Abstraction Sampling in Graphical Models

Filjor Broka
Department of Computer Science
University of California, Irvine
fbroka@ics.uci.edu

Introduction

This research project is in collaboration with my thesis advisor Dr. Rina Dechter and with Dr. Kalev Kask and Dr. Alexander Ihler, also at the University of California, Irvine. We present a new sampling scheme for approximating hard to compute queries over graphical models, such as computing the partition function. The scheme builds upon exact algorithms that traverse a weighted directed state-space graph representing a global function over a graphical model (e.g., probability distribution). With the aid of an abstraction function and randomization, the state space can be compacted (trimmed) to facilitate tractable computation, yielding a Monte Carlo estimate that is unbiased. We present the general idea and analyze its properties analytically and empirically.

Related Works

Imagine that we want to compute a function over a weighted directed graph where the graph is given implicitly, e.g., using a generative state-space search model whose explicit state graph is enormous and does not fit in memory. Clearly algorithms that need to traverse such a state-space graph will fail. Knuth and Chen (Knuth 1975; Chen 1992) proposed a pioneering sampling scheme for estimating quantities that can be expressed as aggregates (e.g., sum) of functions defined over the nodes in the graph. They focused on estimating the number of nodes in such graphs (Knuth 1975). Our work is inspired by their scheme and extends it to a more general class of functions defined over weighted directed graphs or trees. In particular, this extension is applicable to *reasoning tasks over graphical models e.g., probability of evidence or partition function over probabilistic networks* since they can be transformed into tasks over weighted directed state graphs (Dechter and Mateescu 2007). The work of Knuth and Chen (Knuth 1975; Chen 1992) was also extended more recently in the context of predicting the size of search trees in heuristic search (Lelis, Zilles, and Holte 2013) and for search algorithms in graphical models (Lelis, Otten, and Dechter 2014).

Background

A graphical model can be defined by a 4-tuple $\mathcal{M} = (\mathbf{X}, \mathbf{D}, \mathbf{F}, \prod)$, where $\mathbf{X} = \{X_i : i \in I\}$ is a set of variables indexed by a set I, and $\mathbf{D} = \{D_i : i \in I\}$ is the set of finite domains of values for each X_i. $\mathbf{F} = \{\psi_\alpha : \alpha \in scopes(F)\}$ is a set of discrete functions, where $\alpha \subseteq I$ and $X_\alpha \subseteq X$ is the scope of ψ_α. A graphical model represents a global function $Pr(X) \propto \prod_\alpha \psi_\alpha(X_\alpha)$. A popular task is to compute the partition function $Z = \sum_X \prod_\alpha \psi_\alpha(X_\alpha)$.

A graphical model can also be expressed via a weighted state space graph. In an OR search space, the states are partial assignments relative to a variable ordering, where each layer corresponds to a new assigned variable. A graphical model can also be transformed into a more compact AND/OR search space (Dechter and Mateescu 2007), defined relative to a *pseudo tree* of the primal graph, by capturing conditional independence in the model, thus facilitating more effective algorithms (Marinescu and Dechter 2009).

Abstraction Sampling Algorithm

Abstraction sampling for AND/OR trees (AOAS) uses an *abstraction function* that partitions the nodes in the search tree into subsets of *abstract* states under the assumption that nodes in the same abstract state represent similar subproblems and can therefore be *merged* into a "typical" single representative. Given an abstraction function over a weighted directed tree T, our algorithm generates a weighted directed subtree $\tilde{T}$ using a generative randomized process. In particular, the scheme chooses randomly, using importance sampling probabilities, a single representative node from each encountered abstract states and associates it with a *weight* that estimates the total contribution of all states it represents. Importance sampling probabilities are chosen proportional to the current weight of the node, the cost of the path to the root, and an estimate of the total mass of the subtree rooted by this node, given by a heuristic function. An estimator to our query over T can be computed over the generated representative tree $\tilde{T}$, which is supposed to be far smaller. Clearly, if the number of abstract states is bounded, the generated tree is small and the estimator can be computed efficiently. For general AND/OR trees (as opposed to OR trees), we enforce a restriction on abstraction functions, called *properness*.

Properties of Abstraction Sampling

Unbiasedness

THEOREM **1** *Given a weighted directed AND/OR search tree T derived from a graphical model, a value function $Z(n)$ defined recursively over T, and a proper abstraction function over T, the estimate generated by AOAS, $\hat{Z}(r)$, is unbiased.*

Complexity

The *properness* restriction limits compactness of sampled AND/OR trees. More branchings in the pseudo tree imply more abstract states and larger probes. We show that:

THEOREM **2** *Given a pseudo tree, the number of states in a probe by AOAS is $O(n \cdot m^{b+1})$, where n is the number of variables, b bounds the number of branchings along any path of the pseudo tree and m bounds the number of states in the input abstraction function per each variable (and level). For OR trees, $b = 0$, so size is bounded by $O(nm)$.*

Sampling Probabilities

The proof of unbiasedness works for any sampling distribution p. The reason for choosing our specific importance sampling probabilities is to reduce variance.

THEOREM **3 (exact heuristic)** *If the sampling probability in AOAS uses an exact heuristic $h(n) = Z(n)$, namely if it satisfies $p = \frac{w(n')g(n')Z(n')}{w(n')g(n')Z(n')+w(n'')g(n'')Z(n'')}$, then $\hat{Z}$ is exact after a single probe (has zero variance), for any abstraction which is proper.*

Experimental Results

We evaluated our algorithm on instances from 4 benchmarks (DBN, Grids, Linkage, Promedas) by running experiments using different configurations of abstraction functions for one hour each. We used Weighted Mini-Bucket Elimination (WMBE) (Dechter and Rish 2003; Liu and Ihler 2011) heuristic function, whose strength is controlled by a parameter called the i-bound. Higher i-bounds lead to stronger heuristics at the expense of higher computation and memory cost. We experimented with fixed i-bound 10. We experimented with both OR and AND/OR trees. As abstraction function we used a family of abstraction functions called relaxed context-based abstractions, parametrized by a level. 0-level abstraction merges all nodes corresponding to a variable in a single abstract state, leading to a degenerate tree (path), and it thus corresponds to regular importance sampling. Higher level abstractions have increasing number of abstract states, thus leading in general to higher number of nodes expanded during sampling. We tested abstraction levels 0, 4, 8 for OR trees, and levels 0, 1, 2 for AND/OR trees.

We noticed that for benchmarks where regular importance sampling (0-level) has large errors (DBN and Grids), abstraction sampling with high level abstraction shows much stronger performance. For Linkage and Promedas benchmarks, where regular sampling already performs well (small errors), we see that we are not able to improve performance by moving to higher level abstractions.

Summary of Contributions

We developed abstraction sampling algorithm for AND/OR trees (AOAS) and provided theoretical and empirical evaluation. Specifically we show that a) AOAS is unbiased for OR trees under minimal conditions of the abstraction function (i.e., being layered). b) that these conditions are not sufficient for AND/OR trees in general, however when enforcing a restriction of *properness*, unbiasedness is guaranteed for AND/OR trees as well. (c) In an extensive empirical evaluation, using instances from several benchmarks, we illustrate the potential of this scheme, showing that it can significantly improve performance over the baseline of regular importance sampling in benchmarks where the baseline has large errors.

Future Work

Future work would include exploring different abstraction function families, in order to find abstraction functions which are most effective in reducing sampling error. We also plan to extend the existing algorithm from AND/OR trees to AND/OR graphs to benefit from sampling on smaller search spaces.

Acknowledgments

This work was supported in part by NSF grants IIS-1526842 and IIS-1254071.

References

Chen, P.-C. 1992. Heuristic sampling: A method for predicting the performance of tree searching programs. *SIAM Journal on Computing* 21:295–315.

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

Dechter, R., and Rish, I. 2003. Mini-buckets: A general scheme for bounded inference. *J. ACM* 50(2):107–153.

Knuth, D. 1975. Estimating the efficiency of backtracking algorithms. *Math. Comput.* 29:1121–136.

Lelis, L. H. S.; Otten, L.; and Dechter, R. 2014. Memory-efficient tree size prediction for depth-first search in graphical models. In O'Sullivan, B., ed., *Principles and Practice of Constraint Programming - 20th International Conference, CP 2014, Lyon, France, September 8-12, 2014. Proceedings*, volume 8656 of *Lecture Notes in Computer Science*, 481–496. Springer.

Lelis, L. H. S.; Zilles, S.; and Holte, R. C. 2013. Predicting the size of IDA*'s search tree. *Artificial Intelligence* 196(Supplement C):53 – 76.

Liu, Q., and Ihler, A. T. 2011. Bounding the partition function using holder's inequality. In *Proceedings of the 28th International Conference on Machine Learning, ICML 2011, Bellevue, Washington, USA, June 28 - July 2, 2011*, 849–856.

Marinescu, R., and Dechter, R. 2009. AND/OR branch-and-bound search for combinatorial optimization in graphical models. *Artificial Intelligence* 173(16-17):1457–1491.