Combinatorics, Probability and Computing (2019), **28**, pp. 696–719
doi:10.1017/S0963548319000105

CAMBRIDGE
UNIVERSITY PRESS

ARTICLE

Weighted counting of solutions to sparse systems of equations

Alexander Barvinok[1,*,†] and Guus Regts[2,‡]

[1]Department of Mathematics, University of Michigan, Ann Arbor, MI 48109-1043, USA, [2]Korteweg de Vries Institute for Mathematics, University of Amsterdam, P.O. Box 94248, 1090 GE Amsterdam, The Netherlands
*Corresponding author. Email: barvinok@umich.edu

(Received 28 September 2017; revised 4 February 2019; first published online 15 April 2019)

Abstract

Given complex numbers $w_1, \ldots, w_n$, we define the weight $w(X)$ of a set X of 0–1 vectors as the sum of $w_1^{x_1} \cdots w_n^{x_n}$ over all vectors $(x_1, \ldots, x_n)$ in X. We present an algorithm which, for a set X defined by a system of homogeneous linear equations with at most r variables per equation and at most c equations per variable, computes $w(X)$ within relative error $\epsilon > 0$ in $(rc)^{O(\ln n - \ln \epsilon)}$ time provided $|w_j| \leqslant \beta(r\sqrt{c})^{-1}$ for an absolute constant $\beta > 0$ and all $j = 1, \ldots, n$. A similar algorithm is constructed for computing the weight of a linear code over $\mathbb{F}_p$. Applications include counting weighted perfect matchings in hypergraphs, counting weighted graph homomorphisms, computing weight enumerators of linear codes with sparse code generating matrices, and computing the partition functions of the ferromagnetic Potts model at low temperatures and of the hard-core model at high fugacity on biregular bipartite graphs.

2010 MSC Codes: Primary 68Q25; Secondary 68W25, 82B20, 52C07, 52B55

1. Weighted counting of 0–1 vectors

1.1 Weight of a set of 0–1 vectors

Let us fix complex numbers $w_1, \ldots, w_n$, referred to as *weights* in what follows. We define the *weight* $w(x)$ of a 0–1 vector $x \in \{0, 1\}^n$ by

$$w(x) = w_1^{\xi_1} \cdots w_n^{\xi_n} = \prod_{j\,:\,\xi_j=1} w_j, \quad \text{where } x = (\xi_1, \ldots, \xi_n).$$

Here we agree that $0^0 = 1$, so that $w(x)$ is a continuous function of $w_1, \ldots, w_n$ for a fixed x.

We define the *weight* of a finite set $X \subset \{0, 1\}^n$ by

$$w(X) = \sum_{x \in X} w(x) = \sum_{\substack{x \in X,\\ x=(\xi_1,\ldots,\xi_n)}} w_1^{\xi_1} \cdots w_n^{\xi_n}. \tag{1.1}$$

[†]Research partially supported by NSF grant DMS 1361541.
[‡]Research supported by a personal NWO Veni grant.

Given $X \subset \{0, 1\}^n$, the value of $w(X)$ as a function of $w_1, \ldots, w_n$ is also known as the *partition function* or *generating function* of X.

Our first main result is as follows.

Theorem 1.1. *Let $A = (a_{ij})$ be an $m \times n$ integer matrix, and let us define $X \subset \{0, 1\}^n$ by*

$$X = \left\{ x \in \{0, 1\}^n,\ x = (\xi_1, \ldots, \xi_n) : \sum_{j=1}^{n} a_{ij}\xi_j = 0 \quad \text{for } i = 1, \ldots, m \right\}.$$

Suppose that the number of non-zero entries in every row of A does not exceed r for some $r \geqslant 2$ and that the number of non-zero entries in every column of A does not exceed c for some $c \geqslant 1$. There is an absolute constant $\alpha > 0$ such that if $w_1, \ldots, w_n \in \mathbb{C}$ are weights satisfying

$$|w_j| \leqslant \frac{\alpha}{r\sqrt{c}} \quad \text{for } j = 1, \ldots, n,$$

then

$$w(X) \neq 0.$$

One can choose $\alpha = 0.46$.

Geometrically, the set X in Theorem 1.1 is the set of 0–1 vectors in a subspace. We are interested in efficient algorithms to compute $w(X)$ approximately. Theorem 1.1 implies that such an efficient algorithm exists for a non-trivial range of weights $w_1, \ldots, w_n$ provided the matrix A is sufficiently sparse (*i.e.* r and c are sufficiently small), even when the dimension n of the ambient space is allowed to be large. This connection between the sparsity condition for A (frequent in applications and easily verified) and the computational complexity of $w(X)$ appears to be new.

1.2 Computing $w(X)$

Theorem 1.1 implies that $w(X)$ can be efficiently approximated as long as the weights w_j satisfy a slightly stronger inequality,

$$|w_j| \leqslant \frac{\beta}{r\sqrt{c}}, \quad \text{for } j = 1, \ldots, n \tag{1.2}$$

for any $\beta < \alpha$, fixed in advance, so one can choose $\beta = 0.45$. We describe the connection below; see also Section 1.2 of [2].

Without loss of generality we assume that the matrix A has no zero rows and no zero columns (although this assumption is not needed in this section, it will be relevant later in Section 5). Indeed, zero rows of A can be ignored and if, say, the nth column of A is zero, we have

$$w(X) = (1 + w_n)w(\widehat{X}),$$

where $\widehat{X} \subset \{0, 1\}_+^{n-1}$ is the set defined by the system $\widehat{A}x = 0$, where $\widehat{A}$ is the $m \times (n-1)$ matrix obtained from A by deleting the nth column.

For a $\zeta \in \mathbb{C}$, let $\zeta w_1, \ldots, \zeta w_n$ be the scaling of the weights and let $w(X;\zeta)$ be the corresponding weight of X so that $w(X;1) = w(X)$ while $w(X;0) = 1$ (note that $0 \in X$). Theorem 1.1 implies that as long as the weights w_j satisfy (1.2), we have

$$w(X;\zeta) \neq 0 \quad \text{provided } |\zeta| \leqslant \frac{\alpha}{\beta} =: \gamma. \tag{1.3}$$

Note that $\gamma > 1$.

https://doi.org/10.1017/S0963548319000105 Published online by Cambridge University Press

Let us choose a continuous branch of $f(\zeta) = \ln w(X;\zeta)$ for $|\zeta| \leqslant \gamma$, and let

$$T_s(\zeta) = f(0) + \sum_{k=1}^{s} \frac{f^{(k)}(0)}{k!} \zeta^k \tag{1.4}$$

be the Taylor polynomial of f of some degree s computed at $\zeta = 0$. Since (1.3) holds and $w(X;\zeta)$ is a polynomial of degree at most n in ζ, we have

$$|f(1) - T_s(1)| \leqslant \frac{n}{(s+1)\gamma^s(\gamma - 1)}$$

(see Lemma 2.2.1 of [2]). Using that $\gamma > 1$, we conclude that to approximate $f(1) = \ln w(X)$ within an additive error $\epsilon > 0$ by $T_s(1)$, it suffices to choose $s = O(\ln n - \ln \epsilon)$, where the implied constant in the 'O' notation depends only on γ. We then say that $e^{T_s(1)}$ approximates $w(X)$ within relative error ϵ.

We have $f(0) = 0$, and computing $f^{(k)}(0)$ for $k = 1, \dots, s$ reduces to computing

$$\frac{d^k}{d\zeta^k} w(X;\zeta)\Big|_{\zeta=0} \quad \text{for } k = 1, \dots, s \tag{1.5}$$

in $O(s^2)$ time. Indeed, it is not hard to see that the values $f^{(k)}(0)$ are the solutions of a non-degenerate triangular system of linear equations with right-hand side given by (1.5): see Section 2.2.2 of [2]. Furthermore,

$$\frac{d^k}{d\zeta^k} w(X;\zeta)\Big|_{\zeta=0} = k! \sum_{\substack{x \in X, \\ x=(\xi_1,\dots,\xi_n)\,: \\ \xi_1+\cdots+\xi_n=k}} w_1^{\xi_1} \cdots w_n^{\xi_n},$$

so computing (1.5) reduces to the inspection of all points $x \in X$, $x = (\xi_1, \dots, \xi_n)$, satisfying $\xi_1 + \cdots + \xi_n \leqslant s$, which can be done through the exhaustive search in $mn^{O(s)}$ time. Given that $s = O(\ln n - \ln \epsilon)$, this produces an algorithm approximating $w(X)$ within a relative error $\epsilon > 0$ in quasi-polynomial $n^{O(\ln n - \ln \epsilon)}$ time, where the implied constant in the 'O' notation depends only on γ in (1.3). In Section 5 we show that we can compute $f^{(k)}(0)$ in (1.4) faster, in $(rc)^{O(\ln n - \ln \epsilon)}$ time. In particular, if r and c are fixed in advance, we obtain a polynomial-time approximation algorithm.

Next, we consider enumerating 0–1 vectors in affine subspaces, not necessarily containing the origin.

1.3 Non-homogeneous linear equations in 0–1 vectors

We interpret a vector $x = (\xi_1, \dots, \xi_n)$ as a column n-vector. Let A be an $m \times n$ integer matrix as above, let b be an integer m-vector and let

$$X = \{x \in \{0, 1\}^n : Ax = b\}$$

be the set of 0–1 vectors satisfying a system of linear equations with matrix A. In general, it is an NP-hard problem to decide whether X is empty, so there is no hope of computing $w(X)$ efficiently.

Suppose, however, that we are presented with a point $y \in X$, $y = (\eta_1, \dots, \eta_n)$. Every point $x \in X$ can be uniquely written as $x = y + z$, $z = (\zeta_1, \dots, \zeta_n)$, where $Az = 0$ and $\zeta_j \in \{-1, 0\}$ if $\eta_j = 1$ and $\zeta_j \in \{0, 1\}$ if $\eta_j = 0$. Let $a_1, \dots, a_n$ be the columns of A and let $\widehat{A}$ be the matrix obtained from A by replacing a_j with $-a_j$ whenever $\eta_j = 1$. Let

$$Z = \{z \in \{0, 1\}^n : \widehat{A}z = 0\}.$$

https://doi.org/10.1017/S0963548319000105 Published online by Cambridge University Press

Hence every point $x \in X$, $x = (\xi_1, \ldots, \xi_n)$, can be uniquely written as $\xi_j = \eta_j + \sigma_j \zeta_j$, where for $z = (\zeta_1, \ldots, \zeta_n)$ we have $z \in Z$ and

$$\sigma_j = \begin{cases} 1 & \text{if } \eta_j = 0, \\ -1 & \text{if } \eta_j = 1. \end{cases}$$

Then, for the weight of Z, we have

$$w(Z) = \sum_{\substack{z \in Z, \\ z=(\zeta_1,\ldots,\zeta_n)}} \prod_{j=1}^{n} w_j^{\zeta_j} = \sum_{\substack{x \in X, \\ x=(\xi_1,\ldots,\xi_n)}} \prod_{j=1}^{n} w_j^{\sigma_j(\xi_j - \eta_j)} = \sum_{\substack{x \in X, \\ x=(\xi_1,\ldots,\xi_n)}} \prod_{j: \xi_j \neq \eta_j} w_j. \tag{1.6}$$

For $x \in \{0, 1\}^n$, $x = (\xi_1, \ldots, \xi_n)$, let

$$\operatorname{dist}(x, y) = |\{j : \xi_j \neq \eta_j\}|$$

be the Hamming distance between x and y.

In particular, if we choose

$$w_1 = \cdots = w_n = \omega$$

for some ω, we get

$$w(Z) = \sum_{x \in X} \omega^{\operatorname{dist}(x,y)}. \tag{1.7}$$

Assuming that every row of A contains no more than $r \geqslant 2$ non-zero entries and every column of A contains no more than $c \geqslant 1$ non-zero entries, we conclude that the sum (1.7) can be computed within relative error $\epsilon > 0$ in $(rc)^{O(\ln n - \ln \epsilon)}$ time provided

$$|\omega| \leqslant \frac{\beta}{r\sqrt{c}},$$

where $\beta > 0$ is an absolute constant (one can choose $\beta = 0.45$). If r and c are fixed in advance, we have a polynomial-time approximation algorithm of $(n/\epsilon)^{O(1)}$ complexity.

In the next section we consider combinatorial applications of our result. We first consider a variation of Theorem 1.1 that applies to codes.

1.4 Weight of a code

Let $\kappa > 1$ be an integer. We consider n-vectors $x = (\xi_1, \ldots, \xi_n)$ with coordinates ξ_j taking values in the set $\{0, \ldots, \kappa - 1\}$, which we interpret as the set $\mathbb{Z}/\kappa\mathbb{Z}$ of remainders modulo κ. Given n complex numbers $w_1, \ldots, w_n$, we define the *weight* $w(x)$ of a vector $x \in (\mathbb{Z}/\kappa\mathbb{Z})^n$ by

$$w(x) = \prod_{j: \xi_j \neq 0} w_j \quad \text{for } x = (\xi_1, \ldots, \xi_n)$$

and the *weight* $w(X)$ of a set $X \subset (\mathbb{Z}/\kappa\mathbb{Z})^n$ by

$$w(X) = \sum_{x \in X} w(x)$$

(we agree that the weight of the zero vector is 1).

We obtain the following result.

Theorem 1.2. *Let $A = (a_{ij})$ be an $m \times n$ integer matrix and let us define a set $X \subset (\mathbb{Z}/\kappa\mathbb{Z})^n$ by*

$$X = \left\{ x \in (\mathbb{Z}/\kappa\mathbb{Z})^n,\ x = (\xi_1, \ldots, \xi_n) \colon \sum_{j=1}^{n} a_{ij}\xi_j \equiv 0 \mod \kappa \quad \textit{for } i = 1, \ldots, m \right\}.$$

https://doi.org/10.1017/S0963548319000105 Published online by Cambridge University Press

Suppose that the number of non-zero entries in every row of A does not exceed r for some $r \geqslant 2$ and that the number of non-zero entries in every column of A does not exceed c for some $c \geqslant 1$. There is an absolute constant $\alpha > 0$ such that if $w_1, \ldots, w_n \in \mathbb{C}$ are weights satisfying

$$|w_j| \leqslant \frac{\alpha}{(\kappa - 1)r\sqrt{c}} \quad \textit{for } j = 1, \ldots, n,$$

then

$$w(X) \neq 0.$$

One can choose $\alpha = 0.46$.

As in Section 1.2, we obtain an algorithm of $(rc)^{O(\ln \kappa n - \ln \epsilon)}$ complexity to approximate $w(X)$ within relative error $\epsilon > 0$ provided

$$|w_j| \leqslant \frac{\beta}{(\kappa - 1)r\sqrt{c}} \quad \text{for } j = 1, \ldots, n,$$

where $\beta < \alpha$ is fixed in advance (we can choose $\beta = 0.45$). For r and c fixed in advance, the algorithm has polynomial $(\kappa n/\epsilon)^{O(1)}$ complexity.

Organization. We deduce Theorem 1.1 and Theorem 1.2 from a general result asserting that

$$\int_{\mathbb{T}^m} e^{p(z)}\, d\mu \neq 0,$$

for some Laurent polynomials $p : \mathbb{T}^m \longrightarrow \mathbb{C}$ on the torus $\mathbb{T}^m$ endowed with a product probability measure μ (see Theorem 3.1 and Corollary 3.2 below). After that, the proofs of Theorems 1.1 and 1.2 are completed in a more or less straightforward way in Section 4.

In Section 5, we provide details of an approximation algorithm for $w(X)$. We do not discuss an analogous algorithm for codes in Theorem 1.2 as it is very similar. We first consider some concrete combinatorial applications of these results in Section 2 below.

2. Combinatorial applications

We apply Theorem 1.1 to weighted counting of perfect matchings in hypergraphs, computing the partition function of the hard-core model at high fugacity for biregular bipartite graphs and to weighted counting of graph homomorphisms. We apply Theorem 1.2 to computing weight enumerators of linear codes with sparse code generating matrices and to computing the partition function of the ferromagnetic Potts model at low temperatures.

2.1 Perfect matchings in hypergraphs

A *hypergraph* $H = (V, E)$ is a finite set V of *vertices* together with a collection E of non-empty subsets V, called *edges* of the hypergraph. The *degree* of a vertex v is the number of edges $e \in E$ that contain v. A *perfect matching* in H is a set of pairwise disjoint edges $e_1, \ldots, e_n$, such that $e_1 \cup \cdots \cup e_n = V$. Let us introduce a 0–1 variable x_e for each $e \in H$. We encode a collection of edges of H by a 0–1 vector, where

$$x_e = \begin{cases} 1 & \text{if } e \text{ is in the collection,} \\ 0 & \text{otherwise.} \end{cases}$$

Then $e_1, \ldots, e_n$ is a perfect matching if and only if

$$\sum_{e : v \in e} x_e = 1 \quad \text{for all } v \in V. \tag{2.1}$$

https://doi.org/10.1017/S0963548319000105 Published online by Cambridge University Press

In the system (2.1) the number of variables per equation is the maximum degree d of a vertex of H and the number of equations per variable is the maximum cardinality k of an edge. It is an NP-complete problem to find if a given hypergraph contains a perfect matching provided $k \geqslant 3$: see *e.g.* Problem SP1 in [1]. However, as follows from Section 1.3, given *one* perfect matching M_0, we can efficiently approximate a certain statistic over *all* perfect matchings M of H, namely the sum

$$\sum_{M \in \mathcal{M}(H)} \omega^{\operatorname{dist}(M_0, M)}, \tag{2.2}$$

where $\mathcal{M}(H)$ is the set of all perfect matchings, $\operatorname{dist}(M_0, M)$ is the Hamming distance between matchings, that is, the number of edges where the matchings differ and

$$|\omega| \leqslant \frac{\beta}{d\sqrt{k}}.$$

The complexity of the algorithm approximating (2.2) within relative error $\epsilon > 0$ is $(dk)^{O(\ln |E| - \ln \epsilon)}$. If d and k are fixed in advance, the algorithm achieves polynomial $(|E|/\epsilon)^{O(1)}$ complexity. This can be contrasted with the fact that knowing one solution of a problem generally does not help to find another or to count all solutions: see [23] and [24].

Is is shown in [4] that if the hypergraph is *uniform* and *k-partite*, that is, we have $V = V_1 \cup \cdots \cup V_k$ with pairwise disjoint $V_1, \ldots, V_k$ such that $|V_1| = \cdots = |V_k| = n$ and every edge $e \in E$ contains exactly one vertex from each V_i, then one can efficiently approximate (2.2) under the weaker condition

$$|\omega| \leqslant \frac{\beta}{\sqrt{d-1}}$$

for any $\beta < 1$, fixed in advance.

2.2 The hard-core model at high fugacity

Given an undirected graph $G = (V, E)$, a set $S \subset V$ of vertices is called *independent* if no two vertices of S span an edge of G (we agree that $S = \emptyset$ is always independent). The *independence polynomial* of G is a univariate polynomial defined by

$$p_G(\lambda) = \sum_{\substack{S \subset V \\ S \text{ is independent}}} \lambda^{|S|} \tag{2.3}$$

(see *e.g.* Chapter 6 of [2]). It is also known as the partition function of the hard-core model. The parameter λ is known as the *fugacity*.

The problem of (approximately) computing the number of independent sets in a bipartite graph is considered to be computationally hard. It is the basis of the class of #BIS hard problems, and it is known that approximating the $p_G(\lambda)$ on bipartite graphs of maximum degree d is a #BIS hard problem, provided $\lambda > ((d-1)^{d-1})/((d-2)^d)$ [10]. Moreover, as Cai *et al.* [10] informed us, it follows from their construction that computing $p_G(\lambda)$ for sufficiently large λ remains a #BIS-hard problem when restricted to bipartite d-regular graphs G.

However, Jenssen, Keevash and Perkins [16] showed that for $d \geqslant 3$ there exists $\lambda^* = \lambda^*(d) > 0$ such that, for all $\lambda > \lambda^*$ and all d-regular, bipartite, expander graphs G, the value of $p_G(\lambda)$ can be approximated in polynomial time. Here we will use Theorem 1.1 to show that for each fixed $d_1, d_2 \in \mathbb{N}$ such that $d_2 - d_1 \geqslant 1$, there exists $\lambda_0 = \lambda_0(d_1, d_2) > 0$ such that for all $\lambda > \lambda_0$ and any biregular, bipartite graph with degrees d_1, d_2, we can approximate $p_G(\lambda)$ in polynomial time.

To that end, let us fix a biregular bipartite graph $G = (V, E)$ with degrees d_1 and $d_2 \geqslant d_1 + 1$. We write $V = L \cup R$ for the bipartition, and we assume that each vertex in L has degree

d_1 and each vertex in R has degree d_2. For an independent set I we write $I_L := I \cap L$ and $I_R := I \cap R$.

We wish to encode $p_G(\lambda)$ as the weight $w(X)$ of a suitably defined set X. We direct all edges from L to R, thus making G a directed graph. We associate with each vertex $v \in V$ a 0–1 variable x_v and with each edge $(u, v) \in E$ a 0–1 variable x_{uv}. Let X be the solution set to the following system of equations:

$$-x_u + x_v + x_{uv} = 0 \quad \text{for each directed edge } (u, v) \in E. \tag{2.4}$$

Any $x \in X$ uniquely corresponds to an independent set I of G. Indeed, let I be the the sets of vertices $u \in L$ for which $x_u = 0$ and vertices $v \in R$ for which $x_v = 1$. Then, for $u \in I_L$, none of its neighbours will be contained in I, since for each edge (u, v) the value of x_v is forced to be zero. Similarly, for any $v \in I_R$, none of its neighbours will be contained in I since for each edge (u, v), the value of x_u is forced to be 1. Hence the set I is independent. Conversely, if I is an independent set, setting

$$x_u = \begin{cases} 0 & \text{if } u \in I_L, \\ 1 & \text{if } u \in L \setminus I_L, \end{cases} \qquad x_v = \begin{cases} 1 & \text{if } v \in I_R, \\ 0 & \text{if } v \in R \setminus I_R, \end{cases} \qquad x_{uv} = \begin{cases} 0 & \text{if } u \in I_L \text{ or } v \in I_R \\ 1 & \text{if } u \in L \setminus I_L \text{ and } v \in R \setminus I_R \end{cases}$$

gives a solution to (2.4).

Next, we introduce weights w_u for the coordinates x_u with $u \in L$, weights w_v for the coordinates x_v with $v \in R$ and weights w_{uv} for the coordinates x_{uv} with $(u, v) \in E$ as follows:

$$\begin{aligned} w_u &= \omega^{(d_2-d_1)/2} \quad \text{for } u \in L, \\ w_v &= \omega^{(d_2-d_1)/2} \quad \text{for } v \in R, \quad \text{and} \\ w_{uv} &= \omega \quad \text{for } (u, v) \in E. \end{aligned}$$

For a solution $x \in X$ corresponding to an independent set I, we then have

$$\begin{aligned} w(x) &= \Bigg(\prod_{v \in L \setminus I_L} \omega^{(d_2-d_1)/2}\Bigg)\Bigg(\prod_{\substack{\{u,v\} \in E \\ u,v \notin I}} \omega\Bigg)\Bigg(\prod_{u \in I_R} \omega^{(d_2-d_1)/2}\Bigg) \\ &= \omega^{(d_2-d_1)(|L|-|I_L|)/2} \cdot \omega^{|E|-d_1|I_L|-d_2|I_R|} \cdot \omega^{(d_2-d_1)|I_R|/2} \\ &= \omega^{(d_2-d_1)|L|/2+|E|} \cdot \omega^{-(d_1+d_2)|I_L|/2} \cdot \omega^{-(d_1+d_2)|I_R|/2} \\ &= \omega^{(d_1+d_2)|L|/2} \omega^{-(d_1+d_2)|I|/2}. \end{aligned}$$

In other words, for the weight of X, we have

$$w(X) = \omega^{(d_1+d_2)|L|/2} p_G\bigg(\frac{1}{\omega^{(d_1+d_2)/2}}\bigg)$$

for the independence polynomial p_G defined by (2.3).

Now, since in (2.4) the number of variables per equation is 3 and the number of equations per variable is at most d_2, it follows from Theorem 1.1 that if

$$|\lambda| \geqslant (6.7\sqrt{d_2})^{d_1+d_2} > \bigg(\frac{3\sqrt{d_2}}{0.45}\bigg)^{d_1+d_2},$$

then $p_G(\lambda) \neq 0$, and moreover that we can efficiently approximate p_G (in polynomial time if d_2 is fixed in advance). Moreover, we note that with a similar argument, for a d-regular bipartite graph $G = (L \cup R, E)$, we can efficiently approximate the sum

$$\sum_{\substack{I \subset L \cup R \\ I \text{ is independent}}} \lambda^{|I \cap L|}$$

https://doi.org/10.1017/S0963548319000105 Published online by Cambridge University Press

for large λ. This is somewhat similar in spirit to a result of van den Berg and Steiff [5], who showed that for the integer lattice $\mathbb{Z}^d$, assigning $\lambda_1 > 0$ to vertices with even coordinate sum and $\lambda_2 > 0$ to vertices with odd coordinate sum, for all but a countable set of pairs (λ_1, λ_2) the associated Gibbs measure is unique.

2.3 Weighted counting of graph homomorphisms

Let $G_1 = (V_1, E_1)$ be an undirected graph without loops or multiple edges and let $G_2 = (V_2, E_2)$ be an undirected graph without multiple edges, but possibly with loops. We assume that $V_2 = \{1, \ldots, n\}$ and assume that G_1 and G_2 are both connected. A map $\phi : V_1 \longrightarrow V_2$ is called a *homomorphism* if $\phi(u)$ and $\phi(v)$ span an edge of G_2 whenever u and v span an edge of V_1. If V_2 is the complete graph without loops then every homomorphism $\phi : G_1 \longrightarrow G_2$ is naturally interpreted as a colouring of the vertices of G_1 with a set of n colours such that no two vertices spanning an edge of G_1 are coloured with the same colour (such colourings are called *proper*). For any fixed $n \geqslant 3$, it is an NP-complete problem to decide whether a given graph admits a proper n-colouring: see *e.g.* Problem GT5 in [1]. Our goal is to encode all homomorphisms $\phi : G_1 \longrightarrow G_2$ that map a fixed vertex $a \in V_1$ to a fixed vertex, say n, of G_2 as the set of 0–1 solutions to a system of linear equations.

We say that vertices $u, v \in V_1$ are *neighbours* if $\{u, v\} \in E_1$. We orient the edges of G_1 arbitrarily, so that an edge of G_1 is an ordered pair of neighbours (u, v). Let us introduce 0–1 variables x_{ij}^{uv} indexed by (now directed) edges $(u, v) \in E_1$ and ordered pairs $1 \leqslant i, j \leqslant n$ such that $\{i, j\} \in E_2$ (we may have $i = j$). The idea is to use the variables x_{ij}^{uv} to encode a map $\phi : V_1 \longrightarrow V_2$, so that

$$x_{ij}^{uv} = \begin{cases} 1 & \text{if } \phi(u) = i \text{ and } \phi(v) = j, \\ 0 & \text{otherwise.} \end{cases} \tag{2.5}$$

For every ordered pair of neighbours (u, v) and every vertex $i \in V_2$, we define the sum

$$S_i^{u,v} = \sum_{j:\{i,j\} \in E_2} x_{ij}^{uv} \quad \text{if } (u, v) \in E_1 \quad \text{and} \quad S_i^{u,v} = \sum_{j:\{i,j\} \in E_2} x_{ji}^{vu} \quad \text{if } (v, u) \in E_1, \tag{2.6}$$

and for every $u \in V_1$ and every $i \in V_2$, we introduce the following equations:

$$\begin{aligned} &\text{Fix } u \in V_1 \setminus \{a\} \text{ and } i \in V_2. \\ &\text{The sums } S_i^{u,v}\text{, where } v \text{ is a neighbour of } u\text{, are all equal.} \end{aligned} \tag{2.7}$$

The idea, of course, is that the sums (2.6) are all equal to 1 if $\phi(u) = i$ and equal to 0 if $\phi(u) \neq i$. Next, we encode the condition $\phi(a) = n$ by the following system of equations:

$$\text{For all neighbours } v \text{ of } a,\ S_n^{a,v} = 1 \text{ and } S_j^{a,v} = 0 \text{ for } j \neq n. \tag{2.8}$$

Now we claim that for every 0–1 solution $\{x_{ij}^{uv}\}$ of the system (2.7)–(2.8), for any vertex $u \in V_1$, there is a unique vertex $i_u \in V_2$ such that the following equations hold:

$$\text{For all neighbours } v \text{ of } u\text{, we have } S_{i_u}^{u,v} = 1 \text{ and } S_j^{u,v} = 0 \text{ for } j \neq i_u. \tag{2.9}$$

Then, for the map $\phi : V_1 \longrightarrow V_2$ defined by $\phi(u) = i_u$, the conditions (2.5) are satisfied.

Clearly, if a choice $u \longmapsto i_u$ exists, it is unique. Because of (2.8), equations (2.9) hold for $u = a$ and $i_u = n$. Since G_1 is connected, it suffices to show that whenever (2.9) holds for some vertex u, then for every neighbour w of u we can define $i_w \in V_2$ so that (2.9) holds with u replaced by w throughout. Indeed, let w be a neighbour of u such that $(u, w) \in E_1$. It follows by (2.9) that there exists i_w such that

$$x_{i_u i_w}^{uw} = 1 \text{ and } x_{jk}^{uw} = 0 \text{ whenever } j \neq i_u \text{ or } k \neq i_w.$$

https://doi.org/10.1017/S0963548319000105 Published online by Cambridge University Press

From (2.7) it follows that for any neighbour v of w, we have

$$S_{i_w}^{w,v} = S_{i_w}^{w,u} = 1 \text{ and } S_j^{w,v} = S_j^{w,u} = 0 \text{ for } j \neq i_w,$$

as required. The case of neighbours w of u such that $(w, u) \in E_1$ is handled similarly. This proves that 0–1 solutions $\{x_{ij}^{uv}\}$, if any, of the system (2.7)–(2.8), are in one-to-one correspondence with graph homomorphisms $\phi : G_1 \longrightarrow G_2$ such that $\phi(a) = n$.

As we are interested in keeping the system (2.7)–(2.8) as sparse as possible, we arrange equations (2.7) as follows. For a given $u \in V_1$, we list the neighbours v of u in some order $v_1, \ldots, v_m$ and then equate $S_i^{u,v_k} - S_i^{u,v_{k+1}} = 0$ for $k = 1, \ldots, m-1$. When the chosen vertex a is a neighbour, we let $v_1 = a$. This way the system (2.7)–(2.8) has no more than $2d_2$ variables per equation, where d_2 is the largest degree of a vertex of G_2, and no more than 4 equations per variable.

Suppose that we are given a homomorphism $\phi : G_1 \longrightarrow G_2$ satisfying the constraint $\phi(a) = n$ for a fixed vertex a of G_1 and a fixed vertex n of G_2. As in Section 1.3, for an $\omega \in \mathbb{C}$ we consider the sum

$$\sum_{\psi\, :\, \psi(a)=n} \omega^{2\text{dist}(\phi,\psi)}, \tag{2.10}$$

where ψ ranges over all graph homomorphisms satisfying $\psi(a) = n$ and $\text{dist}(\phi, \psi)$ is the number of directed edges where ϕ and ψ disagree. As follows from Section 1.3, we can approximate (2.10) within relative error $\epsilon > 0$ in $d_2^{O(\ln|E_1|+\ln|E_2|-\ln\epsilon)}$ time provided

$$|\omega| \leqslant \frac{\gamma}{d_2} \tag{2.11}$$

for some absolute constant $\gamma > 0$ (we can choose $\gamma = 0.1$). If the largest degree d_2 of a vertex of G_2 is fixed in advance, we obtain a polynomial-time approximation algorithm.

Suppose that G_2 is the complete graph with n vertices and no loops, so that a homomorphism $G_1 \longrightarrow G_2$ is interpreted as a proper n-colouring of G_1 and $d_2 = n - 1$. If $n > d_1$, where d_1 is the largest degree of a vertex of G_1, it is trivial to come up with a homomorphism (proper n-colouring) $\phi : G_1 \longrightarrow G_2$ having a prescribed value on a prescribed vertex. In this case, the sum (2.10) is taken over all proper n-colourings ψ of G_2 and each colouring is counted with weight exponentially small in the number of edges of G_1 whose colouring differ under ϕ and ψ. If we could choose $\omega = 1$ in (2.10), we would have counted all proper n-colourings of G_1 with $n > d_1$ colours, a notoriously difficult problem; see [25] and [11] for a randomized polynomial-time approximation algorithm for counting n-colourings assuming that $n > (11/6)d_1$.

Given a pair of graphs G_1 and G_2, let us modify G_2 to a graph $\widehat{G}_2$ by adding an extra vertex $n + 1$ with a loop and connected to all other vertices of G_2. Then there is always a homomorphism $\phi : G_1 \longrightarrow \widehat{G}_2$ which sends every vertex of G_1 to the newly added vertex $n + 1$. In this case the sum (2.10) with G_2 replaced by $\widehat{G}_2$ and n replaced by $n + 1$ is interpreted as the sum over all homomorphisms of the induced subgraphs of G_1 to G_2.

2.4 Computing weight enumerators of linear codes

If κ is a prime, the set $\mathbb{Z}/\kappa\mathbb{Z}$ is identified with the finite field $\mathbb{F}_\kappa$ with κ elements, and $(\mathbb{Z}/\kappa\mathbb{Z})^n$ is the n-dimensional vector space over $\mathbb{F}_\kappa$. A set $X \subset \mathbb{F}_\kappa^n$ is called a *code*. The univariate polynomial

$$p_X(z) = 1 + \sum_{k=1}^{n} p_k(X) z^k,$$

where $p_k(X)$ is the number of vectors in X with exactly k non-zero coordinates, is called the *weight enumerator* of X: see *e.g.* Chapter 3 of [18].

https://doi.org/10.1017/S0963548319000105 Published online by Cambridge University Press

Suppose that $X \subset \mathbb{F}_\kappa^n$ is defined by a system of linear equations

$$X = \{x \in \mathbb{F}_\kappa^n : Ax = 0\}, \tag{2.12}$$

where $A = (a_{ij})$ is an $m \times n$ matrix with entries $a_{ij} \in \mathbb{F}_\kappa$. Hence $X \subset \mathbb{F}_\kappa^n$ is a subspace, called a *linear code*. Generally, it is hard to compute $p_X(z)$ as it is hard to determine the smallest $k \geqslant 1$ with $p_k(X) \neq 0$: see [6] and [8].

Suppose now that the number of non-zero entries in every row of A does not exceed $r \geqslant 2$ and the number of non-zero entries in every column of A does not exceed $c \geqslant 1$. Let us define weights

$$w_1 = \cdots = w_n = z$$

for some $z \in \mathbb{C}$. Then

$$w(X) = p_X(z),$$

and Theorem 1.2 implies that $p_X(z) \neq 0$ provided $|z| \leqslant \alpha/(\kappa - 1)r\sqrt{c}$ and that $p_X(z)$ can be approximated within relative error $\epsilon > 0$ in $(rc)^{O(\ln \kappa n - \ln \epsilon)}$ time, provided $|z| \leqslant \beta/(\kappa - 1)r\sqrt{c}$, where $\beta < \alpha$ is fixed in advance. Again, if r and c are fixed in advance, we obtain an algorithm of polynomial $m(\kappa n/\epsilon)^{O(1)}$ complexity. Linear codes X (typically binary, *i.e.* for $\kappa = 2$) for which the number of non-zero entries in each row of the matrix A in (2.12) is small are called *low-density parity-check* codes. They have many desirable properties and are of considerable interest: see Section 11 of [20].

Let $C = X^\perp$, $C \subset \mathbb{F}_\kappa^n$, be the subspace (linear code) spanned by the rows of A (we say that A is the *generator matrix* of C). The MacWilliams identity for the weight enumerators of p_X and p_C (see Theorem 3.5.3 of [18]) states that

$$p_X(z) = \frac{1}{\kappa^{\dim C}} \big(1 + (\kappa - 1)z\big)^n p_C\left(\frac{1 - z}{1 + (\kappa - 1)z}\right).$$

It follows that

$$p_C\left(\frac{1 - z}{1 + (\kappa - 1)z}\right) \neq 0 \quad \text{provided } |z| \leqslant \frac{\alpha}{(\kappa - 1)r\sqrt{c}}$$

and that the value of

$$p_C\left(\frac{1 - z}{1 + (\kappa - 1)z}\right)$$

can be efficiently approximated provided

$$|z| \leqslant \frac{\beta}{(\kappa - 1)r\sqrt{c}}.$$

In other words, the weight enumerator $p_C(z)$ of a linear code C with a sparse code generator matrix is non-zero and can be efficiently approximated provided $|1 - z| = O(1/r\sqrt{c})$, where r is an upper bound on the number of non-zero entries in every row, c is an upper bound on the number of non-zero entries in every column of the matrix and the implied constant in the 'O' notation is absolute (in particular, it does not depend on κ).

One notable example of such a code with a sparse generating matrix is the binary *cut code* consisting of the indicators of cuts in a given graph $G = (V, E)$ with set V of vertices and set E of edges (see Section 1.9 of [13] and [8]), that is, indicators of subsets $E_S \subset E$ consisting of the edges with one endpoint in $S \subset V$ and the other in $V \setminus S$. The rows of the code generating matrix are parametrized by vertices $v \in V$ of the graph, the columns are parametrized by the edges e of the graph and the (v, e) entry of the matrix is 1 if v is an endpoint of e and 0 otherwise (hence each row is the indicator of the cut associated with the corresponding vertex). We observe that the code generating matrix of a cut code contains at most $d(G)$ non-zero entries in every row, where $d(G)$ is the largest degree of a vertex of G, and exactly two non-zero entries in every

https://doi.org/10.1017/S0963548319000105 Published online by Cambridge University Press

column. The algorithm obtained for computing the weight of a cut code achieves roughly the same approximation as the algorithms of [21] and of Chapter 7 of [2], where we approach computing weights of cuts via the graph homomorphism partition function.

2.5 Ferromagnetic Potts model at low temperatures

Let $G=(V,E)$ be a connected undirected graph, without loops or multiple edges. Given a real $\beta>0$ and an integer $\kappa>1$, we consider the sum

$$P_{G,\kappa}(\beta)=\sum_{\phi:V\longrightarrow\{0,\ldots,\kappa-1\}}\exp\Big\{\beta\sum_{\{u,v\}\in E}\delta_{\phi(u)\phi(v)}\Big\}, \tag{2.13}$$

where

$$\delta_{ij}=\begin{cases}1 & \text{if } i=j,\\ 0 & \text{if } i\neq j.\end{cases}$$

The expression (2.13) is known as the partition function of the *ferromagnetic* (since $\beta>0$) *Potts model with* κ *colours*: see *e.g.* [14]. Here the numbers $0,1,\ldots,\kappa-1$ are interpreted as colours: we colour the vertices of G with κ colours in all possible ways, and each edge of G with identically coloured endpoints contributes to the inner sum. The number β plays the role of the inverse temperature. Using cluster expansions, it was shown in [15] that for some induced subgraphs G of the lattice $\mathbb{Z}^d$ the sum (2.13) can be approximated in polynomial time provided $\beta>\beta_0(d,\kappa)$ for some constant β_0 (*i.e.* at sufficiently low temperatures). Here we deduce this result for a wide family of graphs and an explicit bound on β_0 from our Theorem 1.2.

First, we rewrite (2.13) in the form

$$P_{G,\kappa}(\beta)=e^{\beta|E|}\sum_{\phi:V\longrightarrow\{0,\ldots,\kappa-1\}}\prod_{\{u,v\}\in E}w(\phi(u),\phi(v)), \tag{2.14}$$

where $w(i,j)=w_\beta(i,j)=e^{\beta(\delta_{ij}-1)}$. Since $\beta>0$, we have $|w(i,j)|\leqslant 1$ and $w(i,j)=1$ if and only if $i=j$.

Next, we write the sum in (2.14) in the form $w(X)$, where X is the set in Theorem 1.2. For that, we interpret colours $0,1,\ldots,\kappa-1$ as remainders modulo κ. We direct the edges of G in an arbitrary way and with every, now directed, edge (u,v) we associate a variable x_{uv} taking values in $\mathbb{Z}/\kappa\mathbb{Z}$. The intended meaning of the variables x_{uv} is that

$$x_{uv}\equiv\phi(v)-\phi(u)\mod\kappa\quad\text{for all } (u,v)\in E, \tag{2.15}$$

so that $x_{uv}\equiv 0$ if and only if the endpoints of the edge $\{u,v\}$ are coloured with the same colour. Given a set $\{x_{uv}:(u,v)\in E\}$, a solution $\phi:V\longrightarrow\mathbb{Z}/\kappa\mathbb{Z}$ to the system (2.15) exists if and only if $\{x_{uv}\}$ satisfy the system of linear equations, constructed as follows. We pick a cycle C in G, orient it arbitrarily, and write

$$\sum_{\substack{\{u,v\}\in C:\\ (u,v)\text{ is co-oriented with } C}}x_{uv}-\sum_{\substack{\{u,v\}\in C:\\ (u,v)\text{ is counter-oriented with } C}}x_{uv}\equiv 0\mod\kappa. \tag{2.16}$$

Moreover, since G is connected, as long as equations (2.16) are satisfied for all cycles C, the system (2.15) has exactly κ solutions, that differ by a shift of an element of $\mathbb{Z}/\kappa\mathbb{Z}$. Indeed, if equations (2.15) are satisfied then clearly (2.16) holds. On the other hand, given a solution to (2.16), we pick a vertex v and assign the value of $\phi(v)$ arbitrarily. Then, for every vertex w we choose a path connecting w to v and assign values of ϕ to the vertices along the path (in a necessarily unique way) so that equations (2.15) are satisfied. Because of (2.16), the value of $\phi(w)$ does not depend on the chosen path.

https://doi.org/10.1017/S0963548319000105 Published online by Cambridge University Press

Let $X \subset (\mathbb{Z}/\kappa\mathbb{Z})^E$ be the set of solutions of the system (2.16). We introduce a weight $w_{uv} = e^{-\beta}$ for each coordinate x_{uv} with $(u, v) \in E$ and write (2.14) as

$$P_{G,\kappa}(\beta) = \kappa\, e^{\beta|E|} w(X),$$

where X is the set of solutions to the system (2.16).

Equations (2.16) are not independent: it suffices to write (2.16) for a set of cycles $\mathcal{C}$ that generate the homology group $H_1(G;\mathbb{Z})$. In view of Theorem 1.2, we would like to choose such a generating set $\mathcal{C}$ of $H_1(G;\mathbb{Z})$ so that the number of edges in each cycle $C \in \mathcal{C}$ does not exceed some $r \geqslant 2$ and the number of cycles $C \in \mathcal{C}$ containing a given edge does not exceed some $c \geqslant 1$, for the smallest possible values of r and c. Then we can approximate the partition function $P_{G,\kappa}(\beta)$ of (2.13)–(2.14) provided

$$\beta \geqslant 0.8 + \ln\left((\kappa - 1)r\sqrt{c}\right) > -\ln 0.45 + \ln\left((\kappa - 1)r\sqrt{c}\right),$$

and for fixed r and c, we get a polynomial-time approximation algorithm.

For example, suppose that G is an induced subgraph of the integer lattice $\mathbb{Z}^d$ (with $d \geqslant 2$) constructed as follows. Given a point $(a_1, \ldots, a_d) \in \mathbb{Z}^d$, we call the set

$$\{(x_1, \ldots, x_d) : a_k \leqslant x_k \leqslant a_k + 1 : k = 1, \ldots, d\}$$

an *elementary cube*. We take finitely many elementary cubes whose union U is a simply connected subset of $\mathbb{R}^d$ and let G be the induced subgraph with vertices in U. Then there is a system of generators, $\mathcal{C}$, of $H_1(G;\mathbb{Z})$ consisting of cycles with $r = 4$ edges each and such that every edge of $C \in \mathcal{C}$ belongs to at most $c = 2(d-1)$ cycles (we choose the cycles on the boundary of two-dimensional faces of the elementary cubes comprising U). Hence, for such a graph G, we obtain a polynomial-time approximation algorithm for $P_{G,\kappa}(\beta)$ provided

$$\beta \geqslant 2.6 + \ln\left((\kappa - 1)\sqrt{d-1}\right) > \ln\frac{4\sqrt{2}}{0.45} + \ln\left((\kappa - 1)\sqrt{d-1}\right).$$

3. Integrating over the torus

We begin our preparations to prove Theorems 1.1 and 1.2.

3.1 Laurent polynomials on the torus

Let

$$\mathbb{S}^1 = \{z \in \mathbb{C} : |z| = 1\}$$

be the unit circle in the complex plane and let

$$\mathbb{T}^m = \mathbb{S}^1 \times \cdots \times \mathbb{S}^1$$

be the direct product of m copies of $\mathbb{S}^1$ (torus), endowed with the product measure $\mu = \mu_1 \times \cdots \times \mu_m$, where μ_i is a Borel probability measure on the ith copy of $\mathbb{S}^1$. We consider Laurent polynomials $p : \mathbb{T}^m \longrightarrow \mathbb{C}$,

$$p(z_1, \ldots, z_m) = \sum_{a \in A} \gamma_a \mathbf{z}^a, \tag{3.1}$$

as random variables on $\mathbb{T}^m$. Here $A \subset \mathbb{Z}^m$ is a finite set of integer vectors, $\gamma_a \in \mathbb{C}$ for all $a \in A$ and

$$\mathbf{z}^a = z_1^{\alpha_1} \cdots z_m^{\alpha_m} \quad \text{provided } a = (\alpha_1, \ldots, \alpha_m),$$

where $z_i^0 = 1$. We are interested in conditions on the coefficients γ_a which ensure that $\mathbb{E}e^p \neq 0$.

For $a \in A$ we define the *support* of a by

$$\operatorname{supp} a = \{i : \alpha_i \neq 0\} \quad \text{where } a = (\alpha_1, \ldots, \alpha_m).$$

Consequently, $|\operatorname{supp} a|$ is the number of non-zero coordinates of $a \in \mathbb{Z}^m$. In this section, we prove the following main result.

Theorem 3.1. *Let $p : \mathbb{T}^m \longrightarrow \mathbb{C}$ be a Laurent polynomial as in (3.1). Suppose that for some $0 \leqslant \theta_1, \ldots, \theta_m < 2\pi/3$ we have*

$$2 \sum_{\substack{a \in A: \\ i \in \operatorname{supp} a}} |\gamma_a| \prod_{j \in \operatorname{supp} a} \frac{1}{\cos(\theta_j/2)} \leqslant \theta_i \quad \text{for } i = 1, \ldots, m. \tag{3.2}$$

Then

$$\mathbb{E} e^p \neq 0.$$

By choosing θ_i in a particular way, we obtain the following corollary.

Corollary 3.2. *There exists an absolute constant $\tau > 0$ such that if $p : \mathbb{T}^m \longrightarrow \mathbb{C}$ is a Laurent polynomial as in (3.1) and*

$$|\operatorname{supp} a| \leqslant c \quad \text{for all } a \in A$$

and some $c \geqslant 1$, and

$$\sum_{\substack{a \in A: \\ i \in \operatorname{supp} a}} |\gamma_a| \leqslant \frac{\tau}{\sqrt{c}} \quad \text{for } i = 1, \ldots, m,$$

then

$$\mathbb{E} e^p \neq 0.$$

One can choose $\tau = 0.56$.

The proof is somewhat similar to that of [3] for $\mathbb{E} e^p$ where $p : \{-1, 1\}^m \longrightarrow \mathbb{C}$ is a polynomial on the Boolean cube.

We start with a simple lemma (a discrete version of this lemma was suggested by Bukh [9]).

Lemma 3.3. *Let $f : \Omega \longrightarrow \mathbb{C}$ be a random variable and let $0 \leqslant \theta < 2\pi/3$ be a real number such that $f(\omega) \neq 0$ for all $\omega \in \Omega$ and the angle between any two complex numbers $f(\omega_1) \neq 0$ and $f(\omega_2) \neq 0$ considered as vectors in $\mathbb{R}^2 = \mathbb{C}$ does not exceed θ. Suppose further that $\mathbb{E}|f| < +\infty$. Then*

$$|\mathbb{E} f| \geqslant \left(\cos \frac{\theta}{2}\right) \mathbb{E}|f|.$$

Proof. First, we claim that 0 does not lie in the convex hull of vectors $f(\omega) \in \mathbb{C} = \mathbb{R}^2$. Otherwise we conclude by the Carathéodory theorem that 0 is a convex combination of some 3 vectors $f(\omega_1)$, $f(\omega_2)$ and $f(\omega_3)$ and the angle between some two of them is at least $2\pi/3$, which is a contradiction. Hence the vectors $f(\omega)$ lie in some convex cone (angle) $K \subset \mathbb{C}$ measuring at most θ and with vertex at 0. Let $\mathcal{L} : \mathbb{R}^2 \longrightarrow \mathbb{R}^2$ be the orthogonal projection onto the bisector of K. Then

$$|\mathbb{E} f| \geqslant |\mathcal{L}(\mathbb{E} f)| = |\mathbb{E}\mathcal{L}(f)| = \mathbb{E}|\mathcal{L}(f)| \geqslant \mathbb{E}\left(|f| \cos \frac{\theta}{2}\right) = \left(\cos \frac{\theta}{2}\right) \mathbb{E}|f|.$$

Here the first (reading from left to right) inequality follows since the length of the orthogonal projection of a vector does not exceed the length of the vector; the next identity follows since $\mathcal{L}$

https://doi.org/10.1017/S0963548319000105 Published online by Cambridge University Press

is a linear operator; the next identity follows since for all $z \in K$ the vectors $\mathcal{L}(z)$ are non-negative multiples of each other; the next inequality follows since

$$|\mathcal{L}(z)| \geqslant \left(\cos \frac{\theta}{2}\right)|z| \quad \text{for all } z \in K;$$

and the final identity follows since the expectation is a linear operator. □

Proof of Theorem 3.1. For a function $f : \mathbb{T}^m \longrightarrow \mathbb{C}$ and a subset $I \subset \{1, \ldots, m\}$, we let $\mathbb{E}_I f$ denote the conditional expectation of f obtained by integrating f over the variables z_i with $i \in I$. Hence, if f is a function of $z_1, \ldots, z_m$ and $I \subset \{1, \ldots, m\}$, then $h_I = \mathbb{E}_I f$ is a function of z_i for $i \notin I$. In particular, $h_I = f$ if $I = \emptyset$ and $h_I = \mathbb{E}f$ if $I = \{1, \ldots, m\}$. If I consists of a single element i, we write $\mathbb{E}_i f$ instead of $\mathbb{E}_{\{i\}} f$. We let

$$\bar{I} = \{1, \ldots, m\} \setminus I$$

denote the complement of I. We will consider functions $f = e^p$ where $p : \mathbb{T}^m \longrightarrow \mathbb{C}$ is a Laurent polynomial.

For $0 \leqslant \theta_1, \ldots, \theta_m < 2\pi/3$, we let $\mathcal{P}_m(\theta_1, \ldots, \theta_m)$ denote the set of m-variate Laurent polynomials p for which the inequalities (3.2) hold. Note that the condition $p \in \mathcal{P}_m(\theta_1, \ldots, \theta_m)$ is a finite system of linear inequalities for $|\gamma_a|$, $a \in A$.

Let us choose $p \in \mathcal{P}_m(\theta_1, \ldots, \theta_m)$, let us fix some values $z_i \in \mathbb{S}^1$ for $I \subset \{1, \ldots, m\}$ and consider p as a function of z_i for $i \notin I$. It is not hard to see that $p \in \mathcal{P}_{m-|I|}(\theta_i : i \notin I)$.

We prove the following statements by induction on m.

Statement 1_m. For any $p \in \mathcal{P}_m(\theta_1, \ldots, \theta_m)$, we have $\mathbb{E}e^p \neq 0$. Moreover, suppose that $p, q \in \mathcal{P}_m(\theta_1, \ldots, \theta_m)$ are two Laurent polynomials that differ in at most one monomial, and the polynomial p is obtained from q by multiplying the coefficient γ_b of some $\mathbf{z}^b$ by some $\zeta \in \mathbb{S}^1$. Then the angle between $\mathbb{E}e^p \neq 0$ and $\mathbb{E}e^q \neq 0$ does not exceed

$$2|\gamma_b| \prod_{i \in \operatorname{supp} b} \frac{1}{\cos(\theta_i/2)}.$$

Statement 2_m. Let $p \in \mathcal{P}_m(\theta_1, \ldots, \theta_m)$ be a Laurent polynomial. Let $I = \{1, \ldots, m\} \setminus \{i\}$ for some $1 \leqslant i \leqslant m$ and let $h_I(z_i) = \mathbb{E}_I e^p$. Then for any $z_i', z_i'' \in \mathbb{S}^1$, we have $h_I(z_i') \neq 0$, $h_I(z_i'') \neq 0$ and the angle between the two complex numbers does not exceed θ_i.

We start by proving Statement 2_1. Then

$$p(z) = \sum_{a \in A} \gamma_a z^a \quad \text{for some finite } A \subset \mathbb{Z}$$

is a univariate Laurent polynomial. For any $z \in \mathbb{S}^1$, we have

$$|\arg e^{p(z)}| \leqslant |\Im\, p(z)| \leqslant |p(z)| \leqslant \sum_{a \in A} |\gamma_a| \leqslant \frac{1}{2}\theta_1$$

and the result is immediate.

Next, we prove that Statements 2_s for $s \leqslant m$ imply Statement 1_m.

Let us choose $p \in \mathcal{P}_m(\theta_1, \ldots, \theta_m)$. For a set $I \subset \{1, \ldots, m\}$, let

$$h_I(z_i : i \notin I) = \mathbb{E}_I e^p.$$

Assuming that $I \neq \{1, \ldots, m\}$, let us pick an $i \notin I$. Then

$$h_{I \cup \{i\}} = \mathbb{E}_i h_I.$$

Let us fix variables $z_j \in \mathbb{S}^1$ with $j \notin I \cup \{i\}$ arbitrarily and consider p as a Laurent polynomial from $\mathcal{P}_r(\theta_k : k \in I \cup \{i\})$ with $r = |I| + 1$. Thus h_I is a function of a single variable $z_i \in \mathbb{S}^1$ and by

https://doi.org/10.1017/S0963548319000105 Published online by Cambridge University Press

Statement 2_r for any two $z_i', z_i'' \in \mathbb{S}^1$, the angle between $h_I(z_i') \neq 0$ and $h_I(z_i'') \neq 0$ does not exceed θ_i. It follows from Lemma 3.3 that $h_{I\cup\{i\}}(z_j : j \notin I \cup \{i\}) \neq 0$ and, moreover,

$$|h_{I\cup\{i\}}| = |\mathbb{E}_i h_I| \geqslant \left(\cos\frac{\theta_i}{2}\right)\mathbb{E}_i|h_I| > 0.$$

Iterating, we obtain

$$|h_{I\cup J}| = |\mathbb{E}_J h_I| \geqslant \left(\prod_{j\in J}\cos\frac{\theta_j}{2}\right)\mathbb{E}_J|h_I| > 0 \quad \text{provided } J \cap I = \emptyset. \tag{3.3}$$

In particular, choosing $J = \bar{I}$, we obtain that $\mathbb{E}e^p \neq 0$.

Suppose now that $p, q \in \mathcal{P}_m(\theta_1, \ldots, \theta_m)$ where p is obtained from q by replacing a single monomial $\gamma_b \mathbf{z}^b$ by $\gamma_b \zeta \mathbf{z}^b$ for some $\zeta \in \mathbb{S}^1$. Let us fix all the remaining coefficients of p and q and consider $\mathbb{E}e^p$ as a function of the coefficient γ_b of $\mathbf{z}^b$ as long as the resulting polynomial remains in $\mathcal{P}_m(\theta_1, \ldots, \theta_m)$ (note that the set of admissible values of $|\gamma_b|$ is convex and includes 0). Since $\mathbb{E}e^p \neq 0$ for all $p \in \mathcal{P}_m(\theta_1, \ldots, \theta_m)$, we can choose a continuous branch of $\ln \mathbb{E}e^p$ as a function of γ_b. Then we have

$$\frac{\partial}{\partial \gamma_b}\ln \mathbb{E}e^p = \frac{(\partial/\partial\gamma_b)\mathbb{E}e^p}{\mathbb{E}e^p} = \frac{\mathbb{E}(\mathbf{z}^b e^p)}{\mathbb{E}e^p}.$$

Let $I = \operatorname{supp} b$. Then

$$|\mathbb{E}(\mathbf{z}^b e^p)| = |\mathbb{E}_I \mathbb{E}_{\bar{I}}(\mathbf{z}^b e^p)| = |\mathbb{E}_I(\mathbf{z}^b \mathbb{E}_{\bar{I}} e^p)| = |\mathbb{E}_I \mathbf{z}^b h_{\bar{I}}| \leqslant \mathbb{E}_I |h_{\bar{I}}|.$$

Similarly,

$$|\mathbb{E}e^p| = |\mathbb{E}_I \mathbb{E}_{\bar{I}} e^p| = |\mathbb{E}_I h_{\bar{I}}| \geqslant \left(\prod_{i\in I}\cos\frac{\theta_i}{2}\right)\mathbb{E}_I|h_{\bar{I}}| > 0$$

by (3.3). Therefore,

$$\left|\frac{\partial}{\partial\gamma_b}\ln\mathbb{E}e^p\right| \leqslant \prod_{i\in\operatorname{supp} b}\frac{1}{\cos(\theta_i/2)}$$

and hence

$$|\ln\mathbb{E}e^p - \ln\mathbb{E}e^q| \leqslant 2|\gamma_b|\prod_{i\in\operatorname{supp} b}\frac{1}{\cos(\theta_i/2)}.$$

Statement 1_m now follows.

Next, we prove that Statement 1_m implies Statement 2_{m+1}.

Let $p \in \mathcal{P}_{m+1}(\theta_1, \ldots, \theta_{m+1})$ be a polynomial and let us choose an $1 \leqslant i \leqslant m+1$. Let $I = \{1, \ldots, m+1\} \setminus \{i\}$ and let $h_I(z_i) = \mathbb{E}_I e^p$. If we fix z_i, we can consider p as a Laurent polynomial in $\mathcal{P}_m(\theta_j : j \neq i)$. Moreover, if we change the value of $z_i = z_i'$ to $z_i = z_i''$, then only the coefficients γ_a of p with $i \in \operatorname{supp} a$ are affected, and each of those coefficients gets multiplied by some $\zeta_a \in \mathbb{S}^1$. Repeatedly applying Statement 1_m, we conclude that $h_I(z_i') \neq 0$, $h_I(z_i'') \neq 0$ and the angle between the two complex numbers does not exceed

$$2\sum_{\substack{a\in A:\\ i\in\operatorname{supp} a}}|\gamma_a|\prod_{j\in\operatorname{supp} a}\frac{1}{\cos(\theta_j/2)},$$

which does not exceed θ_i by the definition of $\mathcal{P}_{m+1}(\theta_1, \ldots, \theta_{m+1})$, and Statement 2_{m+1} follows.

This concludes the induction and proves that $\mathbb{E}e^p \neq 0$. □

https://doi.org/10.1017/S0963548319000105 Published online by Cambridge University Press

Proof of Corollary 3.2. Let us choose

$$\theta_1 = \cdots = \theta_m = \frac{\delta}{\sqrt{c}}$$

for some $0 < \delta < 2\pi/3$ to be determined later. To have the conditions of Theorem 3.1 satisfied, it suffices to have

$$2 \sum_{\substack{a \in A: \\ i \in \operatorname{supp} a}} |\gamma_a| \left(\cos \frac{\delta}{2\sqrt{c}} \right)^{-c} \leqslant \frac{\delta}{\sqrt{c}} \quad \text{for } i = 1, \ldots, m.$$

Since

$$\left(\cos \frac{\delta}{2\sqrt{c}} \right)^{c} \geqslant \cos \frac{\delta}{2} \quad \text{for } 0 \leqslant \delta \leqslant \pi$$

(see [3]), it suffices to have

$$\sum_{\substack{a \in A: \\ i \in \operatorname{supp} a}} |\gamma_a| \leqslant \frac{\delta \cos(\delta/2)}{\sqrt{c}} \quad \text{for } i = 1, \ldots, m.$$

Optimizing over δ, we choose $\delta = 1.72$ and

$$\tau = \frac{\delta \cos(\delta/2)}{2} \approx 0.561,$$

which concludes the proof. □

4. Proofs of Theorems 1.1 and 1.2

First, we prove Theorem 1.1. Let $\mathbb{T}^m = \mathbb{S}^1 \times \cdots \times \mathbb{S}^1$ be the torus as in Section 3 and let us choose μ_i to be the rotation invariant (Haar) probability measure on the ith copy of $\mathbb{S}^1$. Let $\mu = \mu_1 \times \cdots \times \mu_m$ be the Haar probability measure on $\mathbb{T}^m$.

Lemma 4.1. *Let $X \subset \{0, 1\}^n$ be a set and let $w_1, \ldots, w_n$ be complex weights as in Theorem 1.1. Let a_j, $j = 1, \ldots, n$, be the columns of the matrix A, considered as integer m-vectors and let us define a Laurent polynomial $q : \mathbb{T}^m \longrightarrow \mathbb{C}$ by*

$$q(z_1, \ldots, z_m) = \prod_{j=1}^{n} (1 + w_j \mathbf{z}^{a_j}),$$

where

$$\mathbf{z}^a = z_1^{\alpha_1} \cdots z_m^{\alpha_m} \quad \textit{provided } a = (\alpha_1, \ldots, \alpha_m).$$

Then

$$w(X) = \mathbb{E} q.$$

Proof. Since for $a \in \mathbb{Z}^m$, we have

$$\mathbb{E}\mathbf{z}^a = \begin{cases} 1 & \text{if } a = 0, \\ 0 & \text{if } a \neq 0, \end{cases}$$

https://doi.org/10.1017/S0963548319000105 Published online by Cambridge University Press

expanding the product that defines q, we get

$$\mathbb{E}q = \sum_{\substack{\xi_1,\ldots,\xi_n \in \{0,1\}: \\ \xi_1 a_1 + \cdots + \xi_n a_n = 0}} w_1^{\xi_1} \cdots w_n^{\xi_n} = w(X). \qquad \square$$

Proof of Theorem 1.1. Let $q(z_1, \ldots, z_m)$ be the Laurent polynomial of Lemma 4.1, so that $w(X) = \mathbb{E}q$. Assuming that $|w_j| < 1$ for $j = 1, \ldots, n$, we write

$$\ln q = \sum_{j=1}^{n} \ln\left(1 + w_j \mathbf{z}^{a_j}\right) = \sum_{j=1}^{n} \sum_{k=1}^{\infty} (-1)^{k-1} \frac{w_j^k \mathbf{z}^{k a_j}}{k}.$$

For a positive integer N, let us define a Laurent polynomial

$$p_N(z_1, \ldots, z_m) = \sum_{j=1}^{n} \sum_{k=1}^{N} (-1)^{k-1} \frac{w_j^k \mathbf{z}^{k a_j}}{k},$$

which is just a truncation of the series expansion for $\ln q$. Let $\gamma_a \neq 0$ be the coefficient of the Laurent monomial $\mathbf{z}^a$ in p_N. Then

$$|\operatorname{supp} a| \leqslant c,$$

and for $i = 1, \ldots, m$ we have

$$\sum_{a: i \in \operatorname{supp} a} |\gamma_a| \leqslant \sum_{j: a_{ij} \neq 0} \sum_{k=1}^{N} \frac{|w_j|^k}{k} \leqslant r \max_{j=1,\ldots,n} -\ln\left(1 - |w_j|\right) \leqslant \frac{0.56}{\sqrt{c}}$$

as long as

$$|w_j| \leqslant \frac{0.46}{r\sqrt{c}} \quad \text{for } j = 1, \ldots, n. \tag{4.1}$$

(We use that $-\ln(1-x) \leqslant 1.2x$ for $0 \leqslant x \leqslant 0.3$ and that $r \geqslant 2$.) Therefore, by Corollary 3.2, $\mathbb{E}e^{p_N} \neq 0$ as long as (4.1) holds. On the other hand, $\mathbb{E}e^{p_N}$ is an analytic function of $w_1, \ldots, w_n$ in the polydisc (4.1) and $\mathbb{E}e^{p_N}$ converges to $\mathbb{E}q$ uniformly on compact subsets of the polydisc. By the Hurwitz theorem (see *e.g.* Section 7.5 of [17]), we have either $\mathbb{E}q \neq 0$ in the polydisc or $\mathbb{E}q \equiv 0$ in the polydisc. Since for $w_1 = \cdots = w_n = 0$, we have $\mathbb{E}q = 1$, we conclude that $\mathbb{E}q \neq 0$ provided (4.1) holds. □

Proof of Theorem 1.2. We modify the choice of the probability measure μ on $\mathbb{T}^m$ as follows: we choose μ_i to be the uniform probability measure on the roots of unity of degree κ and let $\mu = \mu_1 \times \cdots \times \mu_m$. We note that for $a \in \mathbb{Z}^m$, $a = (\alpha_1, \ldots, \alpha_m)$, we have

$$\mathbb{E}\mathbf{z}^a = \begin{cases} 1 & \text{if } \alpha_i \equiv 0 \mod \kappa \text{ for } i = 1, \ldots, m, \\ 0 & \text{otherwise.} \end{cases}$$

Given an $m \times n$ integer matrix $A = (a_{ij})$, we define $q(z_1, \ldots, z_m)$ by

$$q(z_1, \ldots, z_m) = \prod_{j=1}^{n} \left(1 + w_j \mathbf{z}^{a_j} + \cdots + w_j \mathbf{z}^{(\kappa-1)a_j}\right).$$

https://doi.org/10.1017/S0963548319000105 Published online by Cambridge University Press

Then

$$\mathbb{E}q = \sum_{\substack{\xi_1,\dots,\xi_n:\\ a_{i1}\xi_1+\cdots+\xi_n a_{in}\equiv 0 \mod \kappa\\ \text{for } i=1,\dots,m,\\ \xi_j\in\{0,1,\dots,\kappa-1\}\\ \text{for } j=1,\dots,n}} \prod_{j:\xi_j\neq 0} w_j = w(X).$$

Assuming that $|w_j| < (\kappa-1)^{-1}$ for $j=1,\dots,n$, we expand

$$\ln q = \sum_{j=1}^{n} \ln\left(1 + w_j \mathbf{z}^{a_j} + \cdots + w_j \mathbf{z}^{(\kappa-1)a_j}\right) = \sum_{j=1}^{n}\sum_{s=1}^{\infty}(-1)^s \frac{(w_j\mathbf{z}^{a_j}+\cdots+w_j\mathbf{z}^{(\kappa-1)a_j})^s}{s}.$$

For a positive integer N, let us define a Laurent polynomial

$$p_N(z_1,\dots,z_m) = \sum_{j=1}^{n}\sum_{s=1}^{N}(-1)^s \frac{(w_j\mathbf{z}^{a_j}+\cdots+w_j\mathbf{z}^{(\kappa-1)a_j})^s}{s}.$$

For every Laurent monomial $\mathbf{z}^a$ which appears in p_N with a coefficient $\gamma_a \neq 0$, we have $|\text{supp } a| \leqslant c$. If $i \in \text{supp } a$, then the coefficient of $\mathbf{z}^a$ in the polynomial $(\mathbf{z}^{a_j}+\cdots+\mathbf{z}^{(\kappa-1)a_j})^s$ is non-zero only if $a_{ij}\neq 0$. Hence for $i=1,\dots,m$, we have

$$\sum_{a:i\in\text{supp } a} |\gamma_a| \leqslant \sum_{j:a_{ij}\neq 0}\sum_{s=1}^{N} \frac{\big((\kappa-1)|w_j|\big)^s}{s} \leqslant r \max_{j=1,\dots,n} -\ln\big(1-(\kappa-1)|w_j|\big) \leqslant \frac{0.56}{\sqrt{c}}$$

provided

$$|w_j| \leqslant \frac{0.46}{(\kappa-1)r\sqrt{c}} \quad \text{for } j=1,\dots,n.$$

The proof is then concluded like the proof of Theorem 1.1. □

5. Approximating $w(X)$ faster

Let $X\subset\{0,1\}^n$ be the set defined in Theorem 1.1. We assume that the $m\times n$ matrix A has no zero rows or columns: see Section 1.2. Recall that $r\geqslant 2$ is an upper bound on the number of non-zero entries in a row of A and $c\geqslant 1$ is an upper bound on the number of non-zero entries in a column of A. As in Section 1.2, we define a univariate polynomial $w(X;\zeta)$, that is, the weight of the set X under the scaled weights $\zeta w_1,\dots,\zeta w_n$, so $w(X;\zeta)$ is a polynomial of some degree $d\leqslant n$. We let $f(\zeta)=\ln w(X;\zeta)$ for ζ in a neighbourhood of 0.

Our goal is to show that the term $f^{(k)}(0)$ in the Taylor expansion (1.4) can be computed in $n(rc)^{O(k)}$ time, where we assume the standard RAM machine model with logarithmic-sized words, and additionally we assume that given a column index j of the matrix $A=(a_{ij})$ we can in time $O(c)$ compute the row indices i such that $a_{ij}\neq 0$ (otherwise the running time is bounded by $nm(rc)^{O(k)}$). We note that in this section, all the implied constants in the 'O' notation are absolute. In particular, if $k=O(\ln n-\ln\epsilon)$ as in Section 1.2, and r and c are fixed beforehand, we obtain an algorithm of a polynomial in n/ϵ complexity.

Our algorithm relies heavily on the ideas of [21]; see also [19].

https://doi.org/10.1017/S0963548319000105 Published online by Cambridge University Press

5.1 The idea of the algorithm

Since $w(X;0)=1$, we can write

$$w(X;\zeta)=\prod_{i=1}^{d}\left(1-\frac{\zeta}{\zeta_i}\right),$$

where $\zeta_1,\ldots,\zeta_d\neq 0$ for some $d\leqslant n$ are the roots of $w(X;\zeta)$, listed with multiplicity. Then

$$f(\zeta)=\sum_{i=1}^{d}\ln\left(1-\frac{\zeta}{\zeta_i}\right)$$

and

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

We introduce the *power sums*

$$\sigma_k(A,w)=\zeta_1^{-k}+\cdots+\zeta_d^{-k}. \tag{5.1}$$

Hence our goal is to compute $\sigma_k(A,w)$ in $n(rc)^{O(k)}$ time.

The crucial feature of the power sums $\sigma_k(A,w)$ is that they are *additive functions* of A, as is explained below.

In what follows, we consider the set $\mathcal{M}$ of integer matrices A with rows and columns indexed by non-empty finite subsets of the set $\mathbb{N}$ of positive integers and without zero rows or columns. For non-empty finite subsets $R,C\subset\mathbb{N}$, an $R\times C$ integer-valued matrix $A\in\mathcal{M}$ is a function $A:R\times C\longrightarrow\mathbb{Z}$ and we write the (i,j)th entry of A as $A(i,j)$ for $i\in R$ and $j\in C$. We fix complex weights w_j and define

$$X_A=\left\{(\xi_j:j\in C)\in\{0,1\}^C:\ \sum_{j\in C}A(i,j)\xi_j=0\quad\text{for } i\in R\right\}. \tag{5.2}$$

Similarly, we define univariate polynomials

$$w(X_A;\zeta)=\sum_{\substack{x\in X_A\\ x=(\xi_j:j\in C)}}\prod_{j\in C}(\zeta w_j)^{\xi_j} \tag{5.3}$$

and define power sums $\sigma_k(A,w)$ by (5.1), where $\zeta_1,\ldots,\zeta_d$ are the roots of $w(X_A;\zeta)$, listed with multiplicity.

Let $A_1,A_2\in\mathcal{M}$ be respectively $R_1\times C_1$ and $R_2\times C_2$ matrices. Suppose that $R_1\cap R_2=\emptyset$ and $C_1\cap C_2=\emptyset$. We define the *direct sum* $A=A_1\oplus A_2$ as the $R\times C$ matrix, where $R=R_1\cup R_2$, $C=C_1\cup C_2$ and

$$A(i,j)=\begin{cases}A_1(i,j) & \text{if } i\in R_1 \text{ and } j\in C_1,\\ A_2(i,j) & \text{if } i\in R_2 \text{ and } j\in C_2,\\ 0 & \text{elsewhere.}\end{cases}$$

Clearly, $A\in\mathcal{M}$.

Let $A_1,A_2\in\mathcal{M}$ be matrices such that $A=A_1\oplus A_2$ is defined. We observe that

$$w(X_A;\zeta)=w(X_{A_1};\zeta)w(X_{A_2};\zeta)$$

and hence

$$\sigma_k(A_1\oplus A_2,w)=\sigma_k(A_1,w)+\sigma_k(A_2,w). \tag{5.4}$$

https://doi.org/10.1017/S0963548319000105 Published online by Cambridge University Press

Given an $R \times C$ matrix $A \in \mathcal{M}$ and an $R_1 \times C_1$ matrix $B \in \mathcal{M}$, we define the *index* $\operatorname{ind}(B, A) = 1$ if $R_1 \subset R$, $C_1 \subset C$,

$$A(i, j) = B(i, j) \quad \text{for all } i \in R_1 \text{ and all } j \in C_1$$

and

$$A(i, j) = 0 \quad \text{for all } i \in R \setminus R_1 \text{ and all } j \in C_1.$$

Otherwise, we say that $\operatorname{ind}(B, A) = 0$.

We define a filtration

$$\mathcal{M}_1 \subset \mathcal{M}_2 \subset \cdots \subset \mathcal{M}_k \subset \cdots,$$

where $\mathcal{M}_k \subset \mathcal{M}$ consists of the matrices with at most k columns.

In Lemma 5.2 below we show that we can write

$$\sigma_k(A; w) = \sum_{B \in \mathcal{M}_k} \operatorname{ind}(B, A)\mu_k(B, w) \quad \text{for all } A \in \mathcal{M} \tag{5.5}$$

and some complex numbers $\mu_k(B, w)$. Although the sum in (5.5) contains infinitely many terms, for each $A \in \mathcal{M}$, only finitely many terms are non-zero, so (5.5) is well-defined.

We say that a matrix $B \in \mathcal{M}$ is *connected* if it cannot be represented as a direct sum $B = B_1 \oplus B_2$ for some matrices $B_1, B_2 \in \mathcal{M}$ and *disconnected* otherwise. In Corollary 5.4 below, we deduce from the additivity property (5.4) that $\mu_k(B, w) = 0$ in (5.5) unless B is connected. In Section 5.2 we show for any given $m \times n$ matrix A with at most r non-zero entries in each row and at most c non-zero entries in each column, the number of connected matrices $B \in \mathcal{M}_k$ with $\operatorname{ind}(B, A) = 1$ is at most $n(rc)^{O(k)}$ and that all such matrices B can be found in $n(rc)^{O(k)}$ time. Finally, in Section 5.3 we show that for each connected $B \in \mathcal{M}_k$, one can compute $\mu_k(B, w)$ in $cn2^{O(k)}$ time. This produces an algorithm of $n(rc)^{O(k)}$ complexity for computing $\sigma_k(A, w)$.

Next, we supply the necessary details. We start with a technical result describing how the function $\operatorname{ind}(B, \cdot)$ behaves under multiplication. Let $B_1 \in \mathcal{M}$ be an $R_1 \times C_1$ matrix and let $B_2 \in \mathcal{M}$ be an $R_2 \times C_2$ matrix. If the restrictions of B_1 and B_2 onto $(R_1 \cap R_2) \times (C_1 \cap C_2)$ coincide, we define the *connected sum* $B = B_1 \# B_2$, $B \in \mathcal{M}$, as the $(R_1 \cup R_2) \times (C_1 \cup C_2)$ matrix such that

$$B(i, j) = \begin{cases} B_1(i, j) & \text{if } i \in R_1 \text{ and } j \in C_1, \\ B_2(i, j) & \text{if } i \in R_2 \text{ and } j \in C_2, \\ 0 & \text{otherwise.} \end{cases}$$

In particular, if $R_1 \cap R_2 = \emptyset$ and $C_1 \cap C_2 = \emptyset$, then $B_1 \# B_2 = B_1 \oplus B_2$ is the direct sum of B_1 and B_2.

Lemma 5.1. *Let $B_1 \in \mathcal{M}$ be an $R_1 \times C_1$ matrix and let $B_2 \in \mathcal{M}$ be an $R_2 \times C_2$ matrix.*

Suppose that the following conditions (1)–(3) *are satisfied.*

(1) *For all $i \in R_1 \cap R_2$ and all $j \in C_1 \cap C_2$ we have $B_1(i, j) = B_2(i, j)$.*
(2) *For all $i \in R_1 \setminus R_2$ and all $j \in C_1 \cap C_2$ we have $B_1(i, j) = 0$.*
(3) *For all $i \in R_2 \setminus R_1$ and all $j \in C_1 \cap C_2$ we have $B_2(i, j) = 0$.*

Then $B = B_1 \# B_2$ is defined and

$$\operatorname{ind}(B_1, A)\operatorname{ind}(B_2, A) = \operatorname{ind}(B, A) \quad \textit{for all } A \in \mathcal{M}.$$

If any of conditions (1)–(3) *are violated then*

$$\operatorname{ind}(B_1, A)\operatorname{ind}(B_2, A) = 0 \quad \textit{for all } A \in \mathcal{M}.$$

Proof. Clearly, if (1) is violated then $\operatorname{ind}(B_1, A)\operatorname{ind}(B_2, A) = 0$ for all $A \in \mathcal{M}$. Suppose that (2) is violated. We assume $R_1 \subset R$, for $\operatorname{ind}(B_1, A) = 0$ otherwise. If $\operatorname{ind}(B_2, A) = 1$ then $A(i, j) = 0$ for all

https://doi.org/10.1017/S0963548319000105 Published online by Cambridge University Press

$i \in R_1 \setminus R_2$ and all $j \in C_1 \cap C_2$ and hence $\operatorname{ind}(B_1, A) = 0$ so that $\operatorname{ind}(B_1, A)\operatorname{ind}(B_2, A) = 0$. Similarly, if (3) is violated then $\operatorname{ind}(B_1, A)\operatorname{ind}(B_2, A) = 0$ for all $A \in \mathcal{M}$.

Hence it remains to consider the case when (1)–(3) hold. Without loss of generality we assume that $R_1 \cup R_2$ is a subset of the rows of A and that $C_1 \cup C_2$ is a subset of the columns of A.

If $\operatorname{ind}(B_1, A) = 0$ for some $A \in \mathcal{M}$, then either $B_1(i, j) \neq A(i, j)$ for some $i \in R_1$ and some $j \in C_1$, or $A(i, j) \neq 0$ for some $i \notin R_1$ and some $j \in C_1$. In either case $\operatorname{ind}(B, A) = 0$. Similarly, if $\operatorname{ind}(B_2, A) = 0$ then $\operatorname{ind}(B, A) = 0$. If $\operatorname{ind}(B_1, A) = \operatorname{ind}(B_2, A) = 1$ then $B(i, j) = A(i, j)$ for all $i \in R_1 \cup R_2$ and all $j \in C_1 \cup C_2$, while $A(i, j) = 0$ for all $i \notin R_1 \cup R_2$ and all $j \in C_1 \cup C_2$ and hence $\operatorname{ind}(B, A) = 1$ as well. □

If the conditions (1)–(3) of Lemma 5.1 are satisfied, we say that the matrices B_1 and B_2 are *compatible*, denoted by $B_1 \sim B_2$. Now we are ready to prove the existence of a decomposition (5.5).

Lemma 5.2. *For a positive integer k and a matrix $B \in \mathcal{M}_k$, one can define complex numbers $\mu_k(B, w)$ so that (5.5) holds for all $A \in \mathcal{M}$.*

Proof. We write the polynomial (5.3) in the monomial basis. Assuming that A is an $R \times C$ matrix, we have

$$w(X_A;\zeta) = 1 + \sum_{k=1}^{n} \pi_k(A, w)\zeta^k,$$

where

$$\pi_k(A, w) = \sum_{\substack{x=(\xi_j : j \in C):\\ x \in X_A,\\ \sum_{j \in C} \xi_j = k}} \prod_{j \in C} w_j^{\xi_j},$$

where X_A is defined by (5.2).

We say that a set $S \subset C$ is the *support* of a vector $x \in X_A$, $x = (\xi_j : j \in C)$, provided $\xi_j \neq 0$ if and only if $j \in S$. Clearly, the support of a vector x contributing to $\pi_k(A, w)$ is a set $S \subset C$ satisfying $|S| \leqslant k$, and the vector $x_S = (\xi_j : j \in S)$ satisfies $A_S x_S = 0$, where A_S is the $R \times S$ matrix consisting of the columns of A with indices in S.

This allows us to write

$$\pi_k(A, w) = \sum_{B \in \mathcal{M}_k} \operatorname{ind}(B, A)\lambda_k(B, w), \tag{5.6}$$

where for $R_1 \times C_1$ matrix B we have

$$\lambda_k(B, w) = \sum_{\substack{x=(\xi_j : j \in C_1): x \in X_B,\\ \text{support of } x \text{ is } C_1,\\ \sum_{j \in C_1} \xi_j = k}} \prod_{j \in C_1} w_j^{\xi_j}. \tag{5.7}$$

Although formally the sum (5.6) is infinite, for each $A \in \mathcal{M}$ we have $\operatorname{ind}(B, A) \neq 0$ for only finitely many $B \in \mathcal{M}$, so (5.6) is well-defined.

We observe that

$$\pi_k(A, w) = (-1)^k e_k(\zeta_1^{-1}, \ldots, \zeta_d^{-1}),$$

where e_k is the kth elementary symmetric function and $\zeta_1, \ldots, \zeta_d$ are the roots of $w(X_A;\zeta)$, listed with their multiplicities (recall that the constant term of $w(X_A;\zeta)$ is 1). Therefore, the Newton identities imply that

$$k\pi_k(A, w) = -\sum_{i=1}^{k} \pi_{k-i}(A, w)\sigma_i(A, w) \quad \text{for all } k \geqslant 1, \tag{5.8}$$

https://doi.org/10.1017/S0963548319000105 Published online by Cambridge University Press

where we define

$$\pi_0(A, w) = 1.$$

We define

$$\mu_1(B, w) = -\lambda_1(B, w) \quad \text{for } B \in \mathcal{M}_1.$$

Assuming that $\mu_i(B, w)$ are defined for $B \in \mathcal{M}_i$ and $i = 1, \ldots, k-1$, for $k \geqslant 2$ we define for $B \in \mathcal{M}_k$

$$\mu_k(B, w) = -k\lambda_k(B, w) - \sum_{\substack{B_1 \in \mathcal{M}_{k-i}, B_2 \in \mathcal{M}_i \\ \text{for } 1 \leqslant i \leqslant k-1: \\ B_1 \sim B_2 \text{ and } B_1 \# B_2 = B}} \lambda_{k-i}(B_1, w)\mu_i(B_2, w). \tag{5.9}$$

Here the sum is taken over all distinct ordered pairs of compatible matrices (B_1, B_2) such that $B_1 \# B_2 = B$ (in particular, we may have $B_1 = B_2$). We observe that for each B the sum contains only finitely many terms, so $\mu_k(B, w)$ is well-defined. The identity (5.5) now follows from (5.6), (5.8) and Lemma 5.1. □

Our next goal is to show that in (5.5) we have $\mu_k(B, w) \neq 0$ only for connected matrices B. We start with a general structural result, very similar in spirit to Lemma 4.2 of [12]; see also [21].

Lemma 5.3. *Let us consider a function $f : \mathcal{M} \longrightarrow \mathbb{C}$ defined by*

$$f(A) = \sum_{B \in \mathcal{S}} \mu_B \text{ind}(B, A),$$

where $\mathcal{S} \subset \mathcal{M}$ is a (possibly infinite) set and $\mu_B \in \mathbb{C} \setminus \{0\}$ for all $B \in \mathcal{S}$ (for each $A \in \mathcal{M}$ only finitely many summands are non-zero, so f is well-defined). Suppose that

$$f(A_1 \oplus A_2) = f(A_1) + f(A_2)$$

for any two matrices $A_1, A_2 \in \mathcal{M}$ such that $A_1 \oplus A_2$ is defined. Then each $B \in \mathcal{S}$ is connected, that is, cannot be written as $B = B_1 \oplus B_2$ for some $B_1, B_2 \in \mathcal{M}$.

Proof. Seeking a contradiction, assume that there is a disconnected $B \in \mathcal{S}$. We observe that if $B \in \mathcal{M}$ is connected, then

$$\text{ind}(B, A_1 \oplus A_2) = \text{ind}(B, A_1) + \text{ind}(B, A_2)$$

for any two $A_1, A_2 \in \mathcal{M}$ such that $A_1 \oplus A_2$ is defined (since B is connected, we cannot have $\text{ind}(B, A_1) = \text{ind}(B, A_2) = 1$ provided $A_1 \oplus A_2$ is defined). Therefore, without loss of generality, we assume that all $B \in \mathcal{S}$ are disconnected. Let us choose a $D \in \mathcal{S}$ that has the smallest number of columns. Hence we have $D = D_1 \oplus D_2$ for some D_1 and D_2. Then

$$f(D_i) = \sum_{B \in \mathcal{S}} \mu_B \text{ind}(B, D_i) = 0 \quad \text{for } i = 1, 2,$$

since D_i has fewer columns than any matrix $B \in \mathcal{S}$. Therefore,

$$f(D) = f(D_1) + f(D_2) = 0.$$

On the other hand, $\text{ind}(B, D) = 0$ for all $B \in \mathcal{S} \setminus \{D\}$ and

$$f(D) = \mu_D \text{ind}(D, D) = \mu_D \neq 0,$$

which is a contradiction. □

Corollary 5.4. *In the expansion (5.5), we have $\mu_k(B, w) = 0$ whenever B is disconnected.*

Proof. Follows by (5.4) and Lemma 5.3. □

https://doi.org/10.1017/S0963548319000105 Published online by Cambridge University Press

5.2 Enumerating connected matrices

Given an integer $k \geqslant 1$ and an $R \times C$ matrix $A \in \mathcal{M}$ with at most r non-zero entries in each row and at most c non-zero entries in each column, we want to compile a list of all connected matrices $B \in \mathcal{M}_k$ such that $\operatorname{ind}(B, A) = 1$. First, we observe that an $R_1 \times C_1$ matrix $B \in \mathcal{M}$ such that $\operatorname{ind}(B, A) = 1$ is uniquely determined by its set of columns $C_1 \subset C$, since $R_1 \subset R$ is then the set of rows of A whose restriction onto C_1 are not zero.

We define a graph $G = (C, E)$. The vertices of G are the columns of A, and two vertices c_1 and c_2 span an edge of G if and only if there is a row of A with non-zero entries in columns c_1 and c_2. We note that the degree of each vertex of G does not exceed $d = rc$. To enumerate connected matrices $B \in \mathcal{M}_k$ such that $\operatorname{ind}(B, A) = 1$ is to enumerate sets of vertices of cardinality at most k in C that induce a connected subgraph of G. This is done as in [22]. The crucial observation is that as long as one vertex c is chosen, there are at most

$$k^{-1}\binom{kd}{k-1} \leqslant \frac{(ed)^{k-1}}{2}$$

connected induced subgraphs with $k \geqslant 2$ vertices containing c: see Lemma 2.1 of [7]. Consequently, there are $nd^{O(k)}$ induced connected subgraphs with at most k vertices in G. Once the vertex c is chosen, the subgraphs are enumerated with $d^{O(k)}$ complexity, by successively exploring adjacent vertices: see [22] for details.

5.3 Summary of the algorithm

Given an $m \times n$ matrix A without zero rows and columns, we interpret it as an $R \times C$ matrix $A \in \mathcal{M}$, where $R = \{1, \ldots, m\}$ and $C = \{1, \ldots, n\}$. Given a positive integer k, as in Section 5.2 we compile a list $\mathcal{C}$ of all connected matrices $B \in \mathcal{M}_k$ such that $\operatorname{ind}(B, A) = 1$. We define the filtration

$$\mathcal{C}_1 \subset \mathcal{C}_2 \subset \cdots \subset \mathcal{C}_{k-1} \subset \mathcal{C}_k = \mathcal{C},$$

where $\mathcal{C}_i$ is the set of matrices $B \in \mathcal{C}$ with at most i columns.

Given complex numbers $w_1, \ldots, w_n$, from Lemma 5.2 and (5.7) in particular, we obtain

$$\mu_1(B, w) = -\lambda_1(B, w) = 0 \quad \text{for } B \in \mathcal{C}_1,$$

since by our assumption B has no zero rows.

Suppose that we have computed $\mu_i(B, w)$ for $i = 1, \ldots, k-1$ and all $B \in \mathcal{C}_{k-1}$ for $k \geqslant 2$. To compute $\mu_k(B, w)$ for all $B \in \mathcal{C}_k$, we use formula (5.9). Since every matrix $B \in \mathcal{C}_k$ has at most k columns, there are no more than 4^k pairs of matrices $B_1 \in \mathcal{M}_{k-i}$ and $B_2 \in \mathcal{M}_i$ such that $B_1 \# B_2 = B$ and all such pairs can be found by inspection in $O(4^k)$ time. We then use (5.7) to compute the terms $\lambda_{k-i}(B_1, w)$ for $i = 0, \ldots, k-1$. We note that there are $\binom{k-i-1}{|S|-1} \leqslant 2^{k-i}$ non-negative integer vectors with support S and the sum $k - i$ of the coordinates, so each $\lambda_{k-i}(B_1, w)$ is computed in $c(k-i)n2^{O(k-i)}$ time.

This gives us the list of values $\mu_k(B, w)$ for all $B \in \mathcal{C}_k$. We then compute

$$\sigma_k(A, w) = \sum_{B \in \mathcal{C}_k} \mu_k(B, w),$$

as desired.

Acknowledgements

The authors are grateful to Martin Dyer for pointing out possible connections with [23] and [24], to Alex Samorodnitsky for pointing out possible connections with low-density parity-check codes, to Prasad Tetali for directing us to [5] and to Matthew Jenssen for telling us how to improve the

https://doi.org/10.1017/S0963548319000105 Published online by Cambridge University Press

degree dependence from $d_2 - d_1 \geqslant 2$ to $d_2 - d_1 \geqslant 1$ in Section 2.2. We thank Cai *et al.* [10] for pointing out that their construction implies that computing the partition function in the hard-core model at high fugacity is a #BIS-hard problem in the class of regular bipartite graphs. We thank the anonymous referee for catching an inaccuracy in one of our estimates.

References

[1] Ausiello, G., Crescenzi, P., Gambosi, G., Kann V., Marchetti-Spaccamela, A. and Protasi, M. (1999) *Complexity and Approximation: Combinatorial Optimization Problems and their Approximability Properties*, Springer.

[2] Barvinok, A. (2016) *Combinatorics and Complexity of Partition Functions*, Vol. 30 of Algorithms and Combinatorics, Springer.

[3] Barvinok, A. (2017) Computing the partition function of a polynomial on the Boolean cube. In *A Journey Through Discrete Mathematics* (M. Loebl *et al.*, eds), Springer, pp. 135–164.

[4] Barvinok, A. (2018) Computing permanents of complex diagonally dominant matrices and tensors. *Israel J. Math.*, to appear. arXiv:1801.04191

[5] van den Berg, J. and Steif, J. E. (1994) Percolation and the hard-core lattice gas model. *Stoch. Process. Appl.* **49** 179–197.

[6] Berlekamp, E. R, McEliece, R. J. and van Tilborg, H. C. A. (1978) On the inherent intractability of certain coding problems. *IEEE Trans. Inform. Theory* **24** 384–386.

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

[8] Bruck, J. and M. Naor, M. (1990) The hardness of decoding linear codes with preprocessing. *IEEE Trans. Inform. Theory* **36** 381–385.

[9] Bukh, B. (2015) Personal communication.

[10] Cai, J.-Y., Galanis, A., Goldberg, L. A., Guo, H., Jerrum, M., Štefankovič, D. and Vigoda, E. (2016) #BIS-hardness for 2-spin systems on bipartite bounded degree graphs in the tree non-uniqueness region. *J. Comput. System Sci.* **82** 690–711.

[11] Chen, S., Delcourt, M., Moitra, A., Perarnau, G. and Postle, L. (2019) Improved bounds for randomly sampling colorings via linear programming. In *Proc. Thirtieth Annual ACM–SIAM Symposium on Discrete Algorithms*, SIAM, pp. 2216–2234.

[12] Csikvári, P. and Frenkel, P. E. (2016) Benjamini–Schramm continuity of root moments of graph polynomials. *Europ. J. Combin.* **52** (part B), 302–320.

[13] Diestel, R. (2005) *Graph Theory*, third edition, Vol. 173 of Graduate Texts in Mathematics, Springer.

[14] Friedli, S. and Velenik, Y. (2018) *Statistical Mechanics of Lattice Systems: A Concrete Mathematical Introduction*, Cambridge University Press.

[15] Helmuth, T., Perkins, W. and Regts, G. (2018) Algorithmic Pirogov–Sinai theory. In *Proceedings of the 51st Annual ACM Symposium on the Theory of Computing (STOC 2019).*

[16] Jenssen, M., Keevash, P. and Perkins, W. (2019) Algorithms for #BIS-hard problems on expander graphs. In *Proc. Thirtieth Annual ACM–SIAM Symposium on Discrete Algorithms*, SIAM, pp. 2235–2247.

[17] Krantz, S. G. (1992) *Function Theory of Several Complex Variables*, second edition, Wadsworth & Brooks/Cole Mathematics Series, Wadsworth & Brooks/Cole.

[18] van Lint, J. H. (1999) *Introduction to Coding Theory*, third edition, Vol. 86 of Graduate Texts in Mathematics, Springer.

[19] Liu, J., Sinclair, A. and Srivastava, P. (2019) The Ising partition function: Zeros and deterministic approximation. *J. Statist. Phys.* **174**, 287–315.

[20] Mézard, M. and Montanari, A. (2009) *Information, Physics, and Computation*, Oxford Graduate Texts, Oxford University Press.

[21] Patel, V. and Regts, G. (2017) Deterministic polynomial-time approximation algorithms for partition functions and graph polynomials. *SIAM J. Comput.* **46** 1893–1919.

[22] Patel, V. and Regts, G. (2017) Computing the number of induced copies of a fixed graph in a bounded degree graph. *Algorithmica*. doi:10.1007/s00453-018-0511-9

[23] Valiant, L. G. (1979) The complexity of computing the permanent. *Theoret. Comput. Sci.* **8** 189–201.

[24] Valiant, L. G. and Vazirani, V. V. (1986) NP is as easy as detecting unique solutions. *Theoret. Comput. Sci.* **47** 85–93.

[25] Vigoda, E. (2000) Improved bounds for sampling colorings. *J. Math. Phys.* **41** 1555–1569.

Cite this article: Barvinok A and Regts G (2019). Weighted counting of solutions to sparse systems of equations. *Combinatorics, Probability and Computing*. **28**, 696–719. https://doi.org/10.1017/S0963548319000105