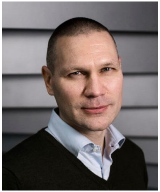
Revisiting Bayesian Network Learning with Small Vertex Cover

Juha Harviainen

Mikko Koivisto



UNIVERSITY OF HELSINKI

UAI 2023

Overview

Introduction

Score-based Structure Learning

- Each parent set G_v of v is assigned a weight $f_v(G_v)$
- The score of a DAG G is the product of vertex-wise scores:

$$f(G) := \prod_{v \in V} f_v(G_v)$$

Bayesian Network Structure Learning (BNSL)

Objective: Compute $\max_G f(G)$

- NP-hard in general[1]
- Often sum of log-scores optimized instead

[1] David M. Chickering. Learning Bayesian networks is NP-complete. *AISTATS'95*.

Sampling and Counting?

- A single DAG might not be enough
- Model averaging and prevalence of features

Bayesian Network Structure Counting (BNSC)

Objective: Compute $\sum_G f(G)$

Bayesian Network Structure Sampling (BNSS)

Objective: Sample G with $\Pr(G) \propto f(G)$

- Counting is $\#$P-hard *[this work]*

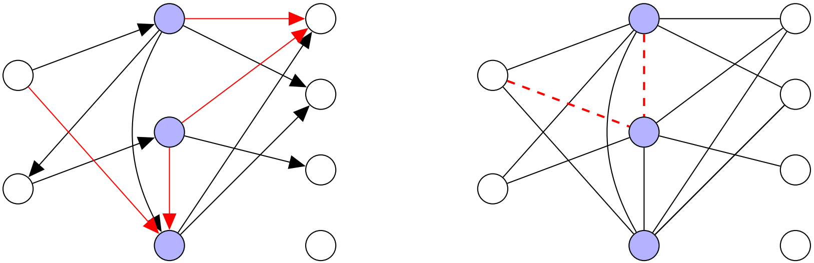
Parameterization

How much do we need to **restrict the set of valid structures** to obtain faster algorithms?

- **Parameterized complexity**: What happens if we limit some aspect of the graphs
- For example, the size of the minimum **vertex cover** (VC)
- Set S is a VC if every edge has at least one endpoint in S

Vertex Cover of Moralized Graph

- **Moralization**: v and w are connected if
  - either $v \to w$ or $v \leftarrow w$, or
  - there is u with $v \to u \leftarrow w$
- Consider only DAGs with VC of size at most k after moralization
- Still hard but polynomial in n if k is fixed[2]



[2] Janne H. Korhonen and Pekka Parviainen. Tractable Bayesian Network Structure Learning with Bounded Vertex Cover Number. *NIPS'15*.

Our Contributions

Nearly a **quadratic speedup** for parameterized structure learning *[this talk]*

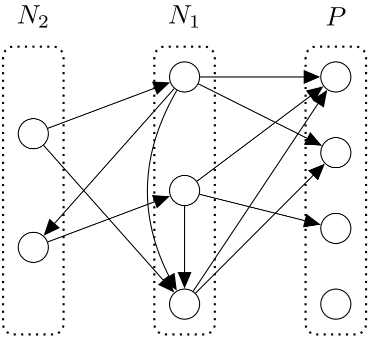
Novel parameterized algorithms for **counting and sampling** structures *[this talk]*

Complexity-theoretical hardness results for counting in general and parameterized cases

Learning

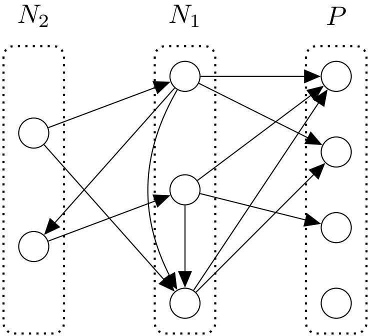
Core and Periphery

- Distribute vertices into core and periphery
- **Core** $N_1 \cup N_2$: VC N_1 of the moralized graph and their parents N_2
- **Periphery** P: other vertices (without children)
- Core and periphery can be optimized independently



Optimization

- Previous work searches over $n^{2k}/(k!)^2$ unordered sets N_1 and N_2
- Core optimized in roughly 2^{2k} operations



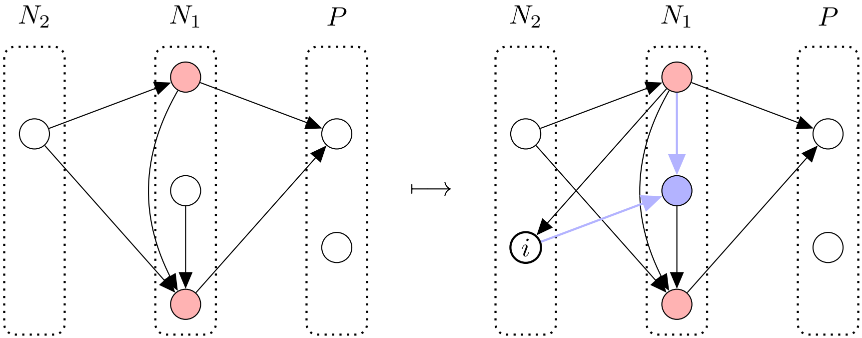
Faster Learning

Outline:

- Brute forcing over n^k **ordered** sets N_1 is sufficient
- Distribute vertices between N_2 and P with dynamic programming
- Maintain information about vertices in N_1 with parents

Faster Learning

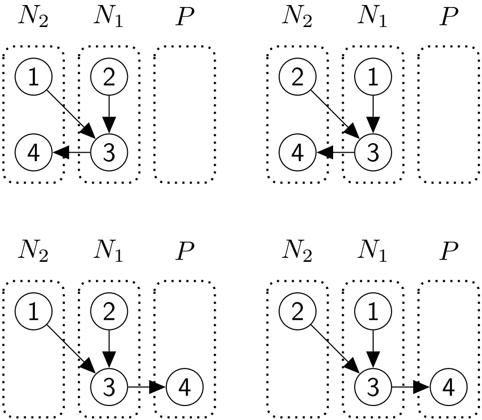
- Fix N_1 and index remaining vertices arbitrarily
- Assume we know best DAG for N_1 and the first $i-1$ of the remaining vertices such that $S \subseteq N_1$ has parents outside N_1
- For each $T \subseteq N_1 \setminus S$ find best DAG with vertex i being a parent of T
- Takes $3^k n^{k+O(1)}$ time (or $2^k n^{k+O(1)}$ is certain cases)



Sampling

Canonical Form?

- How to avoid multiple ways of representing a DAG?



- Canonical form hard to establish

Parent Decompositions

- We settle for **limiting** the number of duplicates

Definition

A DAG and a partition of V into sets N_1, N_2, and P are called a **parent decomposition** if all vertices in N_1 and N_2 have a child.

- At most $2k$ vertices in the core
- By naïve analysis, each DAG has at most 2^{2k} parent decompositions
- More careful analysis gives an upper bound 2^k

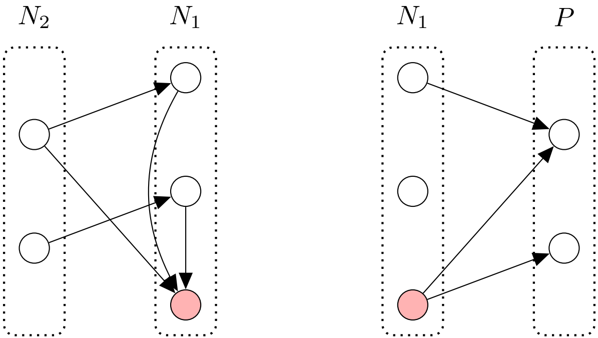
Sampling with Decompositions

Outline:

- Iterate over sets N_1, N_2, and P
  - Compute total weight of each parent decomposition
- Sample a DAG together with a parent decomposition
- Determine a canonical parent decomposition for each DAG
- Accept only that decomposition, reject the sample otherwise

Weights of Decomposition

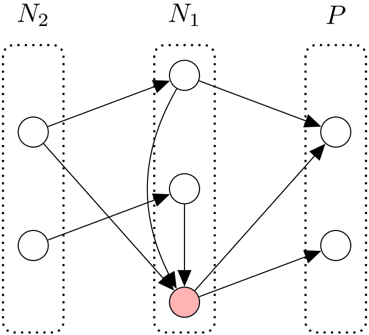
- Fix sets N_1, N_2, and P
- As well as the set $S \subseteq N_1$ of sinks in the core
  - To be a parent decomposition, they must have children in P!
- Sum the cores and peripheries independently and take their product
- **Covering product** for peripheries, **root-layerings**[3] for cores



[3] Topi Talvitie, Aleksis Vuoksenmaa, and Mikko Koivisto. Exact Sampling of Directed Acyclic Graphs from Modular Distributions. *UAI'19*.

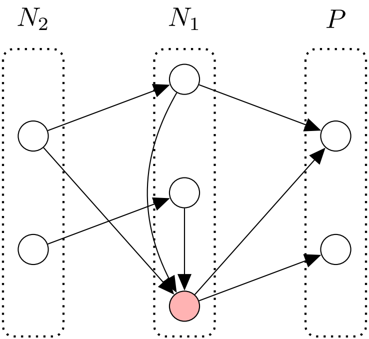
Approximating the Total Weight

- Each DAG satisfying constraints has at most 2^k parent decompositions
- Sum over all N_1, N_2, P, and S gives an 2^k-approximation U
- How to use this for sampling?



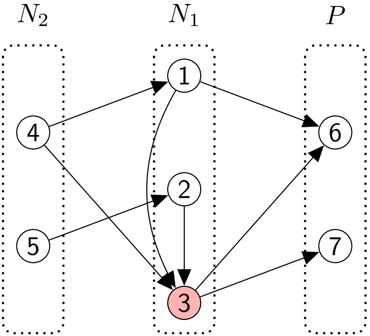
Biased Sampling

- Stochastic backtracking
- Pick N_1, N_2, P, and S at random proportionally to their total weight
- Sample edge structure independently for the core and the periphery
- **Issue:** Same DAG can come from multiple parent decompositions



Rejection Sampling

- **Issue:** Same DAG can come from multiple parent decompositions
- **Solution:** Accept the DAG iff parent decomposition has lexicographically smallest N_2, otherwise reject



Acceptance Probability

- Let $W(N_1, N_2, P, S)$ be the total weight of DAGs with that parent decomposition and set of core sinks
- Further, let G be a DAG with such decomposition
- Probability of sampling G with that decomposition is

$$\frac{W(N_1, N_2, P, S)}{U} \cdot \frac{f(G)}{W(N_1, N_2, P, S)} = \frac{f(G)}{U}$$

- With $\operatorname{par} G$ being the number of parent decompositions of G,
    - sampling probability $\operatorname{par} G \cdot f(G)/U$
    - acceptance probability $f(G)/U$
    - expected acceptance rate $\sum_G f(G)/U$

Better Approximation

- Expected acceptance rate $\sum_G f(G)/U \geq 2^{-k}$
- Each sample a Bernoulli random variable (reject 0, accept 1)
- Multiply empirical acceptance rate by U
- Mean of Bernoulli has good concentration bounds
- Enables approximation at arbitrary precision

Concluding Remarks

Concluding Remarks

- New algorithms for parameterized learning, counting, and sampling
- What is known now:

Problem		Complexity	Class
Optimization		$3^k n^{k+O(1)}$	W[2]-hard[4]
Sampling	Preprocessing	$(4en/k)^{2k} n^{O(1)}$	W[2]-hard*
	Sampling	$4^k n^{O(1)}$	
Counting		$2^{\binom{2k}{2}} 12^k n^{2k+O(1)}$	#W[1]-hard

- What about other parameters?
  - Are the results there best possible?
  - How much restriction needed for FPT sampling or counting?

[4] Niels Grüttemeier and Christian Komusiewicz. Learning Bayesian Networks Under Sparsity Constraints: A Parameterized Complexity Analysis. *J. Artif. Intell. Res. 74*. 2022.

* Sampling enables existence testing

Thank you!