Simple deterministic approximation algorithms for counting matchings *

Mohsen Bayati † David Gamarnik ‡ Dmitriy Katz § Chandra Nair ¶ Prasad Tetali ‖

Abstract

We construct a deterministic fully polynomial time approximation scheme (FPTAS) for computing the total number of matchings in a bounded degree graph. Additionally, for an arbitrary graph, we construct a deterministic algorithm for computing approximately the number of matchings within running time $\exp(O(\sqrt{n}\log^2 n))$, where n is the number of vertices.

Our approach is based on the *correlation decay* technique originating in statistical physics. Previously this approach was successfully used for approximately counting the number of independent sets and colorings in some classes of graphs [BG06],[Wei06],[GK07a]. Thus we add another problem to the small, but growing, class of $\#P$-complete problems for which there is now a deterministic FPTAS.

*Preliminary version of this paper will appear in Proceedings of 39th ACM Symposium on Theory of Computing, San Diego, CA

†Department of Electrical Engineering, Stanford University, Stanford, CA 94305-9510; e-mail: bayati@stanford.edu

‡Operations Research Center and Sloan School of Management, MIT, Cambridge, MA 02139; e-mail: gamarnik@mit.edu

§Operations Research Center, MIT, Cambridge, MA 02139; e-mail: dimdim@mit.edu

¶Microsoft Research, One Microsoft Way, Redmond, WA 98052; e-mail: cnair@microsoft.com

‖School of Mathematics and College of Computing, Georgia Tech, Atlanta, GA 30332-0160; e-mail: tetali@math.gatech.edu

1 Introduction

The focus of the paper is the problem of computing the total number of (full and partial) matchings in a given graph. This problem, along with many other combinatorial counting problems falls into the class of #P-complete complexity class, and thus, modulo a basic complexity theoretic conjecture, cannot be solved (exactly) in polynomial time. Ever since the introduction of the #P complexity class by Valiant [Val], the focus on these problems shifted to finding approximating solutions. Here by far the strongest and the most general method comes from the Markov chain Monte Carlo (MCMC) approach. Based on the equivalence between the counting problem and a related sampling problem [JVV86], this approach focuses on rapidly mixing Markov chains to obtain appropriate random samples. Many randomized approximation schemes for various counting problems were derived in this way – see e.g., [JS97, J03] for several nontrivial applications. Among other results, a fully polynomial *randomized* approximation scheme (FPRAS) for computing the total number of matchings of a given graph was provided by Jerrum and Sinclair [S93, JS97]. Our main thrust in this work is in providing *deterministic* algorithms for counting the number of matchings; this we are able to do efficiently (in fully polynomial time) for the class of bounded degree graphs, and in sub-exponential time for general graphs (see details below).

Recently an alternative approach for constructing approximate counting schemes was developed, leading to *deterministic* approximation schemes for counting problems. The approach is based on the concept of *correlation decay* originating in statistical physics [Dob70] for estimating certain marginal probability distributions (See Proposition 2.2 below). The correlation decay concept, also called *spatial* correlation decay, is related to the notion of uniqueness of Gibbs measures in infinite lattices, hence the interest of statistical physicists in this subject. The approach of using correlation decay property for constructing efficient algorithms for counting was introduced in Weitz [Wei06] and Bandyopadhyay and Gamarnik [BG06], for the problems of counting the number of independent sets and colorings of a graph. The development in [Wei06] was particularly impressive as, unlike in [BG06], no assumptions on the girth of a graph are made and the approach led to the first known *deterministic* fully polynomial time approximation scheme (FPTAS) for a problem in the #P-complete class, namely the problem of counting the number of independent sets in graphs with degree ≤ 5. Weitz's approach was based on a clever reduction of the graph problem to that on a related well-known tree, the *tree of self avoiding walks*. Very recently, the correlation decay approach was also put to use in Gamarnik and Katz [GK07a] for the problem of counting the number of proper colorings of a graph. In this work the step of constructing a self-avoiding walk was bypassed, by instead creating a certain *computation tree* in a dynamic programming type recursion and establishing correlation decay on the corresponding computation tree. As observed in [GK07a], the advantage of establishing correlation decay on a computation tree is that it leads to inverse polynomial errors on the marginal probabilities as opposed to constant errors that one would obtain by using spatial correlation decay directly. For completeness, we reproduce here a section from [GK07a] which explains this crucial difference which ultimately leads to polynomial time approximation algorithms. The relation between correlation decay on a tree and on general graphs was one of the motivations of Weitz' [Wei06] work; and recently particular generalizations of it are given by Jung and Shah [JS] and Nair and Tetali [NT] for certain Markov random field models.

In this paper we use the correlation decay approach for constructing a fully polynomial deterministic approximation scheme for counting the total number of (partial and full) matchings of a graph. Thanks to Vadhan's work [V01], the problem of counting matchings in regular graphs of bounded degree (to be precise, degree at least five) is also known to be #P-complete. In fact we solve a more general problem - the one of computing the partition function corresponding to matchings with a given activity level λ. We establish the correlation decay property on the computation tree. Interestingly, the analysis

becomes far more transparent, when compared to the counterparts in independent sets and colorings. In particular, we establish that the correlation decay property corresponding to counting matchings holds for an arbitrary graph $\mathbb{G}$ and activity λ combination, in the appropriate sense. We show that the rate of the correlation decay (appropriately defined) is $\approx 1 - O(\frac{1}{\sqrt{\Delta}})$, where Δ is the maximum degree of the graph. As a result we construct a deterministic FPTAS for computing the number of matchings in any graph with $\Delta = O(1)$. For the case of arbitrary graphs (with no restriction on the maximum degree) we construct a deterministic approximation scheme which runs in time $\exp(O(\sqrt{n}\log^2 n))$, where n the number of vertices in the graph. The problem of computing the number of matchings of graphs with large girth was addressed recently by Bayati and Nair [BN06] in the context of Belief Propagation algorithms and the validity of the cavity method. The use of tree like recursions, similar to the one in this paper, for computing the matching polynomials can also be found in the work of [God81]. The key new idea is the use of correlation decay to reduce the computation time using the recursions. The fact that the Gibbs distribution corresponding to matchings exhibits a spatial correlation decay was already established by Heilmann and Lieb [HL72] by looking at complex roots of the partition function and later by van den Berg [vdB98] using more probabilistic/combinatorial arguments – in particular, the path of disagreement percolation argument along with the fact that such a path does not bifurcate for the monomer-dimer model in establishing exponential decay of correlation for this model on $\mathbb{Z}^d$. Here we show decay of correlations on the computation tree which is a stronger result. We also note that the work of Kahn and Kim [KK98] is pertinent here. Using Godsil's (self-avoiding) tree construction, they show that in any d-regular graph, the probability that any fixed vertex is *not* in a (uniformly chosen) random matching is asymptotically $1/\sqrt{d}$; moreover, as we realized since doing the present work, their proof establishing this fact (see Claim (2.6) in their paper) also uses (as we do below in Theorem 3.2) a two level recursion and partial derivatives to show convergence to the above probability estimate.

It should be noted that, while an FPRAS for counting matchings is known to exist thanks to the MCMC method, constructing deterministic counterparts is a far more challenging task. For example, while an FPRAS for computing permanent of a matrix is known [JSV04], the best known deterministic approximation scheme only gives factor e^n approximation [LSW00]. In fact the present paper led to a recent improvement of this result [GK07b].

The connection between the correlation decay property and counting is understood within the field of rapidly mixing Markov chains as well. Broadly speaking one expects that the lack of long-range dependence (correlation decay) implies rapid mixing and thus the existence of randomized approximation scheme for counting/sampling problems. Specifically, when the graph satisfies a certain kind of sub-exponential growth condition the correlation decay does imply rapid mixing (see e.g., [DSVW04], [GMP05]). The converse, however, does not hold in general, as shown by Berger et al. [BKMP01], but does hold in some weaker sense, as shown recently by Montanari and Semerjian [MS06].

The present work, along with [BG06],[BN06],[Wei06],[GK07a],[GK07b],[NT] reinforces this connection, as well as, broadly speaking, contributes to the exciting and emerging connection between theoretical computer science, probability theory and statistical physics.

The rest of the paper has the following structure. The definitions and the main result are presented in Section 2. The correlation decay analysis is the subject of Section 3. The approximate counting algorithm and the complexity analysis is presented in Section 4. The discussion on the difference between spatial correlation decay and correlation decay on a computation tree, borrowed from [GK07a] is given in Section 5. Some concluding remarks and further open questions are presented in Section 6.

2 Definitions, preliminaries and the main result

We consider a simple labeled undirected graph $\mathbb{G}$ with n vertices. The vertex set is denoted by $V = \{v_1, \ldots, v_n\}$. E denotes the edge set. $N(v, \mathbb{G}) \subset V$ denotes the set of neighbors of v and $\Delta(v) = |N(v, \mathbb{G})|$ denotes the degree of the vertex v. The degree of the graph is $\Delta \triangleq \max_v \Delta(v)$. We abuse notations by letting $\mathbb{G}\backslash\{v\}$ denote a subgraph of $\mathbb{G}$ induced by nodes $V\backslash\{v\}$. Let $\mathbb{G}_k = \mathbb{G}\backslash\{v_1, \ldots, v_{k-1}\}$ with $\mathbb{G}_0 = \mathbb{G}$.

A matching is a subset $M \subseteq E$ such that no two edges in M share a vertex. We denote by $\mathcal{M} = \mathcal{M}(\mathbb{G})$ the set of all matchings of $\mathbb{G}$. For every $k \leq |V|/2$ let $M(k)$ be the number of size k matchings in $\mathbb{G}$.

Given a fixed parameter $\lambda > 1$, called the *activity*, a natural (Gibbs) probability distribution on the set $\mathcal{M}$ of matchings is defined as:

$$\mathbb{P}_{\mathbb{G}}(M) = \frac{\lambda^{|M|}}{Z(\mathbb{G})},$$

where the normalizing constant $Z(\mathbb{G})$[1] is called the *partition function* corresponding to λ, and is expressed as

$$Z(\mathbb{G}) = \sum_{M \in \mathcal{M}} \lambda^{|M|}.$$

Denote by $\boldsymbol{M}$ a random matching selected according to the Gibbs measure $\mathbb{P}_{\mathbb{G}}$, which clearly depends on the given graph $\mathbb{G}$. The principle goal of this paper is obtaining an approximation of $\mathbb{P}_{\mathbb{G}}(v \notin \boldsymbol{M})$ for every vertex v in polynomial time, for the class of bounded degree graphs $\mathbb{G}$. Here, abusing notation slightly, $v \in M$ means matching M contains an edge incident to v. Thus this quantity simply represents the probability that a random matching selected according to the Gibbs measure does *not* contain an edge adjacent to v.

Our goal is constructing an algorithm which computes $Z(\mathbb{G})$ approximately. The instance size of the problem is $O(\max(|V|, |E|, |\log \lambda|))$.

Definition 2.1. *An approximation algorithm $\mathcal{A}$ is defined to be a Fully Polynomial Time Approximation Scheme (FPTAS) for computing $Z(\mathbb{G})$ if, given arbitrary $\epsilon > 0$, it produces a value $\hat{Z}$ satisfying*

$$\exp(-\epsilon) \leq \frac{\hat{Z}}{Z(\mathbb{G})} \leq \exp(\epsilon),$$

in time which is polynomial in $n, \frac{1}{\epsilon}$.

We now state our main result.

Theorem 2.1. *There exist a deterministic algorithm which provides an FPTAS for computing $Z(\mathbb{G})$ corresponding to a constant activity λ for any graph, whose degree is bounded by a constant, Δ. The above algorithm has complexity $O((\frac{n}{\epsilon})^{\kappa \log \Delta + 1})$, where $\kappa = -2 \Big/ \log\left(1 - \frac{2}{(\sqrt{1+\lambda\Delta}+1)}\right)$.*

Thus, while the running time of the algorithm depends polynomially on $1/\epsilon$ (hence *Fully* Polynomial approximation scheme) the degree of the polynomial depends on both Δ and λ. In the case of no assumptions on the degree of the graph, the complexity of counting approximately the number of matchings becomes $\exp(O(\sqrt{n}\log^2 n))$. The hidden dependence on ϵ is a polynomial in $\frac{1}{\epsilon}$.

[1] For ease of notation we suppress the dependence of $Z(\mathbb{G})$ on λ.

We begin by establishing the following identity. This identity appears in various forms in the context of Markov chain sampling method as well. The identity is also the essence of the cavity method in statistical physics.

Proposition 2.2. *The following identity holds*

$$Z(\mathbb{G}) = \frac{1}{\prod_{1\le k\le |V|} \mathbb{P}_{\mathbb{G}_k}(v_k \notin \boldsymbol{M})}.$$

Proof. Observe that for every graph $\mathbb{G}$ and vertex $v \in \mathbb{G}$

$$\mathbb{P}_{\mathbb{G}}(v \notin \boldsymbol{M}) = \frac{Z(\mathbb{G}\setminus\{v\})}{Z(\mathbb{G})}. \tag{1}$$

Applying this identity recursively to $v_k, \mathbb{G}_k$, and using the convention that the partition function for the graph with no edges equals 1, we obtain the result. □

The above recursion for the partition function (matching polynomial) is the same as the one obtained by Godsil in [God81]. It was also used by Kahn and Kim [KK98] for the analysis of random matchings on regular large degree graphs.

The following corollary is a straightforward application of Proposition 2.2 and therefore we shall focus our attention on constructing an algorithm for computing an approximation of $\mathbb{P}_{\mathbb{G}}(v \notin \boldsymbol{M})$.

Corollary 2.3. *Given any $\epsilon > 0$, if there exists a fully polynomial time algorithm $\mathcal{A}$, which on input $(\mathbb{G}, v)$, computes a value $\hat{p}(v)$ satisfying*

$$\Big|\frac{\hat{p}(v)}{\mathbb{P}_{\mathbb{G}}(v \notin \boldsymbol{M})}\Big| \le \exp(\frac{\epsilon}{n}), \tag{2}$$

then one immediately obtains a fully polynomial time approximation algorithm for $Z(G)$.

3 Correlation decay analysis

A salient feature of the matching problem is that it allows a very simple recursive expression for the value $\mathbb{P}_{\mathbb{G}}(v \notin \boldsymbol{M})$.

Proposition 3.1. *The following holds for every vertex v:*

$$\mathbb{P}_{\mathbb{G}}(v \notin \boldsymbol{M}) = \frac{1}{1+\lambda\sum_{u\in N(v,\mathbb{G})} \mathbb{P}_{\mathbb{G}\setminus\{v\}}(u \notin \boldsymbol{M})}. \tag{3}$$

In particular,

$$\mathbb{P}_{\mathbb{G}}(v \notin \boldsymbol{M}) \ge \frac{1}{1+\lambda\Delta}. \tag{4}$$

Proof. We have

$$Z(\mathbb{G}) = \sum_{M\in\mathcal{M}(\mathbb{G}):v\notin M} \lambda^{|M|} + \sum_{u\in N(v,\mathbb{G})} \sum_{M\in\mathcal{M}(\mathbb{G}):(v,u)\in M} \lambda^{|M|}. \tag{5}$$

Observe that the set of all matchings M such that $v \notin M$ is the set of all matchings in $\mathbb{G}\setminus\{v\}$. Also for every matching M containing (v,u), $M\setminus\{(v,u)\}$ induces a matching in the graph $\mathbb{G}\setminus\{v,u\}$. Conversely,

for every matching M in $\mathbb{G} \setminus \{v,u\}$, $M \cup \{(v,u)\}$ creates a matching in $\mathbb{G}$ containing the edge (v,u). Thus we can rewrite (5) as

$$Z(\mathbb{G}) = Z(\mathbb{G} \setminus \{v\}) + \sum_{u \in N(v,\mathbb{G})} \lambda Z(\mathbb{G} \setminus \{v,u\}). \tag{6}$$

Dividing both parts by $Z(\mathbb{G} \setminus \{v\})$ and using the identity (1) we obtain the result. □

For every subgraph $\hat{\mathbb{G}}$ of the graph $\mathbb{G}$ every vertex $v \in \hat{\mathbb{G}}$ and every $t \in \mathbb{Z}_+$ we introduce a quantity $\Phi_{\hat{\mathbb{G}}}(v,t)$ computed inductively as follows.

1. $\Phi_{\hat{\mathbb{G}}}(v,0) = 1$ for all $\hat{\mathbb{G}}, v$.
2. For every $t \geq 1$,

$$\Phi_{\hat{\mathbb{G}}}(v,t+1) = \frac{1}{1 + \lambda \sum_{u \in N(v,\hat{\mathbb{G}})} \Phi_{\hat{\mathbb{G}} \setminus \{v\}}(u,t)}. \tag{7}$$

The following is an immediate consequence of the definition of Φ: for every $t \geq 0$,

$$\frac{1}{1+\lambda\Delta} \leq \Phi_{\hat{\mathbb{G}}}(v,t) \leq 1, \tag{8}$$

While we have introduced the values $\Phi_{\hat{\mathbb{G}}}(v,t)$ for potentially exponentially many subgraphs of $\mathbb{G}$, it is only a small family of subgraphs of $\mathbb{G}$ for which the value of Φ will be relevant to us. The quantity $\Phi_{\hat{\mathbb{G}}}(v,t)$ will serve as an approximation of $\mathbb{P}_{\hat{\mathbb{G}}}(v \notin \boldsymbol{M})$. The essence of this approximation is described in the following result.

Theorem 3.2 (Correlation Decay)**.** *The following holds for every vertex v and every positive even value t:*

$$\left| \log \mathbb{P}_{\mathbb{G}}(v \notin \boldsymbol{M}) - \log \Phi_{\mathbb{G}}(v,t) \right| \leq \left(1 - \frac{2}{(\sqrt{1+\lambda\Delta}+1)}\right)^{t/2} \log(1+\lambda\Delta).$$

Remark : In the proof below we analyze two steps of the recursions (3) and (7). This leads to a correlation decay rate $\approx (1 - \frac{1}{\sqrt{\lambda\Delta}})$. We could instead use a one-step analysis, but this would give us a decay rate only $\approx (1 - \frac{1}{\lambda\Delta})$. While this would not make a big difference in the case $\lambda, \Delta = O(1)$, it does make a difference in the general case, since Δ could be as large as $n-1$. The case $\lambda = O(n^{\frac{1}{3}})$ was used in [GK07b] for constructing an approximation algorithm for computing a permanent. It also seems that 3-step analysis would not buy us better correlation decay as the rate $\approx (1 - \frac{1}{\sqrt{\lambda\Delta}})$ seems to be tight. This observation is also implicit in [KK98].

Proof. Fix a vertex $v \in \mathbb{G}$, and let $N(v,\mathbb{G}) = \{u_1, \ldots, u_m\}$, $N(u_i, \mathbb{G} \setminus \{v\}) = \{w_1^{(i)}, ..., w_{m_i}^{(i)}\}$. We introduce the following shorthand notations, with x's representing the true probabilities (of certain vertices not being in random matchings) and y's representing the corresponding approximations:

$$x = \log \mathbb{P}_{\mathbb{G}}(v \notin \boldsymbol{M}),\ x_i = \log \mathbb{P}_{\mathbb{G}\setminus\{v\}}(u_i \notin \boldsymbol{M}),\ x_j^{(i)} = \log \mathbb{P}_{\mathbb{G}\setminus\{v,u_i\}}(w_j^{(i)} \notin \boldsymbol{M}), \tag{9}$$

$$y = \log \Phi_{\mathbb{G}}(v,t),\ y_i = \log \Phi_{\mathbb{G}\setminus\{v\}}(u_i, t-1),\ y_j^{(i)} = \log \Phi_{\mathbb{G}\setminus\{v,u_i\}}(w_j^{(i)}, t-2), \tag{10}$$

for $i = 1, \ldots, m,\ j = 1, \ldots, m_i$.

Let $M = \sum_{i=1}^{m} m_i$, $\vec{z} = \{z_1^{(1)}, \ldots, z_{m_1}^{(1)}, \ldots, z_1^{(m)}, ..., z_{m_m}^{(m)}\}$. Let $f : [0,1]^M \to [0,1]$ be given as

$$f(\vec{z}) = \log\Big(1 + \lambda \sum_{i=1}^{m} \frac{1}{1 + \lambda \sum_{j=1}^{m_i} e^{z_j^{(i)}}}\Big).$$

Then we can rewrite (3) and (7) as

$$x = -f(\vec{x}),$$
$$y = -f(\vec{y}).$$

We consider $g(\alpha) = f(\alpha\vec{x} + (1-\alpha)\vec{y})$ as a function of one-dimensional parameter $\alpha \in [0,1]$ and fixed vectors $\vec{x}, \vec{y}$. Applying the mean value theorem, there exists $\alpha \in [0,1]$ such that for $\vec{z}_\alpha = \alpha\vec{x} + (1-\alpha)\vec{y}$,

$$|x - y| = |\nabla f(\vec{z}_\alpha) \cdot (\vec{x} - \vec{y})| \overset{(a)}{\leq} |\nabla f(\vec{z}_\alpha)|_{L_1} \, \|\vec{x} - \vec{y}\|_{L_\infty}, \tag{11}$$

where (a) follows from Hölder's inequality. It is easy to see that

$$\|\nabla f(\vec{z})\|_{L_1} = \frac{1}{1 + \lambda \sum_{i=1}^{m} \frac{1}{1 + \lambda \sum_{j=1}^{m_i} e^{z_j^{(i)}}}} \sum_{i=1}^{m} \lambda \left(\frac{1}{1 + \lambda \sum_{j=1}^{m_i} e^{z_j^{(i)}}}\right)^2 \lambda \sum_{j=1}^{m_i} e^{z_j^{(i)}}.$$

Define $A_i = 1 + \lambda \sum_{j=1}^{m_i} e^{z_j^{(i)}}$. The L_1-norm can be re-written, in terms of A_i as

$$\|\nabla f(\vec{z})\|_{L_1} = \frac{1}{1 + \lambda \sum_{i=1}^{m} \frac{1}{A_i}} \sum_{i=1}^{m} \frac{\lambda(A_i - 1)}{A_i^2} = 1 - \frac{1 + \lambda \sum_{i=1}^{m} \frac{1}{A_i^2}}{1 + \lambda \sum_{i=1}^{m} \frac{1}{A_i}}.$$

It is not difficult to see that the expression $\frac{1 + \lambda \sum_{i=1}^{m} \frac{1}{A_i^2}}{1 + \lambda \sum_{i=1}^{m} \frac{1}{A_i}}$ is minimized, for $0 \leq 1/A_i \leq \infty$, when $1/A_i = \frac{\sqrt{1+\lambda m} - 1}{\lambda m}$. To show this, first observe by taking partial derivatives that the minimum occurs when all of A_i are equal. Then the solution for optimal A_i reduces to a quadratic equation. Substituting for the minimum value, one obtains

$$\|\nabla f(\vec{z})\|_{L_1} = 1 - \frac{1 + \lambda \sum_{i=1}^{m} \frac{1}{A_i^2}}{1 + \lambda \sum_{i=1}^{m} \frac{1}{A_i}} \leq 1 - \frac{2}{(\sqrt{1 + \lambda m} + 1)} \leq 1 - \frac{2}{(\sqrt{1 + \lambda \Delta} + 1)}.$$

Applying this to (11) we obtain

$$\Big|\log \mathbb{P}_{\mathbb{G}}(v \notin \boldsymbol{M}) - \log \Phi_{\mathbb{G}}(v,t)\Big| \leq \Big(1 - \frac{2}{(\sqrt{1 + \lambda\Delta} + 1)}\Big) \max_{i,j} \Big|\log \mathbb{P}_{\mathbb{G}\setminus\{v,u_i\}}(w_j^{(i)} \notin \boldsymbol{M}) - \log \Phi_{\mathbb{G}\setminus\{v,u_i\}}(w_j^{(i)}, t-2)\Big|.$$

Iterating this bound $t/2$ times we obtain that $\Big|\log \mathbb{P}_{\mathbb{G}}(v \notin \boldsymbol{M}) - \log \Phi_{\mathbb{G}}(v,t)\Big|$ is at most $(1 - \frac{2}{(\sqrt{1+\lambda\Delta}+1)})^{t/2}$ times $\max_{\hat{G},u} \Big|\log \mathbb{P}_{\hat{\mathbb{G}}}(u \notin \boldsymbol{M}) - \log \Phi_{\hat{\mathbb{G}}}(u,0)\Big|$, where the maximum is over all subgraph/vertex pairs $(\hat{\mathbb{G}} \subset \mathbb{G}, u \in \hat{\mathbb{G}})$. Applying (4) and the fact $\Phi_{\hat{\mathbb{G}}}(u,0) = 0$, $\max_{\hat{G},u} \Big|\log \mathbb{P}_{\hat{\mathbb{G}}}(u \notin \boldsymbol{M}) - \log \Phi_{\hat{\mathbb{G}}}(u,0)\Big|$ is at most $\log(1 + \lambda\Delta)$. □

4 Algorithm

Our algorithm is based on computing the values $\Phi_{\mathbb{G}}(v,t)$, for $t = O(\log n)$. In our analysis of algorithm complexity, we assume that each arithmetic operation takes one unit of time. This can be done since arithmetic operations introduce at most a polynomial time overhead in the computation.

Lemma 4.1. *The values $\Phi_{\mathbb{G}}(v,t)$ can be computed in time $O(\Delta^t)$. In particular when $t = O(\log n)$ and $\Delta = O(1)$, $\Phi_{\mathbb{G}}(v,t)$ can be computed in polynomial time.*

Proof. The proof follows immediately from recursion (7). □

Now based on this lemma, we propose the following algorithm for estimating the partition function $Z(\mathbb{G})$.

Algorithm CountMATCHINGS

INPUT: A graph/activity pair $(\mathbb{G}, \lambda)$ and a positive integer t.
BEGIN
Set $Z = 1, \hat{\mathbb{G}} = \mathbb{G}$.
While $\hat{G} \neq \emptyset$, find an arbitrary node $v \in \hat{G}$. Compute $\Phi_{\hat{\mathbb{G}}}(v,t)$.
Set $Z = \frac{Z}{\Phi_{\hat{\mathbb{G}}}(v,t)}$, $\hat{G} = \hat{G} \setminus \{v\}$.
END
OUTPUT: Z.

As a final step we show that Φ can be used to approximate the marginal probabilities $\mathbb{P}_{\mathbb{G}}(v \notin \boldsymbol{M})$ with polynomial accuracy.

Lemma 4.2. *Let $\delta = -\log\left(1 - \frac{2}{(\sqrt{1+\lambda\Delta}+1)}\right)$. If $t = 2\lceil(\log n + \log\log(1+\lambda\Delta) - \log\epsilon)/\delta\rceil$, then*

$$e^{-\frac{\epsilon}{n}} \leq \frac{\Phi_{\mathbb{G}}(v,t)}{\mathbb{P}_{\mathbb{G}}(v \notin \boldsymbol{M})} \leq e^{\frac{\epsilon}{n}}.$$

Proof. Theorem 3.2 implies

$$\begin{aligned}\left|\log \frac{\Phi_{\mathbb{G}}(v,t)}{\mathbb{P}_{\mathbb{G}}(v \notin \boldsymbol{M})}\right| &\leq \left(1 - \frac{2}{(\sqrt{1+\lambda\Delta}+1)}\right)^{(\log n + \log\log(1+\lambda\Delta) - \log\epsilon)/\delta} \log(1+\lambda\Delta) \\ &= \frac{\epsilon}{n}.\end{aligned}$$

The result then follows. □

Proof of Theorem 2.1. First we assume the case of bounded degree graph and constant activity: $\lambda, \Delta = O(1)$. The bound on t given by Lemma 4.2 becomes in this case $t = O(\log n)$. The algorithm providing FPTAS is CountMATCHINGS, with input $\mathbb{G}, \lambda$, and $t = O(\log n)$ as in Lemma 4.2. We can combine Lemma 4.1, Lemma 4.2 and Corollary 2.3 and observe that the complexity of the algorithm is bounded by $O(n\Delta^t)$ where $t = 2\lceil(\log n + \log\log(1+\lambda\Delta) - \log\epsilon)/\delta\rceil$. This gives the desired FPTAS with the complexity bounded as stated in the theorem.

In the general case we have for δ defined in Lemma 4.2 that $\delta^{-1} = O(\sqrt{\lambda\Delta})$. Due to our assumption $\lambda = O(n)$, this gives $t = O(\frac{1}{\delta}\log\frac{n}{\epsilon}) = O(\sqrt{\lambda\Delta}\log\frac{n}{\epsilon})$. Thus again applying Lemma 4.1, the complexity of the algorithm CountMATCHINGS is

$$O(n\Delta^t) = O(n^t) = \exp(O(\sqrt{\lambda\Delta}\log^2 n)),$$

where $\Delta = O(n)$ is used. (Here we ignore the explicit dependence on ϵ but it is easy to see that the complexity depends polynomially on $\frac{1}{\epsilon}$.) The special case corresponding to counting matchings $\lambda = 1$ leads to an upper bound $\exp(O(\sqrt{n}\log^2 n))$. □

5 Comparison of the correlation decay on a computation tree and the spatial correlation decay property

As we have mentioned above, the (spatial) correlation decay is known to hold for the Gibbs distribution associated with matchings [HL72],[vdB98] in the following sense: for every node v the marginal probability $\mathbb{P}(v \in \boldsymbol{M})$ is asymptotically independent from similar probabilities associated with nodes on a boundary of the depth-d neighborhood $B(v,d)$ of v in the underlying graph $\mathbb{G}$. It is natural to try to use this result directly as a method for computing approximately the marginal probabilities $\mathbb{P}(v \in \boldsymbol{M})$, for example by computing $\mathbb{P}_{B(v,d)}(\mathbb{P}(v \in \boldsymbol{M}))$ instead, say using brute force computation. Unfortunately, this does not lead to a polynomial time algorithm. In order to obtain ϵ-approximation of the partition function, we need order $O(\epsilon/n)$ approximation of the marginal probabilities, which means the depth d of the neighborhood $B(v,d)$ needs to be at least $O(\log n)$. Here n is the number of nodes. But the resulting cardinality of $B(v,d)$, even for the case of constant degree graphs is $O(\Delta^{\log n}) = n^{O(1)}$ - polynomial in n and the brute-force computation effort would be exponential in n. Notice that even if the underlying graph has a polynomial expansion $|B(v,d)| \leq d^r$, for some power $r \geq 1$, the brute-force computation would still be $O(\exp(\log^r n))$ which is super-polynomial. This is where having correlation decay on computation tree as opposed to spatial decay of correlation is useful.

6 Conclusions

We have constructed a deterministic algorithm for counting approximately the number of matchings of a given graph. The algorithm runs in polynomial time for the class of bounded degree graphs, and in subexponential time $\exp(O(\sqrt{n}\log^2 n))$ for the class of all graphs, where n is the number of nodes.

A natural open question is whether there is an FPTAS for counting matchings in graphs of unbounded degrees? There seem to be some fundamental limitations of the approach proposed in this paper – the correlation decay rate corresponding to the case of matchings seems to be of order $1-O(\frac{1}{\sqrt{\Delta}})$, and thus we speculate that the improvement should come along some combinatorial, (rather than statistical physics), type arguments. In general, it is of interest to see to what extent the correlation decay approach can be used for solving approximately other counting problems for which the MCMC method has been successful. This line of investigation might also bring us a step closer to understanding the extent to which randomized algorithms are more powerful than deterministic algorithms.

Acknowledgements

The authors would like to thank the anonymous referees for their helpful comments. Bayati, Nair and Tetali acknowledge the support of the Theory Group at Microsoft Research, where part of this work was carried out.

References

[BG06] A. Bandyopadhyay and D. Gamarnik, *Counting without sampling. New algorithms for enumeration problems using statistical physics.*, Proceedings of 17th ACM-SIAM Symposium on Discrete Algorithms (SODA), 2006.

[BKMP01] N. Berger, C. Kenyon, E. Mossel, and Y. Peres, *Glauber dynamics on trees and hyperbolic graphs*, Proc. 42nd IEEE Symposium on Foundations of Computer Science (2001).

[BN06] M. Bayati and C. Nair, *A rigorous proof of cavity method for counting matchings*, Annual Allerton Conference on Communication, Control and Computing, 2006.

[Dob70] R. L. Dobrushin, *Prescribing a system of random variables by the help of conditional distributions*, Theory of Probability and its Applications **15** (1970), 469–497.

[DSVW04] M. Dyer, A. Sinclair, E. Vigoda, and D. Weitz, *Mixing in time and space for lattice spin systems: a combinatorial view*, Random Struct. & Alg. **24** (2004), 461–479.

[GK07a] D. Gamarnik and D. Katz, *Correlation decay and deterministic FPTAS for counting list-colorings of a graph*, Proceedings of 18th ACM-SIAM Symposium on Discrete Algorithms (SODA), 2007.

[GK07b] ———, *A deterministic approximation algorithm for computing a permanent of a 0,1 matrix*, Preprint on http://arxiv.org/abs/math.CO/0702039 (2007).

[GMP05] L. A. Goldberg, R. Martin, and M. Paterson, *Strong spatial mixing with fewer colors for lattice graphs*, SIAM J. Comput. **35** (2005), no. 2, 486–517.

[God81] C. D. Godsil, *Matchings and walks in graphs*, J. Graph Th. **5** (1981), 285–297.

[HL72] O.J.Heilman and E.H.Lieb, *Theory of monomer-dimer systems*, Comm. Math. Phys. **25** (1972), 190–232.

[J03] M. Jerrum, *Counting, sampling and integrating: algorithms and complexity*, Lectures in Mathematics, ETH-Zurich, Birkhauser Verlag (2003).

[JS97] M. Jerrum and A. Sinclair, *The Markov chain Monte Carlo method: an approach to approximate counting and integration*, Approximation algorithms for NP-hard problems (D. Hochbaum, ed.), PWS Publishing Company, Boston, MA, 1997.

[JS] K. Jung and D. Shah, *Inference in Binary Pair-wise Markov Random Field through Self-Avoiding Walk,* Preprint on http://arxiv.org/abs/cs.AI/0610111v2.

[JSV04] M. Jerrum, A. Sinclair, and E. Vigoda, *A polynomial-time approximation algorithms for permanent of a matrix with non-negative entries*, Journal of the Association for Computing Machinery **51** (2004), no. 4, 671–697.

[JVV86] M. Jerrum, L. Valiant, and V. Vazirani, *Random generation of combinatorial structures from a uniform distribution*, Theoret. Comput. Sci. **43** (1986), no. 2-3, 169188.

[KK98] J. Kahn and J. H. Kim, *Random matchings in regular graphs*, Combinatorica **8** (1998), 201–226.

[LSW00] N. Linial, A. Samorodnitsky, and A. Wigderson, *A deterministic strongly polynomial algorithm for matrix scaling and approximate permanents*, Combinatorica **20** (2000), no. 4, 545–568.

[MS06] A. Montanari and G. Semerjian, *Rigorous inequalities between length and time scales in glassy systems*, Preprint in arXiv.org (2006).

[NT] C. Nair and P. Tetali, *The correlation decay (CD) tree and strong spatial mixing in multi-spin systems*, Preprint on http://front.math.ucdavis.edu/math.PR/0701494.

[S93] A. Sinclair, *Algorithms for random generation and counting: a Markov chain approach*, Birkhauser Progress in Theoretical Computer Science Series, Birkhauser Verlag (1993).

[V01] S. Vadhan, *The complexity of counting in sparse, regular, and planar graphs*, SIAM Journal on Computing **31** (2001), no. 2, pp. 398–427.

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

[vdB98] J. van den Berg, *On the absence of phase transition in the monomer-dimer model*, CWI reports, PNA-R9813, ISSN 1386-3711 (1998).

[Wei06] D. Weitz, *Counting independent sets up to the tree threshold*, Proc. 38th Ann. Symposium on the Theory of Computing (2006).