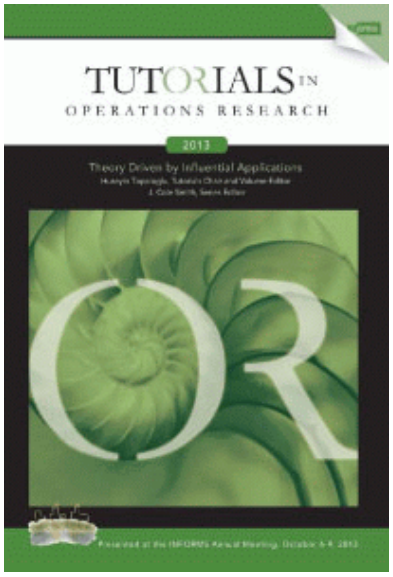
This article was downloaded by: [216.73.216.187] On: 14 July 2026, At: 12:34
Publisher: Institute for Operations Research and the Management Sciences (INFORMS)
INFORMS is located in Maryland, USA



INFORMS TutORials in Operations Research

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Correlation Decay Method for Decision, Optimization, and Inference in Large-Scale Networks

David Gamarnik

To cite this entry: David Gamarnik. Correlation Decay Method for Decision, Optimization, and Inference in Large-Scale Networks. *In* INFORMS TutORials in Operations Research. Published online: 14 Oct 2014; 108-121.
https://doi.org/10.1287/educ.2013.0119

Full terms and conditions of use: https://pubsonline.informs.org/Publications/Librarians-Portal/PubsOnLine-Terms-and-Conditions

This article may be used only for the purposes of research, teaching, and/or private study. Commercial use or systematic downloading (by robots or other automatic processes) is prohibited without explicit Publisher approval, unless otherwise noted. For more information, contact permissions@informs.org.

The Publisher does not warrant or guarantee the article's accuracy, completeness, merchantability, fitness for a particular purpose, or non-infringement. Descriptions of, or references to, products or publications, or inclusion of an advertisement in this article, neither constitutes nor implies a guarantee, endorsement, or support of claims made of that product, publication, or service.

Copyright © 2013, INFORMS

Please scroll down for article—it is on subsequent pages

With 12,500 members from nearly 90 countries, INFORMS is the largest international association of operations research (O.R.) and analytics professionals and students. INFORMS provides unique networking and learning opportunities for individual professionals, and organizations of all types and sizes, to better understand and use O.R. and analytics tools and methods to transform strategic visions and achieve better outcomes.
For more information on INFORMS, its publications, membership, or meetings visit http://www.informs.org

TUTORIALS IN
OPERATIONS RESEARCH
INFORMS 2013

© 2013 INFORMS | ISBN 978-0-9843378-4-2
http://dx.doi.org/10.1287/educ.2013.0119

Correlation Decay Method for Decision, Optimization, and Inference in Large-Scale Networks

David Gamarnik
Massachusetts Institute of Technology, Cambridge, Massachusetts 02140, gamarnik@mit.edu

Abstract Many inference and decision problems on networks involve solving hard algorithmic problems on graphs. Even when the underlying problem admits a polynomial time algorithm, such an algorithm can be impractical simply because of the sheer size of the underlying network, as many technological, communication, and natural networks may easily be of the order of millions or even billions of nodes. Thus one often has to resort to faster distributed (local) algorithms, which can be run in parallel on the underlying networks, providing the required time and computation effort scales. The correlation decay property, which can be established for a broad class of networks exhibiting random topology, random costs, or randomized decisions, is a tool that provides a rigorous justification for the validity of such local algorithms. In this tutorial we illustrate algorithmic methods based on the correlation decay property on several examples. Starting with the PageRank algorithm for search and navigation on the Internet, we will proceed to applications of the correlation decay method to problems in wireless communication, combinatorial optimization problems on graphs, and, finally, to the problem of network learning—a problem in the machine learning field.

Keywords networks; local algorithms; graphical models; long-range independence

Downloaded from informs.org by [216.73.216.187] on 14 July 2026, at 12:34 . For personal use only, all rights reserved.

1. Introduction

Many contemporary technological, communication, and social networks are of enormous scale and are growing in size at a rapid speed. The World Wide Web graph involves about a quarter of a billions of websites. The number of users of Facebook, Twitter, and similar networks is on the order of hundreds of millions. Many naturally occurring networks have enormous sizes as well. For example, the number of neurons in a human brain is of the order of 100 billion. Inference, decision and optimization problems on such graphs often involve solving NP-hard algorithmic problems, which renders these tasks problematic even on the moderate size problems. Even when the underlying problems can be nominally solved using polynomial time algorithms, the scale of the underlying networks often makes such algorithms impractical. A viable alternative is to seek distributed algorithms that can be run in parallel on different parts of the network using only local information, allowing for computation times to be bounded as a function the network size. Traditionally, one of the core computer science disciplines, the field of local algorithms has rapidly expanded recently to other domains including operations research, electrical engineering, economics, physics, and combinatorics as a way of dealing with network scales.

What makes it possible to build a local algorithm that is capable of solving say a global optimization problem defined on some network? It turns out that the correlation decay property, which can be established for a broad class of network models, provides a useful rigorous justification for the validity of such local algorithms. Loosely speaking, the correlation decay principle can be described as follows. Suppose every node i of a network that

we view as a (simple) graph $\mathbb{G}$ with node set V and edge set E is associated with some (possibly random) variable X_i. Loosely speaking, we say that the network $\mathbb{G}$ exhibits the correlation decay property (sometimes also called the long-range independence property) if the random variable X_i is asymptotically independent from the graph $\mathbb{G}$ outside of some small neighborhood of i; namely, fix $r > 0$ and consider an arbitrary change of the graph $\mathbb{G}$ as well as the associated random variables X_j outside of the depth r neighborhood $B(i, r)$ of i. The model exhibits the correlation decay property if the new random variable $\tilde{X}_i$ associated with i is asymptotically close to X_i as r increases, in some appropriate sense. Clearly, having such a property can be potentially very useful for inferring the values X_i, $i \in V$, as one can try to compute the values X_i based on a small size neighborhood $B(i, r)$ containing i as opposed to the entire graph $\mathbb{G}$, resulting in a significant computation reduction. The correlation decay property thus provides the justification for such an approximation. Whether the model exhibits or does not exhibit the correlation decay property depends on model details, and establishing such a property is an interesting technical challenge on its own. The probabilistic aspects that underline the randomness of variables X_i and the associated long-range independence property (or the lack of thereof) can be driven by many stochastic primitives, including randomness of the topology of the underlying graph $\mathbb{G}$, randomness of the cost associated with some optimization problem for which X_i correspond to the optimal decision variables, or randomization of the choices for decision variables X_i themselves, even when the underlying problem is completely deterministic. All variants will be discussed in this tutorial.

The concept of correlation decay principle was considered for the first time in the context of statistical physics models. In particular, the problem of uniqueness of so-called Gibbs measures on infinite graphs, studied for the first time by Dobrushin in his seminal papers (Dobrushin [9]; for book references see Simon [31], Georgii [17]) by means of the correlation decay method, led to the so-called Dobrushin–Shlosman mixing criteria. This criteria is one of the earliest examples of the correlation decay property. Since then, however, the concept of correlation decay expanded beyond statistical physics applications, and in this tutorial we will discuss it in the context of problems closer to the operations research domain.

We will begin with a simple illustration of the idea based on the celebrated PageRank algorithm used for navigation and search on the Internet. We will show that the rank assigned to a node of a graph (websites) depends asymptotically only on a small graph-theoretic neighborhood around this node. As a result, when viewed as probability weights assigned to nodes, the ranks exhibit the correlation decay property. This observation leads to a very simple distributed algorithm for computing ranks of nodes in the Internet graphs and the algorithm for updating them when portions of the graph change. The latter is of particular relevance, since the World Wide Web graph is extremely dynamic; changes occur all the time and one cannot afford to recompute the PageRanks for all of the nodes simultaneously once a few websites are added to or deleted from the Web. This example is the subject of §2. In §3, we turn to the problem of capacity estimation in wireless networks. Using a broadly studied independent set model of a wireless communication network, we will show that, although nominally the problem involves solving a graph counting problem that is #P-hard, a fast local approximation algorithm can be constructed for a certain range of the parameters for which the model exhibits the correlation decay property. A combinatorial optimization problem of finding a largest independent set of a graph is discussed in §4. Although the problem is known to be NP-hard and is also known not to admit a polynomial time approximation algorithm (namely, it is a so-called MAX-SNP-hard problem), we will discover that when the nodes of the graph are equipped with random independent and identically distributed (i.i.d.) weights, the problem in some cases surprisingly becomes tractable and admits a fast local approximation algorithm, and in particular does admit a polynomial time approximation scheme, for certain weight distributions, thanks again to the correlation decay property. Finally, in §5 we turn to a topic that was recently very actively studied in the field of

Downloaded from informs.org by [216.73.216.187] on 14 July 2026, at 12:34 . For personal use only, all rights reserved.

machine learning—graphical model reconstruction based on observed samples. Here the problem formulation is reversed from problems discussed in previous sections, in the sense that the distribution of X_u, $u \in V$ is assumed to be given either explicitly or in the form of an empirical sample, and the goal is to faithfully reconstruct the underlying graph $\mathbb{G}$. We will show that the correlation decay property, when it takes place, can be used to achieve a significant computational reduction in solving this problem.

2. PageRank Algorithm: A Warm-Up Example

One of the key tasks in providing high-quality Internet search outcomes is being able to rank the Web pages according to their importance and relevance. The breakthrough idea that underlies the Google search technology is to use the Internet graph itself as a proxy measure for importance. As a first approximation, one could simply rank the Web pages according to, for example, the number of other pages linked to it. This approach by itself would be too simplistic since one can imagine a Web page with few links to it from other pages, but such that these other pages have large connectivity themselves, thus still making the original Web page potentially highly important. As a way to measure the importance of a Web page beyond its immediate connectivity the following approach is used, which is the main idea behind the PageRank algorithm. Let the Web be described as an undirected graph $\mathbb{G}$ with nodes V, denoted for simplicity by $1, 2, \ldots, n$, and edges E. A more appropriate model to consider is a directed graph, since a link from page u to page v by no means implies a link from v to u. We adopt here this (gross) simplifying assumption for the ease of illustration. A parameter $\epsilon > 0$ is fixed. A nonnegative vector $R = (R_1, \ldots, R_n)$ is defined to be a rank vector if $\sum_i R_i = 1$ and if it satisfies the identity

$$R_i = \frac{\epsilon}{n} + (1-\epsilon) \sum_{j:\,(j,i)\in E} d_j^{-1} R_j,$$

where d_j is the degree of the node j. In matrix form R solves $R = n^{-1}\epsilon J + (1-\epsilon)MR$, where J is the vector of ones and the (j,i)th component of the matrix M is d_j^{-1} if (j,i) is an edge and zero otherwise; namely, R is the unique stationary distribution of the stochastic matrix $n^{-1}\epsilon J + (1-\epsilon)M$. One can also interpret this stationary distribution as a stationary distribution of a "random surfer" who performs a random walk on a graph with transition probability matrix M with probability $1-\epsilon$, but with probability ϵ jumps to a random uniformly chosen point on the graph. This interpretation underlies the random walk (surfer) algorithm—one of the two main approaches for computing approximately the rank vector R (Berkhin [4]). Another main approach, which is the original one proposed by the Google inventors, is based on the following simple power iteration process. Set vector R^0 to be any stochastic vector (for example, set $R_i^0 = 1/n$ for all i). Iteratively, use $R^{t+1} = n^{-1}\epsilon J + (1-\epsilon)MR^t$, until $\|R^{t+1} - R^t\| \le \delta$ for some error tolerance δ and some norm choice $\|\cdot\|$. It is important to note that, in light of the enormous size of the Web graph, solving the rank problem exactly is a challenging task, even though, nominally, the problem of matrix inversion is solvable in polynomial time as a function of the size of the graph. The reason for the power iteration effectiveness comes from the following observation:

$$\begin{aligned}\left|\frac{R_i}{d_i} - \frac{R_i^{t+1}}{d_i}\right| &= (1-\epsilon)d_i^{-1}\left|\sum_{j:\,(j,i)\in E}(d_j^{-1}R_j - d_j^{-1}R_j^t)\right| \\ &\le (1-\epsilon)d_i^{-1}d_i \max_{1\le j\le n}\left|d_j^{-1}R_j - d_j^{-1}R_j^t\right| \\ &= (1-\epsilon)\max_{1\le j\le n}\left|d_j^{-1}R_j - d_j^{-1}R_j^t\right|,\end{aligned}$$

Downloaded from informs.org by [216.73.216.187] on 14 July 2026, at 12:34 . For personal use only, all rights reserved.

where R is the solution of the matrix Equation (1), further implying a convenient contractive property

$$\max_i |d_i^{-1} R_i - d_i^{-1} R_i^{t+1}| \le (1-\epsilon) \max_i |d_i^{-1} R_i - d_i^{-1} R_i^t|.$$

In light of this observation and adopting a reasonable assumption of graph sparsity, namely, that degrees d_i are bounded as a function of the graph size n, a target error tolerance level δ can be reached by taking t so that $(1-\epsilon)^t$ is of the order δ. In particular, the number of iteration t remains bounded as a function of graph size n, again provided that the underlying graph is sparse.

The power iteration algorithm based on computing R^t masks, however, an important structural property of the ranks R in the graph. Observe that the values R_i^t computed by the power iteration algorithm only depend on the nodes j of the graph that have distance at most t from i; namely, if we were to change the graph $\mathbb{G}$ outside the $B(i,t)$ size t neighborhood around i, the value of R_i^t would remain the same. Here the size t neighborhood $B(i,t)$ denotes the subgraph induced by the set of nodes with graph-theoretic distance at most t from i. On the other hand, since the difference of R_i^t and R_i is negligible as t becomes large, then the value of R_i changes very little as we change the graph $\mathbb{G}$ in any way outside of $B(i,t)$; namely, the model exhibits the correlation decay property for the values $X_i = R_i$. This observation enables a simple local algorithm for computing vector R—compute R_i^t for each i by observing the neighborhood $B(i,t)$ only ignoring the rest of the graph $\mathbb{G}\backslash B(i,t)$. Additionally, it simplifies the computation of the page ranks when a portion of a graph changes. In this case it suffices to recompute the page ranks for the "affected" part of the network—the set of nodes that have a distance at most t from the changed part of the graph, where again t is tuned to a desired level of error tolerance δ.

With this simple demonstration of the correlation decay principle at hand, we now turn to computationally more challenging problems in the context of wireless communications, combinatorial optimization, and graphical model reconstruction examples.

3. Correlation Decay Method for Wireless Communication

We now turn to the field of wireless communication and the applications of the correlation decay methods in this area. Our example will differ from the PageRank example in two aspects. First, we will deal with a model where the values X_i exhibiting the correlation decay property are random, and second, most importantly, the state space of the vector $X = (X_i, 1 \le i \le n)$ is exponentially large in n. Thus, unlike the PageRank example, where the computation of the page ranks R can be done in time that is polynomial in the size of the underlying graph, here the underlying computational task will fall into a "problematic" complexity class, namely, the #P-hard class, as we are about to see. In this situation, the correlation decay property provides the saving grace in designing fast inference algorithms.

Wireless communication models deal with analyzing contention for wireless resources, signal frequency and bandwidths being the most commonly considered. Assuming, for simplicity, that a single frequency is used for all users, a very simple (though highly stylized) combinatorial model of wireless communication is the following model based on independent sets of a graph. Consider a finite undirected graph $\mathbb{G}$ with node set $[n] = 1, 2, \ldots, n$ and edge set E. Call requests arrive into nodes i according to an arrival process that is Poisson with rate λ_i. The call durations at a node i are i.i.d. with exponentially distributed holding times with parameter μ_i. No routing is assumed for this model (namely, the system is one hop). The calls are accepted/rejected according to the following simple rule. When a call arrives into some node i at some time t, it is accepted if and only if the node i and all of the nodes j which are neighbors of i are not transmitting a signal at time t; namely, not only the node i needs to be free from calls in progress, but all of the neighboring nodes need to be free as well. Otherwise the call is rejected. In particular, no queueing is assumed for this model, though several papers have incorporated queueing issues as well (Tassiulas and

Downloaded from informs.org by [216.73.216.187] on 14 July 2026, at 12:34 . For personal use only, all rights reserved.

Ephremides [33], Jiang et al. [22]). One of the earliest times the just introduced model was analyzed in the work by Kelly [24, 25], though in the context of computer memory access and allocation, as opposed to wireless communication. Some later works in the context of communication networks include Ramanan et al. [29] and Galvin et al. [11].

Denoting by $X(t) = (X_i(t), 1 \le i \le n)$ the indicator vector for the calls in progress (namely, $X_i(t) = 1$ if the node i is transmitting at time t, and $X_i(t) = 0$ otherwise), we observe that at every time t, the set $I(t) \triangleq \{i\colon X_i(t) = 1\}$ is an independent set in $\mathbb{G}$; that is, no two nodes of $I(t)$ are connected by an edge. The underlying stochastic process turns out to be reversible (Kelly [24]), which leads to the following unique stationary distribution π as $t \to \infty$, which is called the Gibbs distribution:

$$\pi(I) \triangleq \lim_{t\to\infty} \mathbb{P}(X(t) = I) = Z^{-1} \prod_{i\in I} \frac{\lambda_i}{\mu_i}, \tag{1}$$

when I is an independent set, and $\pi(I) = 0$ when I is not an independent set. Here, Z is the normalization constant known as the *partition function* defined by

$$Z = \sum_I \prod_{i\in I} \frac{\lambda_i}{\mu_i},$$

where the sum is over all independent sets I of the graph $\mathbb{G}$. This normalization is the essence of the underlying computational hardness of the problem. Observe that in the special case when $\lambda_i = \mu_i = 1$, Z is the total number of independent sets in a graph. Computing the total number of independent sets of a graph is a canonical #P-hard computational problem (Valiant [35]). At the same time, knowing the value of Z is important for the performance analysis of our model. Let us say we wish to compute the steady-state fraction of calls which are accepted. A call arriving to a node i at time t is accepted if the independent set $I(t)$ observed at time t does not contain nodes $B(i,1) = i \cup N(i)$, where $N(i)$ is the set of neighbors of i. The probability of this event can be written as

$$\mathbb{P}\big(I(t) \cap (i \cup N(i)) = \varnothing\big) = Z^{-1} \sum_I \prod_i \frac{\lambda_i}{\mu_i},$$

where the sum is over all independent sets I, satisfying $I \cap (i \cup N(i)) = \varnothing$. Since the calls arrive at rates λ_i, $1 \le i \le n$, the time averaged fraction of accepted calls is

$$Z^{-1} \sum_i \lambda_i \Big(\sum_I \prod_k \frac{\lambda_k}{\mu_k} \Big),$$

and therefore knowing Z is crucial for evaluating this performance measure. It can be shown with more work that estimating Z is unavoidable for evaluating this and similar performance measures. This means that being able to compute the fraction of accepted calls can be used as for computing Z, and thus it is also a hard computational problem.

Nevertheless, in many cases the value of Z can be computed approximately. The first polynomial time approximation algorithms for counting problems of the type above were constructed using the Markov chain Monte Carlo (MCMC) algorithm (Jerrum and Sinclair [21]) and have been applied to a broad variety of problems including counting independent sets, matchings, graph colorings, computing a permanent of a matrix, etc. Recently, however, a new class of algorithms was proposed that is based on the correlation decay property and thus leads to local, and not just polynomial time, algorithms. As we will elaborate below, it turns out, that when the ratios λ_i/μ_i are not "too large," the model exhibits the correlation decay property. This means that the steady-state probability that a given node i transmits a signal is asymptotically independent of the portion of the graph $\mathbb{G}$, outside of

Downloaded from informs.org by [216.73.216.187] on 14 July 2026, at 12:34 . For personal use only, all rights reserved.

the size r neighborhood $B(i,r)$ of the node i, provided that r is sufficiently large. Specifically, consider modifying a graph $\mathbb{G}$ into some new graph $\tilde{\mathbb{G}}$ such that the subgraph $B(i,r)$ remains the same. Then, there exists B such that when $\max_i \lambda_i/\mu_i \le B$ we have

$$\mathbb{P}_{\mathbb{G}}(i \in I) \approx \mathbb{P}_{\tilde{\mathbb{G}}}(i \in I),$$

where the quality of the approximation $\approx$ is controlled by the size of r. Here we emphasize that the steady-state Gibbs probabilities depend on the underlying graphs $\mathbb{G}$ and $\tilde{\mathbb{G}}$ and thus use the corresponding graph as a subscript for the corresponding probability notation $\mathbb{P}$. The usefulness of this correlation decay property is obvious—to compute $\mathbb{P}_{\mathbb{G}}(i \in I)$, compute instead $\mathbb{P}_{\tilde{\mathbb{G}}}(i \in I)$, where $\tilde{\mathbb{G}}$ is, say, an empty graph outside $B(i,r)$. If d is an upper bound on the degree of the original graph $\mathbb{G}$, then the size of $B(i,r)$ is at most $1 + d + d^2 + \cdots + d^r = O(d^{r+1})$, which *does not depend on* n. The computation of $\mathbb{P}_{\tilde{\mathbb{G}}}(i \in I)$ can be done now on the graph $B(i,r)$ alone in time, which is n-independent and in parallel for all nodes i, say, using a brute-force enumeration. This approach will result in computation time order $O(2^{d^r})$ per node, which while not depending on n, still has a very poor dependence on d. It turns out (Weitz [36]) that one can have even a smarter algorithm that performs the necessary computation only in time $O(d^r)$—exponent gained. The idea is based on establishing the correlation decay property on a so-called computation tree (Weitz [36], Gamarnik and Katz [13]) as opposed to the original graph $B(i,r)$. In the special case when $\lambda_i = \lambda$ and $\mu_i = \mu$, which without the loss of generality we can set to $\mu = 1$, the following bound is known to be tight for the onset of the correlation decay property (Weitz [36]):

$$\lambda < (d-1)^{d-1}/(d-2)^d. \tag{2}$$

The approach was also tested numerically on a grid graphs (finite subgraphs of a lattice $\mathbb{Z}^d$) in Gamarnik and Katz [12], which resulted in significantly improved bounds on the underlying partition function, compared to some earlier numerical estimates based on other techniques. It would be of interest to develop a similar tight bound for the case of general λ_i and μ_i, but we are not aware of such results in the literature.

The approach based on the correlation decay method was developed in Bandyopadhyay and Gamarnik [3] and Weitz [36], but its roots go to much earlier works of Kelly [25], Spitzer [32], and Zachary [38] on Gibbs measures associated with independent sets on trees, and we now recall some of the findings in these papers because they provided the necessary background for the methods developed in Bandyopadhyay and Gamarnik [3] and Weitz [36]. Consider a special case when graph $\mathbb{G}$ is a $(d-1)$-ary tree with depth r; namely, we fix a particular node 0 called the root of the tree. It has exactly $d-1$ children. Each child has exactly $d-1$ children of their own, etc., until r generations. The rth generation nodes (and we have precisely $1 + (d-1) + \cdots + (d-1)^r$ of them) are leaves and do not have children of their own. Note that the degree of the root is $d-1$, and the degree of every other nonleaf node is d. We denote this graph by $\mathbb{T}_{d-1,r}$. Consider the Gibbs distribution in the special case when $\lambda_i = \lambda$ for all nodes for some fixed λ, and $\mu_i = 1$; namely, consider the case for which (2) was mentioned to be the critical value for the correlation decay property. Introduce a notation $p_r = \mathbb{P}_{\mathbb{T}_{d-1,r}}(0 \notin I)$; namely, p_r is the probability that the root node does not belong to an independent set chosen according to the Gibbs distribution π defined by (1). It turns out that p_r satisfies the following simple recursive identity for every $r \ge 0$:

$$p_r = (1 + \lambda p_{r-1}^{d-1})^{-1}.$$

The recursion suggests trying to find a solution to the fixed point equation

$$p = (1 + \lambda p^{d-1})^{-1} \tag{3}$$

and arguing that $p_r \to p$, as $r \to \infty$. That the fixed point equation has a unique solution follows immediately from the fact that the right-hand side is a decreasing function of p. The

Downloaded from informs.org by [216.73.216.187] on 14 July 2026, at 12:34 . For personal use only, all rights reserved.

issue of convergence however is more subtle. However, it turns out that the condition (2) implies precisely the convergence $p_r \to p$ for *every* starting value p_0. Furthermore, the result is tight in the sense that if the condition (2) is violated, the recursions starting at $p_0 = 0$ and $p_0 = 1$ oscillate, and no convergence takes place. Thus we have at least solved the inference problem of computing the probability p_r that the root node is not transmitting a signal in steady state, asymptotically as $r \to \infty$. With a little bit of extra work one can compute similarly the steady-state probability that the root and all of its neighbors are not transmitting a signal—which we need to estimate the probability that a signal arriving into the root node is accepted. This still might seem like a modest progress, as our tree graph $\mathbb{T}_{d-1,r}$ is very specialized and the performance analysis problem was solved only for the root node. Nevertheless, this result can be used in a much wider framework. Consider any graph $\mathbb{G}$ that is d-regular and has r-locally treelike structure. What this means is that every node has degree d, and the r-depth neighborhood $B(i,r)$ of every node $i = 1,\ldots,n$ is graph identical to $(d-1)$-ary tree with depth r, with one modification that the root has degree d, not $d-1$. These graphs are not atypical. A standard result in the theory of random graphs (Bollobas [6], Janson et al. [20], Alon and Spencer [1]) states that a graph chosen uniformly at random from the space of all d-regular graph is nearly $r = O(\log n)$-locally treelike, with probability approaching one as $n \to \infty$. The result obtained above for the case of the tree graph can be bootstrapped to show that, under assumption (2) the probability that a node i in $\mathbb{G}$ is not transmitting a signal is asymptotically

$$\mathbb{P}_{\mathbb{G}}(i \notin I) = (1 + \lambda p^d)^{-1},$$

as $n \to \infty$, where p is the unique solution of the Equation (3). (The change in power for p in the identity above is due to the degree d of i in $\mathbb{G}$ versus degree $d-1$ of the root of $\mathbb{T}_{d,r-1}$.) The critical idea here is that if the convergence to the fixed point takes place, then the value of $\mathbb{P}(i \notin I)$ for the local tree and for the entire graph is approximately the same (the correlation decay takes place), whereas if the recursion results in an oscillating system, then no conclusion can be drawn. The details for this result can be found in Bandyopadhyay and Gamarnik [3]. The result was also extended there for the case of nonregular locally treelike graphs for some bounded degree graphs. The case of nonlocally treelike graphs was done by Weitz [36] using the computation tree approach.

The model above is still very limited in that it ignores many important features of wireless communications such as queueing and delays, multiple frequencies, signal strength, etc. It would be interesting to see the extent to which the correlation decay methods apply to these more realistic models of wireless communication.

4. Correlation Decay Method for Combinatorial Optimization

The algorithmic problem underlying the wireless communication model discussed in the previous section regards the problem of counting the number of independent sets of a graph—a canonical #P-hard counting problem. In this section we discuss applications of the correlation decay methods to the area of combinatorial optimization. Specifically, we will discuss the problem of optimization on graphs, when the objective function is random, but the underlying graph is arbitrary (nonrandom) subject to some restrictions, the bound on a graph degree d being the most typical one.

Many combinatorial optimization problems on graphs not only fall into the NP-hard complexity class, but also often do not admit a polynomial time algorithm that can produce asymptotically optimal solutions. Examples include the problems of finding a largest (in cardinality) independent set in a graph, finding a maximum cut of a graph, finding a maximum number of satisfiable clauses in an instance of a K-SAT problem, etc. For example, assuming $\mathrm{P} \neq \mathrm{NP}$, no polynomial time algorithm exists for finding an independent set within a multiplicative factor $n^{1-\epsilon}$ of the optimal, for any $\epsilon > 0$, where n is the number of nodes

Downloaded from informs.org by [216.73.216.187] on 14 July 2026, at 12:34 . For personal use only, all rights reserved.

Downloaded from informs.org by [216.73.216.187] on 14 July 2026, at 12:34 . For personal use only, all rights reserved.

in the graph (Håstad [19]). Using the terminology of algorithmic complexity, this means that the problem of finding a largest independent set of a graph is MAX-SNP-hard. Even when one restricts the degree of the graph, the problem remains MAX-SNP-hard. For example, it is known that even if the maximum degree of a graph is at most 3, there does not exist a factor 1.0071 approximation algorithm for the maximum independent set problem, unless P = NP (Berman and Karpinski [5]). A similar inapproximability result is known for maximum degrees 4 and 5 (Berman and Karpinski [5]), as well as in the case when the maximum degree bound d grows (Trevisan [34]).

Let us now modify the problem by assuming that each node i of the graph $\mathbb{G} = (V, E)$ has a weight $W_i > 0$. The goal is finding a maximum-weight independent set problem, where the weight $W(I)$ of each independent set I is the sum of the weights of the nodes in it—$W(I) = \sum_{i \in I} W_i$. Furthermore, let us assume that W_i are random. If anything, this should make the problem only harder. After all, a trivial special case $W_i = 1$ corresponds to the problem of finding an independent set with maximum cardinality—the problem that is already hard to solve approximately, as we have discussed above. However, it turns out that if the weights W_i are independent and drawn according to some probability distributions, then, surprisingly, the problems becomes tractable because of the correlation decay phenomena. We now elaborate on this further. Suppose the underlying graph $\mathbb{G}$ has maximum degree ≤ 3, and W_i are i.i.d. with an exponential distribution with rate μ for all nodes. Observe that in this case there exists a unique independent set achieving the largest weight, which we denote by $I(\mathbb{G}, W) \subset V$, where $W = (W_i, i \in V)$ is the vector of weights. This property follows immediately from the continuity of the exponential distribution, which implies that the probability that two distinct independent have the same weight is zero. In light of this we have a well-defined indicator random variables $X_i = 1(i \in I(\mathbb{G}, W))$, $i \in V$—the indicators for whether nodes belong to the largest weighted independent set.

Now let us perform the following "experiment." For each node i, consider the maximum-weight independent set problem but defined now on the induced subgraph $B(i, r)$, where, as in the previous section, $B(i, r)$ is size r graph-theoretic neighborhood around i. Define $X_{i,r}$ similarly to X_i above. It is an indicator whether the largest weight independent set in graph $B(i, r)$ includes i; namely, $X_{i,r} = 1(i \in I(B(i, r), W_{B(i,r)}))$. Here, $W_{B(i,r)}$ naturally denotes a restriction of the weights of the graph to nodes in the neighborhood $B(i, r)$. As it was established in Gamarnik et al. [15], it turns out that X_i and $X_{i,r}$ are likely to be equal as r increases; namely, the model exhibits the correlation decay property. Formally, for every graph sequence $\mathbb{G}_n$ such that every node has degree at most 3,

$$\limsup_{n \to \infty} \lim_{r \to \infty} \mathbb{P}(X_{i,r} \neq X_i) = 0. \tag{4}$$

This is good news. It means that rather than solving the maximum-weight independent set problem on the entire graph, we can solve it for much smaller graphs $B(i, r)$ for each node i and report the result. The radius r can be tuned to make the error probability $\mathbb{P}(X_{i,r} \neq X_i)$ smaller than any tolerance level ϵ. The technique underlying (4) is constructive, and, moreover, the rate of correlation decay is geometric. Denoting the correlation decay rate by $\rho < 1$, this means that there exists a constant $C > 0$ such that the error probability is at most $C\rho^r$. This allows for an explicit bound on r to achieve the target error rate ϵ. Specifically, one should take r large enough so that $C\rho^r < \epsilon$. Notice, that since we have a bound on the degree $d \leq 3$ and r only depends on ϵ and does not depend on the size of the graph, the size of the neighborhood $B(i, r)$ is also graph independent. Based on this one can construct an algorithm that computes the optimal decision $X_{i,r}$ by looking only at the neighborhood $B(i, r)$, as opposed to an entire graph. This problem is by itself nontrivial since it still requires solving a hard optimization problem on an admittedly smaller graph $B(i, r)$. The proposed algorithm is based on a clever breadth-first search iteration, and showing that the so-called computational tree resulting from the breadth-first search itself satisfies the correlation decay property in some appropriate sense. As a result, the resulting running time

of the algorithm is much smaller than one incurred by performing a brute-force search on the graph $B(i,r)$. The resulting algorithm has a linear running time $O(n)$, where n is the number of nodes in the graph, and, most importantly, it is local. As a result, it can be run in parallel for all nodes.

The example above was restricted to graphs with degree at most 3. This is a nontrivial but still very restrictive class of graphs. Can the correlation decay property (4) be generalized to graphs with larger degrees and perhaps different distributions and, finally, perhaps to problems other than the independent set problem? The answer is yes for all of these questions. Although for the case of exponential distribution the correlation decay property provably fails when the graph degree is 5, as shown in Gamarnik et al. [16] (thus the case $d=4$ is still unresolved), the correlation decay property is established for larger d by appropriately modifying the exponential distribution. In particular, as shown in Gamarnik et al. [15], for each d there exists a finite mixture of exponential distributions such that the correlation decay property (4) holds for the maximum-weight independent set problem.

What about problems other than the maximum-weight independent set problem? Consider the following general decision problem defined on a graph $\mathbb{G}=(V,E)$, where again V is the set of nodes and E is the set of edges. We envision agents associated with each node i, each of whom needs to come up with one of two possible decisions X_i, encoded 0 and 1. The graph has degree at most d and is arbitrary otherwise. The objective function $\mathcal{J}$ is assumed to be of the form $\sum_i \mathcal{J}_i(X_i)+\sum_{(i,j)\in E}\mathcal{J}_{i,j}(X_i,X_j)$ and is assumed to be random. Here, J_i is the random reward associated with decision taken by agent i alone, and $J_{i,j}(X_i,X_j)$ is random revenue associated with the decision taken by a pair of agents i and j who are connected by an edge. The goal for agents is to come up with decisions X_i, $i\in V$, that maximize the total reward $\mathcal{J}$. It is sometimes called an economic team decision-making problem (Rusmevichientong and Van Roy [30], Marschak [27], Radner [28]), though it is a general framework for many problems in optimization (specifically combinatorial optimization), statistical inference (maximum likelihood estimation), and several other applications, as discussed in Gamarnik et al. [15]. We refer to this problem as simply the network decision problem. Since many combinatorial optimization problems can be cast as network decision problems, it is NP-hard in general, unless additional assumptions are introduced. This is where the assumption of randomness plays a key role and the correlation decay property can be established. In particular, the following result is established in Gamarnik et al. [15]. Suppose for each node i, the reward for decision 0 is $\mathcal{J}_i(0)=0$, and the reward for decision 1 is $\mathcal{J}_i(1)$ is uniformly distributed over some symmetric interval $[-I_1,I_1]$, independently for all i. Similarly, suppose the rewards associated with edges have a uniform distribution over some interval $[-I_2,I_2]$ for all decisions; namely, for each edge (i,j), $\mathcal{J}_{i,j}(x,y)$, $x,y=0,1$ are uniformly distributed over $[-I_2,I_2]$, independently for all x, y and all edges. Also suppose these rewards are independent from the node-based rewards $J_i(1)$. Letting $\beta=5I_2/(2I_1)$, it is established in Gamarnik et al. [15] that under the assumption

$$\beta(d-1)^2<1,$$

the correlation decay property takes place. This leads to a fast decentralized local algorithm for solving the underlying network decision problem, similar to the one described above for the case of maximum-weight independent set problem. In particular, its running time is only $O(n)$. A similar result can be established for the case of Gaussian random weights, but we omit the details in the interest of space and instead refer the reader to the aforementioned reference.

The ideas of correlation decay have appeared in an implicit forms in an earlier paper by Rusmevichientong and Van Roy [30] in the context of team decision making, when the underlying graph is a line. Furthermore, again implicitly they appear in Kakade et al. [23] in the context of the Fisher model of economic consumption and pricing defined on agents

Downloaded from informs.org by [216.73.216.187] on 14 July 2026, at 12:34 . For personal use only, all rights reserved.

interacting through a random graph. Empirically, the authors have established that the equilibrium prices in their model exhibit the correlation decay property by matching the results obtainable from local approximate computations with correct values computed for small size graphs for which the computation of equilibrium can be achieved. It would be interesting to establish the correlation decay result for this model in a formal rigorous way.

We close this section by mentioning a recent work on negative results on the power of local algorithms in contexts where the model does not exhibit the correlation decay property. In particular, it was established in Gamarnik and Sudan [14] that no local algorithm within a suitably defined framework of local algorithms can find a nearly optimal independent set in sparse random graphs. Although there are many negative results on the power of local algorithms on more contrived examples of graphs or graphs with some prescribed global structure (for example, bipartite graphs), this is the first negative result on a broad family of graphs—sparse random graphs.

5. Correlation Decay Method for the Network Learning Problem

The network questions we have investigated above either in the context of optimization or in the context of inference problems were of the type where the network itself is known. We now switch to the question that has a reversed nature. Suppose we are given random observations X_i, $i \in V$, associated with nodes i in some graph $\mathbb{G} = (V, E)$. Suppose we have an access to the joint distribution of X_i, but at the same time the underlying graph $\mathbb{G}$ is not known. Can we reconstruct the underlying graph? This question, sometimes dubbed as *graph tomography*, is a basis of a very active area of research in the machine learning and statistics fields (Chow and Liu [8], Bresler et al. [7], Anandkumar and Valluvan [2]) and has many applications, for example, in biology (Friedman [10]), in natural language processing (Anandkumar and Valluvan [2]), and in the field of organizational structures and homeland security (Yu et al. [37]).

To discuss the graph learning problem in a rigorous way, we first need to explain what it means that the variables X_i are associated with nodes of a graph and, in particular, the meaning of the edges of a graph. For this purpose we introduce now a model known as a *graphical model* or *Markov random field* (Lauritzen [26], Grimmett [18])—a widely studied object in the field of machine learning and statistics. A graphical model is defined as a graph $\mathbb{G} = (V, E)$, together with a set of (generally dependent) random variables X_i, $i \in V$, satisfying the following property. For every node i, as before, let $N(i)$ be the set of neighbors of i. In our earlier terminology, $N(i)$ is the set of nodes of the graph $B(i, 1)$ minus node i itself. The model is defined to be a graphical model if the random variable X_i is independent from random variables $X_l, l \notin B(i, 1)$, conditional on X_j, $j \in N(i)$. Formally,

$$\mathbb{P}\big(X_i \mid X_j, j \in V \backslash \{i\}\big) = \mathbb{P}\big(X_i \mid X_j, j \in N(i)\big). \tag{5}$$

This property is sometimes called a spatial Markovian property.

What are examples of graphical models? It turns out that one example is just an ordinary Markov chain. Given a Markov chain X_t evolving over discrete times $t = 0, 1, 2, \ldots$, think of this Markov chain instead sitting on a line graph with nodes $0, 1, \ldots, T$ and edges $(0, 1), (1, 2), \ldots, (T-1, T)$ (we assumed a finite graph/time horizon T for convenience). The node i has neighbors $i-1$ and $i+1$ for $i = 1, 2, \ldots, T-1$. Then the property (5) can be rewritten as $\mathbb{P}(X_t \mid X_0, \ldots, X_{t-1}, X_{t+1}, \ldots, X_T) = \mathbb{P}(X_t \mid X_{t-1}, X_{t+1})$—one of the basic properties of Markov chains. The observation applies to the cases $i = 0$ and $i = T$ as well, except in this case each node has only one neighbor. Thus the Markov chain X_t, $0 \le t \le T$ is a graphical model. But, of course, there are more interesting examples of graphical models. As it turns out, both the Gibbs distribution on the set of independent sets discussed in §3 and maximum-weight independent set model with random weights discussed in §4 are

Downloaded from informs.org by [216.73.216.187] on 14 July 2026, at 12:34 . For personal use only, all rights reserved.

examples of graphical models. To convince ourselves of this, start with the second example—maximum-weight independent set I on a graph $\mathbb{G}$, when the weights are random W_i with an arbitrary distribution such that W_i is positive with probability one. As before, let X_i denote the indicator variables for the maximum-weight independent set I; that is, $X_i = 1$ if $i \in I$ and 0 otherwise. Given i, let us consider $\mathbb{P}(X_i \mid X_j, j \in N(i))$. Observe that if at least one of $X_j = 1$ for $j \in N(j)$, then X_i has to be zero by the independent set property, regardless of the values of the variables X_l outside of $N(i)$. On the other hand, if $X_j = 0$ for all $j \in N(i)$, then $X_i = 1$, since we are choosing the maximum-weight independent set, and W_i is positive. Again this holds regardless of the values of X_l for nodes l outside of $N(i)$. Thus, indeed X_i, $i \in V$, is a graphical model.

We can verify the spatial Markovian property for the model considered in §3—randomly chosen independent set—as well. Instead, let us consider a more general model, which we also call graphical model, for the reasons to be justified below. This alternative definition can be introduced as follows. We are given an undirected graph $\mathbb{G} = (V, E)$, a q-size alphabet with elements $[q] \triangleq \{0, 1, \ldots, q-1\}$, a sequence of node potentials Φ_i: $[q] \to \mathbb{R}_+$ for every node $i \in V$, and a sequence of edge potentials $\Phi_{u,v}$: $[q]^2 \to \mathbb{R}_+$ for every edge $(i, j) \in E$. For every vector $\mathbf{x} = (x_i, i \in V)$ with entries $x_i \in [q]$, we associate a probability weight

$$p(\mathbf{x}) = Z^{-1} \prod_{i \in V} \Phi_i(x_i) \prod_{(i,j) \in E} \Phi_{i,j}(x_i, x_j).$$

Here, Z is a normalizing constant called partition function defined by

$$Z = \sum_{\mathbf{x}} \prod_{i \in V} \Phi_i(x_i) \prod_{(i,j) \in E} \Phi_{i,j}(x_i, x_j).$$

We assume that $Z > 0$, since otherwise the model is not well defined. The distribution above is called Gibbs distribution with potentials Φ_i, $\Phi_{i,j}$. This is not a particularly new model, as we have already seen a special case of it in §3. Recall the probability distribution defined on independent sets of a graph described by the identity (1). Notice that it is a special case of the model above for the case $q = 1$, $\Phi_i(0) = 1$, $\Phi_i(1) = \lambda_i/\mu_i$ for every node i, and $\Phi_{i,j}(x_i, x_j) = 0$ when $x_i = x_j = 1$ and 1 otherwise. The definition of $\Phi_{i,j}$ ensures that only independent sets have a positive probability mass attached to them. The definition of Φ_i ensures that each independent set I has probability mass proportional to the product $\prod_{i \in I} \lambda_i/\mu_i$. Hence the equivalence with model (1).

It is a rather straightforward exercise to check that the model above satisfies the spatial Markovian property (5). Perhaps a more surprising fact is that for a broad class of models it is equivalent to it. Specifically, suppose the graph $\mathbb{G} = (V, E)$ is triangle free (no three nodes i, j, l can be found such that $(i, j), (j, l), (i, l) \in E$), suppose the range of values of each X_i is a discrete finite set denoted by $0, 1, \ldots, q-1$, and suppose that a graphical model X_i, $i \in V$ is such that $\mathbb{P}(\mathbf{x}) > 0$ for all vectors $\mathbf{x}$ (this example unfortunately excludes the independent set model since every $\mathbf{x}$ not corresponding to an independent set has probability mass zero). Then there exists a set of node and edge potentials Φ_i, $i \in V$, $\Phi_{i,j}$, $(i, j) \in E$, such that the probability distribution $\mathbb{P}(\cdot)$ underlying the graphical model is the Gibbs distribution with potentials $\Phi_i, \Phi_{i,j}$. Thus in this case the two definitions are equivalent. The equivalency can be extended to graphs with triangles but at the expense of introducing potentials Φ_K associated with cliques (complete subgraphs) of $\mathbb{G}$. In the case of triangle-free graphs, the only cliques are nodes and edges, and thus the node and edge potentials suffice to describe the joint probability distribution.

Given a graphical model $\mathbf{X} = (X_i, i \in V)$ defined on a graph $\mathbb{G}$, the graph reconstruction model is defined as follows: given r i.i.d. samples $\mathbf{X}_1, \ldots, \mathbf{X}_r$ from the joint distribution described by the underlying graphical model, reconstruct the graph $\mathbb{G}$. One of the earliest papers that considered this question was by Chow and Liu in 1960s (Chow and Liu [8]) for the case when the underlying graph is a tree. They proposed a maximum weight spanning tree-based algorithm for reconstructing the underlying tree, with pairwise correlations

Downloaded from informs.org by [216.73.216.187] on 14 July 2026, at 12:34 . For personal use only, all rights reserved.

Downloaded from informs.org by [216.73.216.187] on 14 July 2026, at 12:34 . For personal use only, all rights reserved.

serving as edge weights. In general, although one does not expect a fast algorithm for reconstructing an arbitrary graph (although the author is not aware of any formal hardness results for this problem), in the case of bounded degree graphs, a polynomial time reconstruction algorithm is known. The idea is very simple. Whenever (i,j) is *not* an edge, the set of neighbors of i is a "separator" of X_i from X_j in the sense of conditional independence; namely, thanks to the property (5), we have that X_i and X_j are independent, conditional on $X_l, l \in N(i)$. This suggests a very straightforward algorithm: for every pair of nodes i,j and every set of nodes $S \subset V$ with cardinality $|S| \le d$, compute the sample correlation between X_i and X_j, conditional on X_l, $l \in S$, based on the available sample of vectors $\mathbf{X}_k, 1 \le k \le r$. One fixes an error level $\epsilon > 0$. Given i and j, if a set S is found such that the sample correlation is less than ϵ, then (i,j) is declared *not* to be an edge. Conversely, (i,j) is declared to be an edge if for every subset S, $|S| \le d$, the conditional independence test fails. As it was shown in Bresler et al. [7], this algorithm succeeds in reconstructing the underlying graph $\mathbb{G}$ with high probability when the sample size is $r = O(\log n)$ in $O(n^{d+2} \log n)$ operations. One needs an additional assumption than that the correlation between X_i and X_j is larger than ϵ whenever (i,j) is an edge. (Clearly, an assumption of this sort is necessary, since otherwise the notion of an edge is not particularly meaningful.) The running time estimate is immediate—for every pair of nodes i,j and every subset S, the conditional independence test has to be conducted. Thus the running time is $O(n^{d+2})$. The bound on the sample size r of the order $O(\log n)$ is needed to ensure the soundness of the conditional independence test. For the same reason it appears in the running time. When d is a constant, this leads to a nominally polynomial time algorithm.

In reality, however, the running time of this order is hardly practical when n is large even for moderate values of d. This is where the correlation decay property comes to the rescue. Suppose it is known that the underlying graphical model exhibits the correlation decay property. In particular, suppose the correlation between X_i and X_j is a decreasing function of a graph-theoretic distance between i and j. For example, as can be shown in most models that exhibit the correlation decay property, suppose there exists $0 < \rho < 1$ such that the correlation between X_i and X_j is at most $\rho^{d(i,j)}$, where $d(i,j)$ is the graph-theoretic distance between i and j. In this case, as observed in Bresler et al. [7], one can dramatically reduce the time needed to reconstruct the graph $\mathbb{G}$ by first conducting an unconditional pairwise correlation test between X_i and X_j for all pairs i and j. Whenever the sample correlation is less than ϵ, the (i,j) is declared nonedge. By the correlation decay property, for every i, the set of nodes that fail this correlation test (and thus are still potentially connected to i by an edge) is at most d^D, where D is the smallest integer satisfying $\rho^D < \epsilon$. In particular, this set has a constant (independent of n) cardinality, denoted by K. One then conducts the conditional independent tests on this set K in the way described above, which only requires $O(K^{d+1} \log n)$ running time since the tests are conducted on the set of cardinality K only ($d+1$ as opposed to $d+2$ appears because the node i is fixed). Thus the total running time is only $O(n^2 \log n) + O(n \log n)$, where the first part arises from the first phase of the algorithm, computing correlation between all pairs X_i and X_j, and the second part comes from the "local" conditional independence test, which has a running time $O(nK^{d+1} \log n) = O(n \log n)$. The correlation decay property thus allows us to reduce the running time exponent from $d+2$ to 2—a very significant computation time reduction. As was observed in Anandkumar and Valluvan [2], one can further reduce the computation factor $O(K^{d+1})$ if the underlying graph is locally treelike. This assumption, at least theoretically, is not very restrictive, since bounded degree graphs which are chosen at random (aka Erdős–Rényi graphs) are known to possess the locally treelike property.

Acknowledgments

The author is very grateful to Angelia Nedich for a careful reading of an earlier draft of this tutorial and for providing many useful comments. Part of the research described in the this tutorial was supported by the National Science Foundation [Grant CMMI-0726733].

References

[1] N. Alon and J. Spencer. *Probabilistic Method.* John Wiley & Sons, New York, 1992.

[2] A. Anandkumar and R. Valluvan. Learning loopy graphical models with latent variables: Efficient methods and guarantees. *Annals of Statistics* 41(2):401–435, 2013.

[3] A. Bandyopadhyay and D. Gamarnik. Counting without sampling. Asymptotics of the log-partition function for certain statistical physics models. *Random Structures and Algorithms* 33(4):452–479, 2008.

[4] P. Berkhin. A survey on pagerank computing. *Internet Mathematics* 2(1):73–120, 2005.

[5] P. Berman and M. Karpinski. On some tighter inapproximability results. J. Wiedermann, P. van Emde Boas, and M. Nielsen, eds. *Automata, Languages and Programming*, Lecture Notes in Computer Science, Vol. 1644. Springer, Berlin, 200–209, 1999.

[6] B. Bollobas. *Random Graphs.* Academic Press, London, 1985.

[7] G. Bresler, E. Mossel, and A. Sly. Reconstruction of Markov random fields from samples: Some observations and algorithms. M. Goemans, K. Jansen, J. D. P. Rolim, and L. Trevisan, eds. *Approximation, Randomization and Combinatorial Optimization: Algorithms and Techniques*, Lecture Notes in Computer Science, Vol. 2129. Springer, Berlin, 343–356, 2008.

[8] C. Chow and C Liu. Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory* 14(3):462–467, 1968.

[9] R. L. Dobrushin. Prescribing a system of random variables by the help of conditional distributions. *Theory of Probability and Its Applications* 15(3):469–497, 1970.

[10] N. Friedman. Inferring cellular networks using probabilistic graphical models. *Science* 303(5659):799–805, 2004.

[11] D. Galvin, F. Martinelli, Ramanan, and P. Tetali. The multistate hard core model on a regular tree. *SIAM Journal on Discrete Mathematics* 25(2):894–915, 2011.

[12] D. Gamarnik and D. Katz. Sequential cavity method for computing free energy and surface pressure. *Journal of Statistical Physics* 137(2):205–232, 2009.

[13] D. Gamarnik and D. Katz. Correlation decay and deterministic FPTAS for counting list-colorings of a graph. *Journal of Discrete Algorithms* 12(April):29–47, 2012.

[14] D. Gamarnik and M. Sudan. Limits of local algorithms over sparse random graphs. E-print, arXiv, http://arxiv.org/abs/1304.1831, 2013.

[15] D. Gamarnik, D. Goldberg, and T. Weber. Correlation decay in random decision networks. *Mathematics of Operations Research*, ePub ahead of print August 1, http://dx.doi.org/10.1287/moor.2013.0609, 2013.

[16] D. Gamarnik, T. Nowicki, and G. Swirscsz. Maximum weight independent sets and matchings in sparse random graphs. Exact results using the local weak convergence method. *Random Structures and Algorithms* 28(1):76–106, 2006.

[17] H. O. Georgii. *Gibbs Measures and Phase Transitions.* de Gruyter Studies in Mathematics 9, Walter de Gruyter, Berlin, 1988.

[18] G. Grimmett. *Probability on Graphs: Random Processes on Graphs and Lattices.* Cambridge University Press, Cambridge, UK, 2010.

[19] J. Håstad. Clique is hard to approximate within $n^{1-\epsilon}$. *Acta Mathematica* 182(1):105–142, 1999.

[20] S. Janson, T. Łuczak, and A. Rucinski. *Random Graphs.* John Wiley & Sons, New York, 2000.

[21] M. Jerrum and A. Sinclair. The Markov chain Monte Carlo method: An approach to approximate counting and integration. D. Hochbaum, ed. *Approximation Algorithms for NP-Hard Problems.* PWS Publishing Company, Boston, 482–520, 1997.

[22] L. Jiang, D. Shah, J. Shin, and J. Walrand. Distributed random access algorithm: Scheduling and congestion control. *IEEE Transactions on Information Theory* 56(12):6182–6207, 2010.

[23] S. Kakade, M. Kearns, J. Langford, and L. Ortiz. Correlated equilibria in graphical games. *Proceedings of the 4th ACM Conference on Electronic Commerce*, ACM, New York, 42–47, 2003.

[24] F. Kelly. *Reversibility and Stochastic Networks.* John Wiley & Sons, New York, 1979.

[25] F. Kelly. Stochastic models of computer communication systems. *Journal of the Royal Statistical Society: Series B* 47(3):379–395, 1985.

Downloaded from informs.org by [216.73.216.187] on 14 July 2026, at 12:34 . For personal use only, all rights reserved.

[26] S. L. Lauritzen. *Graphical Models*. Oxford University Press, Oxford, UK, 1996.

[27] J. Marschak. Elements for a theory of teams. *Management Science* 1(2):127–137, 1955.

[28] R. Radner. Team decision problems. *Annals of Mathematical Statistics* 33(3):857–881, 1962.

[29] K. Ramanan, A. Sengupta, I. Ziedins, and P. Mitra. Markov random field models of multicasting in tree networks. *Advances in Applied Probability* 34(1):58–84, 2002.

[30] P. Rusmevichientong and B. Van Roy. Decentralized decision-making in a large team with local information. *Games and Economic Behavior* 43(2):266–295, 2003.

[31] B. Simon. *The Statistical Mechanics of Lattice Gases*, Vol. I. Princeton Series in Physics, Princeton University Press, Princeton, NJ, 1993.

[32] F. Spitzer. Markov random fields on an infinite tree. *Annals of Probability* 3(3):387–398, 1975.

[33] L. Tassiulas and A. Ephremides. Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. *IEEE Transactions on Automatic Control* 37(12):1936–1948, 1992.

[34] L. Trevisan. Non-approximability results for optimization problems on bounded degree instances. *Proceedings of the 33rd Annual ACM Symposium on the Theory of Computing*, ACM, New York, 453–461, 2001.

[35] L. G. Valiant. The complexity of computing the permanent. *Theoretical Computer Science* 8(2):189–201, 1979.

[36] D. Weitz. Counting independent sets up to the tree threshold. *Proceedings of the 38th Annual ACM Symposium on the Theory of Computing*, ACM, New York, 140–149, 2006.

[37] F. Yu, G. Levchuck, D. Pattipati, and F. Tu. A probabilistic computational model for identifying organizational structures from uncertain message data. *10th International Conference on Information Fusion 2007*, IEEE, Piscataway, NJ, 1–8, 2007.

[38] S. Zachary. Countable state space Markov random fields and Markov chains on tree. *Annals of Probability* 11(4):894–903, 1983.

Downloaded from informs.org by [216.73.216.187] on 14 July 2026, at 12:34 . For personal use only, all rights reserved.