The Parameterized Complexity of Approximate Inference in Bayesian Networks

Johan Kwisthout

j.kwisthout@donders.ru.nl
http://www.dcc.ru.nl/johank/

Donders Center for Cognition
Department of Artificial Intelligence

PGM 2016
8th September 2016

Overview

- Exact Bayesian inference is known to be PP-complete [and thus NP-hard] in general (Littman et al, 1998)
- We know quite well when exact inference can be tractable:
  - Limit treewidth of the network and cardinality of the variables
- We know that approximating Bayesian inference is also intractable* in general
  - * Standard complexity disclaimers: $\mathsf{P} \neq \mathsf{NP}, \mathsf{BPP} \neq \mathsf{NP}$
- **When is approximate Bayesian inference tractable?**

Agenda

- Introduction
- Fixed-error randomized tractability
- Parameters of interest
- Example results
- De-randomization result
- Conclusion

Introduction

Problem definitions

MProb

Input: A Bayesian network $\mathcal{B}$ with designated subset of variables **H** and a corresponding joint value assignment **h** to **H**.

Output: $\Pr(\mathbf{h})$.

CProb

Input: A Bayesian network $\mathcal{B}$ with designated non-overlapping subsets of variables **H** and **E** and corresponding joint value assignments **h** to **H** and **e** to **E**.

Output: $\Pr(\mathbf{h} \mid \mathbf{e})$.

Approximation

Additive approximation

AA-MProb

Input: As in MProb, in addition, error bound $\epsilon < 1/2$.

Output: $q(\mathbf{h})$ such that $\Pr(\mathbf{h}) - \epsilon < q(\mathbf{h}) < \Pr(\mathbf{h}) + \epsilon$.

AA-CProb

Input: As in CProb, in addition, error bound $\epsilon < 1/2$.

Output: $q(\mathbf{h} \mid \mathbf{e})$ such that $\Pr(\mathbf{h} \mid \mathbf{e}) - \epsilon < q(\mathbf{h} \mid \mathbf{e}) < \Pr(\mathbf{h} \mid \mathbf{e}) + \epsilon$.

Approximation

Relative approximation

RA-MPROB
Input: As in MPROB, in addition, error bound ϵ.
Output: $q(\mathbf{h})$ such that $\frac{\Pr(\mathbf{h})}{1+\epsilon} < q(\mathbf{h}) < \Pr(\mathbf{h}) \times (1+\epsilon)$.

RA-CPROB
Input: As in CPROB, in addition, error bound ϵ.
Output: $q(\mathbf{h} \mid \mathbf{e})$ such that $\frac{\Pr(\mathbf{h} \mid \mathbf{e})}{1+\epsilon} < q(\mathbf{h} \mid \mathbf{e}) < \Pr(\mathbf{h} \mid \mathbf{e}) \times (1+\epsilon)$.

Approximation

- These approximation problems are **deterministic**: They are guarenteed to always give results within these bounds
- Stochastic (randomized) approximations: Add confidence bound $0 < \delta < 1$
- Approximate result is within error bounds ϵ with probability **at least** δ
- If δ is polynomially bounded away from $1/2$, then the probability of answering correctly can be boosted arbitrarily close to 1 while still requiring only polynomial time

PP and BPP

- PP and BPP are defined as classes of decision problems decidable by a **probabilistic Turing machine**
- PP: Yes-instances are accepted with probability $\delta = 1/2 + 1/c^n$
- BPP: Yes-instances are accepted with probability polynomially bounded away from $1/2$ (i.e., $\delta = 1/2 + 1/n^c$)
- We know that NP $\subseteq$ PP yet assume that BPP $=$ P
- If $\Pi \in$ BPP than Π has an efficient randomized (Monte Carlo) algorithm and δ can be boosted close to 1 in poly time

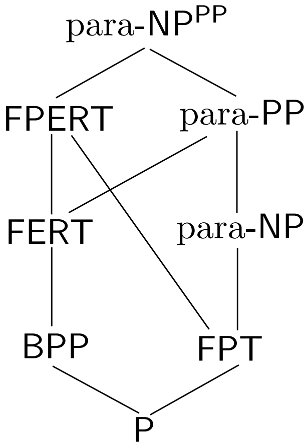
Parameterized complexity

- A (possibly NP-hard) problem Π is called **fixed-parameter tractable** for parameter κ if it can be solved in time, exponential *only* in κ and polynomial in the input size n, that is, in $\mathcal{O}(f(\kappa) \cdot n^c)$
- In practice, this means that instances of Π can be solved **efficiently**, even when the problem is NP-hard in general, if κ is known to be small
- if Π remains NP-hard for all but finitely many values of the parameter κ, then $\{\kappa\}$-Π is para-NP-hard: bounding κ **does not render** the problem tractable

Fixed-error randomized tractability

- Analogously, the class FERT (Kwisthout, 2015) characterizes problems $\{\kappa\}$-Π that can be **efficiently** computed with a randomized algorithm **if κ is bounded**
- Formally, $\{\kappa\}$-$\Pi \in$ FERT if there is a probabilistic Turing machine that accepts Yes-instances of Π with probability $1/2 + \min(f(\kappa), 1/n^c)$ for a constant c and $f : \mathbb{R} \rightarrow \langle 0, 1/2]$
- If Π remains PP-hard even for bounded κ, then $\{\kappa\}$-Π is para-PP-hard
- Corollary: If Π is NP-hard for bounded κ, then $\{\kappa\}$-$\Pi \notin$ FERT

Inclusion diagram



Parameters of interest

What to constrain?

Parameter	Meaning
D_e	*dependence value* of the evidence
B	*local variance bound* of the network
P_h	posterior probability
$\|\mathbf{E}\|$	number of evidence variables
P_e	prior probability of the evidence
d	maximum indegree of the network
l	maximum path length of the network
ϵ	absolute or relative error

Parameters of interest

- The *dependence value of the evidence* (D_e) is a measure of the cumulative strength of the dependencies in the network, given a particular joint value assignment to the evidence variables
- We can define for any node X_i its dependence strength (given evidence $\mathbf{e}$) λ_i as ${}^{u_i}/_{l_i}$, where l_i and u_i are the greatest, resp. smallest numbers such that

$$\forall_{x_i \in \Omega(X_i)} \forall_{\mathbf{p} \in \Omega(\pi(X_i))} l_i \leq \Pr(X_i = x_i \mid \mathbf{p}, \mathbf{e}) \leq u_i$$

- The dependence value of the network (given evidence $\mathbf{e}$) is then given as $D_e = \prod_i \lambda_i$

Parameters of interest

- The *local variance bound* (B) of a network is a measure of the representational expressiveness and the complexity of inference of the network
- It is defined similarly as the dependence value, but is not conditioned on the evidence, and a maximization rather than a product of the local values is computed
- Let λ_i be ${}^{u_i}/_{l_i}$, where l_i and u_i are the greatest, resp. smallest numbers such that

$$\forall_{x_j \in \Omega(X_i)} \forall_{\mathbf{p} \in \Omega(\pi(X_i))} l_i \leq \Pr(X_i = x_i \mid \mathbf{p}) \leq u_i$$

- The local variance bound of the network is then given as $B = \max_i(\lambda_i)$

Example results (1/3)

- For any fixed $\epsilon < 1/2^{n+1}$ absolute approximate inference is PP-hard (rounding to the nearest $1/2^n$ solves $\textsc{MajSat}$)
- For $\epsilon \geq 1/n^c$ we can efficiently approximate the marginal inference problem absolutely by simple forward sampling
- As the number of samples needed to approximate $\textsc{aa-MProb}$ depends (only) on ϵ, this yields the following results:

Corollary (Kwisthout, 2009)

$\{d\}$-$\textsc{aa-MProb}$ *is para-*PP*-hard.*

Result (Henrion, 1986)

$\{\epsilon\}$-$\textsc{aa-MProb} \in$ FERT.

Example results (2/3)

- For relative approximation, however, it is impossible to approximate RA-MPROB in polynomial time with *any* bound ϵ
- This would directly indicate whether $\Pr(\mathbf{h}) = 0$ or not, and hence by a corollary of Cooper's (1990) result, would decide SATISFIABILITY. We thus have that:

Result (Cooper, 1990)

$\{\epsilon\}$-RA-MPROB *is* para-NP-hard.

Corollary

$\{\epsilon\}$-RA-MPROB $\notin$ FERT.

Example results (3/3)

- More interesting to consider computing posterior probabilities conditioned on evidence
- Dagum & Luby (1993) showed that there cannot be any polynomial time absolute approximation algorithm for any $\epsilon < 1/2$ because such an algorithm would decide 3SAT
- Their proof uses a singleton evidence variable ($|\mathbf{E}| = 1$) and at most three incoming arcs per variable ($d = 3$)

Result (Dagum & Luby, 1993)

$\{\epsilon, d, |\mathbf{E}|\}$-AA-CPROB *is* para-NP-hard.

Corollary

$\{\epsilon, d, |\mathbf{E}|\}$-AA-CPROB $\notin$ FERT.

More interesting results

Result (Dagum & Chavez, 1993)

$\{D_e\}$-AA-CProb *is* para-PP-hard.

Result (Dagum & Chavez, 1993)

$\{D_e, \epsilon\}$-RA-CProb $\in$ FERT.

Result (follows from Park & Darwiche, 2004)

$\{B\}$-RA-CProb $\notin$ FERT *for any* $\epsilon < 1$.

Result (Dagum & Luby, 1997)

$\{B, |\mathbf{E}|, \epsilon\}$-RA-CProb $\in$ FERT.

De-randomization

Result (Henrion, 1986)

$\{\epsilon\}$-AA-MPROB $\in$ FERT.

- We can **de-randomize** this stochastic algorithm by simulating all possible sequences of random coin flips and taking a majority decision
- If there are $\mathcal{O}(n)$ such bits, this will yield an exponential-time algorithm (like exact computation)
- Do we actually need so many random bits?

De-randomization

- If $\mathcal{O}(\log n)$ *distinct* random bits would suffice we could effectively simulate the randomized algorithm in poly time
- $\mathcal{O}(\log n)$ random bits can be amplified in polynomial time to a k-wise independent $\mathcal{O}(n)$-bit random string for every fixed k (Luby, 1988)
- Do all the random bits need to be fully independent of each other?
- **No!** For every variable, the random bits should be independent of the random bits **needed to select its parents**
- If the number of parents is bounded (i.e., bounded in-degree d) then d-wise independence suffices

FPT result for approximate inference

Result (Henrion, 1986)

$\{\epsilon\}$-AA-MPROB $\in$ FERT.

Result

$\{d, \epsilon\}$-AA-MPROB $\in$ FPT.

Theoretical question

Whether BPP $\stackrel{?}{=}$ P is a crucial open question in theoretical computer science; its parameterized analog would be whether FERT $\stackrel{?}{=}$ FPT. Here, we need to bound d as well as ϵ: We have an efficient randomized algorithm for every d but no known deterministic one.

Conclusion

- In this paper we investigated the parameterized complexity of approximate Bayesian inference, both by re-interpretation of known results in the formal framework of fixed-error randomized complexity theory, and by adding a few new results
- This gives an insight in what constraints can, or cannot, render approximation tractable
- These constraints are notably different from the traditional constraints (treewidth and cardinality) that are necessary and sufficient to render exact computation tractable
- In future work we wish to extend this line of research to other problems in Bayesian networks, notably the MAP problem