Deterministic polynomial-time approximation algorithms for partition functions and graph polynomials

Viresh Patel[*]   Guus Regts[†]

October 28, 2016

Abstract

In this paper we show a new way of constructing deterministic polynomial-time approximation algorithms for computing complex-valued evaluations of a large class of graph polynomials on bounded degree graphs. In particular, our approach works for the Tutte polynomial and independence polynomial, as well as partition functions of complex-valued spin and edge-coloring models.

More specifically, we define a large class of graph polynomials $\mathcal{C}$ and show that if $p \in \mathcal{C}$ and there is a disk D centered at zero in the complex plane such that $p(G)$ does not vanish on D for all bounded degree graphs graphs G, then for each z in the interior of D there exists a deterministic polynomial-time approximation algorithm for evaluating $p(G)$ at z. This gives an explicit connection between absence of zeros of graph polynomials and the existence of efficient approximation algorithms, allowing us to show new relationships between well-known conjectures.

Our work builds on a recent line of work initiated by Barvinok [2, 3, 4, 5], which provides a new algorithmic approach besides the existing Markov chain Monte Carlo method and the correlation decay method for these types of problems.

Keywords: approximation algorithms, Tutte polynomial, independence polynomial, partition function, graph homomorphism, Holant problem.

1 Introduction

Computational counting is an important area of computer science where one seeks to find efficient algorithms to count certain combinatorial objects such as independent sets, proper colourings, or matchings in a graph. More generally, each combinatorial counting problem has an associated generating function, namely the independence polynomial for independent sets, the chromatic and more generally Tutte polynomial for proper graph colourings, and the matching polynomial for matchings. Such graph polynomials are studied in mathematics and computer science, but also in statistical physics where they are normally referred to as partition functions. A fundamental question asks for which graphs and at which numerical values one can approximately evaluate these polynomials efficiently. Indeed the counting problems correspond to evaluating these graph polynomials or partition functions at particular values.

Many of these counting problems are known to be computationally hard in the sense of being #P-hard, even when one restricts to graphs of maximum degree at most three [10, 19]. On the other hand several efficient randomized approximation algorithms exist for some of these #P-hard problems via the use of the powerful Markov chain Monte Carlo technique. In a major breakthrough, Weitz [46], inspired by ideas from statistical physics, developed the so-called correlation decay method allowing him to obtain the first efficient deterministic approximation algorithm for counting independent sets in graphs of maximum degree at most five. (One expects no such algorithm for graphs of maximum degree larger than five [40], while previously the best known (randomized) algorithm worked only for graphs of maximum degree at most four.) The correlation decay method has subsequently been refined and applied to various other problems; see e.g. [1, 22, 34, 39] and references therein.

In this paper we consider a different approach. The approach is quite robust in that it can be applied to a large class of graph polynomials and gives the first general polynomial-time method to approximate graph polynomials at complex values for bounded degree graphs. Very recently complex evaluations have also been considered by Harvey, Srivastava, and Vondrák [27] for the special case of the independence polynomial. Complex evaluations of graph polynomials, aside from being the natural extensions of real evaluations, arise as interesting counting problems e.g. counting restricted tensions or flows can be modelled as the partition functions of a complex spin system (see [23]) and the number of homomorphisms into any fixed graph can be modelled as the partition function of a complex edge-coloring model (see [43, 44]).

A further important aspect of our work is to highlight the explicit relation between the (absence of complex) roots of a graph polynomial and efficient algorithms to evaluate it. Indeed, in Remark 1.3 below we give the explicit connection between a conjecture of Sokal on zero-free regions of the chromatic polynomial and the notorious algorithmic problem of efficiently approximating the number of proper colourings in a bounded degree graph.

Our approach combines a number of ingredients including ideas from sparse graph limits [17], results on the locations of zeros of graph polynomials and partition functions [38, 37, 29, 7, 8, 36] and an algorithmic development due to Barvinok [2]. The Taylor approximation technique of Barvinok has been used to construct deterministic quasi-polynomial-time approximation algorithms for evaluating a number of graph partition functions (for general graphs); see e.g. work by Barvinok [2, 3, 4, 5], by Barvinok and Soberón [7, 8], and by the second author [36].

The approach can be roughly described as follows. First the problem of evaluating the partition function or graph polynomial is cast as the evaluation of a univariate polynomial. Next, a region is identified where this polynomial does not vanish; hence in this region the logarithm of the polynomial is well-approximated by a low-order Taylor approximation (of order $\log n$, where n in the degree of the polynomial). Finally we must compute this Taylor approximation by efficiently computing the first $O(\log n)$ coefficients of the polynomial. So far this approach has only resulted in algorithms that run in quasi-polynomial time. The main technical contribution of the present paper is a polynomial-time algorithm for computing (essentially) the first $O(\log n)$ coefficients of a large class of graph polynomials whenever we work with bounded degree graphs cf. Theorem 3.1, and we believe it to be of independent interest.

Below we shall state and discuss some concrete results that can be obtained by combining this approach with (known) results on the location of roots of graph polynomials and partition functions. In particular, we obtain new deterministic polynomial-time algorithms (FPTAS) for evaluating the independence polynomial, the Tutte polynomial, and computing partition functions of spin and edge-coloring models in the case of bounded degree graphs. Before we state our algorithmic results, we first need a definition. Since we will approximate polynomials at complex values, we define what it

means to be a good approximation.

Definition 1.1. Let q be a complex number and let $\varepsilon > 0$. We call a complex number ξ a *multiplicative ε-approximation to q* if $e^{-\varepsilon} \leq |q|/|\xi| \leq e^{\varepsilon}$ and if the angle between ξ and q (as seen as vectors in $\mathbb{C} = \mathbb{R}^2$) is at most ε.

1.1 The independence polynomial

The *independence polynomial* of a graph $G = (V, E)$ is denoted by $Z(G)$ and is defined as

$$Z(G)(\lambda) := \sum_{\substack{I \subseteq V \\ I \text{ independent}}} \lambda^{|I|}. \tag{1}$$

In [46] Weitz proved, based on the correlation decay method, that if $0 \leq \lambda < \lambda_c$, where

$$\lambda_c = \frac{(\Delta - 1)^{\Delta - 1}}{(\Delta - 2)^\Delta},$$

then there exists a deterministic algorithm, which given a graph $G = (V, E)$ of maximum degree at most Δ and $\varepsilon > 0$, computes a multiplicative ε-approximation to $Z(G)(\lambda)$ in time $(|V|/\varepsilon)^{O(1)}$. Sly and Sun [40] proved this is tight by showing that, as soon as $\lambda > \lambda_c$, one cannot efficiently approximate $Z(G, \lambda)$ unless NP=RP.

In Section 4 we prove the following result.

Theorem 1.1. *Let $\Delta \in \mathbb{N}$ and let $\lambda \in \mathbb{C}$ be such that $|\lambda| < \lambda^*(\Delta) := \frac{(\Delta - 1)^{\Delta - 1}}{\Delta^\Delta}$. Then there exists a deterministic algorithm, which, given a graph $G = (V, E)$ of maximum degree at most Δ and $\varepsilon > 0$, computes a multiplicative ε-approximation to $Z(G)(\lambda)$ in time $(|V|/\varepsilon)^{O(1)}$.*

Remark 1.1. From the proof of Theorem 1.1 it follows that we can take $O(1)$ to be $D\left(1 - |\lambda|/\lambda^*(\Delta)\right)^{-1} \ln(\Delta) + D'$ for some absolute constants D, D'.

For positive valued λ our result is weaker than Weitz's result since $\lambda_c > \lambda^*$. However our result works for negative[1] and even complex[2] λ. The case $\lambda < 0$ is relevant due to its connection to the Lovász local lemma, cf. [37].

The value λ^* in Theorem 1.1 originates from a paper of Scott and Sokal [37]; they showed that for graphs of maximum degree Δ, the independence polynomial does not vanish at any $\lambda \in \mathbb{C}$ satisfying $|\lambda| \leq \lambda^*$. Also the value of λ^* is tight, as there exists a sequence of trees T_n of maximum degree at most Δ and $\lambda_n < -\lambda^*$ with $\lambda_n \to -\lambda^*$ such that $Z(T_n, \lambda_n) = 0$, cf. [37, Example 3.6].

It would be very interesting to know whether the result of Weitz can be proved using the approach we take here. This would in fact follow from a confirmation of a version of a conjecture of Sokal [41] (the conjecture is stated below Question 2.4 in [41]); see Question 8.2 in the concluding remarks for the exact version that we would need. It would also be very interesting to know if it is hard to approximate $Z(G, \lambda)$ for $\lambda < -\lambda^*$. Some recent progress has been made on this by Harvey, Srivastava and Vondrák [27, Theorem 4.4], showing that it is hard unless NP=RP for λ that are far away from $-\lambda^*$, but as far as we are aware, there are no results known when λ is very close to $-\lambda^*$.

As an extension to Theorem 1.1, we are able to efficiently approximate the independence polynomial on almost the entire complex plane for the special class of claw-free graphs. We make use of a result of Chudnovsky and Seymour [16] stating that the

---

[1] In an unpublished note [42] Srivastava notes that the correlation decay method of Weitz in fact also applies to negative λ as long as $\lambda > -\lambda^*$.

[2] Very recently Harvey, Srivastava and Vondrák [27] used the correlation decay method to obtain efficient approximation algorithms for the multivariate independence polynomial. In fact, our method also applies to the multivariate independence polynomial.

independence polynomial of a claw-free graph has only negative real roots. We prove the following result in Section 4.

Theorem 1.2. *Let $\Delta \in \mathbb{N}$ and let $\lambda \in \mathbb{C}$ be such that λ is not a real negative number. Then there exists a deterministic algorithm, which, given a claw-free graph $G = (V, E)$ of maximum degree at most Δ and $\varepsilon > 0$, computes a multiplicative ε-approximation to $Z(G)(\lambda)$ in time $(|V|/\varepsilon)^{O(1)}$.*

Note that when G is the line graph of some graph H we have that $Z_G(\lambda)$ is equal to the matching polynomial of H. So in particular, Theorem 1.2 implies a result of Bayati, Gamarnik, Katz, Nair, and Tetali [1]. Our proof of it however is entirely different from the proof in [1].

1.2 The Tutte polynomial

The random cluster formulation of the Tutte polynomial of a graph $G = (V, E)$ is a two-variable polynomial, which is denoted by $Z_T(G)$ and is defined by

$$Z_T(G)(q, w) := \sum_{A \subseteq E} q^{k(A)} w^{|A|}, \tag{2}$$

where $k(A)$ denotes the number of components of the graph (V, A). In particular, if $w = -1$, $Z_T(G)(q, -1)$ is equal to the chromatic polynomial of G.

Jerrum and Sinclair [31] showed that when $q = 2$ and $w > 0$ there exists a randomized polynomial-time approximation algorithm for computing evaluations of the Tutte polynomial in general. In [25] Goldberg and Jerrum showed that approximating evaluations of the Tutte polynomial on general graphs for $q > 2$ and $w > 0$ is as hard as counting independent sets in bipartite graphs and in [26] Goldberg and Jerrum showed that for several choices of real parameters (q, w) it is even #P-hard to approximate the evaluation of the Tutte polynomial on general graphs. Goldberg and Guo [24] looked at the complexity of approximately evaluating the Tutte polynomial for general graphs at complex values.

When $w = -1$ and $q \in \mathbb{N}$, $Z_T(G)(q, w)$ gives the number of q-colorings of G. Lu and Yin [34] showed that when $q > 2.58\Delta$ there exists a deterministic polynomial-time algorithm for approximating the Tutte polynomial at $(q, -1)$ on graphs of maximum degree at most Δ. There are many randomized algorithms of the sort above with sharper bounds on q; see e.g. Jerrum [30] and Vigoda [45]. As far as we know there are no general results known for the Tutte polynomial on bounded degree graphs.

We will consider the Tutte polynomial as a univariate polynomial by considering w to be constant. In Section 5 we prove the following result.

Theorem 1.3. *Let $\Delta \in \mathbb{N}$ and let $w \in \mathbb{C}$. Then there exists a constant K (depending on Δ and w) such that if $q \in \mathbb{C}$ is such that $|q| > K$, then there exists a deterministic algorithm, which, given a loopless multigraph $G = (V, E)$ of maximum degree at most Δ and $\varepsilon > 0$, computes a multiplicative ε-approximation to $Z(G)(q, w)$ in time $(|V|/\varepsilon)^{O(1)}$.*

Remark 1.2. From the proof the Theorem 1.3 it follows that we can take $O(1)$ to be $D(1 - K/|q|)^{-1} \Delta \ln(\Delta) + D'$ for some absolute constants D, D'.

The constant K in the theorem above comes from a paper of Jackson, Procacci and Sokal [29] and unfortunately takes half a page to state exactly. However, when w satisfies $|1 + w| \leq 1$ (this includes the chromatic polynomial), then the constant K may be taken to be 6.91Δ.

Remark 1.3. Sokal [28, Conjecture 21] conjectured that $Z_T(G)(q, -1) \neq 0$ as long as $\Re(q) > \Delta(G)$. Combined with our results (and the technique from Section 4.2) a confirmation of the conjecture would imply an efficient approximation algorithm for computing the number of $(\Delta + 1)$-colorings of any graph G of maximum degree at most Δ, a notorious problem in computational counting.

1.3 Partition functions of spin models

Le $A \in \mathbb{C}^{k \times k}$ be a symmetric matrix. In the context of statistical physics A is often called a *spin model* cf. [18]. For a graph $G = (V, E)$, the partition function of A is defined as

$$p(G)(A) = \sum_{\phi: V \to [k]} \prod_{\{u,v\} \in E} A_{\phi(u), \phi(v)}. \tag{3}$$

If A is the adjacency matrix of some graph H, then $p(G)(H)$ is equal to the number of graph homomorphisms from G to H. In [7] $p(G)(A)$ is called the graph homomorphism partition function.

Building on a line of research started by Dyer and Greenhill [20] and Bulatov and Grohe [11], a full dichotomy theorem has been proved for the complexity of exactly computing the partition function of a complex spin model by Cai, Chen and Lu [12]. This dichotomy essentially says that computing the partition function of A exactly is #P hard unless the matrix A has some special structure.

Lin, Liu and Lu [33] proved, using the correlation decay approach, that for fixed $\Delta \in \mathbb{N}$, if a real matrix A is sufficiently close to the all ones matrix (i.e. $|A_{i,j} - 1| \leq O(1)/\Delta$ for all $i, j = 1, \ldots k$), then there exists a $(|V(G)|/\varepsilon)^{O(1)}$-time algorithm for computing a multiplicative ε-approximation to $P(G)(A)$ on graphs of maximum degree at most Δ. Barvinok and Sobéron [7] showed that there exists a $(|V(G)|/\varepsilon)^{O(\ln|V(G)|)}$-time algorithm for complex-valued matrices A that satisfy $|A_{i,j} - 1| \leq O(1)/\Delta$ for all $i, j = 1, \ldots, k$.

Building on the work of Barvinok and Sobéron we prove in Section 6 the following result.

Theorem 1.4. *Let $\Delta, k \in \mathbb{N}$. Then there exists a deterministic algorithm, which, given a graph $G = (V, E)$ of maximum degree at most Δ, a (complex-valued) symmetric $k \times k$ matrix A such that $|A_{i,j} - 1| \leq 0.34/\Delta$ for all $i, j = 1, \ldots, k$, and $\varepsilon > 0$, computes a multiplicative ε-approximation to $p(G)(A)$ in time $(|V|/\varepsilon)^{O(1)}$.*

Remark 1.4. The constant 0.34 can be replaced by 0.45 if $\Delta \geq 3$, and by 0.54 if Δ is large enough, cf. [7].

In [8] Barvinok and Soberón introduced partition functions of graph homomorphisms of G with multiplicities and gave a quasi-polynomial-time algorithm for computing them for certain matrices. In Section 6 we will show that our results also apply to these partition functions.

1.4 Partition functions of edge-coloring models

Edge-coloring models originate in statistical physics and their partition functions have been introduced to the graph theory community by de la Harpe and Jones [18] (where they are called vertex models). We call any map $h : \mathbb{N}^k \to \mathbb{C}$ a *k-color edge-coloring model*. For a graph $G = (V, E)$, the *partition function* of h is defined by

$$p(G)(h) := \sum_{\phi: E \to [k]} \prod_{v \in V} h(\phi(\delta(v))), \tag{4}$$

where $\delta(v)$ denotes the set of edges incident with the vertex v and $\phi(\delta(v))$ denotes the multiset of colors that the vertex v 'sees', which we identify with its incidence vector in $\mathbb{N}^k$ so that we can apply h to it.

Partition functions of edge-coloring models form a rich class of graph parameters including the number of matchings (take $h : \mathbb{N}^2 \to \mathbb{C}$ defined by $h(\alpha) = 1$ if $\alpha_1 \leq 1$ and 0 otherwise), as well as partition functions of spin models, as has been proved by Szegedy [43, 44]. These partition functions can be seen as Holant problems; see e.g.

[14, 15, 13]. They can also be seen as tensor network contractions. We refer the reader to [35] for more background.

Just as for partition functions for spin models much work has been done to establish a complexity dichotomy result for exactly computing Holant problems; see [14, 15, 13]. Not much is known about the complexity of approximating partition functions of edge-coloring models except for a few special cases. As already mentioned, Bayati, Gamarnik, Katz, Nair, and Tetali [1] found an efficient approximation algorithm for counting matchings in bounded degree graphs and Lin, Liu and Lu [33] found efficient approximation algorithms for counting edge covers. Both of these algorithms are based on the correlation decay method.

Building on work of the second author [36] we will prove the following result in Section 7.

Theorem 1.5. *Let* $\Delta, k \in \mathbb{N}$. *Then there exists a deterministic algorithm, which, given a multigraph* $G = (V, E)$ *of maximum degree at most* Δ, *a k-color edge-coloring model h such that* $|h(\phi) - 1| \leq 0.35/(\Delta + 1)$ *for all* $\phi \in \mathbb{N}^k$, *and* $\varepsilon > 0$, *computes a multiplicative* ε-*approximation to* $p(G)(h)$ *in time* $(|V|/\varepsilon)^{O(1)}$.

Remark 1.5. The constant 0.35 may be replaced by 0.47 if $\Delta \geq 3$ and by 0.56 if Δ is large enough; see [36]. Moreover, for readers familiar with the orthogonal group invariance of these partition functions, it is interesting to note that one can use Corollary 6b from [36] to find a much larger family of edge-coloring models for which the partition function can be efficiently approximated.

1.5 Organization

In the next section we shall consider an algorithm due to Barvinok [2] to approximate evaluations of polynomials. Section 3 contains our main technical contribution: we will introduce a class of graph polynomials and give an efficient algorithm for computing their low order coefficients on bounded degree graphs. These two algorithms (or variations of them) will then be combined in Sections 4–7 to prove the results above. These sections can be read independently of one another. Finally, we conclude in Section 8 with some remarks and questions.

2 Approximating evaluations of polynomials

In this section we present an algorithm due to Barvinok [2] to approximate evaluations of polynomials. We take a slightly different approach and give full details for the sake of completeness.

Let $p \in \mathbb{C}[z]$ be a polynomial $p(z) = a_0 + a_1 z + \cdots + a_d z^d$ of degree d and suppose that $p(z) \neq 0$ for all z in an open disk D of radius M. Define the function f on this disk by

$$f(z) := \ln p(z), \tag{5}$$

(where we fix a branch of the logarithm by fixing the principal value of the logarithm at $p(0)$). Recall by Taylor's Theorem that for each $t \in D$, $f(t) = \sum_{j=0}^{\infty} \frac{t^j}{j!} f^{(j)}(0)$. In order to approximate p at $t \in D$, we will find an additive approximation to f at t by truncating the Taylor expansion around $z = 0$. For each $m \in \mathbb{N}$, let

$$T_m(f)(t) := f(0) + \sum_{j=1}^{m} \frac{t^j}{j!} f^{(j)}(0). \tag{6}$$

This can then be transformed to give a multiplicative approximation to p. It will be more convenient for us to use a slightly different form of (6) which we derive below.

Let $\zeta_1, \ldots, \zeta_d \in \mathbb{C}$ be the roots of p. Then we can write $p(z) = a_d(z - \zeta_1) \cdots (z - \zeta_d)$ and $f(z) = \ln(a_d) + \ln(z - \zeta_1) + \cdots + \ln(z - \zeta_d)$. From this we see that $f'(z) = (z - \zeta_1)^{-1} + \cdots + (z - \zeta_d)^{-1}$ and hence for $j \geq 1$,

$$f^{(j)}(0) = -(j-1)! \sum_{i=1}^{d} \zeta_i^{-j}.$$

Thus defining the jth *inverse power sum* to be $p_j := \zeta_1^{-j} + \cdots + \zeta_d^{-j}$ we see that

$$T_m(f)(t) = f(0) - \sum_{j=1}^{m} \frac{p_j t^j}{j} = \ln(a_0) - \sum_{j=1}^{m} \frac{p_j t^j}{j}. \tag{7}$$

In the next proposition we derive a variant of the Newton identities that relate the inverse power sums and the coefficients of the polynomial.

Proposition 2.1. *For the polynomial $p(z) = a_0 + \cdots + a_d z^d$ as above and its inverse power sums p_j as defined above, we have for each $k = 1, 2, \ldots$ that*

$$ka_k = -\sum_{i=0}^{k-1} a_i p_{k-i}.$$

(Here we take $a_i = 0$ if $i > d$.)

Proof. From (7) we know that for $z \in D$ we have $\ln(p(z)) = \ln(a_0) - \sum_{j=1}^{\infty} \frac{p_j z^j}{j}$. Differentiating both sides and multiplying by $p(z)$ we obtain

$$p'(z) = -p(z) \sum_{j=1}^{\infty} p_j z^{j-1}$$

and so

$$\sum_{k=1}^{d} k a_k z^{k-1} = -\sum_{i=0}^{d} a_i z^i \sum_{j=1}^{\infty} p_j z^{j-1}.$$

Comparing coefficients of z^{k-1} on each side gives the desired identity. $\qquad \square$

The next lemma shows that the quality of the approximation (6) and hence (7) depends on the location of the complex roots of p.

Lemma 2.2. *Given $M > 0$ and $t \in \mathbb{C}$ satisfying $|t| < M$, there exists a constant $C = C(t, M) \leq (1 - |t|/M)^{-1}$ such that the following holds. Suppose p is a polynomial of degree d with no roots in the open disk D of radius M. Then for every $\varepsilon > 0$, $\exp(T_m(f)(t))$ is a multiplicative ε-approximation to $p(t)$, where $m = C \ln(d/\varepsilon)$.*

Proof. Let $q := |t|/M$. Then, as $|t| < M$, we have $q < 1$. We will first show that

$$|f(t) - T_m(f)(t)| \leq \frac{dq^{m+1}}{(m+1)(1-q)}. \tag{8}$$

By (7) we have

$$|f(t) - T_m(f)(t)| \leq \left| \sum_{j=m+1}^{\infty} \frac{p_j t^j}{j} \right| \leq \frac{1}{m+1} \sum_{j=m+1}^{\infty} |p_j t^j|. \tag{9}$$

By assumption we know that $|\zeta_i| \geq M$ for each $i = 1, \ldots, d$. Hence $|p_j| \leq d/M^j$ and so $|p_j t^j| \leq dq^j$. Substituting this in into (9) and using that $q < 1$ we obtain (8).

7

Take $m = C \ln(d/\varepsilon)$, where C is chosen such that $C \geq (\ln 1/q)^{-1}$ and $1/m \leq 1 - q$ (so it is easy to check that e.g. $C = (1-q)^{-1}$ suffices). Then the right-hand side of (8) is at most ε. Write $z = T_m(f)(t)$. Then we have $|e^{f(t)-z}| \leq e^{|f(t)-z|} \leq e^{\varepsilon}$ and similarly $|e^{-z+f(t)}| \leq e^{\varepsilon}$. (This follows from the fact that for a complex number $y = a + bi$, we have $|e^y| = e^a \leq e^{|y|}$.) Moreover, the angle between e^z and $e^{f(t)}$ is bounded by $|\Im \ln e^{z-f(t)}| \leq |\ln e^{z-f(t)}| \leq \varepsilon$. This shows that $e^z = \exp(T_m(f)(t))$ is a multiplicative ε-approximation to $p(t)$. $\qquad\Box$

From (7) and Lemma 2.2, if we have an efficient way of computing the inverse power sums p_j from $j = 1$ up to $O(\ln(\deg(p)))$ (which by Proposition 2.1 is essentially equivalent to computing the first $O(\ln(\deg(p)))$ coefficients of p), then we have an efficient way of approximating evaluations of p at points in the disk around zero where p is nonvanishing. We formalize this in the corollary below. In the next section we will show that for certain types of graph polynomials we can compute the inverse power sums efficiently.

Corollary 2.3. *Given $M > 0$ and $t \in \mathbb{C}$ satisfying $|t| < M$, there exists a constant $C = C(t, M) \leq (1 - |t|/M)^{-1}$ such that the following holds. Suppose p is a polynomial given by $p(z) = a_0 + a_1 z + \cdots + a_d z^d$ with no roots in the open disk D of radius M. Suppose further that we are able to compute a_0 and the inverse power sums $p_1, \ldots, p_r$ in time $\tau(r)$ for each $r = 1, \ldots, d$. Then we can compute a multiplicative ε-approximation to $p(t)$ in time $O(\tau(m))$, where $m = C \ln(d/\varepsilon)$.*

Proof. The corollary is immediate from (6), (7) and Lemma 2.2. $\qquad\Box$

3 Computing coefficients of graph polynomials

In this section we present our main technical contribution, which is an efficient way to compute the inverse power sums (and hence the coefficients) of a large class of graph polynomials for bounded degree graphs. Throughout, we will focus on graph polynomials whose coefficients can be expressed as linear combinations of induced subgraph counts. The results in this section are stated only for graphs, but are in fact valid for multigraphs. So the reader could read multigraph instead of graph everywhere in this section. (The *degree* of a vertex in a multigraph is the number of edges incident with the vertex, where a loop is counted twice.)

We start with some definitions after which we state the main result of this section. By $\mathcal{G}$ we denote the collection of all graphs and by $\mathcal{G}_k$ for $k \in \mathbb{N}$ we denote the collection of graphs with at most k vertices. A graph invariant is a function $f : \mathcal{G} \to S$ for some set S that takes the same value on isomorphic graphs. For graphs H, G we denote by $\mathrm{ind}(H, G)$ the number of induced subgraphs of G that are isomorphic to H. Note that if H is equal to the empty graph we have $\mathrm{ind}(H, G) = 1$ for all G. A *graph polynomial* is a graph invariant $p : \mathcal{G} \to \mathbb{C}[z]$, where $\mathbb{C}[z]$ denotes the ring of polynomials in the variable z over the field of complex numbers. Call a graph invariant f *multiplicative* if $f(\varnothing) = 1$ and $f(G_1 \cup G_2) = f(G_1)f(G_2)$ for all graphs G_1, G_2 (here $G_1 \cup G_2$ denotes the disjoint union of the graphs G_1 and G_2).

Definition 3.1. Let p be a multiplicative graph polynomial defined by

$$p(G)(z) := \sum_{i=0}^{d(G)} e_i(G) z^i \tag{10}$$

for each $G \in \mathcal{G}$ with $e_0(G) = 1$. We call p a *bounded induced graph counting polynomial (BIGCP)* if there exists constants $C_1, C_2 \in \mathbb{N}$ such that the following two conditions are satisfied:

8

(i) for every graph G, the coefficients e_i satisfy

$$e_i(G) := \sum_{H \in \mathcal{G}_{C_1 i}} \lambda_{H,i} \text{ind}(H, G) \tag{11}$$

for certain $\lambda_{H,i} \in \mathbb{C}$;

(ii) for each $H \in \mathcal{G}_{C_1 i}$, the coefficients $\lambda_{H,i}$ can be computed in time $O(C_2^{|V(H)|})$.

If, for example, for each i, the coefficient $e_i(G)$ in (10) is equal to the number of independent sets of size i in G, then it is easy to see that p (which is of course the independence polynomial) is a BIGCP. In this case the obvious brute force algorithm to compute the coefficient $e_i(G)$ for an n-vertex graph G runs in time $O(n^i)$ (by checking all i-subsets of $V(G)$) and if $i = O(\ln n)$ then this is quasi-polynomial time. Our main result of this section is a general algorithm for computing inverse power sums of BIGCPs (and hence the coefficients of BIGCP's by Proposition 2.1), which when applied to this example, computes $e_i(G)$ in polynomial time even when $i = O(\ln n)$ as long as the maximum degree of G is bounded.

Theorem 3.1. *Let $C > 0$ and $\Delta \in \mathbb{N}$ and let $p(\cdot)$ be a bounded induced graph counting polynomial. Then there is a deterministic $(n/\varepsilon)^{O(1)}$-time algorithm, which, given any n-vertex graph G of maximum degree at most Δ and any $\varepsilon > 0$, computes the inverse power sums $p_1, \ldots, p_m(G)$ of $p(G)$ for $m = C \ln(n/\varepsilon)$.*

Remark 3.1. The $O(1)$ term in the exponent in the theorem above can crudely be taken to be $10(CC_1 + 1) \ln(C_2 \Delta)$, where C_1 is the constant from the definition of BIGCP.

Before we prove Theorem 3.1 we will first gather some facts about induced subgraph counts and the number of connected induced subgraphs of fixed size that occur in a graph which we will need for the proof.

3.1 Induced subgraph counts

Define $\text{ind}(H, \cdot) : \mathcal{G} \to \mathbb{C}$ by $G \mapsto \text{ind}(H, G)$. So we view $\text{ind}(H, \cdot)$ as a graph invariant. We can take linear combinations and products of these invariants. In particular, for two graphs H_1, H_2 we have

$$\text{ind}(H_1, \cdot) \cdot \text{ind}(H_2, \cdot) = \sum_{H \in \mathcal{G}} c_{H_1, H_2}^H \text{ind}(H, \cdot), \tag{12}$$

where for a graph H, c_{H_1, H_2}^H is the number of pairs of subsets of $V(H)$, (S, T), such that $S \cup T = V(H)$ and $H[S] = H_1$ and $H[T] = H_2$. In particular, given H_1 and H_2, c_{H_1, H_2}^H is nonzero for only a finite number of graphs H.

Computing the parameter $\text{ind}(H, G)$ is generally difficult, but it becomes easier if H is connected (and $V(H)$ is not too large) and G has bounded degree.

Lemma 3.2. *Let H be a connected graph on k vertices and let $\Delta \in \mathbb{N}$. Then*

(i) *there is an $O(n\Delta^{k-1})$-time algorithm, which, given any n-vertex graph G with maximum degree at most Δ, checks whether $\text{ind}(H, G) \neq 0$;*

(ii) *there is an $O(n^2 \Delta^{2(k-1)})$-time algorithm, which, given any n-vertex graph G with maximum degree at most Δ, computes the number $\text{ind}(H, G)$.*

Note that Lemma 3.2 (i) enables us to test for graph isomorphism between bounded degree graphs when $|V(G)| = |V(H)|$.

Proof. Let us list the vertices of $V(H)$, $v_1, \ldots, v_k$ in such a way that for $i \geq 1$ vertex v_i has a neighbour among $v_1, \ldots, v_{i-1}$. Then to embed H into G we first select a target vertex

9

for v_1 and then given that we have embedded $v_1, \ldots, v_{i-1}$ with $i \geq 2$ there are at most Δ choices for where to embed v_i. After k iterations, we have a total of at most $n\Delta^{k-1}$ potential ways to embed H and each possibility is checked in the procedure above. Hence we determine if $\mathrm{ind}(H, G)$ is zero or not in $O(n\Delta^{k-1})$ time.

The procedure above gives a list (of size at most $n\Delta^{k-1}$) of all sets $S \subseteq V(G)$ such that $G[S] = H$, although the list may contain repetitions. It takes time $O(n\Delta^{k-1})^2 = O(n^2\Delta^{2(k-1)})$ to eliminate repetitions, and the length of the resulting list gives the value of $\mathrm{ind}(H, G)$. $\qed$

Next we consider how to enumerate all possible connected induced subgraphs of fixed size in a bounded degree graph graph. We will need the following result of Borgs, Chayes, Kahn, and Lovász [9, Lemma 2.1]:

Lemma 3.3. *Let G be a graph of maximum degree Δ. Fix a vertex v_0 of G. Then the number of connected induced subgraphs of G with k vertices containing the vertex v_0 is at most $\frac{(e\Delta)^{k-1}}{2}$.*

As a consequence we can efficiently enumerate all connected induced subgraphs of logarithmic size that occur in a bounded degree graph G.

Lemma 3.4. *There is a $O(n^2 k^5 (e\Delta)^{2k})$-time algorithm which, given $k \in \mathbb{N}$ and an n-vertex graph $G = (V, E)$ of maximum degree Δ, outputs $\mathcal{T}_k$, the set of all $S \subseteq V$ satisfying $|S| \leq k$ and $G[S]$ connected.*

Proof. By the previous result, we know that $|\mathcal{T}_k| \leq nk(e\Delta)^{k-1}$ for all k.

We inductively construct $\mathcal{T}_k$. For $k = 1$, $\mathcal{T}_k$ is clearly the set of singleton vertices and takes time $O(n)$ to output.

Given that we have found $\mathcal{T}_{k-1}$ we compute $\mathcal{T}_k$ as follows. We first compute the multiset

$$\mathcal{T}_k^* = \{S \cup \{v\} : S \in \mathcal{T}_{k-1} \text{ and } v \in N_G(S)\}.$$

Here $|N_G(S)| \leq |S|\Delta \leq k\Delta$ and takes time $O(k\Delta)$ to find (assuming G is given in adjacency list form). Therefore computing $\mathcal{T}_k^*$ takes time $O(|\mathcal{T}_{k-1}|k\Delta) = O(nk^2(e\Delta)^k)$. Finally we compute the set $\mathcal{T}_k$ by removing the repetitions in $\mathcal{T}_k^*$ (by comparing each element with all previous elements), which takes time $O(n^2 k^4 (e\Delta)^{2k})$.

Starting from $\mathcal{T}_1$, we perform the above iteration k times, requiring a total running time of $O(n^2 k^5 (e\Delta)^{2k})$.

It remains only to show that $\mathcal{T}_k$ contains all the sets we desire. Clearly $\mathcal{T}_{k-1} \subset \mathcal{T}_k$ and assume by induction that $\mathcal{T}_{k-1}$ contains all $T \subset V$ of size $k-1$ with $G[T]$ connected. Given $S \subseteq V$ such that $|S| = k$ and $G[S]$ is connected, take any tree of $G[S]$, remove a leaf v and call the resulting set of vertices S'. Then it is clear that $S' \in \mathcal{T}_{k-1}$ and this implies $S = S' \cup \{v\} \in \mathcal{T}_k$. $\qed$

We call a graph invariant $f : \mathcal{G} \to \mathbb{C}$ *additive* if for each $G_1, G_2 \in \mathcal{G}$ we have $f(G_1 \cup G_2) = f(G_1) + f(G_2)$. The following lemma is a variation of a lemma due to Csikvári and Frenkel [17]; it is fundamental to our approach.

Lemma 3.5. *Let $f : \mathcal{G} \to \mathbb{C}$ be a graph invariant given by $f(\cdot) := \sum_{H \in \mathcal{G}} a_H \mathrm{ind}(H, \cdot)$ (where only finitely many of the a_H are nonzero). Then f is additive if and only if $a_H = 0$ for all graphs H that are disconnected.*

Proof. Let H be connected. Then for $G_1, G_2 \in \mathcal{G}$ we have $\mathrm{ind}(H, G_1 \cup G_2) = \mathrm{ind}(H, G_1) + \mathrm{ind}(H, G_2)$, as H is connected. Thus $\mathrm{ind}(H, \cdot)$ is additive. Clearly, linear combinations of additive graph parameters are again additive. This implies that if f is supported on connected graphs, then f is additive.

Suppose next that f is additive. We need to show that $a_H = 0$ if H is disconnected. By the previous part of the proof, we may assume that $a_H = 0$ for all connected graphs H

10

H. Let now $H = H_1 \cup H_2$ with both H_1 and H_2 nonempty. We may assume by induction that for all graphs H' of order strictly smaller than $k := |V(H)|$ we have $a_{H'} = 0$. Now, by additivity we have

$$f(H) = f(H_1) + f(H_2) = \sum_{H':|V(H')| \geq k} a_{H'}(\text{ind}(H', H_1) + \text{ind}(H', H_2)) = 0,$$

since $|V(H_i)| < k$ for $i = 1, 2$. On the other hand we have

$$f(H) = \sum_{H':|V(H')| \geq k} a_{H'}\text{ind}(H', H) = a_H\text{ind}(H, H).$$

As $\text{ind}(H, H) \neq 0$, this implies that $a_H = 0$ and finishes the proof. $\qquad\square$

3.2 Proof of Theorem 3.1

Recall that $p(\cdot)$ is a bounded induced graph counting polynomial (BIGCP). Given an n-vertex graph G with maximum degree at most Δ, we must show how to compute the first m inverse power sums $p_1, \ldots, p_m$ of $p(G)$ in time $(n/\varepsilon)^{O(1)}$, where $m = C \ln(n/\varepsilon)$. To reduce notation, let us write $p = p(G)$, $d = d(G)$ for the degree of p, and $e_i = e_i(G)$ for $i = 0, \ldots, d$ for the coefficients of p (from (10)). Recall that $p_k := \zeta_1^{-k} + \cdots + \zeta_d^{-k}$, where $\zeta_1, \ldots, \zeta_d \in \mathbb{C}$ are the roots of the polynomial $p(G)$. Noting $e_0 = 1$, Proposition 2.1 gives

$$p_k = -ke_k - \sum_{i=1}^{k-1} e_i p_{k-i}, \tag{13}$$

for each $k = 1, \ldots, d$. By (11), for $i \geq 1$, the e_i can be expressed as linear combinations of induced subgraph counts of graphs with at most $C_1 i$ vertices. Since $p_1 = -e_1$, this implies that the same holds for p_1. By induction, (12), and (13) we have that for each k

$$p_k = \sum_{H \in \mathcal{G}_{C_1 k}} a_{H,k}\text{ind}(H, G), \tag{14}$$

for certain, yet unknown, coefficients $a_{H,k}$.

Since p is multiplicative, the inverse power sums are additive. Thus Lemma 3.5 implies that $a_{H,k} = 0$ if H is not connected. Denote by $\mathcal{C}_i(G)$ the set of connected graphs of order at most i that occur as induced subgraphs in G. This way we can rewrite (14) as follows:

$$p_k = \sum_{H \in \mathcal{C}_{C_1 k}(G)} a_{H,k}\text{ind}(H, G). \tag{15}$$

The next lemma says that we can compute the coefficients $a_{H,k}$ efficiently for $k = 1, \ldots, m$, where $m = C \ln(n/\varepsilon)$.

Lemma 3.6. *There is an $(n/\varepsilon)^{O(1)}$-time algorithm, which given a BIGCP p and an n-vertex graph G, computes and lists the coefficients $a_{H,k}$ in (15) for all $H \in \mathcal{C}_{C_1 k}(G)$ and all $k = 1, \ldots, m = C \ln(n/\varepsilon)$.*

Proof. Using the algorithm of Lemma 3.4, we first compute the sets $\mathcal{T}_{C_1 k}$ consisting of all subsets S of $V(G)$ such that $|S| \leq C_1 k$ and $G[S]$ is connected, for $k = 1 \ldots, m$. This takes time bounded by $(n/\varepsilon)^{O(1)}$. (Note that the algorithm in Lemma 3.4 computes and lists all the sets $\mathcal{T}_i$ for $i = 1, \ldots, C_1 m$.) We next compute and list the graphs in $\mathcal{C}_{C_1 k}(G)$ by considering the set of graphs $\{G[S] \mid S \in \mathcal{T}_{C_1 k}\}$ and removing copies of isomorphic graphs using Lemma 3.2 (i) to test for isomorphism. This takes time at most $(n/\varepsilon)^{O(1)}$ for each k, so the total time to compute and list the $\mathcal{C}_{C_1 k}(G)$ is bounded by $(n/\varepsilon)^{O(1)}$.

To prove the lemma, let us fix $k \leq m$ and show how to compute the coefficients $a_{H,k}$, assuming that we have already computed and listed the coefficients $a_{H,k'}$ for all $k' < k$. Let us fix $H \in \mathcal{C}_{C_1 k}(G)$. By (13), it suffices to compute the coefficient of $\mathrm{ind}(H, \cdot)$ in $p_{k-i} e_i$ for $i = 1, \ldots, k$ (where we set $p_0 = 1$). By (11), (12) and (14) we know that the coefficient of $\mathrm{ind}(H, \cdot)$ in $p_{k-i} e_i$ is given by

$$\sum_{H_1, H_2} c^{H}_{H_1, H_2} a_{H_2, (k-i)} \lambda_{H_1, i} = \sum_{(S,T): S \cup T = V(H)} a_{H[T], (k-i)} \lambda_{H[S], i}. \tag{16}$$

As $|V(H)| \leq C_1 k = O(\ln(n/\varepsilon))$, the second sum in (16) is over at most $4^{C_1 k} = (n/\varepsilon)^{O(1)}$ pairs (S, T). For each such pair, we need to compute $\lambda_{H[S], i}$ and look up $a_{H[T], (k-i)}$. We can compute $\lambda_{H[S], i}$ in time bounded by $C_2^{|S|} = (n/\varepsilon)^{O(1)}$ since p is a BIGCP.

Looking up $a_{H[T], (k-i)}$ in the given list requires us to test isomorphism of $H[T]$ with each graph in $\mathcal{C}_{C_1(k-i)}(G)$ (noting that $a_{H[T], (k-i)} = 0$ if $H[T] \notin \mathcal{C}_{C_1(k-i)}(G)$ by Lemma 3.5). Using Lemma 3.2(i) to test for graph isomorphism, this takes time at most

$$O(|\mathcal{C}_{C_1(k-i)}(G)| C_1(k-i) \Delta^{C_1(k-i)-1}) = O(n/\varepsilon)^{O(1)}.$$

Here we use Lemma 3.3 to bound $|\mathcal{C}_{C_1(k-i)}(G)|$. Together, all this implies that the coefficient of $\mathrm{ind}(H, \cdot)$ in $p_{k-i} e_i$ can be computed in time bounded by $(n/\varepsilon)^{O(1)}$, and so the coefficient $a_{H,k}$ can be computed in time $(n/\varepsilon)^{O(1)}$. Thus all coefficients $a_{H,k}$ for $H \in \mathcal{C}_{C_1 k}(G)$ can be computed and listed in time bounded by $|\mathcal{C}_{C_1 k}(G)|(n/\varepsilon)^{O(1)} = (n/\varepsilon)^{O(1)}$. This can be done for each $k = 1, \ldots, m$ in time $(n/\varepsilon)^{O(1)}$. □

To finish the proof of the theorem, we compute p_k for each $k = 1, \ldots, m$ by adding all the numbers $a_{H,k} \mathrm{ind}(H, G)$ over all $H \in \mathcal{C}_{C_1 k}(G)$. This can be done in time

$$O(m |\mathcal{C}_{C_1 m}(G)| n^2 \Delta^{2(C_1 m - 1)}) = (n/\varepsilon)^{O(1)},$$

where we use that computing $\mathrm{ind}(H, G)$ with $H \in \mathcal{C}_{C_1 k}(G)$ takes time $O(n^2 \Delta^{2(C_1 m - 1)})$ by Lemma 3.2(ii).

Remark 3.2. Let us finally remark that the proof of Theorem 3.1 is very robust. For example, it extends easily to colored versions of induced graph counts. We will moreover extend it to other graph-like structures in Section 7. For the sake of exposition we have chosen to present it in the current form.

4 The independence polynomial

4.1 The independence polynomial on bounded degree graphs

Proof of Theorem 1.1. First note that by a result of Shearer [38] and Scott and Sokal [37, Corollary 5.7], we know that $Z(G)(z) \neq 0$ for all all graphs G of maximum degree at most Δ and all $z \in \mathbb{C}$ that satisfy $|z| \leq \lambda^* = \frac{(\Delta-1)^{\Delta-1}}{\Delta^\Delta}$.

Now fix an n-vertex graph G of maximum degree at most Δ. Let $m = C \ln(n/\varepsilon)$, where $C = C(\lambda, \lambda^*)$ is the constant in Corollary 2.3. As the kth coefficient of $Z(G)$ is equal to $\mathrm{ind}(\bullet^k, G)$, where $\bullet^k$ denotes the graph consisting of k isolated vertices and as $Z(G)$ is clearly multiplicative and has constant term equal to 1, we have that $Z(G)$ is a BIGCP (taking $C_1 = C_2 = 1$). So by Theorem 3.1 we see that for $k = 1, \ldots, m$ we can compute the first m inverse power sums of $Z(G)$ in time $(n/\varepsilon)^{O(1)}$. Noting that the degree of $Z(G)$ is at most n, Corollary 2.3 implies we can compute a multiplicative ε-approximation to $Z(G)(\lambda)$ in time $(n/\varepsilon)^{O(1)}$. This concludes the proof. □

Let us briefly mention how our results apply to the multivariate independence polynomial. For a graph $G = (V, Z)$ and a variable z_v for each $v \in V$ define

$$Z(G)((z_v)_{v \in V}) = \sum_{\substack{I \subset V \\ \text{independent}}} \prod_{v \in I} z_v.$$

Now define a multivariate polynomial q by $q(\lambda) = Z(G)((\lambda z_v)_{v \in V})$. Then the coefficient of λ^k of q can be expressed as the sum over vertex colored induced graph counts on at most k vertices (using the vertices of G as colors) where the coefficient of a graph H with vertex coloring $c : V(H) \to V(G)$ is equal to $\prod_{u \in V(H)} z_{c(u)}$ if H is an independent set of k vertices and zero otherwise. Then by the method from the previous section we can again compute the low order inverse power sums efficiently on bounded degree graphs, cf. Subsection 7.2. Using that the result of Scott and Sokal [37] also applies to the multivariate independence polynomial, it then follows that we can efficiently approximate $Z(G)((z_v)_{v \in V})$ on bounded degree graphs if all z_v satisfy $|z_v| < \lambda^*$.

Evaluating the independence polynomial at negative and complex values gives us new information about the distribution of independent sets in a graph, as illustrated by the following example. We denote by $Z_e(G)(\lambda)$ the polynomial defined in the same way as the independence polynomial except that in the sum (1), we only allow independent sets whose cardinality is even.

Theorem 4.1. *Let $\Delta \in \mathbb{N}$ and let $0 \le \lambda < \lambda^*(\Delta) := \frac{(\Delta-1)^{\Delta-1}}{\Delta^\Delta}$. Then there exists a deterministic algorithm, which, given a graph $G = (V, E)$ of maximum degree at most Δ and $\varepsilon > 0$, computes a multiplicative ε-approximation to $Z_e(G)(\lambda)$ in time $(|V|/\varepsilon)^{O(1)}$.*

Proof. We apply the algorithm of Theorem 1.1 to compute multiplicative ε-approximations $A(\lambda)$ and $A(-\lambda)$ to $Z(G)(\lambda)$ and $Z(G)(-\lambda)$ respectively in time $(|V|/\varepsilon)^{O(1)}$. We have

$$e^{-\varepsilon} Z(G)(\lambda) \le A(\lambda) \le e^\varepsilon Z(G)(\lambda) \quad \text{and} \quad e^{-\varepsilon} Z(G)(-\lambda) \le A(-\lambda) \le e^\varepsilon Z(G)(-\lambda).$$

Taking half the sum of these equations and noting that $Z_e(G)(\lambda) = \frac{1}{2}(Z(G)(\lambda) + Z_G(-\lambda))$, we see that $\frac{1}{2}(A(\lambda) + A(-\lambda))$ is a multiplicative ε-approximation to $Z_e(G)(\lambda)$ provided both $Z(G)(\lambda)$ and $Z(G)(-\lambda)$ have the same sign.

Clearly $Z(G)(\lambda) > 0$ since the coefficients of $Z(G)$ are nonnegative real numbers. Also $Z(G)(-\lambda) > 0$ because we know by the result of Scott and Sokal [37] that $Z(G)$ does not vanish in the interval $[-\lambda^*, \lambda^*]$, and we know $Z(G)$ is positive in the interval $[0, \lambda^*]$ since all the coefficients of $Z(G)$ are nonnegative real numbers. Hence $Z(G)$ is positive on the whole interval $[-\lambda^*, \lambda^*]$ and in particular $Z(G)(\lambda) > 0$. □

4.2 The independence polynomial on claw-free graphs

In this subsection, we illustrate a technique of Barvinok for approximating graph polynomials on larger regions of the complex plane by making careful polynomial transformations. We use this technique to prove Theorem 1.2, which shows that we can approximate the independence polynomial of claw-free graphs on almost the entire complex plane. First we require a few preliminary results.

Proposition 4.2. *If G is a claw-free graph of maximum degree Δ and ζ is a root of the independence polynomial $Z(G)$ of G then $\zeta \in \mathbb{R}$ with $\zeta < -\frac{1}{e(\Delta-1)}$.*

Proof. The fact that $\zeta \in \mathbb{R}$ is a result of Chudnovsky and Seymour [16]. The fact that ζ must be negative follows because all the coefficients of $Z(G)$ are positive. Now the result of Sokal and Scott [37] states that

$$|\zeta| \ge \lambda^*(\Delta) = \frac{(\Delta-1)^{\Delta-1}}{\Delta^\Delta} > \frac{1}{e(\Delta-1)},$$

from which the proposition follows. $\square$

We also require the following lemma of Barvinok [5].

Lemma 4.3. *For $\rho \in (0, 1)$ we define*

$$\alpha = \alpha(\rho) = 1 - \exp(-\rho^{-1}), \qquad \beta = \beta(\rho) = \frac{1 - \exp(-1 - \rho^{-1})}{1 - \exp(-\rho^{-1})} > 1,$$

$$N = N(\rho) = (1 + \rho^{-1}) \exp(1 + \rho^{-1}), \qquad \sigma = \sigma(\rho) = \sum_{i=1}^{N} \frac{\alpha^i}{i}.$$

The polynomial

$$\phi(z) = \phi_\rho(z) = \frac{1}{\sigma} \sum_{i=1}^{N} \frac{(\alpha z)^i}{i}$$

has the following properties:

(i) $\phi(0) = 0$ and $\phi(1) = 1$ and ϕ has degree N;

(ii) *If $z \in \mathbb{C}$ with $|z| \leq \beta$ then $\phi_\rho(z) \in S_\rho$, where*

$$S_\rho := \{z \in \mathbb{C} \mid -\rho \leq Re(z) \leq 1 + 2\rho \quad and \quad -2\rho \leq Im(z) \leq 2\rho\}.$$

Proposition 4.4. *Fix $\lambda = re^{i\theta} \in \mathbb{C}$ with $\theta \in (-\pi, \pi)$. Let S_ρ be as in the previous lemma, and let $\mathbb{R}^-$ denote the negative real line. Then*

$$\lambda S_\rho \cap \mathbb{R}^- \subseteq \begin{cases} [-2\rho r, 0] & \text{if } \theta \in [-\frac{\pi}{2}, \frac{\pi}{2}]; \\ [-2\rho r / |sin\theta|, 0] & \text{otherwise.} \end{cases}$$

Proof. S_ρ is a bounded strip parallel to the real axis in the complex plane, so λS_ρ is the same strip enlarged by a factor r and rotated by an angle θ. The proposition then follows from elementary trigonometry. $\square$

Proof of Theorem 1.2. Recall that we are given a claw-free graph G of maximum degree Δ and $\lambda \in \mathbb{C}$ that is not a negative real number and we wish to find a multiplicative ε-approximation to $Z(G)(\lambda)$.

Set $n := |V(G)|$ and let $\lambda = re^{i\theta}$ with $\theta \in (-\pi, \pi)$. Set

$$\rho = \begin{cases} 1/6r(\Delta - 1) & \text{if } \theta \in [-\frac{\pi}{2}, \frac{\pi}{2}]; \\ |\sin\theta|/6r(\Delta - 1) & \text{otherwise,} \end{cases}$$

and consider the polynomial $g(z) = Z(G)(\lambda\phi_\rho(z))$. Note that the degree of g is $O(n)$ since the degree of $Z(G)$ is at most n and the degree of ϕ_ρ is a constant $N(\rho)$.

We will use Corollary 2.3 to find a multiplicative ε-approximation to $g(1) = Z(G)(\lambda)$ in time $(n/\varepsilon)^{O(1)}$. In order to apply Corollary 2.3 to draw this conclusion, it is enough to check that (i) g has no roots in the disk $|z| \leq \beta := \beta(\rho)$ and that (ii) the first $m = C\ln(d/\varepsilon)$ inverse power sums of g can be computed in time $(n/\varepsilon)^{O(1)}$, where $d = O(n)$ is the degree of g and $C = C(\beta, 1)$ is the constant in the statement of Corollary 2.3.

It remains to check (i) and (ii). To see (i), note first that by Lemma 4.3, ϕ_ρ maps the disk $D = \{z \in \mathbb{C} \mid |z| \leq \beta\}$ into S_ρ. By Proposition 4.4 and our choice of ρ, we have $\lambda\phi_\rho(D) \cap \mathbb{R}^- \subseteq (-\frac{1}{3(\Delta-1)}, 0]$. By Proposition 4.2 we know that if $Z(G)(\zeta) = 0$ then $\zeta \in \mathbb{R}$ with $\zeta < -\frac{1}{e(\Delta-1)}$. In particular this implies $g(\cdot) = Z(G)(\lambda\phi_\rho(\cdot))$ has no root in the disk D.

For (ii), we show that we can compute the first m coefficients of g in time $(n/\varepsilon)^{O(1)}$, which is sufficient by Proposition 2.1. Given a polynomial $p(z) = \sum_{i=0}^{d} a_i z^i$, write

$p_{[m]}(z) := \sum_{i=0}^{m} a_i z^i$. Then we note that $g_{[m]} = (Z(G) \circ (\lambda\phi))_{[m]} = (Z(G)_{[m]} \circ (\lambda\phi_{[m]}))_{[m]}$, where we crucially use the fact that ϕ has no constant term since $\phi(0) = 0$. In words, to obtain $g_{[m]}(z)$ we substitute $\lambda\phi_{[m]}(z)$ into $Z(G)_{[m]}(z)$ and keep the first m terms. Thus, in $O(m^3)$-time we can obtain the first m coefficients of g if we know the first m coefficients of $Z(G)$. As $Z(G)$ is a BIGCP, we can compute its first m inverse power sums in time $(n/\varepsilon)^{O(1)}$ (as in the proof of Theorem 1.1), from which we can find its first m coefficients in time $O(m^2)$ by Proposition 2.1. This finishes the proof. $\qquad\square$

We remark that, for the $(n/\varepsilon)^{O(1)}$ running time in the algorithm above, the $O(1)$ in the exponent depends on λ and grows exponentially fast in $r = |\lambda|$. However, this dependence can be brought down to $O(|\lambda|^{1/2})$ by adapting Lemma 4.3 as described by Barvinok [6].

5   The Tutte polynomial

Here we give a proof of Theorem 1.3.

Proof of Theorem 1.3. By a result of Jackson, Procacci and Sokal, cf. [29, Theorem 1.2] (which is valid for loopless multigraphs) we know that there exists a constant $K > 0$ depending on Δ and w such that for all q with $|q| > K$ we have $Z_T(G)(q, w) \neq 0$ for all graphs G of maximum degree at most Δ. This is exactly opposite to what we need to apply Corollary 2.3, so let us define the graph polynomial p_T by

$$p_T(G)(z) := z^{|V|} Z_T(G)(1/z, w), \tag{17}$$

for any graph $G = (V, E)$. Note that $p_T(G)$ has degree $n := |V|$ and that if x is a multiplicative ε-approximation to $p_T(G)(1/q, w)$, then $q^n x$ is a multiplicative ε-approximation to $Z_T(G)(q, w)$, so it is sufficient to find the former.

We will show that for any n-vertex graph G of maximum degree at most Δ, we can compute the first m inverse power sums of $p_T(G)$ in time $(n/\varepsilon)^{O(1)}$, where $m = C\ln(n/\varepsilon)$ and $C = C(1/q, 1/K)$ is the constant in Corollary 2.3. Corollary 2.3 then implies we can compute a multiplicative ε-approximation to $p_T(G)(1/q)$ and hence to $Z_T(G)(q, w)$ in time $(n/\varepsilon)^{O(1)}$.

We will show that $p_T(G)$ is a BIGCP so that by Theorem 3.1 we can conclude that we can compute the first m inverse power sums in time $(n/\varepsilon)^{O(1)}$.

Since the Tutte polynomial $Z_T(G)(z, w)$ (as a polynomial in z) is a monic and multiplicative graph polynomial (of degree $n = |V(G)|$), we know that the constant term of p_T equals 1 and that p_T is multiplicative. So it suffices to show conditions (i) and (ii) in Definition 3.1. The coefficient of z^k in $p_T(G)$ equals the coefficient of z^{n-k} in $Z_T(G)(z, w)$ and is by definition equal to the sum over all subsets A of E such that A induces a graph with exactly $n - k$ components, where each subset is counted with weight $w^{|A|}$. Let us call a component of a graph *nontrivial* if it consists of more than one vertex. Suppose some subset of the edges $A \subseteq E$ induces $n - k$ components of which c are nontrivial. Then we have $n - k - c$ isolated vertices and so the graph F, consisting of the union of these nontrivial components, has $n - (n - k - c) = k + c$ vertices and $k(F) = c$ components. Thus we have a correspondence between subsets A of E that induce a graph with exactly $n - k$ components and subgraphs F of G with no isolated vertices satisfying $k(F) = V(F) - k$. Therefore the coefficient of z^{n-k} in $Z_T(G)$ can be expressed as

$$\sum_{\substack{F \subseteq G \,:\, \delta(F) \geq 1 \\ |k(F)| = |V(F)| - k}} w^{|E(F)|} = \sum_{H} \sum_{\substack{F \subseteq H \,:\, \delta(F) \geq 1 \\ V(F) = V(H) \\ |k(F)| = |V(F)| - k}} w^{|E(F)|} \mathrm{ind}(H, G). \tag{18}$$

In fact the first sum can be taken over graphs H with at most $2k$ vertices. This is because $V(H) = V(F)$ and

$$|V(F)| = k(F) + k \leq \frac{V(F)|}{2} + k.$$

as F has no isolated vertices.

From (18), we can compute the coefficient of $\text{ind}(H,G)$ by checking all subsets of $E(H)$ in time $O(2^{E(H)}) = O(2^{\Delta|V(H)|})$. This implies that p_T is a BIGCP (taking $C_1 = 2$ and $C_2 = 2^\Delta$). $\qquad\square$

Remark 5.1. Csikvári and Frenkel [17] introduced *graph polynomials of bounded exponential type* and showed that these polynomials have bounded roots on bounded degree graphs. This was utilized in [36] to give quasi-polynomial-time approximation algorithm for evaluations of these polynomials. The Tutte polynomial with the second argument fixed is an example of such a polynomial. We remark here that the proof given above for the Tutte polynomial also easily extends to graph polynomial of bounded exponential type. So the algorithm in [36] can be adapted to run in polynomial time on bounded degree graphs.

6   Partition functions of spin models

In this section we will state and prove a generalization of Theorem 1.4 and we will indicate how our method applies to partition functions of graph homomorphisms with multiplicities.

6.1   Partition functions for edge-colored graphs

Suppose a graph $G = (V, E)$ has an edge-coloring $\psi : E \to [c]$, which need not be a proper coloring. Suppose also that for each $i = 1, \ldots, c$ we have a symmetric $k \times k$-matrix A^i. Let us write $\mathcal{A} = (A^1, \ldots, A^c)$. Then we can extend the definition of the partition function of a spin model as follows:

$$p(G)(\mathcal{A}) = \sum_{\phi: V \to [k]} \prod_{e = \{u,v\} \in E} A^{\psi(e)}_{\phi(u), \phi(v)}. \tag{19}$$

We will refer to $p(G)(\mathcal{A})$ as the *partition function of* $\mathcal{A}$. In [22] this is called a Markov random field (if the A_i are nonnegative) and in [34] this is called a multi spin system. Clearly, if $c = 1$ this just reduces to the partition function of a spin model. We have the following result, which implies Theorem 1.4.

Theorem 6.1. *Let $\Delta, k \in \mathbb{N}$. Then there exists a deterministic algorithm, which, given a c-edge-colored graph $G = (V, E)$ of maximum degree at most Δ, symmetric $k \times k$ matrices $A_1, \ldots, A_c$ such that $|A^s_{i,j} - 1| \leq 0.34/\Delta$ for all $i, j = 1, \ldots, k$ and $s = 1, \ldots, c$, and $\varepsilon > 0$, computes a multiplicative ε-approximation to $p(G)(\mathcal{A})$ in time $(|V|/\varepsilon)^{O(1)}$.*

Remark 6.1. The implicit constant in the big O only depends on Δ and k, cf. Remark 3.1. In particular, the number of colors c does not play a role in the complexity of the algorithm. The constant 0.34 may be replaced by 0.45 if $\Delta \geq 3$ and by 0.54 if Δ is large enough. See [7].

Proof. Let J be the all ones matrix. For $z \in \mathbb{C}$, let $\mathcal{A}'(z) := (J + z(A^1 - J), \ldots, J + z(A^c - J))$. Define a univariate polynomial q by

$$q(G)(z) = k^{-|V|} p(G)(\mathcal{A}'(z)). \tag{20}$$

Then $q(G)(0) = 1$ and $q(G)(1) = k^{-|V|}p(G)(\mathcal{A})$. Barvinok and Soberón [7, Theorem 1.6] showed that there exists a constant $\delta > 0$ such that $q(G)(z) \neq 0$ for all z satisfying $|z| \leq 1 + \delta$.

We will show that for any n-vertex graph G of maximum degree at most Δ, we can compute the first m inverse power sums of $q(G)$ in time $(n/\varepsilon)^{O(1)}$, where $m = C \ln(n/\varepsilon)$ and $C = C(1, 1 + \delta)$ is the constant in Corollary 2.3. Noting that the degree of $q(G)$ is at most $|E| \leq n\Delta/2$, Corollary 2.3 implies we can compute a multiplicative ε-approximation to $q(G)(1)$ in time $(n/\varepsilon)^{O(1)}$. So it remains to show that we can compute the first m inverse power sums of $q(G)$ in time $(n/\varepsilon)^{O(1)}$.

By definition, $q(G)(z)$ satisfies

$$q(G)(z) = k^{-n} \sum_{\phi:V \to [k]} \prod_{e=\{u,v\} \in E} (J + (z(A^{\psi(e)} - J)))_{\phi(u),\phi(v)}$$

$$= k^{-n} \sum_{i=0}^{|E|} z^i \left(\sum_{\substack{F \subseteq E \\ |F|=i}} \sum_{\phi:V \to [k]} \prod_{e=\{u,v\} \in F} (A^{\psi(e)} - J)_{\phi(u),\phi(v)} \right). \tag{21}$$

For a subset F of E, define $G[F]$ to be the edge-colored graph induced by the edges in F. The vertex set of $G[F]$ consists of those vertices incident with edges in F and hence has size at most $2|F|$. For each $\ell \in \mathbb{N}$, let $\mathcal{G}_\ell(c)$ be the collection of graphs on at most ℓ vertices whose edges are colored with colors from $[c]$. For two edge-colored graphs H_1, H_2 with colors from $[c]$ we denote by $\text{ind}_c(H_1, H_2)$ the number of induced graphs of H_2 that are isomorphic as edge-colored graphs to H_1.

Then we see that the coefficient of z^i in (21) can be written as follows:

$$k^{-n} \sum_{\substack{H \in \mathcal{G}_{2i}(c) \\ |E(H)|=i}} k^{n-|V(H)|} \left(\sum_{\phi:V(H) \to [k]} \prod_{e=\{u,v\} \in E(H)} (A^{\psi(e)} - J)_{\phi(u),\phi(v)} \right) \text{ind}_c(H, G)$$

$$= \sum_{\substack{H \in \mathcal{G}_{2i}(c) \\ |E(H)|=i}} \left(k^{-|V(H)|} \sum_{\phi:V(H) \to [k]} \prod_{e=\{u,v\} \in E(H)} (A^{\psi(e)} - J)_{\phi(u),\phi(v)} \right) \text{ind}_c(H, G). \tag{22}$$

If $i \leq m$, the inner sum in (22) can be computed in time $O(k^{2m}) = (n/\varepsilon)^{O(1)}$. Clearly, q has constant term equal to 1 and it is multiplicative. In case $c = 1$, i.e., in case we are dealing with ordinary graphs, this implies that q is a BIGCP (with constant $C_1 = 2$ and $C_2 = k$) and so Theorem 3.1 implies that we can compute the first m inverse power sums of q in time bounded by $O(n/\varepsilon)^{O(1)}$.

In the general case we note that Theorem 3.1 remains valid if ind is replaced by ind_c in Definition 3.1. To see this, it suffices to note that all lemmas and proofs in Section 3 directly carry over to the edge-colored case. This finishes the proof. $\square$

6.2 Partition functions of graph homomorphisms with multiplicities

Let again $G = (V, E)$ be equipped with an edge-coloring $\psi : E \to [c]$, which need not be a proper coloring. Suppose also that for each $i = 1, \ldots, c$ we have a symmetric $k \times k$-matrix A^i. Let us write $\mathcal{A} = (A^1, \ldots, A^c)$. Let $n = |V|$ and let $\mu = (\mu_1, \ldots, \mu_k)$ with $\mu_i \in \mathbb{Z}_{\geq 1}$ for each i be such that that $\sum_{i=1}^k \mu_i = n$. We call such μ a *composition* of n in k parts. Barvinok and Soberón [8] define the *partition function of graph homomorphisms with multiplicities μ* as

$$p_\mu(G)(\mathcal{A}) = \sum_{\substack{\phi:V \to [k] \\ |\phi^{-1}(i)|=\mu_i}} \prod_{e=\{u,v\} \in E} A^{\psi(e)}_{\phi(u),\phi(v)}. \tag{23}$$

We refer to [8] for more details and background on this type of partition function.

Building on a result from Barvinok and Soberón [8, Section 2] and using exactly the same proof as above we directly establish the following:

Theorem 6.2. *Let $\Delta, k \in \mathbb{N}$. Then there exists a deterministic algorithm, which, given an c-edge colored graph $G = (V, E)$ of maximum degree at most Δ, a composition μ of $|V|$ in k parts, symmetric $k \times k$ matrices $A_1, \ldots, A_c$ such that $|A_{i,j}^s - 1| \leq 0.1/\Delta$ for all $i, j = 1, \ldots, k$ and $s = 1, \ldots, c$, and $\varepsilon > 0$, computes an ε-approximation to $p_\mu(G)(\mathcal{A})$ in time $(|V|/\varepsilon)^{O(1)}$.*

7   Partition functions of edge-coloring models

In this section we state and prove a generalization of Theorem 1.5. It is along the same lines as the generalization of Theorem 1.4 in the previous section. The proof also goes along the same line, but as we will see below there are some details that are different.

7.1   Partition functions for vertex-colored graphs

Let $G = (V, E)$ be a graph that is equipped with a vertex coloring $\psi : V \to [c]$ (ψ need not be a proper coloring). Suppose that we have k-color edge-coloring models $h^1, \ldots, h^c$. Let us write $\mathcal{H} = (h^1, \ldots, h^c)$. Often the pair $(G, \{h_1, \ldots, h^c\}, \psi)$ is called a *signature grid*, cf. [14, 15, 13]. Then we can extend the definition of the partition function of an edge-coloring model as follows:

$$p(G)(\mathcal{H}) = \sum_{\phi : E \to k} \prod_{v \in V} h^{\psi(v)}(\phi(\delta(v))). \tag{24}$$

We will refer to $p(G)(\mathcal{H})$ as the *partition function* of $\mathcal{H}$. It is also called the *Holant problem* of the signature grid $(G, \{h_1, \ldots, h^c\}, \psi)$ cf. [14, 15, 13]. We have the following result, which implies Theorem 1.5.

Theorem 7.1. *Let $\Delta, k \in \mathbb{N}$. Then there exists a deterministic algorithm, which, given a c-vertex colored graph $G = (V, E)$ of maximum degree at most Δ, k-color edge-coloring models $h^1, \ldots, h^c$ that satisfy $|h^s(\phi) - 1| \leq 0.35/(\Delta + 1)$ for all $\phi \in \mathbb{N}^k$ and $s = 1, \ldots, c$, and $\varepsilon > 0$, computes a multiplicative ε-approximation to $p(G)(\mathcal{H})$ in time $(|V|/\varepsilon)^{O(1)}$.*

Remark 7.1. Just as for edge-colored graphs, the number of colors c does not play a role in the time complexity in the theorem above. Additionally, the constant 0.35 may be replaced by 0.47 if $\Delta \geq 3$ and by 0.56 if Δ is large enough; see [36]. Moreover, for readers familiar with the orthogonal group invariance of these partition functions one can use Corollary 6b from [36] to find a larger family of edge-coloring models for which the partition function can be efficiently approximated.

Proof. Let J denote the constant ones function $J : \mathbb{N}^k \to \mathbb{C}$ (defined by $J(\phi) = 1$ for all $\phi \in \mathbb{N}^k$). Let for $z \in \mathbb{C}$, $\mathcal{H}(z) := (J + z(h^1 - J), \ldots, J + z(h^c - J))$. Consider the following univariate polynomial:

$$q(G)(z) := k^{-|E|} p(G)(\mathcal{H}(z)). \tag{25}$$

Observe that $q(G)(1) = k^{-|E|} p(G)(\mathcal{H})$ and that $q(G)$ is a polynomial of degree at most $n := |V|$. So, just as in the previous section, the problem of approximating the partition function $p(G)(\mathcal{H})$ is replaced by approximating an evaluation of a univariate polynomial.

By Corollary 6a from [36] (which is valid for multigraphs) there exists $\delta > 0$ such that $q(G)(z) \neq 0$ for all $z \leq 1 + \delta$. We will show (in Theorem 7.2) that for any n-vertex graph G of maximum degree at most Δ, we can compute the first m inverse power sums

of $q(G)$ in time $(n/\varepsilon)^{O(1)}$, where $m = C\ln(n/\varepsilon)$ and $C = C(1, 1+\delta)$ is the constant in Corollary 2.3. Noting that the degree of $q(G)$ is at most n, Corollary 2.3 implies we can compute a multiplicative ε-approximation to $q(G)(1)$ in time $(n/\varepsilon)^{O(1)}$.

Ideally we would like to do this using Theorem 3.1 just as in the proof of Theorem 6.1. Since partition functions of edge-coloring models are multiplicative, the polynomial q is also multiplicative and it has constant term equal to 1. So to be able to apply Theorem 3.1 we need only check that the coefficients of q can be expressed as linear combinations of induced graph counts. This is in fact proved in [36] for $c = 1$, but in that expression it is not clear whether the coefficients $\lambda_{H,i}$ in (11) can be computed efficiently. So instead of directly applying Theorem 3.1 we will have to do a little more work, which we postpone to the next section. $\qquad\square$

7.2 Computing coefficients of $q(G)(z)$

By definition,

$$q(G)(z) = k^{-|E|} \sum_{\phi:E\to[k]} \prod_{v\in V} (J + z(h^{\psi(v)} - J))(\phi(\delta(v)))$$

$$= k^{-|E|} \sum_{i=0}^{n} z^i \left(\sum_{\substack{U\subseteq V \\ |U|=i}} \sum_{\phi:E\to[k]} \prod_{u\in U} (h^{\psi(u)} - J)(\phi(\delta(u))) \right). \tag{26}$$

We need the concept of a *fragment*, which is a pair (H, κ), where H is a *c*-vertex colored graph and where κ is a map $\kappa : V(H) \to \{0, 1, \dots, \Delta\}$. We think of $\kappa(u)$ as a number of *half edges* incident with u. Note that the graph G itself can be thought of as a fragment by taking the map $\kappa : V(G) \to \{0, \dots, \Delta\}$ to be $\kappa(v) = 0$ for all $v \in V(G)$.

For $U \subseteq V$ we let $G(U)$ be the fragment $(G[U], \kappa)$ where $\kappa(u)$ is equal to the number of edges that connect u with $V \setminus U$. Clearly, for each U of size i the second sum on the right in (26) only depends on the isomorphism class of the fragment $G(U)$. (An *isomorphism* from a fragment (H, κ) to a fragment (H', κ') is an isomorphism α of the underlying graphs that preserves vertex colors and such that for each $u \in V(H)$, $\kappa(u) = \kappa'(\alpha(u))$.) For a fragment $F = (H, \kappa)$ let $E(F)$ denote the set of edges of F including half edges and let $V(F)$ denote the vertex set of the underlying graph H. Then define,

$$p(F)(\mathcal{H}) := \sum_{\phi:E(F)\to[k]} \prod_{v\in V(F)} h^{\psi(v)}(\phi(\delta(v))). \tag{27}$$

Here we implicitly assume that the $\psi : V(F) \to [c]$ is the same as the vertex colouring $\psi : V(H) \to [c]$ of H. Define for a fragment $F = (H, \kappa)$, $\mathrm{ind}^*(F, G)$ to be the number of sets U of size $|V(F)|$ such that $G(U)$ is isomorphic to F. Writing $\mathcal{H} - J = (h^1 - J, \dots, h^c - J)$, we can rewrite (26) as

$$q(G)(z) = k^{-|E|} \sum_{i=0}^{n} z^i \left(\sum_{\substack{F=(H,\kappa) \\ |V(H)|=i}} k^{|E|-|E(F)|} p(F)(\mathcal{H} - J)\mathrm{ind}^*(F, G) \right)$$

$$= \sum_{i=0}^{n} z^i \left(\sum_{\substack{F=(H,\kappa) \\ |V(H)|=i}} k^{-|E(F)|} p(F)(\mathcal{H} - J)\mathrm{ind}^*(F, G) \right), \tag{28}$$

where the sum runs over fragments. Let us denote the coefficient of z^i in (28) by e_i. In [36] it is proved that in case $c = 1$, $\mathrm{ind}^*(F, G)$ can be expressed as a linear combination of the parameters $\mathrm{ind}(H, G)$ for certain graphs H. As mentioned above, the coefficients in this expression may not be easy to compute (at least we do not know how to do this). So will have to work with the parameters $\mathrm{ind}^*(F, \cdot)$ instead. This is not a severe

problem, since essentially if we replace ind in (11) by ind*, then Theorem 3.1 remains valid. Indeed, we have the following theorem.

Theorem 7.2. *Let $C > 0$ and $\Delta \in \mathbb{N}$. Then there is a deterministic $(n/\varepsilon)^{O(1)}$-time algorithm, which, given any n-vertex graph G of maximum degree at most Δ and any $\varepsilon > 0$, computes the inverse power sums $p_1, \dots, p_m$ of $q(G)$ for $m = C\ln(n/\varepsilon)$.*

The proof of Theorem 7.2 follows the same line as the proof of Theorem 3.1. Essentially we need to replace graphs by fragments in the proof and check that everything remains valid. For completeness we will give the proof.

We first need to note that for a fragment $F_1 = (H_1, \kappa_1)$ the graph parameter ind$^*(F_1, \cdot)$ can be extended to the collection of all fragments as follows: for a fragment $F_2 = (H_2, \kappa_2)$ we let ind$^*(F_1, F_2)$ denote the number of embeddings of H_1 as induced subgraph in H_2, preserving vertex colors, such that for each vertex v of H_1 we have that the number of neighbours of v in $V(H_2) \setminus V(H_1)$ is equal to $\kappa_1(v) - \kappa_2(v)$. Then for two fragments F_1 and F_2 we have

$$\text{ind}^*(F_1, \cdot) \cdot \text{ind}^*(F_2, \cdot) = \sum_F c_{F_1, F_2}^F \text{ind}^*(F, \cdot), \tag{29}$$

where the sum runs over all fragments F and where for a fragment F, c_{F_1, F_2}^F denotes the number of pairs of subsets (S, T) of $V(F)$ such that $S \cup T = V(F)$ and $F_1 = F(S)$ and $F_2 = F(T)$. (Here $F(S)$ is the fragment induced by S, i.e., if $F = (H, \kappa)$, then $F(S) = (H[S], \alpha)$ where for $s \in S$ we set $\alpha(s) = \deg_H(s) - \deg_{H[S]}(s) + \kappa(s)$.) We call a fragment $F = (H, \kappa)$ *connected* if the graph H is connected. We now adapt some of the statements and proofs of the results in Section 3 to include fragments.

We start with some definitions. By $\mathcal{F}$ we denote the collection of all fragments and by $\mathcal{F}_k$ for $k \in \mathbb{N}$ we denote the collection of fragments with at most k vertices. (Recall that we implicitly assume that the vertices of our fragments are colored with the colors $1, \dots, c$.) For two fragments $F_1 = (H_1, \kappa_1)$ and $F_2 = (H_2, \kappa_2)$, $F_1 \cup F_2 := (H, \kappa)$, where $H = H_1 \cup H_2$ and $\kappa : V(H_1 \cup V(H_2) \to \{0, 1, \dots, \Delta\}$ is the map whose restriction to $V(H_1)$ is κ_1 and whose restriction to $V(H_2)$ is κ_2. An *invariant of fragments* is a function $f : \mathcal{F} \to S$ for some set S that takes the same value on isomorphic fragments. Call an invariant of fragments f *multiplicative* if $f(\varnothing) = 1$ and $f(F_1 \cup F_2) = f(F_1)f(F_2)$ for all fragments F_1, F_2. The maximum degree of a fragment $F = (H, \kappa)$ is equal to the maximum of $\deg(v) + \kappa(v)$ over $v \in V(G)$.

Lemma 7.3. *Let $F = (H, \kappa)$ be a connected fragment on k vertices and let $\Delta \in \mathbb{N}$. Then there is an $O(n^2\Delta^{2(k-1)})$-time algorithm, which, given any n-vertex fragment $\hat{F}$ with maximum degree at most Δ, computes the number ind$^*(F, \hat{F})$.*

Note that Lemma 7.3 enables us to test for isomorphism of fragments between bounded degree fragments when $|V(F)| = |V(\hat{F})|$.

Proof. This follows immediately from the proof of Lemma 3.2. We apply the proof of Lemma 3.2 to the underlying graphs and then remove any potential embedding that either violates the vertex coloring constraints or the constraints that κ imposes. $\square$

We call an invariant of fragment $f : \mathcal{F} \to \mathbb{C}$ *additive* if for each $F_1, F_2 \in \mathcal{F}$ we have $f(F_1 \cup F_2) = f(F_1) + f(F_2)$. The following variation of a lemma due to Csikvári and Frenkel [17] has exactly the same proof as Lemma 3.5; one just needs to replace graph by fragment everywhere in the proof.

Lemma 7.4. *Let $f : \mathcal{F} \to \mathbb{C}$ be an invariant of fragments given by $f(\cdot) := \sum_{F \in \mathcal{F}} a_F \text{ind}^*(F, \cdot)$ (where only finitely many of the a_F are nonzero). Then f is additive if and only if $a_F = 0$ for all fragments F that are disconnected.*

We now sketch the proof of Theorem 7.2.

7.2.1 Proof of Theorem 7.2

Let $\zeta_1, \ldots, \zeta_d \in \mathbb{C}$ be the roots of the polynomial $q(G)$ and recall that for $\ell \in \mathbb{N}$, p_ℓ is the ℓth inverse power sum of the ζ_i. Here d denotes the degree of $q(G) = \sum_{i=0}^{d} e_i z^i$, which is at most n. By (28), for $i \geq 1$, the e_i can be expressed as linear combinations of induced fragments counts of fragments with at most ℓ vertices. Since $e_1 = -p_1$, this implies that the same holds for p_1. By induction, (29) and (13) (using that $e_0 = 1$) we have that for each ℓ

$$p_\ell = \sum_{F \in \mathcal{F}_\ell} a_{F,\ell} \mathrm{ind}^*(F, G), \tag{30}$$

for certain, yet unknown, coefficients $a_{F,\ell}$.

Since $\hat{q}$ is multiplicative, the power sums are additive. Thus Lemma 7.4 implies that $a_{F,\ell} = 0$ if F is not connected. Denote by $\mathcal{C}'_\ell(G)$ the set of connected fragments F of order at most ℓ such that $\mathrm{ind}^*(F, G) \neq 0$. This way we can rewrite (30) as follows:

$$p_\ell = \sum_{F \in \mathcal{C}'_\ell(G)} a_{F,\ell} \mathrm{ind}^*(F, G). \tag{31}$$

The next lemma says that we can compute the coefficients $a_{F,\ell}$ efficiently for $\ell = 1, \ldots, m$, where $m = C \ln(n/\varepsilon)$.

Lemma 7.5. *There is an $O(n/\varepsilon)^{O(1)}$-time algorithm, which given an n-vertex graph G and $\varepsilon > 0$, computes and lists the coefficients $a_{F,\ell}$ in (31) for all $F \in \mathcal{C}'_\ell(G)$ and all $\ell = 1, \ldots, m = C \ln(n/\varepsilon)$.*

Proof. Using the algorithm of Lemma 3.4, we first compute the sets $\mathcal{T}_\ell$ consisting of all subsets S of $V(G)$ such that $|S| \leq \ell$ and $G[S]$ is connected, for $\ell = 1 \ldots, m$. This takes time bounded by $(n/\varepsilon)^{O(1)}$. We next compute and list the fragments in $\mathcal{C}'_\ell(G)$ by considering the set of fragments $\{G(S) \mid S \in \mathcal{T}_\ell\}$ and removing copies of isomorphic fragments using Lemma 7.3 to test for isomorphism. This takes time at most $(n/\varepsilon)^{O(1)}$ for each ℓ, so the total time to compute and list the $\mathcal{C}'_\ell(G)$ is bounded by $(n/\varepsilon)^{O(1)}$.

To prove the lemma, let us fix $\ell \leq m$ and show how to compute the coefficients $a_{F,\ell}$, assuming that we have already computed and listed the coefficients $a_{F,\ell'}$ for all $\ell' < \ell$. Let us fix $F \in \mathcal{C}'_\ell(G)$. By the Newton identities (13), it suffices to compute the coefficient of $\mathrm{ind}^*(F, \cdot)$ in $p_{\ell-i} e_i$ for $i = 1, \ldots, \ell$ (where we set $p_0 = 1$). By (28), (29) and (30) we know that the coefficient of $(-1)^i \mathrm{ind}(F, \cdot)$ in $p_{\ell-i} e_i$ is given by

$$\sum_{\substack{F_1 \in \mathcal{F}_i, F_2 \in \mathcal{F}_{\ell-i} \\ |V(F_1)| = i}} c^F_{F_1, F_2} a_{F_2, (\ell-i)} \frac{p(F_1)(J - \mathcal{H})}{k^{|E(F_1)|}} = \sum_{\substack{S, T \subseteq V(F) \\ S \cup T = V(F) \\ |S| = i, |T| \leq \ell - i}} a_{F(T), (\ell-i)} \frac{p(F(S))(J - \mathcal{H})}{k^{|E(F(S))|}}. \tag{32}$$

For each such pair (S, T), we need to compute $\frac{p(F(S))(J - \mathcal{H})}{k^{|E(F(S))|}}$ and look up $a_{F(T), (\ell-i)}$. We can compute $\frac{p(F(S))(J - \mathcal{H})}{k^{|E(F(S))|}}$ in time bounded by $O(k^{\Delta \ell}) = (n/\varepsilon)^{O(1)}$.

Looking up $a_{F(T), (\ell-i)}$ in the given list requires us to test isomorphism of $F(T)$ with each fragment in $\mathcal{C}'_{\ell-i}(G)$ (noting that $a_{F(T), (\ell-i)} = 0$ if $F(T) \notin \mathcal{C}'_{(\ell-i)}(G)$ by Lemma 7.4). Using Lemma 7.3 to test for isomorphism, this takes time at most

$$O(|\mathcal{C}'_{(\ell-i)}(G)|(\ell-i)^2 \Delta^{2(\ell-i-1)}) = O(n/\varepsilon)^{O(1)}.$$

Here we use Lemma 3.3 to bound $|\mathcal{C}'_{(\ell-i)}(G)| \leq |\mathcal{T}_{(\ell-i)}(G)|$. Together, all this implies that the coefficient of $\mathrm{ind}^*(F, \cdot)$ in $p_{\ell-i} e_i$ can be computed in time bounded by $(n/\varepsilon)^{O(1)}$, and so the coefficient $a_{F,\ell}$ can be computed in time $(n/\varepsilon)^{O(1)}$. Thus all coefficients $a_{F,\ell}$ for $F \in \mathcal{C}'_\ell(G)$ can be computed and listed in time bounded by $|\mathcal{C}'_\ell(G)|(n/\varepsilon)^{O(1)} = (n/\varepsilon)^{O(1)}$. This can be done for each $\ell = 1, \ldots, m$ in time $(n/\varepsilon)^{O(1)}$. $\square$

To finish the proof of the theorem, we compute p_ℓ for each $\ell = 1, \ldots, m$ by adding all the numbers $a_{F,\ell} \text{ind}^*(F, G)$ over all $F \in \mathcal{C'}_\ell(G)$. This can be done in time

$$O(m|\mathcal{C'}_m(G)|n^2\Delta^{2(m-1)}) = (n/\varepsilon)^{O(1)},$$

where we have used that computing $\text{ind}^*(F, G)$ with $F \in \mathcal{C'}_\ell(G)$ takes time bounded by $O(n^2\Delta^{2(m-1)})$ by Lemma 7.3. This finishes the proof.

8 Concluding remarks and open questions

In this paper we have presented a direct connection between the absence of complex roots for a large class of graph polynomials (BIGCPs) and the existence of (deterministic) algorithms to efficiently approximate evaluations of these polynomials. We have illustrated its use by giving deterministic polynomial-time approximation algorithms for evaluations of the Tutte polynomial, the independence polynomial and graph polynomials obtained from spin and edge-coloring models at complex numbers on bounded degree graphs.

This connection naturally leads to the question of how hard it is to approximate evaluations of these graph polynomials close to (complex) roots. Of course this question is rather vague. So let us formulate a more concrete question.

Question 8.1. Recall the constant $\lambda^*(\Delta) = \frac{(\Delta-1)^{\Delta-1}}{\Delta^\Delta}$ for $\Delta \in \mathbb{N}$. What is the complexity of approximating $Z_G(-\lambda)$ for $\lambda > \lambda^*(\Delta)$ for graphs G of maximum degree at most Δ?

We note that Harvey, Srivastava and Vondrák [27] have already made some progress on this question. But the interesting case, when λ is close to $\lambda^*(\Delta)$, is, as far as we know, still open.

As is noted in the introduction our result for the independence polynomial at positive λ does not allow us to efficiently approximate the independence polynomial at λ for $\lambda^* \leq \lambda < \lambda_c$. (This can be done with the correlation decay approach cf. Weitz [46].) But using the approach in Section 4.2 it would follow from a positive answer to the following question, which is a restatement of a version of a conjecture of Sokal [41].

Question 8.2. Let $\varepsilon > 0$ and $\Delta \in \mathbb{N}$. Does there exists $\delta > 0$ (possibly depending on Δ) such that if $\lambda \in \mathbb{C}$ satisfies

$$|\Im(\lambda)| \leq \delta, \text{ and } 0 \leq \Re(\lambda) \leq (1-\varepsilon)\frac{(\Delta-1)^{\Delta-1}}{(\Delta-2)^\Delta}, \tag{33}$$

then $Z_G(\lambda) \neq 0$ for all graphs G of maximum degree at most Δ.

We also iterate a another conjecture of Sokal [28, Conjecture 21], which, if true, would by the methods of the present paper imply that we have an efficient algorithm for approximately counting the number of $(\Delta+1)$-colorings in any graph of maximum degree at most Δ.

Question 8.3. Let $\Delta \in \mathbb{N}$. Is it true that $Z_T(G)(-1, q) \neq 0$ for any $q \in \mathbb{C}$ with $\Re(q) > \Delta$ and any graph G of maximum degree at most Δ?

Another question that arises naturally is the following. Barvinok [2, 5] found quasi-polynomial-time approximation algorithms for computing the permanent of certain matrices, based on absence of zeros. Our method for computing inverse power sums of BIGCPs on bounded degree graphs presented in Section 3 does not seem to apply to permanents. It would be very interesting to find a more general method that also applies to permanents.

Acknowledgements

References

[1] M. Bayati, D. Gamarnik, D. Katz, C. Nair and P. Tetali, Simple deterministic approximation algorithms for counting matchings. In *Proceedings of the thirty-ninth annual ACM symposium on Theory of computing* (pp. 122–127), ACM, 2007.

[2] A. Barvinok, Computing the permanent of (some) complex matrices, *Foundations of Computational Mathematics* (2014) 1–14.

[3] A. Barvinok, Computing the partition function for cliques in a graph, *Theory of Computing* **11** (2015), Article 13 pp. 339–355

[4] A. Barvinok, Computing the partition function of a polynomial on the Boolean cube, arXiv preprint, *arXiv:1503.07463* (2015).

[5] A. Barvinok, Approximating permanents and hafnians, ArXiv preprint, *arXiv:1601.07518* (2016).

[6] A. Barvinok, Personal communication (2016).

[7] A. Barvinok and P. Soberón, Computing the partition function for graph homomorphisms, arXiv preprint, *arXiv:1406.1771* (2014). To appear in *Combinatorica*, doi: 10.1007/s00493-016-3357-2.

[8] A. Barvinok and P. Soberón, Computing the partition function for graph homomorphisms with multiplicities, *Journal of Combinatorial Theory, Series A* **137** (2016) 1–26.

[9] C. Borgs, J. Chayes, J. Kahn and L. Lovász, Left and right convergence of graphs with bounded degree, *Random Structures and Algorithms* **42** (2013) 1–28.

[10] R. Bubley, M. Dyer, C. Greenhill and M. Jerrum: On approximately counting colorings of small degree graphs, *SIAM Journal on Computing* **29** (1999) 387–400.

[11] A. Bulatov and M. Grohe, The complexity of partition functions, *Theoretical Computer Science* **348** (2005) 148–186.

[12] J. Cai, X. Chen and P. Lu, Graph homomorphisms with complex values: A dichotomy theorem, *SIAM Journal on Computing* **42** (2013) 924–1029.

[13] J. Cai, H. Guo and T. Williams, A complete dichotomy rises from the capture of vanishing signatures, In: *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pp. 635–644. ACM, 2013.

[14] J. Cai, S. Huang and P. Lu, From Holant to #CSP and Back: Dichotomy for Holant^c Problems. In ISAAC, pages 253–265, 2010.

[15] J. Cai, P. Lu and M. Xia, Computational complexity of Holant problems, *SIAM Journal on Computing* **40** (2011) 1101–1132.

[16] M. Chudnovsky and P. Seymour, The roots of the independence polynomial of a clawfree graph, *Journal of Combinatorial Theory, Series B* **97** (2007) 350–357.

[17] P. Csikvári and P. E. Frenkel, Benjamini–Schramm continuity of root moments of graph polynomials, *European Journal of Combinatorics* **52** (2016) 302–320.

[18] P. de la Harpe, and V.F.R. Jones, Graph invariants related to statistical mechanical models: examples and problems, *Journal of Combinatorial Theory*, Series B **57** (1993) 207–227.

[19] M. Dyer and C. Greenhill, On Markov chains for independent sets, *Journal of Algorithms* **35** (2000) 17–49.

[20] M. Dyer and C. Greenhill, The complexity of counting graph homomorphisms, *Random Structures and Algorithms* **17** (2000) 260–289.

[21] A. Galanis, D. Stefankovic, E. Vigoda and L. Yang: Ferromagnetic Potts model, Refined #BIS-hardness and related results. In RANDOM 2014, LNCS 6845, pp. 677–691 2014. Full version available at `http://arxiv.orgabs/1311.4839`.

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

[23] D. Garijo, A. Goodall and J. Nešetřil, Jaroslav, On the number of *B*-flows of a graph, *European Journal of Combinatorics* **35** (2014) 273–285.

[24] L.A. Goldberg and H. Guo, The complexity of approximating complex-valued Ising and Tutte partition functions, arXiv preprint arXiv:1409.5627 (2014).

[25] L.A. Goldberg and M. Jerrum, Approximating the partition function of the ferromagnetic Potts model, *Journal of the ACM* **59** (2012) 1–25.

[26] L.A. Goldberg, and M. Jerrum, The complexity of computing the sign of the Tutte polynomial (and consequent #P-hardness of approximation) In *Automata, Languages, and Programming*, pp. 399–410, Springer Berlin Heidelberg, 2012.

[27] N.J. Harvey, P. Srivastava and J. Vondrák, Computing the independence polynomial in Shearer's region for the LLL, arXiv preprint, arXiv:1608.02282, 2016.

[28] B. Jackson, Zeros of chromatic and flow polynomials of graphs, Journal of Geometry **76** (2003) 76–95.

[29] B. Jackson, A. Procacci and A.D. Sokal, Complex zero-free regions at large $|q|$ for multivariate Tutte polynomials (alias Potts-model partition functions) with general complex edge weights, *Journal of Combinatorial Theory, Series B* **103** (2013) 21–45.

[30] M. Jerrum, A very simple algorithm for estimating the number of *k*-colorings of a low-degree graph, *Random Structures and Algorithms* **7** (1995) 157–165.

[31] M. Jerrum and A. Sinclair, Polynomial-time approximation algorithms for the Ising model, *SIAM Journal on computing* **22** (1993) 1087–1116.

[32] T. Lee and T. Yang, Statistical theory of equations of state and phase transitions. I. Theory of condensation, *Physical Review* **87** (1952): 404.

[33] C. Lin, J. Liu and P. Lu, A simple FPTAS for counting edge covers, in: *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp341–348, SIAM, 2014.

[34] P. Lu and Y. Yin, Improved FPTAS for multi-spin systems, in: *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pp 639–654, Springer Berlin Heidelberg, 2013.

[35] G. Regts, *Graph Parameters and Invariants of the Orthogonal Group*, PhD thesis, University of Amsterdam, 2013.

[36] G. Regts, Zero-free regions of partition functions with applications to algorithms and graph limits, arXiv preprint, *arXiv.1507.02089* (2015). To appear in *Combinatorica*.

[37] A.D. Scott and A.D. Sokal, The repulsive lattice gas, the independent-set polynomial, and the Lovász local lemma, *Journal of Statistical Physics* **118** (2005) 1151–1261.

[38] J.B. Shearer, On a problem of Spencer, *Combinatorica* **5** (1998) 241–245.

[39] A. Sinclair, P. Srivastava and M. Thurley, Approximation algorithms for two-state anti-ferromagnetic spin systems on bounded degree graphs, *Journal of Statistical Physics* **155** (2014) 666–686.

[40] A. Sly and N. Sun, The computational hardness of counting in two-spin models on d-regular graphs, in *Proceedings of the 53rd Annual Symposium on Foundations of Computer Science* (FOCS), 2012 IEEE, pp. 361–369. IEEE, 2012.

[41] A. Sokal, A personal list of unsolved problems concerning lattice gases and anti-ferromagnetic Potts models, *Markov Processes And Related Fields* **7** (2001) 21–38.

[42] P. Srivastava, Approximating the hard core partition function with negative activities. Available at `http://www.its.caltech.edu/ piyushs/docs/approx.pdf`

[43] B. Szegedy, Edge-coloring models and reflection positivity, *Journal of the American Mathematical Society* **20** (2007) 969–988.

[44] B. Szegedy, Edge coloring models as singular vertex-coloring models, in: *Fete of Combinatorics and Computer Science* (G.O.H. Katona, A. Schrijver, T.Szönyi, editors), Springer, Heidelberg and János Bolyai Mathematical Society, Budapest (2010) 327–336.

[45] E. Vigoda, Improved bounds for sampling colorings, *Journal of Mathematical Physics* **41** (2000), 1555–1569.

[46] D. Weitz, Counting independent sets up to the tree threshold, in *Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*, STOC 06, pages 140–149, New York, NY, USA, 2006. ACM.