Electronic Colloquium on Computational Complexity, Report No. 145 (2023)

Total Variation Distance Estimation Is as Easy as Probabilistic Inference*

Arnab Bhattacharyya
National University of Singapore

Sutanu Gayen
Indian Institute of Technology Kanpur

Kuldeep S. Meel
University of Toronto

Dimitrios Myrisiotis
National University of Singapore

A. Pavan
Iowa State University

N. V. Vinodchandran
University of Nebraska-Lincoln

September 20, 2023

Abstract

In this paper, we establish a novel connection between total variation (TV) distance estimation and probabilistic inference. In particular, we present an efficient, structure-preserving reduction from relative approximation of TV distance to probabilistic inference over directed graphical models. This reduction leads to a fully polynomial randomized approximation scheme (FPRAS) for estimating TV distances between distributions over any class of Bayes nets for which there is an efficient probabilistic inference algorithm. In particular, it leads to an FPRAS for estimating TV distances between distributions that are defined by Bayes nets of bounded treewidth. Prior to this work, such approximation schemes only existed for estimating TV distances between product distributions. Our approach employs a new notion of *partial* couplings of high-dimensional distributions, which might be of independent interest.

1 Introduction

Machine learning and data science heavily rely on probability distributions that are widely used to capture dependencies among large number of variables. Such *high-dimensional distributions* naturally appear in various domains including neuroscience [ROL02, CTY06], bioinformatics [BB01], text and image processing [Mur22], and causal inference [Pea09]. Substantial research has been devoted to developing models that represent high-dimensional probability distributions succinctly. One prevalent approach is through graphical models. In a graphical model, a graph describes the conditional dependencies among variables and the probability distribution is factorized according to the adjacency relationships in the graph [KF09]. When the underlying graph is a directed graph, the model is known as a Bayesian network or Bayes net.

Two fundamental computational tasks on distributions are *distance computation* and *probabilistic inference*. In this work, we establish a novel connection between these two seemingly different computational tasks. Using this connection, we design new relative error approximation algorithms for estimating the statistical distance between Bayes net distributions with bounded treewidth.

*The author list has been sorted alphabetically by last name; this should not be used to determine the extent of authors' contributions.

ISSN 1433-8092

Distance computation. The distance computation problem is the following: Given descriptions of two probability distributions P and Q, compute $\rho(P, Q)$ for a distance measure ρ. A distance measure of central importance is the *total variation (TV) distance* (also known as *statistical distance* or *statistical difference*). Let P and Q are distributions over a finite domain $\mathcal{D}$. The total variation distance between P and Q, denoted by $d_{\mathrm{TV}}(P, Q)$, is defined as

$$d_{\mathrm{TV}}(P, Q) = \max_{S \subseteq \mathcal{D}} (P(S) - Q(S)) .$$

The total variation distance satisfies many basic properties which makes it a versatile and fundamental measure for quantifying the dissimilarity between probability distributions. First, it has an explicit probabilistic interpretation: The TV distance between two distributions is the maximum gap between the probabilities assigned to a single event by the two distributions. Second, it satisfies many mathematically desirable properties: It is bounded in $[0, 1]$, it is a metric, and it is invariant with respect to bijections. Total variation distance also measures the minimum probability that $X \neq Y$ among all couplings (X, Y) between P and Q. Because of these reasons, the total variation distance is a central distance measure employed in a wide range of areas including probability and statistics, machine learning, information theory, cryptography, data privacy, and pseudorandomness.

Probabilistic inference. There are several related computational tasks that fall under the umbrella of the term probabilistic inference. We use the following: Given (a representation of) random variables $X_1, \ldots, X_n$ and (a representation of) sets $S_1, \ldots, S_n$ such that for all i the set S_i is a subset of the range of X_i, compute the probability $\mathbf{Pr}[X_1 \in S_1, \ldots, X_n \in S_n]$.

Probabilistic inference in graphical models is a fundamental computational task with a wide range of applications that spans disciplines including statistics, machine learning, and artificial intelligence (e.g., [WJ+08]). Various algorithms have been proposed for this problem, encompassing both exact approaches like message passing [Pea88], variable elimination [Dec99], and junction-tree propagation [LS88], as well as approximate techniques such as loopy belief propagation, variational inference-based methods [WJ+08], and particle-based algorithms (refer to Chapter 13 of [KF09] and the references therein). Computational hardness results have also been established in several works [Coo90, LMP01, Rot96, KBvdG10].

1.1 Our contributions

Our main contribution is a *structure-preserving* reduction from the TV distance estimation problem to the probabilistic inference problem over Bayes nets. In particular, we exhibit an efficient probabilistic reduction such that for two Bayes nets P and Q defined over a directed acyclic graph (DAG) G, the reduction makes probabilistic inference queries to a Bayes net $\mathcal{L}$ defined over the *same* DAG G and returns a relative approximation of the $d_{\mathrm{TV}}(P, Q)$.

Theorem 1 (Informal)**.** *There is a polynomial-time randomized algorithm that takes a DAG G, two Bayes nets P and Q over G, and parameters ε, δ as inputs and behaves as follows. The algorithm makes probabilistic inference oracle queries to a Bayes net over the same DAG G and outputs an $(1 + \varepsilon)$-relative approximation of $d_{\mathrm{TV}}(P, Q)$ with probability at least $1 - \delta$.*

Remark 2. It is known that probabilistic inference computation over Bayes nets is a #P-hard problem and hence exact d_{TV} computation reduces to probabilistic inference over Bayes nets [Coo90]. A salient feature of our reduction is that it *preserves the structure* of the Bayes net. This leads to efficient d_{TV} estimation algorithms for any class of Bayes nets that admits efficient probabilistic inference algorithms. Note that exact d_{TV} computation is #P-complete even for product distributions for which inference computation is straightforward [BGM+23].

As a corollary, we obtain a fully polynomial time randomized approximation scheme (FPRAS) for relatively approximating the TV distance between Bayes nets over any class of DAGs for which an efficient probabilistic inference algorithm exists. The well-known variable elimination algorithm can be used for efficient probabilistic inference for Bayes nets over DAGs with treewidth $O(\log n)$. This leads to a new FPRAS for TV distance estimation for Bayes nets over DAGs with logarithmic treewidth.

Corollary 3 (Informal)**.** *There is an FPRAS for estimating the TV distance between two Bayes nets of treewidth $O(\log n)$ that are defined over the same DAG of n nodes.*

Prior to our work, such approximation schemes were known only for product distributions, which are Bayes nets over a graph with no edges [FGJW23]. In particular, designing an FPRAS for estimating TV distance between Bayes nets over trees (which are graphs with treewidth 1) was an open question. Our result resolves this question. It is known that for Bayes nets over general DAGs we cannot hope to have an FPRAS for relatively approximating TV distance. In particular, [BGM+23] shows that it is NP-hard to decide whether $d_{\mathrm{TV}}(P, Q)$ is zero or not when P and Q are arbitrary Bayes nets over DAGs of in-degree 2. In spite of this impossibility result, Corollary 3 shows that it is indeed possible to obtain an FPRAS for a large class of Bayes nets, namely Bayes nets of $O(\log n)$ treewidth.

Our next set of results focuses on the case when one of the distributions is the uniform distribution. We first prove that the exact computation of the TV distance between a Bayes net distribution and the uniform distribution is #P-complete. To complement this result, we show that there is an FPRAS that estimates the TV distance between the uniform distribution and *any* Bayes net distribution.

Theorem 4. *It is #P-complete to compute the TV distance between a Bayes net that has bounded in-degree and the uniform distribution.*

Theorem 5 (Informal)**.** *There is an FPRAS for estimating the TV distance between a Bayes net and the uniform distribution.*

1.2 Related work

Koller and Friedman [KF09] provide a comprehensive overview of probabilistic graphical models. They discuss the general principles and philosophies behind graphical models, including Bayesian networks and Markov networks.

Distance computation. Recently, Bhattacharyya, Gayen, Meel, Myrisiotis, Pavan, and Vinodchandran [BGM+23] initiated the study of the computational complexity aspects of TV distance over graphical models. In that work, they proved that exactly computing the TV distance between product distributions is #P-complete, that it is NP-hard to decide whether the TV distance between two Bayes nets of in-degree 2 is equal to 0 or not, and also gave an FPTAS for approximating the TV distance between an arbitrary product distribution and the uniform distribution. In a subsequent work, Feng, Guo, Jerrum and Wang [FGJW23] gave an FPRAS for approximating the TV distance between two arbitrary product distributions.

TV distance estimation was also studied previously from a more complexity-theoretic and cryptographic viewpoint. Sahai and Vadhan [SV03] established in a seminal work that additively approximating the TV distance between two distributions that are samplable by Boolean circuits is hard for SZK (Statistical Zero Knowledge). Goldreich, Sahai, and Vadhan [GSV99] showed that the problem of deciding whether a distribution samplable by a Boolean circuit is close or

far from the uniform distribution is complete for the complexity class NISZK (Non-Interactive Statistical Zero Knowledge).

Additive approximation of TV distance is much easier. Canonne and Rubinfeld [CR14] showed how to additively estimate TV distance between distributions that can be efficiently sampled and whose probability mass functions can be efficiently evaluated. Clearly, Bayes nets satisfy both conditions (where "efficient" means as usual polynomial in the number of parameters). Bhattacharyya, Gayen, Meel and Vinodchandran [BGMV20] extended this idea to develop polynomial-time algorithms for additively approximating the TV distance between two bounded in-degree Bayes nets using a polynomial number of samples from each.

Probabilistic inference. There is a significant body of work dedicated to exact probabilistic inference. As we mentioned earlier, some algorithmic paradigms that have been developed for the task of probabilistic inference are message passing [Pea88], variable elimination [Dec99], and junction-tree propagation [LS88]. Recently, Klinkenberg, Blumenthal, Chen, and Katoen [KBCK23] presented an exact Bayesian inference method for inferring posterior distributions encoded by probabilistic programs featuring possibly unbounded looping behaviors. Similarly, Klinkenberg, Winkler, Chen, and Katoen [KWCK23], explore the theory of generating functions and investigate its usage in the exact quantitative reasoning of probabilistic programs.

With the advent of big data and the increasing complexity of models, traditional exact inference methods may become computationally infeasible. Approximate inference techniques, such as variational inference and sampling methods like Markov Chain Monte Carlo, provide efficient and scalable alternatives to tackle these challenges. Minka [Min01] introduces the expectation propagation algorithm for approximate Bayesian inference. This method unifies two previous techniques: Assumed-density filtering, an extension of the Kalman filter, and loopy belief propagation. Murphy, Weiss, and Jordan [MWJ13] investigate the effectiveness of loopy belief propagation. They present empirical results showing the performance of the algorithm and discuss its limitations and trade-offs. Ranganath, Gerrish, and Blei [RGB14] introduce black box variational inference, a flexible and scalable approach for approximate Bayesian inference. Their paper presents a general framework for approximating posterior distributions and discusses applications in latent variable models. Blei, Kucukelbir, and McAuliffe [BKM17] provide a comprehensive review of variational inference, a family of methods for approximate Bayesian inference. They cover the principles of variational inference, present different algorithmic approaches, and discuss its applications in machine learning.

1.3 Organization

The rest of the paper is organized as follows. We provide a technical overview of our results in Section 2 and some background material in Section 3. We prove the main results as follows: We show Theorem 1 in Section 4; Corollary 3 in Section 4.3; Theorem 4 in Section 5.1; Theorem 5 in Section 5.2. We conclude in Section 6.

2 Technical overview

We present in this section some intuition regarding the technical aspects of our results.

2.1 Proof of Theorem 1

Our approach is to carefully define an estimator function f and a distribution π so that $\mathbf{E}_\pi[f] = d_{\mathrm{TV}}(P,Q)/Z$ where Z is an efficiently computable normalization constant. The algorithm proceeds by estimating $\mathbf{E}_\pi[f]$, multiplies it by Z, and returns the value. Probabilistic inference queries are used to compute Z and to sample from the distribution π.

There are several challenges in this approach, including setting up the estimator function f and the distribution π so that: (i) $\mathbf{E}_\pi[f]$ is large enough for an empirical estimate to be a good relative approximation and (ii) probabilistic inference queries can be used to efficiently sample from π and to compute Z.

Our starting point is a well-known connection between the TV distance and *couplings*.

Definition 6. Suppose P and Q are two arbitrary distributions on a common finite set $[\ell]$ (where $\ell > 0$). A *coupling* of P and Q is a distribution on pairs (X,Y) such that $X \sim P$ and $Y \sim Q$. An *optimal coupling* of P and Q is a distribution on pairs (X,Y) such that (1) $X \sim P$, $Y \sim Q$, and (2) for any $w \in [\ell]$, $\mathbf{Pr}[X = Y = w] = \min(P(w)\,, Q(w))$.

Couplings are closely related to TV distance. For any coupling (X,Y) between P and Q, $d_{\mathrm{TV}}(P,Q) \le \mathbf{Pr}[X \neq Y]$. Additionally, for an optimal coupling as defined above, $\mathbf{Pr}[X \neq Y]$ exactly equals $d_{\mathrm{TV}}(P,Q)$.

Therefore, a natural strategy to compute $d_{\mathrm{TV}}(P,Q)$ is to use the probabilistic inference oracle to compute $\mathbf{Pr}_{\mathcal{O}}[X \neq Y]$ over an optimal coupling $\mathcal{O}$ of P and Q. Indeed, assuming for simplicity that the alphabet size $\ell = 2$,

$$\mathbf{Pr}[X \neq Y] = 1 - \mathbf{Pr}[(X_i, Y_i) \in \{(1,1),(2,2)\} \text{ for all } i]\,,$$

and so, if $\mathcal{O}$ was also a Bayes net over the same DAG as P and Q, we could use the given probabilistic inference oracle to exactly compute $\mathbf{Pr}_{\mathcal{O}}[X \neq Y]$.

Perhaps surprisingly at first glance, optimal couplings generally do not have the same structure as the base distributions. In fact, even for n-variate *product distributions* P and Q, optimal couplings do not factorize. A natural candidate for an optimal coupling of product distributions is a *local coupling*, a joint distribution $\mathcal{L}$ on $(X,Y) = (X_1,\ldots,X_n,Y_1,\ldots,Y_n)$ where each (X_i,Y_i) is independently sampled from an optimal coupling of the i-th marginals of P and Q. However, local couplings are generally[1] not optimal.

In a very recent work, Feng, Guo, Jerrum, and Wang [FGJW23] showed that, when P and Q are product distributions, approximating $\mathbf{Pr}_{\mathcal{L}}[X \neq Y]$ nevertheless always leads to a good approximation for $\mathbf{Pr}_{\mathcal{O}}[X \neq Y]$ where $\mathcal{L}$ and $\mathcal{O}$ are local and optimal couplings respectively of P and Q. More precisely, denoting by α the ratio $\mathbf{Pr}_{\mathcal{O}}[X \neq Y]/\,\mathbf{Pr}_{\mathcal{L}}[X \neq Y]$, what [FGJW23] showed are that for any two n-dimensional product distributions P and Q:

(i) α is at least $\Omega(1/n)$;

(ii) there is an unbiased estimator $\hat{\alpha} \in [0,1]$ of α that can be efficiently evaluated.

The Chernoff bound then implies that averaging $O(n)$ independent copies of $\hat{\alpha}$ gives a relative approximation of α. Since $\mathbf{Pr}_{\mathcal{L}}[X \neq Y] = 1 - \prod_{i=1}^{n}(1 - d_{\mathrm{TV}}(P_i,Q_i))$ is easy to compute, an FPRAS for $d_{\mathrm{TV}}(P,Q) = \alpha \cdot \mathbf{Pr}_{\mathcal{L}}[X \neq Y]$ follows for product distributions.

Generalizing this approach to Bayes nets over general DAGs poses several challenges. The main issues arise even when we consider very simple Bayes nets—directed path graphs. Let P

[1] For example, say $P = \mathsf{Ber}(2/3) \otimes \mathsf{Ber}(2/3)$, while $Q = \mathsf{Ber}(1/3) \otimes \mathsf{Ber}(1/3)$. Here, if $\mathcal{O}$ is optimal, $\Pr_{\mathcal{O}}[(X_1,X_2) \neq (Y_1,Y_2)] = d_{\mathrm{TV}}(P,Q) = 1/3$, while if $\mathcal{L}$ is local, $\Pr_{\mathcal{L}}[(X_1,X_2) \neq (Y_1,Y_2)] = 1 - \Pr_{\mathcal{L}}[X_1 = Y_1] \cdot \Pr_{\mathcal{L}}[X_2 = Y_2] = 1 - (1 - d_{\mathrm{TV}}(\mathsf{Ber}(2/3), \mathsf{Ber}(1/3)))^2 = 5/9$.

and Q be Bayes nets over a directed path of length n. That is, for both P and Q, the i-th and $(i-2)$-th marginals are conditionally independent given the $(i-1)$-th marginal. In terms of factorizations, for all $w \in [\ell]^n$:

$$P(w) = \prod_{i=1}^{n} \mathop{\mathbf{Pr}}_{P}[w_i|w_{i-1}] \qquad \text{and} \qquad Q(w) = \prod_{i=1}^{n} \mathop{\mathbf{Pr}}_{Q}[w_i|w_{i-1}].$$

The inputs to the distance estimation problem are the conditional probability distributions $P_{i|i-1}(b|c) = \mathbf{Pr}_P[w_i = b|w_{i-1} = c]$ and $Q_{i|i-1}(b|c) = \mathbf{Pr}_Q[w_i = b|w_{i-1} = c]$ for all $i \in [n]$ and $c \in [\ell]$. The goal is to output an $(1+\varepsilon)$-approximation of $d_{\mathrm{TV}}(P, Q)$ with probability at least $1 - \delta$.

As in the case of product distributions, suppose we seek a coupling $\mathcal{L}$ of P and Q that also forms a Bayes net over the directed path. In other words, we would like a coupling $\mathcal{L}$ generating $(X_1, \ldots, X_n, Y_1, \ldots, Y_n)$ such that each (X_i, Y_i) is independent of (X_{i-2}, Y_{i-2}) conditioned on (X_{i-1}, Y_{i-1}). However, there is an immediate problem: Namely, X_i and X_{i-2} may be dependent given X_{i-1} through the path $X_{i-2} \to Y_{i-1} \to X_i$, and similarly Y_i and Y_{i-2} may be dependent given Y_{i-1} through the path $Y_{i-2} \to X_{i-1} \to Y_i$. Hence, it may not be possible[2] to ensure that $(X_1, \ldots, X_n)$ form a copy of P and $(Y_1, \ldots, Y_n)$ form a copy of Q, as is required for a coupling.

Our main conceptual innovation is that we *drop the requirement that $\mathcal{L}$ forms a coupling* and allow $\mathcal{L}$ to be a *local partial coupling*. A local partial coupling of P and Q is a distribution $\mathcal{L}$ over $(X, Y) \in [\ell]^n \times [\ell]^n$ satisfying the following three properties:

(i) $\mathcal{L}$ is a Bayes net over a directed path of length n with marginals (X_i, Y_i) at node i;

(ii) $X \sim P$;

(iii) for any $b, c_1, c_2 \in [\ell]$, it is the case that

$$\mathbf{Pr}[X_i = Y_i = b|X_{i-1} = c_1, Y_{i-1} = c_2] = \min(P_{i|i-1}(b|c_1), Q_{i|i-1}(b|c_2)).$$

Note that the above conditions do not place a condition on the distribution of Y. When P and Q are arbitrary Bayes net distributions described via DAGs, these conditions can be generalized from the path to general DAGs in a straightforward manner.

Such an $\mathcal{L}$ can always be constructed by using (iii) to define the conditional probability of $X_i = Y_i = b$ and by adjusting the rest of the probability mass to ensure that for all b, c_1, c_2:

$$\mathbf{Pr}[X_i = b|X_{i-1} = c_1, Y_{i-1} = c_2] = P_{i|i-1}(b|c_1).$$

The fact that P is a Bayes net on a path then implies that $X \sim P$.

Define $Z = \mathbf{Pr}_{\mathcal{L}}[X \neq Y]$ for $\mathcal{L}$ as above. Since $\mathcal{L}$ is a path Bayes net, as mentioned earlier, we can use probabilistic inference over the path (by classic variable elimination) to efficiently compute Z exactly. The main technical result we establish about Z is that

$$Z \leq 2n \cdot d_{\mathrm{TV}}(P, Q).$$

Our proof of this inequality is elementary but crucially uses properties (ii) and (iii) in the definition of $\mathcal{L}$ above. Now we can follow the same strategy as [FGJW23] by defining an unbiased estimator $\hat{\alpha}$ for $\alpha = d_{\mathrm{TV}}(P, Q)/Z$ and empirically estimating the expectation of $\hat{\alpha}$.

To define this estimator, we observe that we can define $d_{\mathrm{TV}}(P, Q) = \sum_w g^*(w)$ while $Z = \sum_w g(w)$, where

$$g^*(w) = P(w) - \min(P(w), Q(w)), \qquad g(w) = P(w) - \prod_i \min(P_{i|i-1}(w_i|w_{i-1}), Q_{i|i-1}(w_i|w_{i-1})).$$

[2]Note that this issue does not arise for product distributions as there are no paths to speak of.

Here, $0 \leq g^*(w) \leq g(w)$ for all w. We then use a classic importance sampling technique (mentioned, e.g., in [CS97]) to estimate ratios of sums:

$$\frac{\sum_w g^*(w)}{\sum_w g(w)} = \mathop{\mathbf{E}}_{w \sim \pi}\left[\frac{g^*(w)}{g(w)}\right]$$

where the distribution π has mass function $\pi(w) = g(w)/\sum_w g(w)$. Using the fact that $g^*(w)/g(w)$ lies in the interval $[0,1]$ and has expectation over π equal to $\alpha \geq 1/2n$, we can conclude our analysis by the Chernoff bound. This approach requires that we are able to sample efficiently from π, which we show is possible using calls to a probabilistic inference oracle.

2.2 Proofs of the rest of the results

We outline here the main proof ideas of the rest of our results.

Proof of Corollary 3. The proof of Corollary 3 is an application of Theorem 1. To make use of Theorem 1, we establish that probabilistic inference (i.e., computing $\mathbf{Pr}[X_1 \in S_1, \ldots, X_n \in S_n]$) can be efficiently implemented for Bayes nets of constant alphabet size and logarithmic treewidth (Lemma 26). It is known that a tree decomposition of graphs that have logarithmic treewidth can be computed in polynomial time [RS84]. The variable elimination algorithm of [ZP94] shows that inference can be done in polynomial time given a tree decomposition, provided that the treewidth of the Bayes net is logarithmic.

Proof of Theorem 4. Theorem 4 is proved by showing a reduction from #SAT to computing the TV distance between an appropriately defined Bayes net and the uniform distribution. This is achieved by creating a Bayes net that captures the circuit structure of a Boolean formula F of which we want to compute its number of satisfying assignments. The CPTs of this Bayes net mimic the function of the logical gates (AND, OR, NOT) of F.

Proof of Theorem 5. Theorem 5 is proved by giving an algorithm that exploits the following property of TV distance. Let P be a Bayes net over n variables that has maximum in-degree d and alphabet size ℓ. In this case $d_{\mathrm{TV}}(P, \mathbb{U})$ is equal to

$$\begin{aligned}\frac{1}{2}\sum_x |P(x) - \mathbb{U}(x)| &= \sum_x \max(0, P(x) - \mathbb{U}(x)) = \sum_x \mathbb{U}(x)\max\left(0, \frac{P(x)}{\mathbb{U}(x)} - 1\right) \\ &= \mathop{\mathbf{E}}_{x \sim \mathbb{U}}\left[\max\left(0, \frac{P(x)}{\mathbb{U}(x)} - 1\right)\right] = \mathop{\mathbf{E}}_{x \sim \mathbb{U}}[\max(0, P(x)\,\ell^n - 1)]\,.\end{aligned}$$

This yields a natural estimator for $d_{\mathrm{TV}}(P, \mathbb{U})$, whereby we draw samples $x_1, \ldots, x_m \sim \mathbb{U}$ and then compute and output

$$\frac{1}{m}\sum_{i=1}^{m} \max(0, P(x_i)\,\ell^n - 1)\,.$$

The crux of our analysis is to show that the quantity $\max(0, P(x)\,\ell^n - 1)$ is between 0 and $1 + O\Big(d_{\mathrm{TV}}(P, \mathbb{U})\,\ell^{d+1} n\Big)$. This enables us to use a value of m that is in $O\Big(\mathrm{poly}\Big(n\ell^d, 1/\varepsilon, \log(1/\delta)\Big)\Big)$, whereby ε is the accuracy error and δ is the confidence error of the FPRAS. Note that the running time is polynomial in the input length, as any description of the Bayes net P has size at least $n + \ell^{d+1}$.

3 Preliminaries

We use $[n]$ to denote the set $\{1, \ldots, n\}$ and log to denote $\log_2$. Throughout the paper, we shall assume that all probabilities are represented as rational numbers of the form a/b. We denote the uniform distribution by $\mathbb{U}$.

The following concentration inequality will be useful in our proofs.

Lemma 7 (Hoeffding's inequality)**.** *Let $X_1, \ldots, X_n$ be independent random variables such that $a_i \leq X_i \leq b_i$ for all $1 \leq i \leq n$. Then*

$$\mathbf{Pr}\left[\left|\sum_{i=1}^{n} X_i - \mathbf{E}\left[\sum_{i=1}^{n} X_i\right]\right| \geq t\right] \leq 2\exp\left(-\frac{2t^2}{\sum_{i=1}^{n}(b_i - a_i)^2}\right).$$

We shall use the following notion of an approximation algorithm.

Definition 8 (FPRAS)**.** A function $f : \{0,1\}^* \to \mathbb{R}$ admits a *fully polynomial-time randomized approximation scheme (FPRAS)* if there is a *randomized* algorithm $\mathcal{A}$ such that for all n and all inputs $x \in \{0,1\}^n$, $\varepsilon > 0$, and $\delta > 0$, $\mathcal{A}$ outputs a $(1+\varepsilon)$-relative approximation of $f(x)$, i.e., a value v that lies in the interval $[f(x)/(1+\varepsilon), (1+\varepsilon)f(x)]$, with probability $1-\delta$. The running time of $\mathcal{A}$ is polynomial in $n, 1/\varepsilon, 1/\delta$.

3.1 Bayes nets

For a directed acyclic graph (DAG) G and a node v in G, let $\Pi(v)$ denote the set of parents of v.

Definition 9 (Bayes nets)**.** A *Bayes net* is specified by a directed acyclic graph (DAG) over a vertex set $[n]$ and a collection of probability distributions over symbols in $[\ell]$, as follows. Each vertex i is associated with a random variable X_i whose range is $[\ell]$. Let $\Pi(X_i)$ denote the set of random variables associated with $\Pi(i)$ (by overloading the notation Π). Each node i of G has a CPT (Conditional Probability Table) that describes the following: For every $x \in [\ell]$ and every $y \in [\ell]^k$, where k is the size of $\Pi(i)$, the CPT has the value of $\Pr[X_i = x | \Pi(X_i) = y]$ stored. Given such a Bayes net, its associated probability distribution P is given by the following: For all $x \in [\ell]^n$,

$$P(x) = \Pr_P[X = x] = \prod_{i=1}^{n} \Pr_P\left[X_i = x_i | X_{\Pi(X_i)} = x_{\Pi(X_i)}\right].$$

Here X is the joint distribution $(X_1, \ldots, X_n)$ and $x_{\Pi(X_i)}$ is the projection of x to the indices in $\Pi(i)$.

Note that $P(x)$ can be computed in linear time by using the CPTs of P to retrieve the probabilities $\mathbf{Pr}_P\left[X_i = x_i | X_{\Pi(X_i)} = x_{\Pi(X_i)}\right]$.

An important and useful notion is that of the moralization of a Bayes net.

Definition 10 (Moralization of Bayes nets)**.** Let B be a Bayes net over a DAG G. The *moralization of B* is the undirected graph that is obtained from G as follows. For every node u of G and any pair (v, w) of its parents $\Pi(u)$ if v and w are not connected by some edge in G, then add the edge (v, w). (Note that after this step the parents of every node of G form a clique.) Finally, make all edges of G undirected.

We shall require the following simple observation.

Lemma 11. *Given a Bayes net over n nodes, its moralization can be computed in time $O(\text{poly}(n))$.*

Proof. Let B be a Bayes net over a DAG G that has n nodes. Let v be a node of G and let $\Pi(v)$ be the set of the parents of v. We can construct a clique among the nodes of $\Pi(v)$ in time $O(n^2)$, since $|\Pi(v)| \leq n$. Therefore we can construct all of the required cliques in time $n \cdot O(n^2) = O(n^3)$. Finally, we can make all directed edges of G undirected in time $O(n^2)$. This yields a total running time of $O(n^3)$. □

3.2 Total variation distance

The following notion of distance is central in this work.

Definition 12 (Total variation distance)**.** For probability distributions P, Q over a finite sample space $\mathcal{D}$, the *total variation distance* of P and Q is

$$d_{\mathrm{TV}}(P, Q) = \max_{S \subseteq \mathcal{D}} (P(S) - Q(S)) \,.$$

Note that $d_{\mathrm{TV}}(P, Q)$ also equals $\frac{1}{2} \sum_{w \in \mathcal{D}} |P(w) - Q(w)| = \sum_{w \in \mathcal{D}} \max(0, P(w) - Q(w))$.

3.3 Probabilistic inference

The notions of probabilistic inference and probabilistic inference oracle (for Bayes nets) are central in this work.

For us, *probabilistic inference* is the following computational task: Given (a representation of) random variables $X_1, \ldots, X_n$ and (a representation of) sets $S_1, \ldots, S_n$ such that for all i the set S_i is a subset of the range of X_i, compute the probability $\mathbf{Pr}[X_1 \in S_1, \ldots, X_n \in S_n]$.[3]

Let us now define probabilistic inference (oracle) queries.

Definition 13 (Probabilistic inference query over Bayes nets)**.** A *probabilistic inference query* takes a description of a Bayes net distribution P over n nodes and alphabet size ℓ and descriptions of sets $S_1, \ldots, S_n$, where for all $1 \leq i \leq n$, $S_i \subseteq [\ell]$, and returns in time $O(1)$ the value of $\mathbf{Pr}_P[X_1 \in S_1, \ldots, X_n \in S_n]$.

3.4 Treewidth and tree decompositions

We require the definition of treewidth.

Definition 14. A *tree decomposition of an undirected graph* $G = (V, E)$ is a tree T with nodes $X_1, \ldots, X_n$, where each X_i is a subset of V, satisfying the following properties (the term node is used to refer to a vertex of T to avoid confusion with vertices of G):

1. The union of all sets X_i equals V. That is, each graph vertex is contained in at least one tree node.

2. If X_i and X_j both contain a vertex v, then all nodes X_k of T in the (unique) path between X_i and X_j contain v as well. Equivalently, the tree nodes containing vertex v form a connected subtree of T.

3. For every edge (v_1, v_2) in the graph, there is a subset X_i that contains both v_1 and v_2. That is, vertices are adjacent in the graph only when the corresponding subtrees have a node in common.

[3] Note that a notion of probabilistic inference that has previously been considered [KBvdG10] is the following: Given random variables $X_1, \ldots, X_n$, a set $I = \{i_1, \cdots, i_k\} \subseteq [n]$, values $x_{i_1}, \ldots, x_{i_k}$ that belong to the ranges of $X_{i_1}, \ldots, X_{i_k}$, respectively, and an event E, compute the probability $\mathbf{Pr}[(X_{i_1}, \ldots, X_{i_k}) = (x_{i_1}, \ldots, x_{i_k}) \,|E]$.

The *width of a tree decomposition* is the size of its largest set X_i minus one. The *treewidth* $\mathrm{tw}(G)$ *of a graph* G is the minimum width among all possible tree decompositions of G.

We shall also extend the notion of treewidth to Bayes nets, as follows.

Definition 15. The *treewidth of a Bayes net* is defined to be equal to the treewidth of its moralization.

We require the following two theorems, Theorem 16 and Theorem 17, respectively; Theorem 16 is about a tree decomposition algorithm and Theorem 17 is about the variable elimination algorithm.

Theorem 16 (Tree decomposition [RS84])**.** *There is a* $O(w3^{3w}n^2)$*-time algorithm that finds a tree decomposition of width* $4w+1$*, if the treewidth of the input graph is at most* w*.*

We will make use of the variable elimination algorithm to efficiently implement probabilistic inference queries for bounded treewidth Bayes nets.

Theorem 17 (Variable elimination; following Zhang and Poole [ZP94])**.** *There is an algorithm, called the* variable elimination algorithm*, for the following task: Given a Bayes net* B *over variables* $X_1, \ldots, X_n \in [\ell]$*, sets* $S_1, \ldots, S_n \subseteq [\ell]$*, the moralization* M_B *of* B*, and a tree decomposition* $\mathcal{T}$ *of width* w *of* M_B*, compute the probability* $\mathbf{Pr}_B[X_1 \in S_1, \ldots, X_n \in S_n]$*. The running time of this algorithm is* $O(n\ell^w)$*.*

4 Structure-preserving reduction from TV distance estimation to probabilistic inference

In this section, we prove Theorem 1 and Corollary 3. In the following, let $T(G, \ell)$ be the running time of some implementation of a probabilistic inference oracle for a Bayes net over a DAG G that has alphabet size ℓ.

Theorem 1 (Formal)**.** *There is a polynomial-time randomized algorithm that takes a DAG* G*, two Bayes nets* P *and* Q *over* G *(as CPTs) that have alphabet size* ℓ*, and parameters* ε, δ *as inputs and behaves as follows.*

The algorithm makes probabilistic inference queries for a Bayes net over the same DAG G *that has alphabet size* ℓ^2 *and outputs an* $(1+\varepsilon)$*-relative approximation of* $d_{\mathrm{TV}}(P, Q)$ *with probability at least* $1-\delta$*. The running time of this algorithm is* $T(G, \ell^2) \cdot O(n^3\varepsilon^{-2}\ell \log \delta^{-1})$ *and the number of its probabilistic inference queries is* $O(n^3\varepsilon^{-2}\ell \log \delta^{-1})$*.*

The rest of the section is devoted to proving Theorem 1 and is organized as follows. We first introduce the ingredients that are necessary for describing the algorithm. In Section 4.1, we show how the algorithm can be implemented using probabilistic inference queries. Finally, in Section 4.2 we establish its correctness.

Let P and Q be two Bayes net distributions defined over a DAG G with n nodes and alphabet $[\ell]$. Without loss of generality, assume that the nodes are topologically ordered as in the sequence $1, 2, \ldots, n$.

Our approach is to define an estimator function f and a distribution π so that $\mathbf{E}_\pi[f] = d_{\mathrm{TV}}(P, Q)/Z$ where Z is a normalization constant. The algorithm proceeds by estimating $\mathbf{E}_\pi[f]$, multiplies it by Z, and returns the value. The algorithm uses probabilistic inference queries to compute Z and to sample from the distribution π.

Let w be an element of the sample space, i.e, a n-symbol string over $[\ell]$. Given $1 \le i \le n$, $\Pi(i)$ denotes the set of parents of i in G and let $w_{\Pi(i)}$ denote the projection w at the parents of node i in G. We first define a function h over $[\ell]^n \times [n]$ as follows:

$$h(w,i) := \min\Big(P_{i|\Pi(i)}\Big(w_i|w_{\Pi(i)}\Big), Q_{i|\Pi(i)}\Big(w_i|w_{\Pi(i)}\Big)\Big).$$

Descriptions of f, Z, and π. The *estimator function* f is defined as follows:

$$f(w) := \frac{\max(0, P(w) - Q(w))}{g(w)} \qquad \text{where} \qquad g(w) := P(w) - \prod_{i=1}^{n} h(w,i)$$

for all w. It is straightforward to show that f is computable in time $O(n)$. We define $Z := \sum_{w\in[\ell]^n} g(w)$ to be a normalization constant. Finally, the distribution π is specified by the probability function $\pi(w) := g(w)/Z$ for all w.

Description of $\mathcal{L}$. We now define a Bayes net distribution $\mathcal{L}$ over the graph G which is used to make inference queries by the algorithm. The distribution $\mathcal{L}$ is over the alphabet $[\ell]^2$ and is a joint distribution (X, Y) where X and Y take value over $[\ell]^n$. We specify a CPT for (X, Y). For this, we need to specify for every i and $b, z \in [\ell]$ the probability $\Pr[(X_i, Y_i) = (b, z)]$ conditioned on the values $\Pi(i)$ take. We will first describe the probability where both X_i and Y_i take the same value b. For every $c_1, c_2 \in [\ell]$,

$$\mathbf{Pr}\Big[(X_i, Y_i) = (b,b) \mid \Big(X_{\Pi(i)}, Y_{\Pi(i)}\Big) = (c_1, c_2)\Big] = \min\Big(P_{i|\Pi(i)}(b|c_1), Q_{i|\Pi(i)}(b|c_2)\Big).$$

Define the remaining probabilities to ensure that the marginal X is distributed according to P. That is, for every $z \neq b$ assign $\mathbf{Pr}\Big[(X_i, Y_i) = (b, z) \mid \Big(X_{\Pi(i)}, Y_{\Pi(i)}\Big) = (c_1, c_2)\Big]$ so that the following holds:

$$\begin{aligned}\sum_{z:z\neq b} &\mathbf{Pr}\Big[(X_i, Y_i) = (b, z) \mid \Big(X_{\Pi(i)}, Y_{\Pi(i)}\Big) = (c_1, c_2)\Big] \\ &= P_{i|\Pi(i)}(b|c_1) - \min\Big(P_{i|\Pi(i)}(b|c_1), Q_{i|\Pi(i)}(b|c_2)\Big)\end{aligned}$$

Now we are ready to describe the algorithm (see Algorithm 1).

Algorithm 1 FPRAS for d_{TV} estimation using a probabilistic inference oracle.

Require: Bayes nets P, Q over DAG G with n nodes, parameters ε, δ.

Ensure: The output Est is an $(1+\varepsilon)$-approximation of $d_{\mathrm{TV}}(P, Q)$, with probability at least $1 - \delta$.

Construct the Bayes net distribution $\mathcal{L}$ over G
Compute Z by making one *probabilistic inference query* using $\mathcal{L}$
$m \leftarrow Cn^2\varepsilon^{-2}\log\delta^{-1}$ (for some sufficiently large $C > 0$)
$F \leftarrow 0$
for $i \leftarrow 1$ **to** m **do**
    Sample $w^i \sim \pi$ by making *probabilistic inference queries* using $\mathcal{L}$
    $F \leftarrow F + f(w^i)$
end for
$\mathsf{Est} \leftarrow ZF/m$
return Est

4.1 Implementing the algorithm with probabilistic inference queries

This subsection is devoted to showing that the sampling from the distribution π and the computation of the normalization constant Z can be done by making probabilistic inference queries. Recall that $\mathcal{L}$ is joint distribution (X, Y). We start with the following crucial observation which states that the marginal X (in $\mathcal{L}$) is distributed according to the distribution P.

Observation 18. *For every $b, c_1, c_2 \in [\ell]$,*

$$\mathbf{Pr}\Big[X_i = b | \Big(X_{\Pi(i)}, Y_{\Pi(i)}\Big) = (c_1, c_2)\Big] = P_{i|\Pi(i)}(b|c_1)\,.$$

Proof. We have

$$\begin{aligned}
\mathbf{Pr}\Big[X_i = b | \Big(X_{\Pi(i)}, Y_{\Pi(i)}\Big) = (c_1, c_2)\Big] &= \sum_{z\in[\ell]} \mathbf{Pr}\Big[(X_i, Y_i) = (b, z) \mid \Big(X_{\Pi(i)}, Y_{\Pi(i)}\Big) = (c_1, c_2)\Big] \\
&= \mathbf{Pr}\Big[(X_i, Y_i) = (b, b) \mid \Big(X_{\Pi(i)}, Y_{\Pi(i)}\Big) = (c_1, c_2)\Big] \\
&\quad + \sum_{z:z\neq b} \mathbf{Pr}\Big[(X_i, Y_i) = (b, z) \mid \Big(X_{\Pi(i)}, Y_{\Pi(i)}\Big) = (c_1, c_2)\Big] \\
&= \min\Big(P_{i|\Pi(i)}(b|c_1)\,, Q_{i|\Pi(i)}(b|c_2)\Big) \\
&\quad + P_{i|\Pi(i)}(b|c_1) - \min\Big(P_{i|\Pi(i)}(b|c_1)\,, Q_{i|\Pi(i)}(b|c_2)\Big) \\
&= P_{i|\Pi(i)}(b|c_1)\,.
\end{aligned}$$

□

Therefore, X factorizes like P with its conditional probabilities matching that of P and hence $X \sim P$. This realizes the notion of a local partial coupling as was earlier discussed in Section 2.1 and satisfies all three properties: (i) $\mathcal{L}$ is a Bayes net distribution over the same DAG G (that is used to describe distributions P and Q), (ii) $X \sim P$, and (iii) in the joint distribution (X, Y), the conditional probabilities are equal to the minimum of the two conditional probabilities associated to P and Q as it is the case in standard couplings.

In Claim 19 we relate the normalization constant Z of the distribution π to the marginals X and Y of the distribution $\mathcal{L}$. Moreover, we also relate the generalized normalization constant

$$Z_{b_1,\ldots,b_k} := \sum_{w:(w_1,\ldots,w_k)=(b_1,\ldots,b_k)} g(w)\,,$$

for $b_1, \ldots, b_k \in [\ell]$, to the marginals X and Y of the distribution $\mathcal{L}$. We need this generalized normalization constant to show that sampling from the distribution π (Claim 21) can be efficiently done via probabilistic inference queries.

Claim 19. *It is the case that*

$$Z = \mathbf{Pr}[X \neq Y] \qquad \text{and} \qquad Z_{b_1,\ldots,b_k} = \mathbf{Pr}[X \neq Y, X_1 = b_1, \ldots, X_k = b_k]$$

for any $b_1, \ldots, b_k \in [\ell]$.

Proof. Since $X \sim P$ and for that matter $P(w) = \mathbf{Pr}[X = w]$, we have

$$\begin{aligned}
g(w) &= P(w) - \prod_{i=1}^{n} \min\Big(P_{i|\Pi(i)}\Big(w_i|w_{\Pi(i)}\Big)\,, Q_{i|\Pi(i)}\Big(w_i|w_{\Pi(i)}\Big)\Big) \\
&= P(w) - \prod_{i=1}^{n} \mathbf{Pr}\Big[(X_i, Y_i) = (w_i, w_i) \mid \Big(X_{\Pi(i)}, Y_{\Pi(i)}\Big) = \Big(w_{\Pi(i)}, w_{\Pi(i)}\Big)\Big]
\end{aligned}$$

$$
\begin{aligned}
&= P(w) - \mathbf{Pr}[X = Y = w] \\
&= \mathbf{Pr}[X = w] - \mathbf{Pr}[X = Y = w] \\
&= \mathbf{Pr}[X = w] - \mathbf{Pr}[Y = w | X = w] \cdot \mathbf{Pr}[X = w] \\
&= \mathbf{Pr}[X = w] \left(1 - \mathbf{Pr}[Y = w | X = w]\right) \\
&= \mathbf{Pr}[X = w] \, \mathbf{Pr}[Y \neq w | X = w] \\
&= \mathbf{Pr}[X = w, Y \neq w] \,.
\end{aligned}
$$

Therefore, we have that

$$Z = \sum_w g(w) = \mathbf{Pr}[X \neq Y]$$

and

$$
\begin{aligned}
Z_{b_1,\ldots,b_k} &= \sum_{w:(w_1,\ldots,w_k)=(b_1,\ldots,b_k)} g(w) \\
&= \sum_{w:(w_1,\ldots,w_k)=(b_1,\ldots,b_k)} \mathbf{Pr}[X = w, Y \neq w] \\
&= \mathbf{Pr}[X \neq Y, X_1 = b_1, \ldots, X_k = b_k] \,.
\end{aligned}
$$

□

The following claim says that Z and $Z_{b_1,\ldots,b_k}$ can be easily computed given access to a probabilistic inference oracle for $\mathcal{L}$.

Claim 20. *It is the case that Z and $Z_{b_1,\ldots,b_k}$ (for any $b_1, \ldots, b_k \in [\ell]$) can be computed in time $O(1)$ by making $O(1)$ probabilistic inference queries to the Bayes net distribution $\mathcal{L}$.*

Proof. Note that $Z = \mathbf{Pr}[X \neq Y]$ is equal to $1 - \mathbf{Pr}[X = Y]$. Therefore it suffices to compute $\mathbf{Pr}[X = Y]$ by using a probabilistic inference oracle. This can done by observing that $\mathbf{Pr}[X = Y]$ is equal to $\mathbf{Pr}[(X_1, Y_1) \in S_1, \ldots, (X_n, Y_n) \in S_n]$ for $S_1 = \cdots = S_n = \{(1,1), \ldots, (\ell, \ell)\}$.

Now note that $Z_{b_1,\ldots,b_k} = \mathbf{Pr}[X \neq Y, X_1 = b_1, \ldots, X_k = b_k]$ is equal to

$$\mathbf{Pr}[X_1 = b_1, \ldots, X_k = b_k] - \mathbf{Pr}[X = Y, X_1 = b_1, \ldots, X_k = b_k] \,.$$

What is left is to show how to compute these two probabilities by using a probabilistic inference oracle. We have that $\mathbf{Pr}[X_1 = b_1, \ldots, X_k = b_k]$ is equal to $\mathbf{Pr}[(X_1, Y_1) \in S_1, \ldots, (X_n, Y_n) \in S_n]$ for $S_i = \{(b_i, 1), \ldots, (b_i, \ell)\}$ for all $1 \leq i \leq k$ and $S_{k+1} = \cdots = S_n = [\ell]^2$.

Similarly, we have that

$$\mathbf{Pr}[X = Y, X_1 = b_1, \ldots, X_k = b_k] = \mathbf{Pr}[(X_1, Y_1) \in S_1, \ldots, (X_n, Y_n) \in S_n]$$

for $S_i = \{(b_i, b_i)\}$ for all $1 \leq i \leq k$ and $S_{k+1} = \cdots = S_n = \{(1,1), \ldots, (\ell, \ell)\}$. □

We will now show that probabilistic inference queries allow for efficient sampling from π.

Claim 21. *Sampling from the distribution π can be implemented in time $O(n\ell)$ by making $O(n\ell)$ probabilistic inference queries.*

Proof. We describe how to sample from π iteratively, symbol by symbol. Assume that we have sampled the first $k-1$ symbols, that is, assume that we have already sampled $w_1, \ldots, w_{k-1}$ to be equal to $b_1, \ldots, b_{k-1} \in [\ell]$. We describe now how to sample w_k. For every possible value $b \in [\ell]$ of w_k, we compute the marginal

$$\mu_b := \pi(b_1, \ldots, b_{k-1}, b) = \frac{\sum_{w:(w_1 \cdots w_k)=(b_1 \cdots b_{k-1} b)} g(w)}{Z} = \frac{Z_{b_1 \cdots b_{k-1} b}}{Z}$$

by two invocations of Claim 20. Then, we sample w_k based on the values $\{\mu_b\}_{b=1}^{\ell}$.

Let $S(n)$ be the number of steps to sample n symbols from π. The above procedure gives the recurrence relation $S(n) = O(\ell) + S(n-1)$ which yields $S(n) = O(n\ell)$. Since we perform at most two probabilistic inference queries for every coordinate and every symbol, the total number of probabilistic inference queries is equal to $S(n) = O(n\ell)$. □

4.2 Analysis of the algorithm

Next, we establish some useful properties of the function f and the distribution π.

Claim 22. *For any w, it is the case that $0 \leq f(w) \leq 1$.*

Proof. We separately show $0 \leq f(w)$ and $f(w) \leq 1$. To establish $0 \leq f(w)$, since the numerator is non-negative, it suffices to show that $g(w) = P(w) - \prod_{i=1}^{n} h(w,i) \geq 0$ or equivalently $P(w) \geq \prod_{i=1}^{n} h(w,i)$.

We have

$$\begin{aligned} P(w) &= \prod_{i=1}^{n} P_{i|\Pi(i)}\left(w_i | w_{\Pi(i)}\right) \\ &\geq \prod_{i=1}^{n} \min\left(P_{i|\Pi(i)}\left(w_i|w_{\Pi(i)}\right), Q_{i|\Pi(i)}\left(w_i|w_{\Pi(i)}\right)\right) = \prod_{i=1}^{n} h(w,i) \end{aligned}$$

by the definition of h.

For showing $f(w) \leq 1$, it suffices to show that $P(w) - Q(w) \leq g(w)$ (since $0/g(w) = 0 \leq 1$). Since $g(w) = P(w) - \prod_{i-1}^{n} h(w,i)$, it suffices to show that $Q(w) \geq \prod_{i=1}^{n} h(w,i)$. An argument identical to the above, where we showed that $P(w) \geq \prod_{i=1}^{n} h(w,i)$, will show this. □

We next relate the expected value of the function f with respect to the distribution π to $d_{\mathrm{TV}}(P,Q)$.

Claim 23. *It is the case that $\mathbf{E}_{\pi}[f(w)] = d_{\mathrm{TV}}(P,Q)/Z$.*

Proof. We have that $\mathbf{E}_{\pi}[f(w)]$ is equal to

$$\begin{aligned} \mathop{\mathbf{E}}_{\pi}\left[\frac{\max(0, P(w) - Q(w))}{g(w)}\right] &= \sum_{w} \pi(w) \frac{\max(0, P(w) - Q(w))}{g(w)} \\ &= \sum_{w} \frac{g(w)}{Z} \frac{\max(0, P(w) - Q(w))}{g(w)} \\ &= \frac{1}{Z} \sum_{w} \max(0, P(w) - Q(w)) \\ &= \frac{d_{\mathrm{TV}}(P,Q)}{Z}. \end{aligned}$$

□

We need the following claim that ensures the estimand is large enough to facilitate Monte Carlo sampling.

Lemma 24. *It is the case that $Z \leq 2n \cdot d_{\mathrm{TV}}(P,Q)$.*

Proof. By Claim 19, it suffices to show that $\mathbf{Pr}[X \neq Y] \leq 2n \cdot d_{\mathrm{TV}}(P,Q)$. We split the event $(X \neq Y)$ into n disjoint events $\{E_i\}_{i=1}^{n}$. Without loss of generality, assume that $1, 2, \ldots, n$ is the

topological ordering of the vertices of G. Event E_i is defined as $(\bigwedge_{1\le j\le i-1} X_j = Y_j) \wedge (X_i \neq Y_i)$. Note that the E_i's are disjoint. Thus $\mathbf{Pr}[X \neq Y] = \sum_i \mathbf{Pr}[E_i]$. We have that

$$\mathbf{Pr}[E_i] \le \mathbf{Pr}\left[(X_i \neq Y_i) \wedge (X_{\Pi(i)} = Y_{\Pi(i)})\right] = \sum_{\sigma} \mathbf{Pr}\left[(X_i \neq Y_i) \wedge (X_{\Pi(i)}, Y_{\Pi(i)}) = (\sigma,\sigma)\right]$$

where σ is an assignment for $\Pi(i)$ (note that the length of σ is equal to the in-degree of i). Henceforth, for notational brevity, we shall omit the dependence on i. Thus,

$$\begin{aligned}
\mathbf{Pr}[X \neq Y] &= \sum_i \mathbf{Pr}[E_i] \\
&\le \sum_i \sum_{\sigma} \mathbf{Pr}\Big[X_i \neq Y_i \wedge \Big(X_{\Pi(i)}, Y_{\Pi(i)}\Big) = (\sigma,\sigma)\Big] \\
&= \sum_i \sum_{\sigma} \mathbf{Pr}\Big[X_i \neq Y_i \mid \Big(X_{\Pi(i)}, Y_{\Pi(i)}\Big) = (\sigma,\sigma)\Big] \mathbf{Pr}\Big[\Big(X_{\Pi(i)}, Y_{\Pi(i)}\Big) = (\sigma,\sigma)\Big].
\end{aligned}$$

We require the following claim.

Claim 25. *For any σ, we have*

$$\mathbf{Pr}\Big[X_i \neq Y_i \mid \Big(X_{\Pi(i)}, Y_{\Pi(i)}\Big) = (\sigma,\sigma)\Big] = d_{\mathrm{TV}}\Big(P_{i|\Pi(i)}(\cdot|\sigma), Q_{i|\Pi(i)}(\cdot|\sigma)\Big).$$

Proof. We have that $\mathbf{Pr}\Big[X_i \neq Y_i \mid \Big(X_{\Pi(i)}, Y_{\Pi(i)}\Big) = (\sigma,\sigma)\Big]$ is equal to

$$\begin{aligned}
1 - \mathbf{Pr}\Big[X_i = Y_i \mid \Big(X_{\Pi(i)}, Y_{\Pi(i)}\Big) = (\sigma,\sigma)\Big] &= 1 - \sum_{c\in[\ell]} \mathbf{Pr}\Big[(X_i, Y_i) = (c,c) \mid \Big(X_{\Pi(i)}, Y_{\Pi(i)}\Big) = (\sigma,\sigma)\Big] \\
&= 1 - \sum_{c\in[\ell]} \min\Big(P_{i|\Pi(i)}(c|\sigma), Q_{i|\Pi(i)}(c|\sigma)\Big) \\
&= \sum_{c\in[\ell]} P_{i|\Pi(i)}(c|\sigma) - \sum_{c\in[\ell]} \min\Big(P_{i|\Pi(i)}(c|\sigma), Q_{i|\Pi(i)}(c|\sigma)\Big) \\
&= \sum_{c\in[\ell]} \Big(P_{i|\Pi(i)}(c|\sigma) - \min\Big(P_{i|\Pi(i)}(c|\sigma), Q_{i|\Pi(i)}(c|\sigma)\Big)\Big) \\
&= \sum_{c\in[\ell]} \max\Big(0, P_{i|\Pi(i)}(c|\sigma) - Q_{i|\Pi(i)}(c|\sigma)\Big) \\
&= d_{\mathrm{TV}}\Big(P_{i|\Pi(i)}(\cdot|\sigma), Q_{i|\Pi(i)}(\cdot|\sigma)\Big).
\end{aligned}$$

□

By Claim 25 we have that $\mathbf{Pr}[X \neq Y]$ is at most

$$\begin{aligned}
&\sum_i \sum_{\sigma} \mathbf{Pr}\Big[\Big(X_{\Pi(i)}, Y_{\Pi(i)}\Big) = (\sigma,\sigma)\Big] d_{\mathrm{TV}}\Big(P_{i|\Pi(i)}(\cdot|\sigma), Q_{i|\Pi(i)}(\cdot|\sigma)\Big) \\
&\quad\le \sum_i \sum_{\sigma} \mathbf{Pr}\Big[X_{\Pi(i)} = \sigma\Big] \frac{1}{2} \sum_c \Big|P_{i|\Pi(i)}(c|b) - Q_{i|\Pi(i)}(c|\sigma)\Big| \\
&\quad\le \sum_i \sum_{\sigma} P_{\Pi(i)}(\sigma) \frac{1}{2} \sum_c \Big|P_{i|\Pi(i)}(c|\sigma) - Q_{i|\Pi(i)}(c|\sigma)\Big| \quad \text{(since } X \sim P \text{ by Observation 18)} \\
&\quad= \sum_i \sum_{\sigma} \frac{1}{2} \sum_c \Big|P_{\Pi(i)}(\sigma)\, P_{i|\Pi(i)}(c|\sigma) - P_{\Pi(i)}(\sigma)\, Q_{i|\Pi(i)}(c|\sigma)\Big| \\
&\quad= \sum_i \sum_{\sigma} \frac{1}{2} \sum_c \Big|P_{\Pi(i)}(\sigma)\, P_{i|\Pi(i)}(c|\sigma) - Q_{\Pi(i)}(\sigma)\, Q_{i|\Pi(i)}(c|\sigma)
\end{aligned}$$

$$
\begin{aligned}
&\quad + Q_{\Pi(i)}(\sigma)\, Q_{i|\Pi(i)}(c|\sigma) - P_{\Pi(i)}(\sigma)\, Q_{i|\Pi(i)}(c|\sigma)\Big| \\
&\le \sum_i \sum_\sigma \frac{1}{2} \sum_c \Big| P_{\Pi(i)}(\sigma)\, P_{i|\Pi(i)}(c|\sigma) - Q_{\Pi(i)}(\sigma)\, Q_{i|\Pi(i)}(c|\sigma)\Big| \\
&\quad + \sum_i \sum_\sigma \frac{1}{2} \sum_c \Big| Q_{\Pi(i)}(\sigma)\, Q_{i|\Pi(i)}(c|\sigma) - P_{\Pi(i)}(\sigma)\, Q_{i|\Pi(i)}(c|\sigma)\Big| \\
&= \sum_i \sum_\sigma \frac{1}{2} \sum_c \Big| P_{i,\Pi(i)}(c,\sigma) - Q_{i,\Pi(i)}(c,\sigma)\Big| \\
&\quad + \sum_i \sum_\sigma \frac{1}{2} \Big| Q_{\Pi(i)}(\sigma) - P_{\Pi(i)}(\sigma)\Big| \sum_c Q_{i|\Pi(i)}(c|\sigma) \\
&= \sum_i \frac{1}{2} \sum_\sigma \sum_c \Big| P_{i,\Pi(i)}(c,\sigma) - Q_{i,\Pi(i)}(c,\sigma)\Big| \\
&\quad + \sum_i \sum_\sigma \frac{1}{2} \Big| Q_{\Pi(i)}(\sigma) - P_{\Pi(i)}(\sigma)\Big| \\
&= \sum_i d_{\mathrm{TV}}\Big(P_{i,\Pi(i)}, Q_{i,\Pi(i)}\Big) + \sum_i d_{\mathrm{TV}}(P_i, Q_i) \\
&\le 2n \cdot d_{\mathrm{TV}}(P,Q).
\end{aligned}
$$

The last inequality follows because the inequalities $d_{\mathrm{TV}}\Big(P_{i,\Pi(i)}, Q_{i,\Pi(i)}\Big) \le d_{\mathrm{TV}}(P,Q)$ and $d_{\mathrm{TV}}(P_i, Q_i) \le d_{\mathrm{TV}}(P,Q)$ hold. □

We are now ready to prove the correctness and provide a running time bound for Algorithm 1. We have, from Hoeffding's inequality (Lemma 7), that

$$
\begin{aligned}
\mathbf{Pr}[|\mathsf{Est} - d_{\mathrm{TV}}(P,Q)| > \varepsilon d_{\mathrm{TV}}(P,Q)] &= \mathbf{Pr}\left[\left|\frac{Z}{m}\sum_{i=1}^m f\left(w^i\right) - Z \mathop{\mathbf{E}}_{\pi}[f(w)]\right| > \varepsilon d_{\mathrm{TV}}(P,Q)\right] \\
&= \mathbf{Pr}\left[\left|\sum_{i=1}^m f\left(w^i\right) - m \mathop{\mathbf{E}}_{\pi}[f(w)]\right| > \frac{m\varepsilon}{Z} d_{\mathrm{TV}}(P,Q)\right] \\
&\le 2\exp\left(-\frac{2m^2\varepsilon^2 d_{\mathrm{TV}}^2(P,Q)}{Z^2 \sum_{i=1}^m (0-1)^2}\right) \\
&\le 2\exp\left(-\frac{2m^2\varepsilon^2 d_{\mathrm{TV}}^2(P,Q)}{4n^2 d_{\mathrm{TV}}^2(P,Q)\, m}\right) \\
&= 2\exp\left(-\frac{m\varepsilon^2}{2n^2}\right)
\end{aligned}
$$

which is at most δ whenever $m = \Omega(n^2\varepsilon^{-2}\log\delta^{-1})$. The second inequality follows from Lemma 24.

Thus the running time of Algorithm 1 is $O(mn\ell) = O(n^3\varepsilon^{-2}\ell\log\delta^{-1})$, since we draw m samples from π, we can sample from π in time $O(n\ell)$, and evaluate f in time $O(n)$. Finally, the number of probabilistic inference queries is at most $O(n^3\varepsilon^{-2}\ell\log\delta^{-1})$.

4.3 Application: An FPRAS for estimating the TV distance between Bayes nets of bounded treewidth

In this subsection we prove Corollary 3.

Corollary 3 (Formal)**.** *There is an FPRAS for estimating the TV distance between two Bayes nets of treewidth $w = O(\log n)$ and alphabet size $\ell = O(1)$, which are defined over the same DAG of n nodes. In particular, if ε and δ are the accuracy and confidence errors of the FPRAS, respectively, the FPRAS runs in time $\mathrm{poly}(n) \cdot O(\varepsilon^{-2} \log \delta^{-1})$.*

The proof of Corollary 3 will follow from the lemma below, Lemma 26, and an application of Theorem 1. We first prove Lemma 26.

Lemma 26. *Probabilistic inference is efficient for all Bayes nets over n variables which have alphabet size $\ell = O(1)$ and treewidth $O(\log n)$.*

Proof. Let B be a Bayes net over variables $X_1, \ldots, X_n$ that has alphabet size $\ell = O(1)$ and treewidth $w = O(\log n)$. Let $S_1, \ldots, S_n \subseteq [\ell]$ be sets. The probabilistic inference task that we want to perform is to compute the probability $\mathbf{Pr}_B[X_1 \in S_1, \ldots, X_n \in S_n]$.

First, we construct the moralization of B (see Definition 10), namely M_B, in time $O(\mathrm{poly}(n))$ by invoking Lemma 11. Then, we use Theorem 16 to compute a tree decomposition $\mathcal{T}$ of M_B of width at most $4w + 1 \leq 5w$ in time $O(w3^{3w}n^2)$. Finally, we use the variable elimination algorithm of Theorem 17 on B, $S_1, \ldots, S_n$, M_B, and $\mathcal{T}$ to compute $\mathbf{Pr}_B[X_1 \in S_1, \ldots, X_n \in S_n]$ in time $O(n\ell^{5w})$.

The running time of this procedure is $O(\mathrm{poly}(n)) + O(w3^{3w}n^2) + O(n\ell^{5w}) = O(\mathrm{poly}(n))$, whereby we have used the facts that $\ell = O(1)$ and $w = O(\log n)$. This concludes the proof. □

The proof of Corollary 3 now follows by invoking Theorem 1 for $\ell = O(1)$ and $T(G, \ell^2) = O(\mathrm{poly}(n))$.

5 TV distance between a Bayes net and the uniform distribution

5.1 #P-completeness

The main result of this subsection is Theorem 4. Recall that a function f from $\{0,1\}^*$ to non-negative integers is in the class #P if there is a polynomial time non-deterministic Turing machine M so that for any x, it is the case that $f(x)$ is equal to the number of accepting paths of $M(x)$.

We now prove Theorem 4.

Proof of Theorem 4. In what follows, we separately show membership in #P and #P-hardness.

Membership in #P. Let P be a Bayes net distribution over the Boolean domain $\{0,1\}^n$. The goal is to design a nondeterministic machine $\mathcal{N}$ so that the number of accepting paths of $\mathcal{N}$ (normalized by an appropriate quantity) equals $d_{\mathrm{TV}}(P, \mathbb{U})$. We will assume that the probabilities specified in the CPTs of the Bayes net for P are fractions. Let M be equal to 2^n times the product of the denominators of all the probabilities in the CPTs. The non-deterministic machine $\mathcal{N}$ first guesses an element $i \in \{0,1\}^n$ in the sample space of P, computes $|P(i) - 1/2|$ by using the CPTs, then guesses an integer $0 \leq z \leq M$, and finally accepts if and only if $1 \leq z \leq M|P(i) - 1/2|$. (Note that $M|P(i) - 1/2| = |M \cdot P(i) - M/2|$ is an integer.) It follows that

$$d_{\mathrm{TV}}(P, \mathbb{U}) = \frac{1}{2} \sum_{i \in \{0,1\}^n} \left| P(i) - \frac{1}{2} \right| = \frac{\text{number of accepting paths of } \mathcal{N}}{2M}$$

since the number of accepting paths of $\mathcal{N}$ is equal to $\sum_{i\in\{0,1\}^n}(M\,|P(i)-1/2|)$ which is equal to $M\sum_{i\in\{0,1\}^n}|P(i)-1/2|$ or $2Md_{\mathrm{TV}}(P,Q)$.

#P-hardness. For the #P-hardness part, the proof gives a Turing reduction from the problem of counting the satisfying assignments of a CNF formula (which is #P-hard to compute) to computing the total variation distance between a Bayes net distribution and the uniform distribution. In what follows, by a graph of a formula we mean the DAG that captures the circuit structure of F, whereby the nodes are either AND, OR, NOT, or variable gates, and the edges correspond to wires connecting the gates.

Let F be a CNF formula viewed as a Boolean circuit. Assume F has n input variables $x_1,\ldots,x_n$ and m gates $\Gamma=\{y_1,\ldots,y_m\}$, where Γ is topologically sorted with y_m being the output gate. We will define a Bayes net distribution on some DAG G which, intuitively, is the graph of F.

The vertex set of G is split into two sets $\mathcal{X}$ and $\mathcal{Y}$, and a node Z. The set $\mathcal{X}=\{X_i\}_{i=1}^n$ contains n nodes with node X_i corresponding to variable x_i and the set $\mathcal{Y}=\{Y_i\}_{i=1}^m$ contains m nodes with each node Y_i corresponding to gate y_i. So totally there are $n+m+1$ nodes. There is a directed edge from node V_i to node V_j if the gate/variable corresponding to V_i is an input to V_j.

The distribution P on G is given by a CPT defined as follows. Each X_i is a uniformly random bit. For each Y_i, its CPT is deterministic: For each of the setting of the parents Y_j, Y_k, namely y_j, y_k, the variable Y_i takes the value of the gate y_i for that setting of its inputs y_j, y_k. Finally, let Z be the value of Y_m OR-ed with a random bit.

Note that the formula F computes a Boolean function on the input variables. Let $f:\{0,1\}^n\to\{0,1\}$ be this function. We extend f to $\{0,1\}^m$ (i.e., $f:\{0,1\}^n\to\{0,1\}^m$) to also include the values of the intermediate gates.

With this notation in mind, for any binary string XYZ of length $n+m+1$ it is the case that P has a probability 0 if $Y\neq f(X)$. Let $A:=\{x\mid F(x)=1\}$ and $R:=\{x\mid F(x)=0\}$.

To finish the proof, we will write the number of satisfying assignments of F, namely $|A|$, as a polynomial-time computable function of $d_{\mathrm{TV}}(P,\mathbb{U})$: We have

$$2\cdot d_{\mathrm{TV}}(P,\mathbb{U})=\sum_{X,Y,Z}|P-\mathbb{U}|=\underbrace{\sum_{\substack{X,Y,Z\\ Y\neq f(X)}}|P-\mathbb{U}|}_{(1)}+\underbrace{\sum_{\substack{X,Y,Z\\ Y=f(X)}}|P-\mathbb{U}|}_{(2)}$$

where we have abused the notation $P,\mathbb{U}$ to denote the probabilities $P(X,Y,Z)\,,\mathbb{U}(X,Y,Z)$. We will calculate (1) and (2) separately. For (1) we have:

$$\sum_{\substack{X,Y,Z\\ Y\neq f(X)}}|P-\mathbb{U}|=\sum_{\substack{X,Y,Z\\ Y\neq f(X)}}\left|0-\frac{1}{2^{n+m+1}}\right|=\frac{2^{n+1}(2^m-1)}{2^{n+m+1}}=1-\frac{1}{2^m}.$$

For (2), we have

$$\sum_{\substack{X,Y,Z\\ Y=f(X)}}|P-\mathbb{U}|=\underbrace{\sum_{\substack{X,f(X),Z\\ X\in A}}|P-\mathbb{U}|}_{(3)}+\underbrace{\sum_{\substack{X,f(X),Z\\ X\in R}}|P-\mathbb{U}|}_{(4)}$$

and now we calculate the terms (3) and (4) separately. For (3), we have:

$$\sum_{\substack{X,f(X),Z\\X\in A}} |P-\mathbb{U}| = \sum_{\substack{X,f(X),0\\X\in A}} |P-\mathbb{U}| + \sum_{\substack{X,f(X),1\\X\in A}} |P-\mathbb{U}|$$
$$= \sum_{\substack{X,f(X),0\\X\in A}} \left|0-\frac{1}{2^{n+m+1}}\right| + \sum_{\substack{X,f(X),1\\X\in A}} \left|\frac{1}{2^n}-\frac{1}{2^{n+m+1}}\right| = \frac{|A|}{2^{n+m+1}} + \frac{|A|\cdot(2^{m+1}-1)}{2^{n+m+1}} = \frac{|A|}{2^n}$$

and for (4) we have

$$\sum_{\substack{X,f(X),Z\\X\in R}} |P-\mathbb{U}| = \sum_{\substack{X,f(X),0\\X\in R}} |P-\mathbb{U}| + \sum_{\substack{X,f(X),1\\X\in R}} |P-\mathbb{U}|$$
$$= \sum_{\substack{X,f(X),0\\X\in R}} \left|\frac{1}{2^{n+1}}-\frac{1}{2^{n+m+1}}\right| + \sum_{\substack{X,f(X),1\\X\in R}} \left|\frac{1}{2^{n+1}}-\frac{1}{2^{n+m+1}}\right| = \frac{|R|\cdot(2^m-1)\cdot 2}{2^{n+m+1}}.$$

Thus

$$2\cdot d_{\mathrm{TV}}(P,\mathbb{U}) = (1)+(2) = (1)+(3)+(4)$$
$$= 1-\frac{1}{2^m}+\frac{|A|}{2^n}+\frac{|R|\cdot(2^m-1)\cdot 2}{2^{n+m+1}} = 2\left(1-\frac{1}{2^m}+\frac{|A|}{2^{m+n+1}}\right)$$

since $|A|+|R|=2^n$. For that matter, $d_{\mathrm{TV}}(P,\mathbb{U}) = \frac{|A|}{2^{n+m+1}} + \left(1-\frac{1}{2^m}\right)$ or

$$|A| = 2^{n+m+1}\left(d_{\mathrm{TV}}(P,\mathbb{U}) - \left(1-\frac{1}{2^m}\right)\right).$$

That concludes the proof.

5.2 Estimation in fully polynomial time

We prove Theorem 5.

Theorem 5 (Formal)**.** *There is an FPRAS for estimating the TV distance between a Bayes net P and the uniform distribution. Let n be the number of nodes of P, let ℓ be the size of its alphabet, and let d be its maximum in-degree. Then the running time of this FPRAS is $O\left(n^3\ell^{2d+2}\varepsilon^{-2}\log\delta^{-1}\right)$ whereby ε is the accuracy error and δ is the confidence error of the FPRAS.*

Remark 27. Note that the running time of the FPRAS of Theorem 5 is polynomial in the input length, as the description of the Bayes net P in terms of the CPTs has size at least $n+\ell^{d+1}$.

We shall now prove Theorem 5. We require the following lemma (which we will prove later).

Lemma 28. *For all x, it is the case that*

$$1-O\left(d_{\mathrm{TV}}(P,\mathbb{U})\,\ell^{d+1}n\right) \le P(x)\,\ell^n \le 1+O\left(d_{\mathrm{TV}}(P,\mathbb{U})\,\ell^{d+1}n\right)$$

whenever $d_{\mathrm{TV}}(P,\mathbb{U}) \le \frac{1}{16\ell^{d+1}}$.

The proof of Theorem 5 now resumes as follows. First, let us assume that $d_{\mathrm{TV}}(P,\mathbb{U}) \le \frac{1}{16\ell^{d+1}}$ so that Lemma 28 holds. We have that $d_{\mathrm{TV}}(P,\mathbb{U})$ is equal to

$$\frac{1}{2}\sum_x |P(x)-\mathbb{U}(x)| = \sum_x \max(0, P(x)-\mathbb{U}(x))$$

$$
\begin{aligned}
&= \sum_x \mathbb{U}(x) \max\left(0, \frac{P(x)}{\mathbb{U}(x)} - 1\right) \\
&= \mathop{\mathbf{E}}_{x\sim\mathbb{U}}\left[\max\left(0, \frac{P(x)}{\mathbb{U}(x)} - 1\right)\right] \\
&= \mathop{\mathbf{E}}_{x\sim\mathbb{U}}[\max(0, P(x)\,\ell^n - 1)]\,.
\end{aligned}
$$

This yields a natural estimator for $d_{\mathrm{TV}}(P, \mathbb{U})$, namely Est, as follows:

1. Sample $x_1, \ldots, x_m \sim \mathbb{U}$ for some value of m that we will fix later;
2. compute $\max(0, P(x_i)\,\ell^n - 1)$ for all $1 \le i \le m$;
3. output $(1/m)\sum_{i=1}^m \max(0, P(x_i)\,\ell^n - 1)$.

We will now prove the correctness and upper bound the running time of this procedure. We have from Hoeffding's inequality (Lemma 7) and Lemma 28 that

$$
\begin{aligned}
&\mathbf{Pr}[|\mathsf{Est} - d_{\mathrm{TV}}(P, \mathbb{U})| > \varepsilon d_{\mathrm{TV}}(P, \mathbb{U})] \\
&\quad= \mathbf{Pr}\left[\left|\frac{1}{m}\sum_{i=1}^m \max(0, P(x_i)\,\ell^n - 1) - \mathop{\mathbf{E}}_{x\sim\mathbb{U}}[\max(0, P(x)\,\ell^n - 1)]\right| > \varepsilon d_{\mathrm{TV}}(P, \mathbb{U})\right] \\
&\quad= \mathbf{Pr}\left[\left|\sum_{i=1}^m \max(0, P(x_i)\,\ell^n - 1) - m\mathop{\mathbf{E}}_{x\sim\mathbb{U}}[\max(0, P(x)\,\ell^n - 1)]\right| > m\varepsilon d_{\mathrm{TV}}(P, \mathbb{U})\right] \\
&\quad\le 2\exp\left(-\frac{2m^2\varepsilon^2 d_{\mathrm{TV}}^2(P, \mathbb{U})}{\sum_{i=1}^m (0 - O(d_{\mathrm{TV}}(P, \mathbb{U})\,\ell^{d+1}n))^2}\right) \\
&\quad= 2\exp\left(-\frac{2m^2\varepsilon^2 d_{\mathrm{TV}}^2(P, \mathbb{U})}{m\cdot O(d_{\mathrm{TV}}^2(P, \mathbb{U})\,\ell^{2d+2}n^2)}\right) \\
&\quad= 2\exp\left(-\frac{m\varepsilon^2}{O(\ell^{2d+2}n^2)}\right)
\end{aligned}
$$

which is at most δ whenever $m = \Omega\left(n^2\ell^{2d+2}\varepsilon^{-2}\log\delta^{-1}\right)$.

The running time of this procedure is $O(mn) = O\left(n^3\ell^{2d+2}\varepsilon^{-2}\log\delta^{-1}\right)$, since we draw m samples and P can be evaluated on any sample in time $O(n)$.

If $d_{\mathrm{TV}}(P, \mathbb{U}) > \frac{1}{16\ell^{d+1}}$, then it suffices to additively approximate $d_{\mathrm{TV}}(P, \mathbb{U})$ up to error $\varepsilon/\left(16\ell^{d+1}\right)$. This can be done by Monte Carlo sampling using $m = \Omega\left(\ell^{2d+2}\varepsilon^{-2}\log\delta^{-1}\right)$ samples and $O(mn) = O\left(n\ell^{2d+2}\varepsilon^{-2}\log\delta^{-1}\right)$ time.

We now prove Lemma 28.

Proof of Lemma 28. Let us denote the maximum in-degree of P by d. Let X_0 be an arbitrary node with its parents as $X_1, \ldots, X_d$.

We have that $\gamma := d_{\mathrm{TV}}(P, \mathbb{U})$ is at least

$$
\begin{aligned}
&d_{\mathrm{TV}}((X_0, \ldots, X_d), (Y_0, \ldots, Y_d)) \\
&\quad= \frac{1}{2}\sum_{v_0}\cdots\sum_{v_d} |\mathbf{Pr}[(X_0, \ldots, X_d) = (v_0, \ldots, v_d)] - \mathbf{Pr}[(Y_0, \ldots, Y_d) = (v_0, \ldots, v_d)]| \\
&\quad= \frac{1}{2}\sum_{v_0}\cdots\sum_{v_d} \left|\mathbf{Pr}[X_0 = v_0 | X_1 = v_1, \ldots, X_d = v_d]\,\mathbf{Pr}[X_1 = v_1, \ldots, X_d = v_d] - \frac{1}{\ell^{d+1}}\right|
\end{aligned}
$$

or

$$\frac{1}{2}\sum_{v_0}\cdots\sum_{v_d}\left|\mathbf{Pr}[X_0=v_0|X_1=v_1,\ldots,X_d=v_d]\,\mathbf{Pr}[X_1=v_1,\ldots,X_d=v_d]-\frac{1}{\ell^{d+1}}\right|=\gamma$$

or

$$\left|\mathbf{Pr}[X_0=v_0|X_1=v_1,\ldots,X_d=v_d]\,\mathbf{Pr}[X_1=v_1,\ldots,X_d=v_d]-\frac{1}{\ell^{d+1}}\right|\leq 2\gamma \tag{1}$$

for any $v_0,\ldots,v_d$. We observe the following.

Claim 29. *We have that* $1/\ell^d-\gamma\leq\mathbf{Pr}[X_1=v_1,\ldots,X_d=v_d]\leq 1/\ell^d+\gamma$.

Proof. Since $d_{\mathrm{TV}}(P,\mathbb{U})=\gamma$ and $\mathbf{Pr}[Y_1=v_1,\ldots,Y_d=v_d]=1/\ell^d$, the claim is immediate. □

By Equation (1) and Claim 29 we have the following.

Corollary 30. *For* $\gamma<1/\left(2\ell^d\right)$ *we have that*

$$|\mathbf{Pr}[X_0=v_0|X_1=v_1,\ldots,X_d=v_d]-1/\ell|\leq 8\gamma\ell^d.$$

Proof. By Equation (1) we have

$$\frac{1}{\ell^{d+1}}-2\gamma\leq\mathbf{Pr}[X_0=v_0|X_1=v_1,\ldots,X_d=v_d]\,\mathbf{Pr}[X_1=v_1,\ldots,X_d=v_d]\leq\frac{1}{\ell^{d+1}}+2\gamma$$

or

$$\begin{aligned}\frac{\frac{1}{\ell^{d+1}}-2\gamma}{\mathbf{Pr}[X_1=v_1,\ldots,X_d=v_d]}&\leq\mathbf{Pr}[X_0=v_0|X_1=v_1,\ldots,X_d=v_d]\\&\leq\frac{\frac{1}{\ell^{d+1}}+2\gamma}{\mathbf{Pr}[X_1=v_1,\ldots,X_d=v_d]}\end{aligned}$$

or, by making use of Claim 29,

$$\frac{\frac{1}{\ell^{d+1}}-2\gamma}{\frac{1}{\ell^d}+\gamma}\leq\mathbf{Pr}[X_0=v_0|X_1=v_1,\ldots,X_d=v_d]\leq\frac{\frac{1}{\ell^{d+1}}+2\gamma}{\frac{1}{\ell^d}-\gamma}$$

or

$$\frac{\frac{1}{\ell}-2\ell^d\gamma}{1+\ell^d\gamma}\leq\mathbf{Pr}[X_0=v_0|X_1=v_1,\ldots,X_d=v_d]\leq\frac{\frac{1}{\ell}+2\ell^d\gamma}{1-\ell^d\gamma}.$$

We now have

$$\begin{aligned}\mathbf{Pr}[X_0=v_0|X_1=v_1,\ldots,X_d=v_d]&\leq\frac{\frac{1}{\ell}+2\ell^d\gamma}{1-\ell^d\gamma}\\&\leq\left(\frac{1}{\ell}+2\ell^d\gamma\right)\left(1+2\ell^d\gamma\right)\\&=\frac{1}{\ell}+2\ell^{d-1}\gamma+2\ell^d\gamma+4\ell^{2d}\gamma^2\\&\leq\frac{1}{\ell}+2\ell^d\gamma+2\ell^d\gamma+4\ell^d\gamma\\&=\frac{1}{\ell}+8\ell^d\gamma\end{aligned}$$

since $1/(1-x) \leq 1+2x$ for $x < 1/2$ (here $x = \ell^d\gamma < 1/2$), and

$$\begin{aligned}
\mathbf{Pr}[X_0 = v_0 | X_1 = v_1, \ldots, X_d = v_d] &\geq \frac{\frac{1}{\ell} - 2\ell^d\gamma}{1+\ell^d\gamma} \\
&\geq \left(\frac{1}{\ell} - 2\ell^d\gamma\right)\left(1-\ell^d\gamma\right) \\
&= \frac{1}{\ell} - \ell^{d-1}\gamma - 2\ell^d\gamma + 2\ell^{2d}\gamma^2 \\
&\geq \frac{1}{\ell} - \ell^d\gamma - 2\ell^d\gamma \\
&\geq \frac{1}{\ell} - 8\gamma\ell^d
\end{aligned}$$

since $1/(1+x) \geq 1-x$ for $x < 1/2$ (here $x = \ell^d\gamma < 1/2$). □

The result now follows from the observation that

$$\left(1/\ell - 8\gamma\ell^d\right)^n \leq P(x) = \prod_{i=1}^{n} \mathbf{Pr}\left[X_i = x_i | X_{\Pi(X_i)} = x_{\Pi(X_i)}\right] \leq \left(1/\ell + 8\gamma\ell^d\right)^n$$

or

$$\left(1 - 8\gamma\ell^{d+1}\right)^n \leq P(x)\,\ell^n \leq \left(1 + 8\gamma\ell^{d+1}\right)^n$$

or

$$1 - 16\gamma\ell^{d+1}n \leq P(x)\,\ell^n \leq 1 + 16\gamma\ell^{d+1}n,$$

whereby we used the facts that $(1-\alpha)^k \geq (1-2\alpha k)$ and $(1+\alpha)^k \leq (1+2\alpha k)$ whenever $\alpha < 1/2$ and $k > 0$, and the fact that $\gamma < 1/\left(16\ell^{d+1}\right)$ or $8\gamma\ell^{d+1} < 1/2$.

Finally, we have

$$1 - 16d_{\mathrm{TV}}(P,\mathbb{U})\,\ell^{d+1}n \leq P(x)\,\ell^n \leq 1 + 16d_{\mathrm{TV}}(P,\mathbb{U})\,\ell^{d+1}n$$

as desired.

6 Conclusion

We have established a general connection between probabilistic inference and TV distance computation. In particular, we proved that TV distance estimation can be reduced to probabilistic inference. This enables us to prove the existence of a novel FPRAS for estimating the TV distance between Bayes nets of small treewidth.

Moreover, we made some significant progress in understanding the complexity of computing the TV distance between an arbitrary Bayes net and the uniform distribution: We showed that the exact computation is #P-hard, while there is an FPRAS for the same task.

We outline the following open problems: Can we prove similar results for TV distance estimation between undirected graphical models? Another problem of interest is to study other notions of distance, such as Wasserstein metrics.

Acknowledgements

The work of AB was supported in part by National Research Foundation Singapore under its NRF Fellowship Programme (NRF-NRFFAI-2019-0002) and an Amazon Faculty Research Award. The work of SG was supported by an initiation grant from IIT Kanpur and a SERB

award CRG/2022/007985. Pavan's work is partly supported by NSF award 2130536. Vinodchandran's work is partly supported by NSF award 2130608. This work was supported in part by National Research Foundation Singapore under its NRF Fellowship Programme [NRF-NRFFAI1-2019-0004] and an Amazon Research Award. Part of the work was done during Meel, Pavan, and Vinodchandran's visit to Simons Institute for the Theory of Computing.

References

[BB01] Pierre Baldi and Søren Brunak. *Bioinformatics: The machine learning approach*. MIT press, 2001.

[BGM+23] Arnab Bhattacharyya, Sutanu Gayen, Kuldeep S. Meel, Dimitrios Myrisiotis, A. Pavan, and N. V. Vinodchandran. On approximating total variation distance. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI 2023, 19th-25th August 2023, Macao, SAR, China*, pages 3479–3487. ijcai.org, 2023.

[BGMV20] Arnab Bhattacharyya, Sutanu Gayen, Kuldeep S. Meel, and N. V. Vinodchandran. Efficient distance approximation for structured high-dimensional distributions via learning. In *Proc. of NeurIPS*, 2020.

[BKM17] David M. Blei, Alp Kucukelbir, and Jon D. McAuliffe. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, 2017.

[Coo90] Gregory F Cooper. The computational complexity of probabilistic inference using Bayesian belief networks. *Artificial intelligence*, 42(2-3):393–405, 1990.

[CR14] Clément L. Canonne and Ronitt Rubinfeld. Testing probability distributions underlying aggregated data. In Javier Esparza, Pierre Fraigniaud, Thore Husfeldt, and Elias Koutsoupias, editors, *Automata, Languages, and Programming - 41st International Colloquium, ICALP 2014, Copenhagen, Denmark, July 8-11, 2014, Proceedings, Part I*, volume 8572 of *Lecture Notes in Computer Science*, pages 283–295. Springer, 2014.

[CS97] Ming-Hui Chen and Qi-Man Shao. On Monte Carlo methods for estimating ratios of normalizing constants. *The Annals of Statistics*, 25(4):1563–1594, 1997.

[CTY06] Nick Chater, Joshua B Tenenbaum, and Alan Yuille. Probabilistic models of cognition: Conceptual foundations. *Trends in cognitive sciences*, 10(7):287–291, 2006.

[Dec99] Rina Dechter. Bucket elimination: A unifying framework for reasoning. *Artificial Intelligence*, 113(1-2):41–85, 1999.

[FGJW23] Weiming Feng, Heng Guo, Mark Jerrum, and Jiaheng Wang. A simple polynomial-time approximation algorithm for the total variation distance between two product distributions. *TheoretiCS*, 2, 2023.

[GSV99] Oded Goldreich, Amit Sahai, and Salil P. Vadhan. Can statistical zero knowledge be made non-interactive? or On the relationship of SZK and NISZK. In *Proc. of CRYPTO*, pages 467–484, 1999.

[KBCK23] Lutz Klinkenberg, Christian Blumenthal, Mingshuai Chen, and Joost-Pieter Katoen. Exact Bayesian inference for loopy probabilistic programs. *CoRR*, abs/2307.07314, 2023.

[KBvdG10] Johan Kwisthout, Hans L Bodlaender, and Linda C van der Gaag. The necessity of bounded treewidth for efficient inference in Bayesian networks. In *ECAI*, volume 215, pages 237–242, 2010.

[KF09] Daphne Koller and Nir Friedman. *Probabilistic graphical models: Principles and techniques*. MIT press, 2009.

[KWCK23] Lutz Klinkenberg, Tobias Winkler, Mingshuai Chen, and Joost-Pieter Katoen. Exact probabilistic inference using generating functions. *CoRR*, abs/2302.00513, 2023.

[LMP01] Michael L Littman, Stephen M Majercik, and Toniann Pitassi. Stochastic Boolean satisfiability. *Journal of Automated Reasoning*, 27:251–296, 2001.

[LS88] Steffen L Lauritzen and David J Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society: Series B (Methodological)*, 50(2):157–194, 1988.

[Min01] Thomas Minka. Expectation propagation for approximate Bayesian inference. In *UAI*, 2001.

[Mur22] Kevin P Murphy. *Probabilistic machine learning: An introduction*. MIT press, 2022.

[MWJ13] Kevin P. Murphy, Yair Weiss, and Michael I. Jordan. Loopy belief propagation for approximate inference: An empirical study. *CoRR*, abs/1301.6725, 2013.

[Pea88] Judea Pearl. *Probabilistic reasoning in intelligent systems: Networks of plausible inference*. Morgan Kaufmann, 1988.

[Pea09] Judea Pearl. *Causality*. Cambridge university press, 2009.

[RGB14] Rajesh Ranganath, Sean Gerrish, and David M. Blei. Black box variational inference. *Arxiv preprint arXiv:1401.0118*, 2014.

[ROL02] Rajesh PN Rao, Bruno A Olshausen, and Michael S Lewicki. *Probabilistic models of the brain: Perception and neural function*. MIT press, 2002.

[Rot96] Dan Roth. On the hardness of approximate reasoning. *Artificial Intelligence*, 82(1-2):273–302, 1996.

[RS84] Neil Robertson and Paul D. Seymour. Graph minors III. Planar tree-width. *J. Comb. Theory, Ser. B*, 36(1):49–64, 1984.

[SV03] Amit Sahai and Salil P. Vadhan. A complete problem for statistical zero knowledge. *J. ACM*, 50(2):196–249, 2003.

[WJ+08] Martin J Wainwright, Michael I Jordan, et al. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2):1–305, 2008.

[ZP94] Nevin Lianwen Zhang and David L. Poole. A simple approach to Bayesian network computations, 1994.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il