Parallel Algorithms and Generalized Frameworks for Learning Large-Scale Bayesian Networks

Poster Summary

Ankit Srivastava
School of Computational Science and Engineering
Georgia Institute of Technology
Atlanta, GA, USA
asrivast@gatech.edu

Srinivas Aluru (advisor)
School of Computational Science and Engineering
Georgia Institute of Technology
Atlanta, GA, USA
aluru@cc.gatech.edu

ABSTRACT

Bayesian networks (BNs) are an important subclass of graphical machine learning (ML) models that enable probabilistic reasoning about interactions between variables of interest. Their interpretability makes them an ideal model for making high-stakes decisions in fields where explainability is desirable. However, learning BNs with even few thousand variables using existing software libraries requires an infeasible amount of time. This has prevented BNs from becoming a viable alternative to other ML models. To address this, we have developed scalable high-performance libraries for learning large-scale BNs. In this poster, we present our work on parallelizing a variety of popular BN learning algorithms, including a method for constructing parameter-sharing specialization of BNs – module networks. Our experiments show that the optimized open-source implementations of our parallel algorithms reduce the time required for learning networks with tens of thousands of variables from multiple months to a few hours by efficiently utilizing thousands of cores.

CCS CONCEPTS

• **Computing methodologies → Bayesian network models**.

KEYWORDS

Bayesian networks, module networks, score-based learning, parallel machine learning, gene networks

ACM Reference Format:
Ankit Srivastava and Srinivas Aluru (advisor). 2021. Parallel Algorithms and Generalized Frameworks for Learning Large-Scale Bayesian Networks: Poster Summary. In *The International Conference for High Performance Computing, Networking, Storage and Analysis (SC '21), November 14–19, 2021, St. Louis, MO, USA.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Bayesian networks (BNs) are a subclass of probabilistic graphical models that employ directed acyclic graphs to compactly represent exponential-sized joint probability distributions over a set of random variables [10]. Since BNs enable interpretable reasoning about interactions between the variables of interest [24], they have already been successfully employed in multiple fields with high human-impact [8, 18, 22]. Furthermore, the recent focus on the need for explainability in the decisions made by machine learning (ML) models [6] has led to a push for the use of inherently interpretable models like BNs for making high-stakes decisions [11].

Given a data set sampled from a joint probability distribution, learning the exact BN structure is NP-hard [5]. Correspondingly, a wide range of heuristic algorithms have been developed for learning BN structure. However, the heuristics are also computationally intensive and can take months to sequentially learn large-scale networks [16]. The existing libraries for learning BNs that support multiple heuristics are either completely sequential (e.g., *pcalg* [7]) or support only limited parallelism (e.g., *bnlearn* [13] and *Tetrad* [12]). We posit that the lack of a high-performance library for expeditious learning of large-scale BNs is a major hurdle in their viability as an alternative to other ML models.

2 METHODOLOGY

To fill the void discussed in Section 1, we developed scalable parallel algorithms for a wide variety of popular BN learning heuristics. First, we developed a generalized framework for parallelizing the algorithms classified as *constraint-based*. Using this framework, we proposed efficient parallel versions of five different sequential algorithms in the category: *GS* [9], *IAMB* [20], *Inter-IAMB* [20], *MMPC* [20, 21], and *SI-HITON* [1]. Then, we proposed the first parallel approach for the construction of module networks (MoNets) [14] – a parameter-sharing specialization of BNs – using the *score-based* method known as *Lemon-Tree* [4].

The design of these parallel algorithms utilize two key ideas: 1) fine-grained definition of the total available work through tuples, and 2) distribution of tuples for load balancing. All our parallel algorithms are designed to produce the exact same networks as the corresponding sequential versions, irrespective of the parallelism used. We implemented these algorithms using *C++* and *MPI* and optimized them for good performance in practice.

3 RESULTS

We evaluated our implementations for an important application of BNs – construction of gene-regulatory networks from gene expression data sets. In our experiments, we learned the genome-scale networks for two model organisms, *Saccharomyces cerevisiae* and *Arabidopsis thaliana*, from big data sets [2, 19]. Our implementations show significant sequential speedup over the prior state-of-the-art (up to 60X over *bnlearn* for learning BNs and 3.8X over *Lemon-Tree* [3] for learning MoNets). Further, they scale well for learning networks with tens of thousands of variables and reduce the time

SC '21, November 14–19, 2021, St. Louis, MO, USA
2021. ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

required for learning the corresponding BNs from more than a week using *bnlearn* to less than 38 seconds using 2048 cores, and that for learning MoNets from more than a year using *Lemon-Tree* to less than 2.8 hours using 4096 cores.

These results improve and add to our previous works that have been accepted in peer-reviewed conferences [16, 17]. We have also released the corresponding implementations as part of open-source libraries – *ramBLe* [15] and *ParsiMoNe* [23]. Our implementations are agnostic to the underlying application and can be used by ML and application-domain researchers for expeditious construction of large-scale BNs and MoNets.

4 FUTURE WORK

We are planning to make our libraries more accessible by making them available as *Python* and *R* packages. Additionally, we are planning to implement more learning heuristics and investigate accelerators like GPUs for improving the performance of our implementations further.

ACKNOWLEDGMENTS

To Sriram P. Chockalingam for his guidance. This research is supported in part by the National Science Foundation under OAC-1828187 and OAC-1854828.

REFERENCES

[1] Constantin F Aliferis, Alexander Statnikov, Ioannis Tsamardinos, Subramani Mani, and Xenofon D Koutsoukos. 2010. Local causal and Markov blanket induction for causal discovery and feature selection for classification part I: algorithms and empirical evaluation. *Journal of Machine Learning Research* 11, 1 (2010).

[2] Maneesha Aluru and Sriram Chockalingam. 2021. *A. thaliana Gene Expression Dataset for Development Processes.* https://doi.org/10.5281/zenodo.4672797

[3] Eric Bonnet. 2015. Lemon-Tree - Module Network Inference software. https://github.com/erbon7/lemon-tree.

[4] Eric Bonnet, Laurence Calzone, and Tom Michoel. 2015. Integrative multi-omics module network inference with Lemon-Tree. *PLoS Comput Biol* 11, 2 (2015), e1003983.

[5] David Maxwell Chickering, David Heckerman, and Christopher Meek. 2004. Large-sample learning of Bayesian networks is NP-hard. *Journal of Machine Learning Research* 5, Oct (2004), 1287–1330.

[6] David Gunning and David W Aha. 2019. DARPA's Explainable Artificial Intelligence Program. *AI Magazine* 40, 2 (2019), 44–58.

[7] Markus Kalisch, Martin Mächler, Diego Colombo, Marloes H Maathuis, and Peter Bühlmann. 2012. Causal inference using graphical models with the R package pcalg. *Journal of Statistical Software* 47, 11 (2012), 1–26.

[8] Evangelia Kyrimi, Scott McLachlan, Kudakwashe Dube, Mariana R Neves, Ali Fahmi, and Norman Fenton. 2020. A Comprehensive Scoping Review of Bayesian Networks in Healthcare: Past, Present and Future. *arXiv preprint arXiv:2002.08627* (2020).

[9] Dimitris Margaritis and Sebastian Thrun. 2000. Bayesian network induction via local neighborhoods. In *Advances in neural information processing systems*. MIT press, 505–511.

[10] Judea Pearl. 1985. Bayesian networks: A model of self-activated memory for evidential reasoning. In *Proceedings of the 7th conference of the Cognitive Science Society, University of California, Irvine, CA, USA*. 15–17.

[11] Cynthia Rudin. 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence* 1, 5 (2019), 206–215.

[12] Richard Scheines, Peter Spirtes, Clark Glymour, Christopher Meek, and Thomas Richardson. 1998. The TETRAD project: Constraint based aids to causal model specification. *Multivariate Behavioral Research* 33, 1 (1998), 65–117.

[13] Marco Scutari. 2010. Learning Bayesian Networks with the bnlearn R Package. *Journal of Statistical Software* 35, i03 (2010), 22 pages.

[14] Eran Segal, Dana Pe'er, Aviv Regev, Daphne Koller, Nir Friedman, and Tommi Jaakkola. 2005. Learning module networks. *Journal of Machine Learning Research* 6, 4 (2005).

[15] Ankit Srivastava. 2020. ramBLe - A Parallel Framework for Bayesian Learning. https://github.com/asrivast28/ramBLe.

[16] Ankit Srivastava, Sriram Chockalingam, Maneesha Aluru, and Srinivas Aluru. 2021. Parallel Construction of Module Networks. In *2021 SC21: International Conference for High Performance Computing, Networking, Storage and Analysis (SC)*. ACM.

[17] Ankit Srivastava, Sriram Chockalingam, and Srinivas Aluru. 2020. A Parallel Framework for Constraint-based Bayesian Network Learning via Markov Blanket Discovery. In *2020 SC20: International Conference for High Performance Computing, Networking, Storage and Analysis (SC)*. IEEE Computer Society, 74–88.

[18] Franco Taroni, Alex Biedermann, Silvia Bozza, Paolo Garbolino, and Colin Aitken. 2014. *Bayesian networks for probabilistic inference and decision analysis in forensic science*. John Wiley & Sons.

[19] Konstantine Tchourine, Christine Vogel, and Richard Bonneau. 2018. Condition-specific modeling of biophysical parameters advances inference of regulatory networks. *Cell reports* 23, 2 (2018), 376–388.

[20] Ioannis Tsamardinos, Constantin F Aliferis, Alexander R Statnikov, and Er Statnikov. 2003. Algorithms for large scale Markov blanket discovery.. In *FLAIRS conference*, Vol. 2. AAAI Press, 376–380.

[21] Ioannis Tsamardinos, Laura E Brown, and Constantin F Aliferis. 2006. The max-min hill-climbing Bayesian network structure learning algorithm. *Machine learning* 65, 1 (2006), 31–78.

[22] Charlotte S Vlek, Henry Prakken, Silja Renooij, and Bart Verheij. 2016. A method for explaining Bayesian networks for legal evidence with scenarios. *Artificial Intelligence and Law* 24, 3 (2016), 285–324.

[23] Ankit Srivastava. 2021. ParsiMoNe - Parallel Construction of Module Networks. https://github.com/asrivast28/ParsiMoNe.

[24] Changhe Yuan, Heejin Lim, and Tsai-Ching Lu. 2011. Most relevant explanation in Bayesian networks. *Journal of Artificial Intelligence Research* 42 (2011), 309–352.