Learning the structure of Bayesian Networks using constraint programming

Fulya Trösser, [1] **Simon de Givry,** [1] **George Katsirelos**[2]

[1]Université de Toulouse, INRAE, UR MIAT, F-31320, Castanet-Tolosan, France
[2]UMR MIA-Paris, INRAE, AgroParisTech, Univ. Paris-Saclay, 75005 Paris, France
{fulya.ural, simon.de-givry}@inrae.fr, gkatsi@gmail.com

Towards the goal of explainable AI, Bayesian networks offer a rich framework for probabilistic reasoning. Bayesian Network Structure Learning (BNSL) from discrete observations corresponds to finding a compact model which best explains the data. It defines an NP-hard problem with a superexponential search space of Directed Acyclic Graphs (DAG). Several constraint-based (exploiting local conditional independence tests) and score-based (exploiting a global objective formulation) BNSL methods have been developed in the past.

Complete methods for score-based BNSL include dynamic programming (Silander and Myllymäki 2006), heuristic search (Yuan and Malone 2013; Fan and Yuan 2015), maximum satisfiability (Berg, Järvisalo, and Malone 2014), branch-and-cut (Bartlett and Cussens 2017) and constraint programming (van Beek and Hoffmann 2015). Here, we focus on the latter two.

GOBNILP (Bartlett and Cussens 2017) is a state-of-the-art solver for BNSL. It implements branch-and-cut in an integer linear programming (ILP) solver. At each node of the branch-and-bound tree, it generates cuts that improve the linear relaxation. A major class of cuts generated by GOBNILP are *cluster cuts*, which identify sets of parent sets that cannot be used together in an acyclic graph. In order to find cluster cuts, GOBNILP solves an NP-hard subproblem created from the current optimal solution of the linear relaxation.

CPBayes (van Beek and Hoffmann 2015) is a constraint programming-based (CP) method for BNSL. It uses a CP model that exploits symmetry and dominance relations present in the problem, subproblem caching, and a pattern database to compute lower bounds, adapted from heuristic search (Fan and Yuan 2015). van Beek and Hoffmann showed that CPBayes is competitive with GOBNILP in many instances. In contrast to GOBNILP, the inference mechanisms of CPBayes are very lightweight, which allows it to explore many orders of magnitude more nodes per time unit, even accounting for the fact that computing the pattern databases before search can sometimes consume considerable time. On the other hand, the lightweight pattern-based bounding mechanism can take into consideration only limited information about the current state of the search. Specifically, it can take into account the current total ordering implied by the DAG under construction, but no information that has been derived about the potential parent sets of each vertex, i.e., the current domains of parent set variables.

In this work, we derive a lower bound that is computationally cheaper than that computed by GOBNILP. We give a polynomial-time algorithm that discovers a subclass of cluster cuts that provably improve the linear relaxation. We then give a greedy algorithm for solving the linear relaxation, inspired by similar algorithms for MaxSAT and Weighted Constraint Satisfaction Problems (WCSP). Finally, we give an algorithm that enforces generalised arc consistency on the acyclicity constraint, based on previous work by van Beek and Hoffmann, but with improved complexity and practical performance. Our implementation of these techniques in the solver CPBayes leads to significantly improved performance, both in the size of the search tree explored and in runtime.

References

Bartlett, M.; and Cussens, J. 2017. Integer Linear Programming for the Bayesian network structure learning problem. *Artificial Intelligence*, 258–271.

Berg, J.; Järvisalo, M.; and Malone, B. 2014. Learning optimal bounded treewidth Bayesian networks via maximum satisfiability. In *Artificial Intelligence and Statistics*, 86–95. PMLR.

Fan, X.; and Yuan, C. 2015. An Improved Lower Bound for Bayesian Network Structure Learning. In *Proc. of AAAI-15*. Austin, Texas.

Silander, T.; and Myllymäki, P. 2006. A Simple Approach for Finding the Globally Optimal Bayesian Network Structure. In *Proc. of UAI'06*. Cambridge, MA, USA.

van Beek, P.; and Hoffmann, H.-F. 2015. Machine learning of Bayesian networks using constraint programming. In *Proc. of International Conference on Principles and Practice of Constraint Programming*, 429–445. Cork, Ireland.

Yuan, C.; and Malone, B. 2013. Learning Optimal Bayesian Networks: A Shortest Path Perspective. *J. of Artificial Intelligence Research*, 48: 23–65.