be decomposed into a binary tree of applications of the aggregation operator. Figure 2 illustrates this for a case where n, the size of the population of the extra parameter, is a power of two. The results at each level of the tree are identical. When a parent parameterized random variable represents a set of random variables that can be treated as independent (we may need to delay conditioning on observations and elimination of some variables to keep them independent), we can eliminate parent parameterized random variable using a *square-and-multiply* method (Piṅgala, 200BC) whose time complexity is logarithmic in the population size of the extra parameter.

4. Empirical test

In the following experiment, we investigated how the population size of parameters affects inference in the presence of aggregation. We compared the performance of variable elimination (VE) (Zhang & Poole, 1994), variable elimination with noisy-OR factorization (Díez & Galán, 2003) (VE-FCT), C-FOVE, C-FOVE with lifted factorization of Díez and Galán (2003) (C-FOVE-FCT), and C-FOVE with aggregation parfactors (AC-FOVE). We used Java implementations of the above algorithms on an Intel Core 2 Duo 2.66GHz processor with 1024MB of memory made available to the JVM.

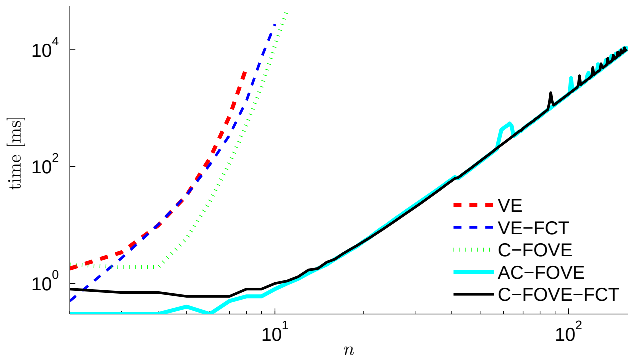

Figure 3. Performance on the smoking-friendship model.

For our test we used a directed first-order model from (Carbonetto et al., 2009) that explains how people alter their smoking habits within their social network. Given the population size n, the equivalent propositional graphical model has $3n^2 + n$ nodes and $12n^2 - 9n$ arcs. We varied n from 2 to 200 and for each value, we computed a marginal probability of a single individual being a smoker. Figure 3 shows the results of the experiment. VE, VE-FCT and C-FOVE algorithms failed to solve instances with the population size greater than 8, 10, and 11, respectively. The C-FOVE-FCT and AC-FOVE algorithms performed the best and were able to handle efficiently much larger instances (up to the population size equal to 158). Both algorithms at one point could use counting elimination (de Salvo Braz et al., 2007), while C-FOVE was forced to propositionalize. It is important to remember that C-FOVE-FTC can only be applied to MAX and MIN-based aggregation.

5. Related work

All directed first-order probabilistic formalisms include some, more or less sophisticated, forms of aggregation. Jaeger (2002) provides in-depth analysis of issues related to aggregation. Our inference algorithm can handle a subset of aggregation operators presented in his work.

Previous work on lifted probabilistic inference didn't address aggregation explicitly; however *counting formulas* of Milch et al. (2008) can be used to represent aggregation. The length of the representation of counting formulas depends on the population size, therefore they are not very well suited for specifying aggregation in relational probabilistic models. Inference performed during empirical tests described in Section 4 involved counting formulas, but not in the context of aggregation.

6. Future work

While presented the algorithm can handle a wide range of aggregation cases, there still exist models that can't be handled efficiently by lifted inference, for example a "lattice" structure shown in (Poole, 2008). These models pose an interesting challenge for future research.

Acknowledgments

The authors wish to thank Peter Carbonetto, Mark Crowley and Michael Chiang for valuable comments. This work was supported by NSERC grant to David Poole

References

Buntine, W. L. (1994). Operations for learning with graphical models. *J Artif Intell Res*, *2*, 159–225.

Carbonetto, P., Kisyński, J., Chiang, M., & Poole, D. (2009). *Learning a contingently acyclic, probabilistic relational model of a social network* (Technical Report TR-2009-08). University of British Columbia, Department of Computer Science.

De Raedt, L., Frasconi, P., Kersting, K., & Muggleton, S. H. (Eds.). (2008). *Probabilistic inductive logic programming*. Springer.

de Salvo Braz, R., Amir, E., & Roth, D. (2007). Lifted first-order probabilistic inference. In L. Getoor and B. Taskar (Eds.), *Introduction to statistical relational learning*, 433–450. MIT Press.

Díez, F. J. (1993). Parameter adjustment in Bayes networks. The generalized noisy OR-gate. *Proc. 9th UAI* (pp. 99–105).

Díez, F. J., & Galán, S. F. (2003). Efficient computation for the noisy MAX. *Int J Intell Syst*, *18*, 165–177.

Getoor, L., & Taskar, B. (Eds.). (2007). *Introduction to statistical relational learning*, 433–450 . MIT Press.

Jaeger, M. (2002). Relational Bayesian networks: a survey. *Electronic Articles in Computer and Information Science*, *6*.

Kisyński, J., & Poole, D. (2009). Lifted aggregation in directed first-order probabilistic models. *Proc. 21st IJCAI*. To appear.

Milch, B., Zettlemoyer, L. S., Kersting, K., Haimes, M., & Kaelbling, L. P. (2008). Lifted probabilistic inference with counting formulas. *Proc. 23rd AAAI* (pp. 1062–1068).

Pearl, J. (1986). Fusion, propagation and structuring in belief networks. *Artif Intell*, *29*, 241–288.

Piṅgala (200B.C.). *Chandah-sûtra*.

Poole, D. (2003). First-order probabilistic inference. *Proce. 18th IJCAI* (pp. 985–991).

Poole, D. (2008). The Independent Choice Logic and beyond. In L. De Raedt, P. Frasconi, K. Kersting and S. H. Muggleton (Eds.), *Probabilistic inductive logic programming*, 222–243. Springer.

Zhang, N. L., & Poole, D. (1994). A simple approach to Bayesian network computations. *Proc. 10th AI* (pp. 171–178).

Zhang, N. L., & Poole, D. (1996). Exploiting causal independence in Bayesian network inference. *J Artif Intell Res*, *5*, 301–328.