

---

# Approximability of Probability Distributions

---

**Alina Beygelzimer**

Department of Computer Science  
University of Rochester, Rochester, NY 14627  
beygel@cs.rochester.edu

**Irina Rish**

IBM T. J. Watson Research Center  
Hawthorne, NY 10532  
rish@us.ibm.com

## Abstract

We consider the question of how well a given distribution can be approximated with probabilistic graphical models. We introduce a new parameter, *effective treewidth*, that captures the degree of approximability as a tradeoff between the accuracy and the complexity of approximation. We present a simple approach to analyzing achievable tradeoffs that exploits the threshold behavior of monotone graph properties, and provide experimental results that support the approach.

## 1 Introduction

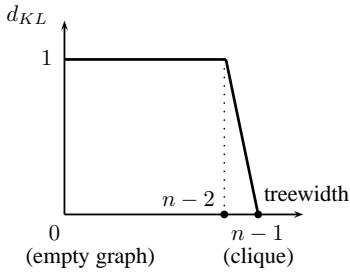
A major concern in probabilistic reasoning using graphical models, such as Bayesian networks, is the computational complexity of inference, which is generally NP-hard [5]. Typical approaches use approximation algorithms that trade accuracy for efficiency. However, the following important questions remain open: How can we characterize the accuracy/efficiency trade-off of a given distribution, i.e. its *degree of approximability*? How do we distinguish between distributions that are easy to approximate and those that are hard?

An equally important motivation for studying these questions comes from learning probabilistic graphical models from data. Our goal is to learn models that not only fit the data well but also yield efficient inference. Note that traditional model selection criteria, such as BIC/MDL, aim at fitting the data well and minimizing the *representation* complexity of the learned model (i.e., the total number of parameters). However, as demonstrated in [2], such criteria are unable to capture the *inference* complexity of the model. In particular, it was shown that two models that fit the data equally well and have similar representation complexity may have quite different graph structures, making one model *exponentially* slower to reason with than the other.

The complexity of exact inference algorithms in Bayesian networks, such as the junction tree algorithm [12], or closely related variable-elimination techniques [7], is exponential in the size of the largest dependency created during inference, which corresponds to the size of the largest clique induced during graph triangulation associated with the inference. This parameter (formally defined later) is known as the *treewidth*. As shown by [11], some form of triangulation is a necessary components of any scheme for (exact) belief updating based on local calculations. Thus, the treewidth arises as a natural measure of inference complexity in graphical models, and it is the measure we adopt here. In order to measure the accuracy of an approximate distribution  $Q(X)$  with respect to the target distribution  $P(X)$  we use the information-theoretic notion of *Kullback-Leibler divergence* (*KL-divergence*)  $d_{KL}(P, Q)$ .

The following questions naturally arise: If we tolerate a certain inaccuracy in our model, what is the best inference complexity we can hope to achieve? Or, vice versa, what is the best achievable approximation accuracy given a constraint on the complexity (i.e., a bound on the treewidth)? Intuitively, a distribution  $P(X)$  is "easy" (approximable) if its close to a distribution  $Q(X)$  representable by a low-treewidth Bayesian network, so that a small sacrifice in accuracy yields a significant gain in efficiency. The following example should make the motivation clear.

**Motivating Example** Consider the parity function on  $n$  binary random variables, and let our target distribution  $P$  be the uniform distribution on the values to which it assigns 1 (i.e., on  $n$ -bit strings with an odd number of 1s). It is easy to see that any approximation  $Q$  that decomposes over a Bayesian network whose moralized graph (formally defined in the next section) misses at least one edge, is precisely as inaccurate as the one that assumes all variables to be independent (i.e., has no edges).



This follows from the fact that the probability distribution induced on any proper subset of the variables is uniform, and thus for any subset  $\{X_{i_1}, \dots, X_{i_k}\}$  of  $k < n$  variables,  $P(X_{i_1} | X_{i_2}, \dots, X_{i_k}) = P(X_{i_1})$ , uniform on  $\{0, 1\}$ . It is then readily seen that  $\sum_{\mathbf{x}} P(\mathbf{x}) \log Q(\mathbf{x}) = 2^{-(n-1)} \sum_{\mathbf{x}: P(\mathbf{x}) > 0} \log \prod_{i=1}^n Q(x_i | x_{i_1}, \dots, x_{i_r}) = \log \prod_{i=1}^n Q(x_i) = \log 2^n = -n$ ,<sup>1</sup> and  $d_{KL}(P, Q) = -H(P) + n = 1$  since  $H(P) = n - 1$ . Thus, unless we can afford the complexity of the complete graph, there

is *absolutely* no sense (i.e., absolutely no gain in accuracy and a potentially exponential loss of efficiency) in using a model more complex than the empty graph. Intuitively, this gives an example of a nonapproximable distribution.

On the other hand, one can easily construct a distribution with large but weak dependencies. Exact representation of such distribution requires a complex Bayesian network (e.g., a complete graph); however, a small sacrifice in accuracy may yield a very simple model. For example, consider a distribution  $P$  over  $n$  Boolean variables, in which variables  $X_1, \dots, X_{n-1}$  are independent and uniformly distributed; if all  $X_1, \dots, X_{n-1}$  are true,  $X_n$  is true with probability  $1/4$  (and false with probability  $3/4$ ); otherwise  $X_n$  is true with probability  $1/2$  (regardless of the values of  $X_1, \dots, X_{n-1}$ ). It is easy to see that any exact representation of  $P$  (e.g.,  $X_n$  is a child node of  $X_1, \dots, X_{n-1}$ ) requires a complete graph (after moralization). On the other hand, it is easy to show that  $P$  is very close (has KL-divergence  $2^{-(n+1)}$  vanishing exponentially with  $n \rightarrow \infty$ ) to the joint distribution  $Q(X) = \prod_i P(X_i)$  of  $n$  independent, uniformly distributed variables, represented by the disconnected graph<sup>2</sup>.

In practice, of course, distributions are typically between the extremes. The tradeoff between the complexity and accuracy is monotonic; however, it may be far from linear. The goal is to exploit these nonlinearities in choosing the best available tradeoff, given a target distribution we wish to approximate, or a data set sampled from a distribution we wish to learn. We propose an approach that uses the existence of thresholds in monotone graph properties. The theory of random graphs was initiated by Erdős and Rényi [8], and one of the main observations they made was that many natural monotone properties appear (or disappear, depending on the direction of monotonicity) rather suddenly; i.e., a sharp tran-

<sup>1</sup>Second to last equality is due to the well-known fact that  $d_{KL}(P, Q)$  is minimized by forcing conditional probabilities of  $Q$  to coincide with those computed from  $P$ .

<sup>2</sup>Note that  $P(X)$  differs from  $Q(X)$  in only two states, when all  $X_1, \dots, X_{n-1}$  are true, and  $X_n$  is either true or false.

sition from the property being very unlikely to it being very likely happens as the edge probability is increased a little (or decreased, if the property is monotone decreasing).

This paper makes the following contributions. First, we show that both important properties of random graphical models, the property of “being efficient” (i.e., having treewidth at most some fixed integer  $k$ ) and the property of “being accurate” (i.e., being at distance at most some  $\delta$  from the target distribution), are monotone and demonstrate a threshold behavior, giving us two families of threshold curves parameterized by  $k$  and by  $\delta$ , respectively. Second, we introduce the notion of *effective treewidth*  $k(\delta)$ , which denotes the smallest achievable treewidth  $k$  given a constraint  $\delta$  on KL-divergence from the target (level of inaccuracy). The effective treewidth captures the approximability of the distribution, and is determined by relative position of the threshold curves, an inherent property of the target distribution. Finally, we provide an efficient sampling-based approach that actually finds a model achieving  $k(\delta)$  with high probability. We estimate the threshold curves and, using their relative position, identify a class of treewidth-bounded models such that the models in the class are *still* simple, yet this class *already* contains (with high probability) a sufficiently good approximations to the target distribution (otherwise, we suggest that the distribution is inherently hard to approximate).

## 2 Preliminaries and Related Work

Let  $P$  be a probability distribution on a set of  $n$  discrete random variables  $X = \{X_1, X_2, \dots, X_n\}$ . A *Bayesian network* [12] is a directed acyclic graph (DAG)  $G$  where the nodes correspond to the random variables and the edges represent direct dependencies among them. The dependencies are quantified by associating each node  $X_i$  with a local conditional probability distribution  $P(X_i | \Pi_i)$ , where  $\Pi_i$  is the set of parents of  $X_i$  (nodes pointing to  $X_i$ ) in  $G$ . The joint probability distribution encoded by a Bayesian network is given by the product  $P(X_1, \dots, X_n) = \prod_{i=1}^n P(X_i | \Pi_i)$ . We say that a distribution  $P(X)$  *decomposes* over a DAG  $G$  if there exist local conditional probability distributions corresponding to  $G$  such that  $P(X)$  can be written in such a form.

In general, exact probabilistic inference in Bayesian networks is NP-hard [5]. For singly-connected networks (i.e., networks with no undirected cycles), there is a linear time local belief-propagation algorithm [12]. In order to use this algorithm in the presence of cycles, one typically constructs a *junction tree* of the network and runs the algorithm on this tree [12]. Constructing a junction tree involves triangulating the graph, i.e., adding edges so that every cycle of length greater than three has a chord (i.e., an edge between a pair of non-adjacent nodes). Each triangulation corresponds to some order of eliminating variables when summing terms out during inference [7]. Exact inference can then be done in time and space linear in the representation of clique marginals in the junction tree, which is *exponential* in the size of the largest clique induced during triangulation. This number (minus one) is called the *width* of a given triangulation. The minimum width over all possible triangulations is called the *treewidth* of the graph<sup>3</sup>.

Given a target distribution  $P(\mathbf{X})$  and an approximation  $Q(\mathbf{X})$ , the *information divergence* (Kullback-Leibler divergence, or KL-divergence) between  $P$  and  $Q$  is defined as  $d_{KL}(P, Q) = \sum_{\mathbf{x}} P(\mathbf{x}) \log \frac{P(\mathbf{x})}{Q(\mathbf{x})}$ , where  $\mathbf{x}$  ranges over all possible assignments to the variables in  $\mathbf{X}$  (See [6].) Notice that  $d_{KL}(P, Q)$  is not necessarily symmetric.

Given a set of independent samples from a distribution  $P(X)$ , the general goal is to learn a Bayesian network model of this distribution that involves dependencies only on limited

<sup>3</sup>The triangulation procedure is defined for undirected graphs, so we must first make the network undirected while preserving the set of independence assumptions; this can be done by *moralizing* the network, i.e., connecting (“marrying”) the parents of every node by a clique and then dropping the direction of all edges.

subsets of the variables. Restricting the size of dependencies controls both overfitting and the complexity of inference in the resulting model. The samples are in the form of tuples  $\langle x_1, \dots, x_n \rangle$  each corresponding to a particular assignment  $\langle X_1 = x_1, \dots, X_n = x_n \rangle$ . A natural way of controlling the complexity of the learned model is to limit ourselves to a class of treewidth-bounded networks. Let  $\mathcal{D}_k$  denote the class of distributions decomposable on graphs with treewidth at most  $k$  ( $0 \leq k < n$ ), with  $\mathcal{D}_1$  corresponding to the set of tree-decomposable distributions. The distribution within  $\mathcal{D}_k$  minimizing the information divergence from the target distribution  $P$  is called the *projection* of  $P$  onto  $\mathcal{D}_k$ .

**Learning bounded-treewidth models** Chow and Liu [4] showed that the projection of an arbitrary distribution  $P(X)$  on the class of tree-decomposable distributions  $\mathcal{D}_1$  is simply a maximum weight spanning tree, where the edge weight is the mutual information between the corresponding variables. Notice that candidate spanning trees can be compared without any knowledge of  $P$  beyond that given by pairwise statistics, which yields an efficient algorithm. The additive decomposition of  $d_{KL}$  used in the proof, can be easily extended to “wider” networks. Fix a network structure  $G$ , and let  $Q$  be a distribution decomposable over  $G$ . Then

$$d_{KL}(P, Q) = \sum_{\mathbf{x}} P(\mathbf{x}) \log \frac{P(\mathbf{x})}{Q(\mathbf{x})} = - \sum_{i=1}^n \sum_{x_i, \pi_i} P(x_i, \pi_i) \log Q(x_i | \pi_i) - H(P),$$

where  $\pi_i$  ranges over all possible values of  $\Pi_i$ . If  $P$  is the empirical distribution induced by the given sample  $S$  of size  $N$  (i.e., defined by frequencies of events in the sample), then the first term can be shown to be  $-LL(Q)/N$ , where  $LL(Q) = \log P(S|Q)$ , the log-likelihood of the model  $Q$ <sup>4</sup>. Thus minimizing  $d_{KL}(P, Q)$  is equivalent to maximizing the log-likelihood  $LL(Q)$ . Standard arguments (see, for example, [12]) show that the first term is maximized by forcing all conditional probabilities  $Q(x_i | \pi_i)$  to coincide with those computed from  $P$  (e.g., relative frequencies in the sample if  $P$  is the empirical distribution). Hence if  $G$  is fixed, the projection onto the set of  $G$ -decomposable distributions is uniquely defined, and we will identify  $G$  with this projection (ignoring some notational abuse). A more challenging problem is finding the best DAG  $G$  in some treewidth-bounded class  $\mathcal{D}_k$  that yields a model closest to  $P$ ; this clearly reduces to finding the minimum-weight hypertree [10]. For *undirected* graphs (i.e., Markov models), Srebro [13] showed the reverse reduction, thus proving the NP-hardness of learning bounded-treewidth Markov networks, and provided an approximation method. It is also important to note that our approach is complementary to the ones of [4, 13] as we address the orthogonal problem of identifying the optimal treewidth-bounded *class of models* given the data.

**Threshold behavior of random graphs** We use the model of random directed acyclic graphs (DAGs) defined by Barak and Erdős [1]. Consider the probability space  $G(n, p)$  of random undirected graphs on  $n$  nodes with edge probability  $p$  (i.e., every pair of nodes is connected with probability  $p$ , independently of every other pair). Let  $G_{n,p}$  stand for a random graph from this probability space. We will also occasionally use  $G_{n,m}$  to denote a graph chosen randomly from among all graphs with  $n$  nodes and  $m$  edges. When  $p = m / \binom{n}{2}$ , the two models are practically identical. A random DAG in the Barak-Erdős model is obtained from  $G_{n,p}$  by orienting the edges according to the ordering of vertices, i.e., all edges are directed from higher to lower indexed vertices.

A *graph property*  $\mathcal{P}$  is naturally associated with the set of graphs having  $\mathcal{P}$ . A property is *monotone increasing* if it is preserved under edge addition: If a graph  $G$  satisfies the property, then every graph on the same set of nodes containing  $G$  as a subgraph must satisfy it as well. It is easy to see (and intuitively clear) that if  $\mathcal{P}$  is a monotone increasing property then the probability that  $G_{n,p}$  satisfies  $\mathcal{P}$  is a non-decreasing function of  $p$ . A

---

<sup>4</sup>Since the true distribution  $P$  is given only by the sample, we let  $P$  also denote the empirical distribution induced by the sample, ignoring some abuse of notation.

*monotone decreasing* property is defined similarly. For example, the property of having treewidth at most some fixed integer  $k$  is monotone decreasing: adding edges can only increase the treewidth. The theory of random graphs was initiated by Erdős and Rényi [8], and one of the main observations they made was that many natural monotone properties appear rather suddenly, i.e., as we increase  $p$ , there is a sharp transition from a property being very unlikely to it being very likely. Friedgut [9] proved that *every* monotone graph property of undirected graphs has such a threshold behavior. Random DAGs (corresponding to random partially ordered sets) have received less attention than random undirected graphs, partially because of the additional structure that prevents the completely independent choice of edges. Nonetheless, many properties of random DAGs were also shown to have threshold functions. (See, for example, [3] and references therein.) However, we don't know of any general result for random DAGs analogous to that of Friedgut [9].

### 3 Formalization and Main Idea

First we introduce two properties of networks essential for the rest of the paper.

**Accuracy** Recall that the information divergence of a given DAG  $G$  from the target distribution  $P$  is given by  $d_{KL}(P, G) = W(G) - H(P)$ , where  $W(G) = -\sum_{i=1}^n \sum_{x_i, \pi_i} P(x_i, \pi_i) \log P(x_i | \pi_i)$ . (In our case,  $P$  is the empirical distribution induced by the given sample  $S$  of size  $N$ . As mentioned before,  $W(G) = -LL(G)/N \geq 0$ .) Fix a distance parameter  $\delta > 0$ , and consider the property  $\mathcal{P}_\delta$  of  $n$ -node DAGs of having  $W(G) \geq \delta$ . Notice that  $\mathcal{P}_\delta$  is monotone increasing: Adding edges to a graph can only bring the graph closer to the target distribution, since any distribution decomposable on the original graph is also decomposable on the augmented one. Thus if  $G$  is a subgraph of  $G'$ , then  $W(G) \geq \delta$  only if  $W(G') \geq \delta$ .

**Complexity** Fix an integer  $k$ , and consider the property of  $n$ -node DAGs of having treewidth of their moralized graph at most  $k$ . Call this property  $\mathcal{P}_k$  and observe that it is a structural property of a DAG, which does *not* depend on the target distribution and its projection onto the DAG. It is also a monotone decreasing property, since if a graph has treewidth at most  $k$ , then certainly any of its subgraphs does.

Recall that we identify each graph with the projection of the target distribution onto the graph. We call a pair  $(k, \delta)$  *achievable* for a distribution  $P$ , if there exists a distribution  $Q$  decomposable on a graph with treewidth at most  $k$  such that  $d_{KL}(P, Q) \leq \delta$ . The *effective treewidth* of  $P$ , with respect to a given  $\delta$ , is defined as the smallest  $k(\delta)$  such that the pair  $(k, \delta)$  is achievable, i.e., if all distributions at distance at most  $\delta$  from  $P$  are not decomposable on graphs with treewidth less than  $k(\delta)$ . This formulation gives the level of inevitable complexity (i.e., treewidth)  $k$ , given the desired level of accuracy  $\delta$ . We will also be interested in average-case analogs of these definitions. Fix  $\epsilon > 0$ . We will say that a pair  $(k, \delta)$  is  $\epsilon$ -*achievable* for  $P$  if at least an  $\epsilon$ -fraction of all DAGs in  $\mathcal{D}_k$  certify that  $(k, \delta)$  is achievable. Thus we not only care about the existence of an approximation with given  $\delta$  and  $k$ , but also in the *number* of such approximations.

Given an empirical distribution  $P$  as above and a distance parameter  $\delta > 0$ , the goal is to find the smallest treewidth bound  $k$  such that the pair  $(k, \delta)$  is achievable for  $P$ . It should, of course, be also feasible to actually find a model achieving this tradeoff. This consideration leads to a somewhat different goal, which we achieve (and justify) below.

Consider the curve given by  $\mu_\delta(p) = \Pr[W(G_{n,p}) \geq \delta]$ . Let  $p_\delta$  be the critical value of the property  $\mathcal{P}_\delta$  defined by  $\mu_\delta(p_\delta) = 1/2$ . Roughly, the probability that a random DAG  $G_{n,p}$  satisfies  $\mathcal{P}_\delta$  jumps from zero to one around  $p = p_\delta$ . Similarly, for each treewidth bound  $k$ , let  $\mu_k(p) = \Pr[\text{width}(G_{n,p}) \leq k]$ , and let  $p_k$  be such that  $\mu_k(p_k) = 1/2$ .

Suppose that we knew the value of  $p_\delta$ . For reasons that will become clear in a moment, our goal will be to find the smallest  $k$  such that  $p_k > p_\delta + o(1)$ . More precisely, we want the

smallest  $k$  satisfying  $\Pr[\text{width}(G_{n,p_\delta}) \leq k] \geq 1/2 + \epsilon$ , for some constant  $\epsilon > 0$ . Let  $k^*$  denote this  $k$ . To find  $k^*$ , we will generate  $m$  independent random DAGs in  $G(n, p_\delta)$  and use them to approximate  $\Pr[\text{width}(G_{n,p_\delta}) \leq k]$  for all  $k, 0 \leq k < n$ . Chernoff bounds can be used in the straightforward way (see Appendix A) to show that  $m = \frac{\ln(2/\gamma)}{2\rho^2}$  samples suffice to get within an additive error  $\rho$  with probability at least  $1 - \gamma$ . Notice that  $m$  is independent of  $n$  and  $\delta$ ; and if  $\rho$  and  $\gamma$  are constant, then so is  $m$ . The desired  $k^*$  is just the smallest  $k$  whose estimate is at least  $1/2 + \epsilon + \rho$ .

Now, we know that at least half of the DAGs in  $G(n, p_\delta)$  satisfy  $\mathcal{P}_\delta$ . On the other hand, at least a  $(1/2 + \epsilon)$ -fraction of DAGs in  $G(n, p_\delta)$  satisfy  $\mathcal{P}_{k^*}$ . Thus there must trivially exist DAGs in  $G(n, p_\delta)$  satisfying both. In fact, at least an  $\epsilon$ -fraction of them surely will. Moreover, there is a very simple probabilistic algorithm for finding one: We just need to sample  $O(1/\epsilon)$  DAGs in  $G(n, p_\delta)$  and choose the closest one. Clearly we are overcounting, since the same DAGs may contribute to both probabilities. If we cut below  $1/2$ , however, we may be unlucky enough to find ourselves in the situation where most DAGs in  $G(n, p_\delta)$  satisfying  $\mathcal{P}_\delta$  are not the ones that have small treewidth; and, intuitively, this does not seem too unrealistic, since the graphs in  $G(n, p_\delta)$  with small treewidth will not necessarily fit the distribution better than the ones with large treewidth. The value of  $k^*$  defined as above certainly gives an upper bound on the smallest achievable treewidth. In practice, especially when  $n$  is large, it may be desirable to cut lower than  $1/2$ . The appropriate cutoff value can be found by simply doing a binary search on the interval  $(0, 1/2]$ .

It remains to find the critical value  $p_\delta$ , which can be done using sampling. Note that the values related to treewidth are independent of the target distribution and can be precomputed for the given  $n$ . In order to find  $p_\delta$ , we can do a binary search on the interval  $[0, 1]$ : If the current edge probability is  $p$ , we will approximate  $\mu_\delta(G_{n,p})$  using the sampler from Appendix A, and branch accordingly based on the estimate. The search is continued until  $p$  gets sufficiently close to satisfying  $\mu_\delta(G_{n,p}) = 1/2$ .

A simple example should help make the goals clear. A distribution is called *k-wise independent* if any subset of  $k$  variables is mutually independent (however, there may exist dependencies on larger subsets). Figure 1 shows the curves for a 3-wise independent distribution on 8 random variables.

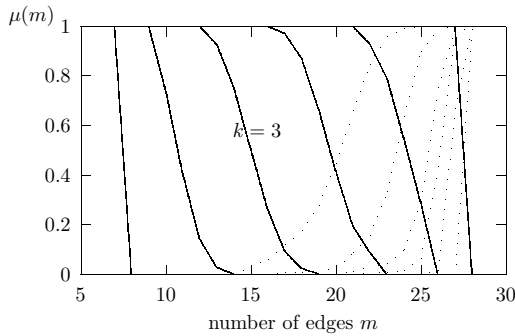


Figure 1: Threshold curves for a 3-wise independent distribution on 8 random variables.

$k = \{1, \dots, 6\}$  (from left to right respectively). For  $k = 7$ , the curve is just  $\mu_m(\mathcal{P}_k) = 1$ . The monotone increasing curves correspond to the property of having  $d_{KL}$  at most  $\delta$ . The leftmost curve is for  $\delta = 0.07$ , and it decreases by 0.01 as we go from left to right; the smaller  $\delta$ , the higher the quality of approximation, thus the smaller the probability of attaining it. The empty graph (treewidth 0) had divergence 0.073. As  $m$  increases, the probability of having small treewidth decreases, while the probability of getting close to the target increases. (Since  $n$  is small, we computed the divergence exactly.) As the random graph evolves, we want to capture the moment when the first probability is *still* high, while

We can hardly expect graphs with treewidth at most 2 to do well on this distribution, since all triples are independent, and their marginals do not reveal any higher-order structure; as we will see this is indeed the case. The  $x$ -axis in Figure 1 corresponds to the number of edges  $m$ , the  $y$ -axis denotes the probability that  $G_{n,m}$  satisfies the property corresponding to a given curve. The monotone decreasing curves correspond to the properties  $\mathcal{P}_k$  for

the second is *already* high. As expected, graphs with treewidth at most 2 are as inaccurate as the empty graph since all triples are independent. Given the desired level of closeness  $\delta$ , we want to find the smallest treewidth  $k$  such that the corresponding curves meet above some cut-off probability. For example, to get within  $d_{KL}$  at most 0.7, we may suggest, say, projecting onto graphs with treewidth 4 (cutting at 0.4). The cut-off value determines the efficiency of finding a model with such  $k$  and  $\delta$  (see discussion above).

**Estimating  $d_{KL}$**  Fix a bounded-treewidth DAG  $G$ . The target distribution is the empirical distribution  $P$  induced by the given sample. Recall that  $d_{KL}(P, G)$  decomposes into the sum of conditional entropies induced by  $G$  (minus the entropy of  $P$ ). Höffgen [10] showed how to estimate these conditional entropies with any fixed additive precision  $\rho$  using polynomially many samples. Fix  $0 < \rho, \gamma < 1$ . More precisely, he showed that a sample of size  $m = m(\gamma, \rho) = O\left(\left(\frac{n}{\rho}\right)^2 \log^2 \frac{n}{\rho} \log \frac{n^{k+1}}{\gamma}\right)$  suffices to obtain good estimations of all induced conditional entropies with probability at least  $1 - \gamma$ , which in turn suffices to estimate  $d_{KL}(P, G)$  with the additive precision  $\rho$ .

**Estimating Treewidth** Finding the treewidth of an arbitrary graph is known to be NP-hard. However, in practice, efficient heuristic algorithms are commonly used to find a suboptimal elimination ordering that often provides a good estimation of the treewidth (e.g., min-degree or max-cardinality orderings). We used here the maximum-cardinality heuristic. The computational time for networks on several hundred nodes is insignificant. Note that the values related to treewidth are independent of the target distribution and can be computed in advance using even more sophisticated algorithms that provide better treewidth estimates.

## 4 Experimental Results

Due to page limit, we discuss an application of the method to a single network known as ALARM (originating from anesthesia monitoring domain); similar results very obtained on other networks (e.g., randomly generated ones). The network has 37 nodes, 46 directed edges, 19 additional undirected edges induced by moralization; the treewidth is 4. A sample of size  $N = 10^4$  was generated using ancestral sampling, inducing the empirical distribution with support on 5570 unique variable assignments. Figure 2 shows the curve illustrating the (estimated) tradeoffs available for  $P$ . For each treewidth bound  $k$ , the curves gives an estimate of the best achievable value of  $W = d_{KL} - H(P)$ . (Recall that  $W$  relates to the log-likelihood as follows:  $LL = -N \cdot W$ .)

The estimate is based on generating 400 random DAGs with 37 nodes and  $m$  edges, for every possible  $m$ . Several points on the curve are worthy of note. The upper-left point  $(0, 23.4)$  corresponds to the model that assumes all 37 variables to be independent. On the other extreme, the lower-right point  $(36, 0)$  corresponds to the clique on 37 nodes, which of course can model  $P$  perfectly, but with exponential complexity. The closer the area under the curve to zero, the easier the distribution (in the sense discussed in this paper). Here we see that the highest gain in accuracy from allowing the model to be more complex occurs up to treewidth 4, less so 5 and 6; by further increasing the treewidth we do not gain much in accuracy.

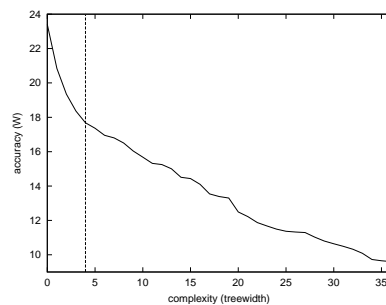


Figure 2: Tradeoff curve for ALARM.

In this sense we succeed, since the distribution was simulated from a treewidth-4 model. (Note, however, that it does not imply that the empirical distribution itself decomposes on a treewidth-4 model. The simplest example of this is when the true distribution is

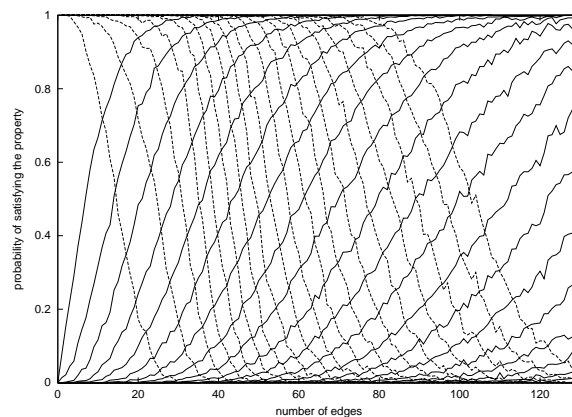


Figure 3: Threshold curves for ALARM

uniform.) Such tradeoff curves are similar to commonly used ROC (Receiver Operating Characteristic) curves; the techniques for finding the cutoff value in ROC curves can be used here as well. Instead of plotting the best achievable distance, we can plot the best distance achievable by at least an  $\epsilon$ -fraction of models in the class, parameterizing the tradeoff curve by  $\epsilon$ . Figure 3 shows the threshold curves. The axes have the same meaning as in Figure 1. Varying sample size and the number of randomly generated DAGs does not change the behavior of the curves in any meaningful way; not surprisingly, increasing these parameters results in smoother curves.

The experiments support the following conclusions: the properties capturing the complexity and accuracy of a model indeed demonstrate a threshold behavior, which can be exploited in determining the best tradeoff for the given distribution; the simple approach based on generating random graphs and using them to approximate the thresholds is indeed capable of capturing the effective width of a distribution.

## References

- [1] A. Barak and P. Erdős. On the maximal number of strongly independent vertices in a random acyclic directed graph. *SIAM J. Algebraic and Discrete Methods*, 5:508–514, 1984.
- [2] A. Beygelzimer and I. Rish. Inference complexity as a model-selection criterion for learning bayesian networks. In *Proceedings of the Eighth International Conference on Principles of Knowledge Representation and Reasoning (KR2002)*, Toulouse, France, 2002.
- [3] B. Bollobás and G. Brightwell. The structure of random graph orders. *SIAM J. Discrete Mathematics*, 10(2):318–335, 1997.
- [4] C. Chow and C. Liu. Approximating discrete probability distributions with dependence trees. *IEEE Trans. on Inf. Theory*, 14:462–467, 1968.
- [5] G.F. Cooper. The computational complexity of probabilistic inference using Bayesian belief networks. *A. Intelligence*, 42(2–3):393–405, 1990.
- [6] T. Cover and J. Thomas. *Elements of information theory*. John Wiley & Sons Inc., New York, 1991.
- [7] R. Dechter. Bucket elimination: A unifying framework for probabilistic inference. In *Proc. Twelfth Conf. on Uncertainty in Artificial Intelligence*, pages 211–219, 1996.
- [8] P. Erdős and A. Rényi. On the evolution of random graphs. *Bull. Inst. Internat. Statist.*, 38:343–347, 1961.
- [9] E. Friedgut and G. Kalai. Every monotone graph property has a sharp threshold. *Proc. of the AMS*, 124(10):2993–3002, 1996.
- [10] K. Höffgen. Learning and robust learning of product distributions. In *Proceedings of the 6th Annual Workshop on Computational Learning Theory*, pages 77–83, 1993.
- [11] F. V. Jensen and F. Jensen. Optimal junction trees. In *Proceedings of the Tenth Conference on Uncertainty and Artificial Intelligence*, 1994.
- [12] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers, 1988.
- [13] N. Srebro. Maximum likelihood bounded Tree-Width markov networks. In *Proceedings of the 17th Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 504–511, 2001.

Note: Appendix may be ignored at the discretion of the Program Committee.

## A The Sampler

To estimate  $\Pr[W(G_{n,p}) \geq \delta]$ , given  $p$  and  $\delta$ , the algorithm just generates  $m$  independent copies of  $G_{n,p}$ , denoted  $G_1, \dots, G_m$ , and outputs  $S_m = \frac{1}{m} \sum_{i=1}^m U_i$  as its estimate of  $\mu = \Pr[W(G_{n,p}) \geq \delta]$ , where  $U_i$  is a random variable indicating whether  $W(G_i) \geq \delta$ . The approximation is parameterized by the error probability  $\gamma$  and accuracy  $\rho$ ; i.e., we want to get a value at most  $\rho$  away from  $\Pr[W(G_{n,p}) \geq \delta]$  with probability at least  $1 - \gamma$  (over the choice of  $G_1, \dots, G_m$ ). By the Chernoff Bound,

$$\Pr[|S_m - \mu| > \rho] < 2e^{-2m\rho^2}.$$

Setting  $m = \frac{\ln(2/\gamma)}{2\rho^2}$  makes the above error probability at most  $\gamma$ , as required. Thus  $O(\frac{\ln(1/\gamma)}{\rho^2})$  samples suffice. The same argument can be used for estimating the probability that  $\text{width}(G_{n,p}) \leq k$ , given  $p$  and  $k$ .

If we have an estimate of  $\mathbf{E}[\text{width}(G_{n,p_\delta})]$ , then Markov's inequality immediately gives an upper bound on  $k^*$  (see Section 3). Indeed, we have

$$\Pr[\text{width}(G_{n,p_\delta}) \leq k] \geq 1 - \frac{\mathbf{E}[\text{width}(G_{n,p_\delta})]}{k}.$$

We have  $\mu_k(p_\delta) \geq 1 - \frac{\mathbf{E}[\text{width}(G_{n,p_\delta})]}{k} \geq \frac{1}{2}$ , yielding  $1 - \mathbf{E}[\text{width}(G_{n,p_\delta})]/k \geq 1/2$ , or  $k \geq 2\mathbf{E}[\text{width}(G_{n,p_\delta})]$ . The upper bound on  $k^*$  is given by the smallest integral  $k$  satisfying the inequality. Note that for estimating the expectation  $\mathbf{E}[\text{width}(G_{n,p})]$ , the number of samples required by the above algorithm is no longer independent of  $n$ . Although  $\text{width}(G_{n,p})$  is bounded (which is crucial, since otherwise no reasonable sampling method could guarantee any approximation), this bound depends on  $n$ , leading to the sample size  $m = \frac{n^2 \ln(2/\gamma)}{2\varepsilon^2}$ .