The Complexity of Distinguishing Markov Random Fields

Andrej Bogdanov[1⋆], Elchanan Mossel[2⋆⋆], and Salil Vadhan[⋆⋆⋆]

[1] Institute for Theoretical Computer Science, Tsinghua University
[2] Dept. of Statistics and Dept. of Computer Sciences, U.C. Berkeley
[3] School of Engineering and Applied Sciences, Harvard University

Abstract. Markov random fields are often used to model high dimensional distributions in a number of applied areas. A number of recent papers have studied the problem of reconstructing a dependency graph of bounded degree from independent samples from the Markov random field. These results require observing samples of the distribution at all nodes of the graph. It was heuristically recognized that the problem of reconstructing the model where there are hidden variables (some of the variables are not observed) is much harder.

Here we prove that the problem of reconstructing bounded-degree models with hidden nodes is hard. Specifically, we show that unless NP = RP,

- It is impossible to decide in randomized polynomial time if two models generate distributions whose statistical distance is at most 1/3 or at least 2/3.
- Given two generating models whose statistical distance is promised to be at least 1/3, and oracle access to independent samples from one of the models, it is impossible to decide in randomized polynomial time which of the two samples is consistent with the model.

The second problem remains hard even if the samples are generated efficiently, albeit under a stronger assumption.

1 Introduction

We study the computational complexity of reconstructing a Markov random field of bounded degree from independent and identically distributed samples at a subset of the nodes.

The problem of reconstructing Markov random fields (MRF) has been recently considered as Markov random fields provide a a very general framework

⋆ andrejb@tsinghua.edu.cn. This work was supported in part by the National Natural Science Foundation of China Grant 60553001, and the National Basic Research Program of China Grants 2007CB807900 and 2007CB807901.

⋆⋆ mossel@stat.berkeley.edu. Supported by a Sloan fellowship in Mathematics, by NSF Career award DMS-0548249, NSF grant DMS-0528488 and ONR grant N0014-07-1-05-06.

⋆⋆⋆ salil@eecs.harvard.edu. Work done while visiting U.C. Berkeley, supported by the Miller Institute for Basic Research in Science, a Guggenheim Fellowship, US-Israel BSF grant 2002246, and ONR grant N00014-04-1-0478.

for defining high dimensional distributions. Much of the interest emanates from the use of such models in biology, see e.g. [1] and a list of related references [2].

Reconstructing Markov random fields where the generating model is a bounded-degree *tree* is one of the major computational problems in evolutionary biology, see e.g. [3, 4]. For tree models the problem of sampling from a given model or calculating the probability of observing a specific sample for a given model are well known to be computationally feasible using simple recursions (also termed "dynamic programming" and "peeling"). Moreover, in the last decade it was shown that the problem of reconstructing a tree model given samples at a subset of the nodes is computationally feasible under mild non-degeneracy conditions, see e.g. [5–7] for some of the best results of this type. (These results often assume that the samples are observed at the leaves of the tree, but they easily extend to the case where some of the observables are internal nodes.)

Following extensive experimental work, Abbeel *et al.* [8] considered the problem of reconstructing bounded-degree (non-tree) graphical models based on factor graphs, and proposed an algorithm with polynomial time and sample complexity. The goal of their algorithm was not to reconstruct the true structure, but rather to produce a distribution that is close in Kullback-Leibler divergence to the true distribution.

In a more recent work [9], it was shown that the generating graph of maximal degree d on n nodes can be efficiently reconstructed in time $n^{O(d)}$ under mild non-degeneracy conditions. Other results on reconstructing the graph have appeared in [10].

Note that all of the results for non-tree models assume that there are no hidden variables. This is consistent with our results described next which show that the problem of reconstructing models with hidden variables is computationally hard.

1.1 Definitions and Main Results

Fix an alphabet Σ. An *undirected model* M over Σ^n consists of an undirected graph G with n vertices and a collection of weight functions $w_e : \Sigma^2 \to \mathbb{R}^{\geq 0}$, one for each edge $e \in E(G)$. The degree of the model is the degree of the underlying graph. To each undirected model M we associate the probability distribution μ_M on Σ^n given by

$$\Pr_{X \sim \mu_M}[X = a] = \frac{\prod_{(u,v) \in E(G)} w_{(u,v)}(a_u, a_v)}{Z_M} \tag{1}$$

where Z_M is the *partition function*

$$Z_M = \sum_{a \in \Sigma^n} \prod_{(u,v) \in E(G)} w_{(u,v)}(a_u, a_v).$$

This probability distribution μ_M is called the *Markov Random Field* of M. (Throughout, we will only work with models where $Z_M \neq 0$ so that μ_M is well-defined.)

As an example, consider the special case that $\Sigma = \{0, 1\}$ and all the weight functions are the NAND function. Then an assignment a has nonzero weight iff it is the characteristic vector of an independent set in the graph, Z_M counts the number of independent sets in the graph, and μ_M is the uniform distribution on the independent sets in the graph. For even this special case, it is NP-hard to compute Z_M given M is NP-hard, even approximately [11] and in bounded-degree graphs [12]. Due to the close connection between approximate counting and sampling [13], it follows that given a bounded-degree model M, it is infeasible to sample from the distribution μ_M (unless NP = RP). Here, we are interested in computational problems of the reverse type: given samples, determine M. Nevertheless, our techniques are partly inspired by the line of work on the complexity of counting and sampling.

We note that in standard definitions of Markov Random Fields, there is a weight function w_C for every *clique* C in the graph (not just edges), and the probability given to an assignment a is proportional to the product of the weights of all cliques in the graph. Our definition corresponds to the special case where all cliques of size greater than 2 have weight functions that are identically one. This restriction only makes our hardness results stronger. (Note that in bounded-degree graphs, there are only polynomially many cliques and they are all of bounded size, so our restriction has only a polynomial effect on the representation size.)

Markov Random fields model many stochastic processes. In several applications of interest one is given samples from the distribution μ_M and is interested in "reconstructing" the underlying model M. Often the observer does not have access to all the vertices of M, but only to a subset $V \subseteq \{1, \ldots, n\}$ of "revealed" vertices. We call this a model with *hidden nodes* $M \mid V$ and denote the corresponding distribution by $\mu_{M|V}$.

We are interested in the computational complexity of reconstructing the model M given samples from $\mu_{M|V}$. Of course, the model M may not be uniquely specified by $\mu_{M|V}$ (e.g. M may have a connected component that is disjoint from V), so one needs to formalize the question more carefully. Since we are interested in proving hardness results, we take a minimalist view of reconstruction: Any algorithm that claims to reconstruct M given samples from $\mu_{M|V}$ should in particular be able to distinguish two models M and M' when their corresponding distributions $\mu_{M|V}$ and $\mu_{M'|V}$ are statistically far apart.

As a first step towards understanding this question, we consider the following computational problem:

Problem dDIST

INPUT: Two models M_0 and M_1 over Σ^n of degree d, a set $V \subseteq \{1, \ldots, n\}$.

PROMISE: Z_{M_0} and Z_{M_1} are nonzero.

YES INSTANCES: The statistical distance between $\mu_{M_0|V}$ and $\mu_{M_1|V}$ is at most $1/3$.

NO INSTANCES: The statistical distance between $\mu_{M_0|V}$ and $\mu_{M_1|V}$ is at least $2/3$.

Here, the *statistical distance* (a.k.a. total variation distance) between two distributions μ and ν on a set Ω is the quantity

$$\mathrm{sd}(\mu,\nu) = \max_{T\subseteq\Omega} \left|\Pr_{X\sim\mu}[X\in T] - \Pr_{X\sim\nu}[X\in T]\right|.$$

The computational problem dDIST, and all others we consider in this paper, are *promise problems*, which are decision problems where the set of inputs are restricted in some way, and we do not care what answer is given on inputs that are neither yes or no instances or violate the promise. Languages are special cases where all strings are either yes or no instances. For more about promise problems, see the survey by Goldreich [14].

Next, we consider a problem that seems much more closely related to (and easier than) reconstructing a model from samples. Here, the distinguisher is given two candidate models for some probabilistic process, as well as access to samples coming from this process. The goal of the distinguisher then is to say which is the correct model for this process.

Problem dSAMP

INPUT: Two models M_0 and M_1 over Σ^n of degree d, a set $V \subseteq \{1,\ldots,n\}$.

PROMISE: Z_{M_0} and Z_{M_1} are nonzero, and the statistical distance between $\mu_0 = \mu_{M_0|V}$ and $\mu_1 = \mu_{M_1|V}$ is at least $1/3$.

PROBLEM: Given oracle access to a sampler S that outputs independent samples from either μ_0 or μ_1, determine which is the case.

More precisely, the distinguishing algorithm D is required to satisfy the condition

$$\Pr[D^{S_b}(M_0,M_1,V) = b] > 2/3 \qquad \text{for } b\in\{0,1\} \tag{2}$$

where S_b denotes the sampler for μ_b and the probability is taken both over the randomness of the sampler and the randomness of D.

Our main results are that both of these problems are hard:

Theorem 1. *If there is a deterministic (resp., randomized) polynomial-time algorithm for* 3DIST*, then* NP = P *(resp.,* NP = RP*). This holds even if we restrict to models over the alphabet* $\Sigma = \{0,1\}$*.*

Theorem 2. *If there is a randomized polynomial-time algorithm for* 3SAMP*, then* NP = RP*. This holds even if we restrict to models over the alphabet* $\Sigma = \{0,1\}$*.*

These characterizations are the best possible: If NP = RP, both DIST and SAMP have efficient algorithms. See Appendix A.

The proofs of the two theorems are based on the fact that the Markov Random Field of a suitably chosen model can approximate the uniform distribution over satisfying assignments of an arbitrary boolean circuit. By revealing one node, we can then use an algorithm for either dDIST or dSAMP to distinguish the case that the first variable is 1 in all satisfying assignments from the case that the first variable is 0 in all satisfying assignments, which is an NP-hard problem.

2 Sampling Satisfying Assignments with a Markov Random Field

In this section, we establish the key lemma that is used in all of our hardness results — given a boolean circuit C, we can construct a model whose Markov Random Field corresponds to the uniform distribution on satisfying assignments of C.

Lemma 1. *There is a polynomial-time algorithm R that on input a circuit $C : \{0,1\}^n \to \{0,1\}$ produces an undirected model M of degree 3 over alphabet $\{0,1\}$ with a collection of special vertices $v_1, \ldots, v_n$ such that $Z_M \neq 0$ and if C is satisfiable, then the statistical distance between a random satisfying assignment of C and the Markov Random Field of M restricted to $v_1, \ldots, v_n$ is at most 2^{-n}.*

This proof is an extension of the standard reduction from circuit satisfiability to independent set: For each gate in the circuit and every possible assignment to the wires at this gate we have a vertex in the graph, and we put an edge between vertices corresponding to inconsistent assignments. (For the output gate, we remove those vertices corresponding to non-satisfying assignments.) Then the uniform distribution on *maximum* independent sets in the graph corresponds exactly to the uniform distribution on satisfying assignments in the circuit. However, the independent set model also gives weight to independent sets that are not maximum.

The weight corresponding to maximum independent sets can be magnified using the "blow-up" technique of [13, 11], where we clone every vertex polynomially many times and replace each edge with complete bipartite graph between the clones of the endpoints. However, this results in a graph of polynomially large degree. In order to obtain a degree 3 model, we use the more general weight functions allowed in a Markov Random Field to achieve the same blow-up effect with many fewer edges. Specifically, we can force all clones of a vertex to have the same value by connecting them in a cycle with appropriate weight functions, and can also use the weights to magnify the weight of large sets. Then we can spread out the edges of the original graph among the clones in a way that the degree increases only by 1.

Proof. Consider the following polynomial-time algorithm that, on input a circuit C of size s, produces an undirected model M over alphabet $\{0,1\}$. We assume without loss of generality that each gate has fanin two and that all NOT gates are at the input level. For each gate g of C, including the input gates, and each consistent assignment α of values to the wires incident to this gate, the model M has $r = 8s$ vertices $v_{g,\alpha,1}, \ldots, v_{g,\alpha,r}$. (Note that for each gate g, there are at most $2^3 = 8$ possible assignments α.) For the output gate, we only consider assignments consistent with the circuit accepting. For every i, connect the vertices $v = v_{g,\alpha,i}$ and $u = v_{g,\alpha,i+1}$ by an edge with the following weighted "inner

constraint":

$$w_{in}(a_u, a_v) = \begin{cases} 1 & \text{if } a_u = a_v = 0 \\ 2 & \text{if } a_u = a_v = 1 \\ 0 & \text{otherwise.} \end{cases}$$

For any pair of gates g, h where either $g = h$ or g and h are connected, and any pair of assignments α for g and β for h that are inconsistent, add the following "outer constraint' between $v = v_{g,\alpha,i}$ and $u = v_{h,\beta,j}$, where i (resp. j) is the first index that has not been used in any outer constraint for g (resp. h):

$$w_{out}(a_u, a_v) = \begin{cases} 0 & \text{if } a_u = a_v = 1 \\ 1 & \text{otherwise.} \end{cases}$$

The first type of constraint ensures that all representatives of the same gate-assignment pair are given the same value, and favors values that choose the assignment. The second type of constraint ensures that the assignments to the vertices of the model are consistent with circuit evaluation.

Assume that C is satisfiable, and look at the distribution induced by the Markov Random Field of M on the vertices $v_1, \ldots, v_n$, where $v_i = v_{x_i,1,1}$ represent the inputs of C. For every satisfying assignment α of C, consider the corresponding assignment α' of M that assigns value 1 to all vertices representing gate-assignment pairs consistent with the evaluation of C on input α, and 0 to all others. This gives α' relative weight 2^{sr} in the Markov Random Field.

We now argue that the combined weight of all other assignments of M cannot exceed $2^{-s} \cdot 2^{sr}$, and the claim follows easily from here. By construction, every assignment of M with nonzero weight assigns 1 to at most one group of vertices $v_{g,\alpha,1}, \ldots, v_{g,\alpha,r}$ for every gate g, and if the assignment does not represent a satisfying assignment of C then at least one gate must have no group assigned 1. For each group assigned 1, there are at most 8 ways to choose the assignment from each group, and each such assignment contributes a factor of 2^r to the weight, so the total weight of non-satisfying assignments is at most

$$\sum_{k=0}^{s-1} \binom{s}{k} \cdot (8 \cdot 2^r)^k \leq 2^s \cdot 8^s \cdot 2^{(s-1)r} \leq 2^{-s} \cdot 2^{sr}$$

by our choice of r. □

3 Hardness of 3DIST and 3SAMP

In this section we prove Theorems 1 and 2. For both, we will reduce from the following NP-hard problem.

Problem CKTDIST

INPUT: A circuit C (with AND, OR, NOT gates) over $\{0,1\}^n$.

PROMISE: C is satisfiable.

YES INSTANCES: All satisfying assignments of C assign the first variable 1.

NO INSTANCES: All satisfying assignments of C assign the first variable 0.

Lemma 2. *If* CKTDIST *has a polynomial-time (resp., randomized polynomial-time) algorithm, then* NP = P *(resp.,* NP = RP*).*

Proof. This follows from a result of Even, Selman, and Yacobi [15], who showed that given two circuits (C_0, C_1) where it is promised that exactly one is satisfiable, it is NP-hard to distinguish the case that C_0 is satisfiable from the case that C_1 is satisfiable. This problem is easily seen to be equivalent to CKTDIST by setting $C(b,x) = C_b(x)$. (The interest of [15] in this problem was the fact that it is in the promise-problem analogue NP $\cap$ coNP, whereas there cannot be NP-hard languages in NP $\cap$ coNP unless NP = coNP.) □

Now we use Lemma 1 to reduce CKTDIST to 3DIST and 3SAMP.

Proof (of Theorem 1). To prove Theorem 1, let's assume for sake of contradiction that there is an efficient algorithm D for 3DIST. For simplicity, we assume that D is deterministic; the extension to randomized algorithms is straightforward.

Given a satisfiable circuit C, we will to use the distinguishing algorithm D to distinguish the case that all satisfying assignments assign the first variable 1 from the case that all satisfying assignments assign the first variable 0. First, using Lemma 1, we turn the circuit C into an undirected model M and let v be the variable corresponding to the first variable of C. Then $\mu_{M|\{v\}}$ is a Bernoulli random variable that outputs 1 with probability approximately equal (within $\pm 2^{-n}$) to the fraction of satisfying assignments that assign the first variable 1.

Next, let M' be any model where the node v is always assigned 1 in $\mu_{M'}$. (For example, we can have a single edge (u,v) with weight function $w_{(u,v)}(a_u, a_v) = a_u a_v$.)

Then $\mu_{M|\{v\}}$ and $\mu_{M'|\{v\}}$ have statistical distance at most $2^{-n} \leq 1/3$ if C is a NO instance of CKTDIST, and have statistical distance at least $1 - 2^{-n} \geq 2/3$ if C is a YES instance. Thus, $D(M, M', \{v\})$ correctly decides CKTDIST, and NP = P. □

Proof (of Theorem 2). Similarly to the previous proof, we reduce CKTDIST to 3SAMP: Given a circuit C, define the circuits $C_0(x_1, x_2, \ldots, x_n) = C(x_1, \ldots, x_n)$ and $C_1(x_1, x_2, \ldots, x_n) = C(\neg x_1, x_2, \ldots, x_n)$. Note that if all satisfying assignments of C assign the first variable value b, then all satisfying assignments to C_b assign the $x_1 = 0$ and all satisfying assignments to $C_{\neg b}$ assign $x_1 = 1$. Now, we apply Lemma 1 to construct models M_0 and M_1 corresponding to C_0 and C_1, and we reveal only the vertex $V = \{v_1\}$ corresponding to the variable x_1. (Note that $\mu_{M_0|V}$ and $\mu_{M_1|V}$ have statistical distance at least $1 - 2 \cdot 2^{-n}$.) Given a randomized algorithm A for 3SAMP, we run $A^S(M_0, M_1)$ where S is the sampler

that always outputs 0. If all satisfying assignments of C assign $x_1 = b$ then S is 2^{-n}-close in statistical distance to $S_b \sim \mu_{M_b|V}$. Thus

$$\Pr[A^S(M_0, M_1) = b] \geq \Pr[A^{S_b}(M_0, M_1) = b] - \text{poly}(n) \cdot 2^{-n} \geq 2/3 - o(1)$$

and the construction gives a randomized algorithm for CKTDIST. □

4 On the samplability of the models

One possible objection to the previous results is that the Markov Random Fields in question are not required to be samplable. In some of the applications we have in mind, the model represents a natural (physical, biological, sociological,...) process. If we believe that nature itself is a computationally efficient entity, then it makes sense to assume that the models we are trying to reconstruct will be efficiently samplable. It is natural to ask if the problem of distinguishing Markov Random Fields remains hard in this setting too.

Problem EFFSAMP

INPUT: Two models M_0 and M_1 over Σ^n of degree d, a set $V \subseteq \{1, \ldots, n\}$, and a parameter s in unary.

PROMISE: Z_{M_0} and Z_{M_1} are nonzero, the statistical distance between $\mu_0 = \mu_{M_0|V}$ and $\mu_1 = \mu_{M_1|V}$ is at least $1/3$, and both μ_{M_0} and μ_{M_1} are 2^{-n}-close in statistical distance to distributions samplable by circuits of size at most s.

PROBLEM: Given oracle access to a sampler S that outputs independent samples from either μ_0 or μ_1, determine which is the case.

We have the following hardness result for EFFSAMP. Here CZK is the class of decision problems that have "computational zero-knowledge proofs". (See [16] for a definition.)

Theorem 3. *If* EFFSAMP *has a polynomial-time randomized algorithm, then* CZK = BPP.

A slightly weaker version of this theorem says that if EFFSAMP has a polynomial-time randomized algorithm, then one-way functions, or equivalently pseudorandom generators [17], do not exist. (See [16] for definitions of both one-way functions and pseudorandom generators.) To prove this, we observe that an algorithm for EFFSAMP can be used to break any candidate pseudorandom generator G: Convert G into an undirected model $M_0 \mid V$ using Lemma 1, and let $M_1 \mid V$ be a model whose Markov Random Field is uniform. Then the algorithm for EFFSAMP can be used to tell if a sample came from the pseudorandom generator or from the uniform distribution, thereby breaking the generator. Theorem 3 is stronger because it is known that if one-way functions exist, then CZK = PSPACE ≠ BPP [18–20].

To prove the actual theorem, we use a result of Ostrovsky and Wigderson [21], which says that if CZK ≠ BPP then there must exist an "auxiliary-input pseudorandom generator", which can also be broken by the same argument.

Proof. Suppose that CZK $\neq$ BPP. Then by Ostrovsky and Wigderson [21], there exists an *auxiliary-input one-way function*: This is a polynomial-time computable function $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ such that for every polynomial p and polynomial-size circuit C, there exist infinitely many a such that

$$\Pr_{x\sim\{0,1\}^n}[f(a, C(a, f(a,x))) = f(a,x)] < 1/p(n)$$

where n is the length of a. By Håstad et al. [17], it follows that there is also an *auxiliary-input pseudorandom generator*: This is a polynomial-time computable function $G : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^{n+1}$ such that for every polynomial-size circuit family D and every polynomial p, there exist infinitely many a such that

$$\left|\Pr_{y\sim\{0,1\}^{n+1}}[D(a,y)] - \Pr_{x\sim\{0,1\}^n}[D(a, G(a,x))]\right| < 1/p(n).$$

It follows by a standard hybrid argument that for every polynomial-size oracle circuit D whose oracle provides independent samples from a given distribution we have that

$$\left|\Pr[D^U(a)] - \Pr[D^{G_a}(a)]\right| < 1/p(n).$$

for infinitely many a, where U is (a sampler for) the uniform distribution on $\{0,1\}^{n+1}$ and G_a is the output distribution of $G(a,x)$ when x is chosen uniformly from $\{0,1\}^n$. We show that if EFFSAMP has a polynomial-time randomized algorithm A, then for every polynomial-time computable G there is a circuit D such that

$$\left|\Pr[D^U(a)] - \Pr[D^{G_a}(a)]\right| > 1/4.$$

for every a. Fix an a of length n, let $C_a(x,y)$ be the circuit

$$C_a(x,y) = \begin{cases} 1 & \text{if } y = G(a,x) \\ 0 & \text{otherwise} \end{cases}$$

Apply Lemma 1 to circuit C_a to obtain a model M_a, and let V be the set of nodes of M_a corresponding to the input y of C_a. Then the Markov Random Field of M_a is 2^{-n} close to the distribution G_a. Let $M' \mid V$ be a model whose Markov Random Field is the uniform distribution over $\{0,1\}^{n+1}$. Then $D^?(a) = A^?(M_a, M')$ is the desired circuit. □

Acknowledgments

We thank the anonymous referees for helpful comments on the presentation.

References

1. Friedman, N.: Infering cellular networks using probalistic graphical models. Science (2004)
2. Kasif, S.: Bayes networks and graphical models in computational molecular biology and bioinformatics, survey of recent research. http://genomics10.bu.edu/bioinformatics/kasif/bayes-net.html (2007)

3. Felsenstein, J.: Inferring Phylogenies. Sinauer, New York, New York (2004)
4. Semple, C., Steel, M.: Phylogenetics. Volume 22 of Mathematics and its Applications series. Oxford University Press (2003)
5. Erdös, P.L., Steel;, M.A., Székely, L.A., Warnow, T.A.: A few logs suffice to build (almost) all trees (part 1). Random Structures Algorithms **14**(2) (1999) 153–184
6. Mossel, E.: Distorted metrics on trees and phylogenetic forests. IEEE Computational Biology and Bioinformatics **4** (2007) 108–116
7. Daskalakis, C., Mossel, E., Roch, S.: Optimal phylogenetic reconstruction. In: Proceedings of the thirty-eighth annual ACM symposium on Theory of computing (STOC 2006). (2006) 159–168
8. Abbeel, P., Koller, D., Ng, A.Y.: Learning factor graphs in polynomial time and sampling complexity. Journal of Machine Learning Research **7** (2006) 1743–1788
9. Bresler, G., Mossel, E., Sly, A.: Reconstruction of Markov random fields from samples: Some easy observations and algorithms (2008) These proceedings. Preliminary version on arxiv http://front.math.ucdavis.edu/0712.1402.
10. Wainwright, M.J., Ravikumar, P., Lafferty, J.D.: High dimensional graphical model selection using ℓ_1-regularized logistic regression. In: Proceedings of the NIPS. (2006)
11. Sinclair, A.: Algorithms for Random Generation and Counting: A Markov chain Approach. Progress in Theoretical Computer Science. Birkhäuser (1993)
12. Luby, M., Vigoda, E.: Fast convergence of the Glauber dynamics for sampling independent sets. Random Struct. Algorithms **15**(3-4) (1999) 229–241
13. Jerrum, M., Valiant, L.G., Vazirani, V.V.: Random generation of combinatorial structures from a uniform distribution. Theor. Comput. Sci. **43** (1986) 169–188
14. Goldreich, O.: On promise problems: a survey. In: Theoretical computer science. Volume 3895 of Lecture Notes in Comput. Sci. Springer, Berlin (2006) 254–290
15. Even, S., Selman, A.L., Yacobi, Y.: The complexity of promise problems with applications to public-key cryptography. Information and Control **61**(2) (1984) 159–173
16. Goldreich, O.: Foundations of cryptography. Cambridge University Press, Cambridge (2001) Basic tools.
17. Håstad, J., Impagliazzo, R., Levin, L.A., Luby, M.: A pseudorandom generator from any one-way function. SIAM Journal on Computing **28**(4) (1999) 1364–1396 (electronic)
18. Goldreich, O., Micali, S., Wigderson, A.: Proofs that yield nothing but their validity, or All languages in NP have zero-knowledge proof systems. Journal of the Association for Computing Machinery **38**(3) (1991) 691–729
19. Impagliazzo, R., Yung, M.: Direct minimum-knowledge computations (extended abstract). In Pomerance, C., ed.: Advances in Cryptology—CRYPTO '87. Volume 293 of Lecture Notes in Computer Science., Springer-Verlag, 1988 (16–20 August 1987) 40–51
20. Ben-Or, M., Goldreich, O., Goldwasser, S., Håstad, J., Kilian, J., Micali, S., Rogaway, P.: Everything provable is provable in zero-knowledge. In Goldwasser, S., ed.: Advances in Cryptology—CRYPTO '88. Volume 403 of Lecture Notes in Computer Science., Springer-Verlag, 1990 (21–25 August 1988) 37–56
21. Ostrovsky, R., Wigderson, A.: One-way functions are essential for non-trivial zero knowledge. In: Proc. 2nd Israel Symp. on Theory of Computing and Systems, IEEE Computer Society Press (1993) 3–17
22. Sahai, A., Vadhan, S.: A complete problem for statistical zero knowledge. Journal of the ACM **50**(2) (March 2003) 196–249 Extended abstract in *FOCS '97*.

23. Babai, L., Moran, S.: Arthur-Merlin games: A randomized proof system and a hierarchy of complexity classes. Journal of Computer and System Sciences **36** (1988) 254–276

A Converse theorem

Theorem 4. *If* NP = RP*, then for every d there are randomized polynomial-time algorithms for* dDIST *and* dSAMP*.*

To prove Theorem 4, we use the following results of Jerrum, Valiant, Vazirani [13].

Theorem 5. *Assume* NP = RP*. Then there exists*

1. *A randomized polynomial-time sampling algorithm* Sample *that on input a satisfiable circuit $C : \{0,1\}^n \to \{0,1\}$ and $\varepsilon > 0$ (represented in unary), has an output distribution that is ε-close in statistical distance to the uniform distribution on the satisfying assignments of C. and and*
2. *A randomized polynomial-time sampling algorithm* Count *that on input a circuit $C : \{0,1\}^n \to \{0,1\}$ and $\varepsilon > 0$ (represented in unary) such that with high probability*

$$|C^{-1}(1)| \leq \text{Count}(C, \varepsilon) \leq (1+\varepsilon)|C^{-1}(1)|$$

Now we assume NP = RP and describe algorithms for dDIST and dSAMP.

Algorithm for DIST: Using part (1) of Theorem 5, we can sample from a distribution close to the Markov Random Field $M \mid V$. To see this, consider the circuit C that takes as inputs an assignment $x \in \Sigma^n$, and numbers $t_e \in \mathbb{N}$, one for each edge e of M and outputs

$$C(x, e, w) = \begin{cases} 1, & \text{if } t_e \leq w_e(x_e) \text{ for all } e \\ 0, & \text{otherwise.} \end{cases}$$

Conditioned on $C(x, e, w) = 1$, for a uniformly chosen triple (x, e, w) the input $x \sim \Sigma^n$ follows exactly the distribution μ_M. Using the above theorem, there is then an algorithm which on input (M, V) outputs a sample from a distribution that is 1/9-close (in statistical distance) to $\mu_{M|V}$. Let us use $C_{M,V}$ as the sampling circuit obtained by hardwiring M and V as inputs to the algorithm A.

Now given an input M_0, M_1, V for dDIST, we produce the circuits $C_0 = C_{M_0,V}$ and $C_1 = C_{M_1,V}$. Note that if $\text{sd}(\mu_0, \mu_1) > 2/3$ then the statistical distance between the output distributions of these two circuits is $> 2/3 - 1/9 = 5/9$, and if $\text{sd}(\mu_0, \mu_1) < 1/3$ then the distance is $< 1/3 + 1/9 = 4/9$. The problem of distinguishing circuits with large statistical distance from those with small statistical distance is known to be in the complexity class AM [22], which collapses to BPP under the assumption that NP = RP [23].

Algorithm for SAMP: First, we may assume that the statistical distance between the distributions μ_0 and μ_1 is as large as $9/10$: Instead of working with the original models, take 40 independent copies of each model; now each sample of this new model will correspond to 40 independent samples of the original model. The statistical distance increases from $1/3$ to $9/10$ by the following inequality:

Claim. Let μ, ν be arbitrary distributions, and μ^k, ν^k consist of k independent copies of μ, ν, respectively. Then

$$1 - \exp(k \cdot \mathrm{sd}(\mu, \nu)^2/2) \leq \mathrm{sd}(\mu^k, \nu^k) \leq k \cdot \mathrm{sd}(\mu, \nu).$$

Using part (2) of Theorem 5, for every partial configuration $a \in \Sigma^V$, we can efficiently compute approximations $p_0(a), p_1(a)$ such that

$$p_0(a) \leq \mu_0(a) \leq 2p_0(a) \qquad \text{and} \qquad p_1(a) \leq \mu_1(a) \leq 2p_1(a),$$

where $\mu_i(a) = \Pr_{X \sim \mu_i}[X = a]$. Now consider the following algorithm D: On input M_0, M_1, V, generate a sample a from S, output 0 if $p_0(a) > p_1(a)$ and 1 otherwise. Then, assuming the counting algorithm of Theorem 5 returns the correct answer, we have:

$$\begin{aligned}\Pr[D^{S_0}(M_0, M_1, V) = 0] &\geq \sum\nolimits_{a:\mu_0(a) > 2\mu_1(a)} \mu_0(a) \\ &\geq \sum\nolimits_{a:\mu_0(a) > \mu_1(a)} \mu_0(a) - \sum\nolimits_{a:2\mu_1(a) \geq \mu_0(a) > \mu_1(a)} \mu_0(a).\end{aligned}$$

The first term is at least as large as $\mathrm{sd}(\mu_0, \mu_1) \geq 9/10$. For the second term, we have

$$\begin{aligned}\sum\nolimits_{a:2\mu_1(a) \geq \mu_0(a) > \mu_1(a)} \mu_0(a) &\leq \sum\nolimits_{a:2\mu_1(a) \geq \mu_0(a) > \mu_1(a)} 2\mu_1(a) \\ &\leq 2 \cdot \sum\nolimits_{a:\mu_0(a) > \mu_1(a)} \mu_1(a) \\ &\leq 2 \cdot (1 - \mathrm{sd}(\mu_0, \mu_1)) = 1/5.\end{aligned}$$

It follows that $\Pr[D^{S_0}(M_0, M_1, V) = 0] > 2/3$, and by the same argument $\Pr[D^{S_1}(M_0, M_1, V) = 1] > 2/3$.