Symposium on Theoretical Aspects of Computer Science 2009 (Freiburg), pp. 493–504
www.stacs-conf.org

A COMPLEXITY DICHOTOMY FOR PARTITION FUNCTIONS WITH MIXED SIGNS

LESLIE ANN GOLDBERG[1] AND MARTIN GROHE[2] AND MARK JERRUM[3] AND MARC THURLEY[2]

[1] Department of Computer Science, University of Liverpool, Liverpool L69 3BX, UK

[2] Institut für Informatik, Humboldt-Universität zu Berlin, 10099 Berlin, Germany

[3] School of Mathematical Sciences, Queen Mary, University of London, Mile End Road, London E1 4NS, UK

Abstract. *Partition functions*, also known as *homomorphism functions*, form a rich family of graph invariants that contain combinatorial invariants such as the number of k-colourings or the number of independent sets of a graph and also the partition functions of certain "spin glass" models of statistical physics such as the Ising model.

Building on earlier work by Dyer and Greenhill [7] and Bulatov and Grohe [6], we completely classify the computational complexity of partition functions. Our main result is a dichotomy theorem stating that every partition function is either computable in polynomial time or #P-complete. Partition functions are described by symmetric matrices with real entries, and we prove that it is decidable in polynomial time in terms of the matrix whether a given partition function is in polynomial time or #P-complete.

While in general it is very complicated to give an explicit algebraic or combinatorial description of the tractable cases, for partition functions described by a Hadamard matrices — these turn out to be central in our proofs — we obtain a simple algebraic tractability criterion, which says that the tractable cases are those "representable" by a quadratic polynomial over the field $\mathbb{F}_2$.

1. Introduction

We study the complexity of a family of graph invariants known as *partition functions* or *homomorphism functions* (see, for example, [10, 17, 18]). Many natural graph invariants can be expressed as homomorphism functions, among them the number of k-colourings, the number of independent sets, and the number of nowhere-zero k-flows of a graph. The functions also appear as the partition functions of certain "spin-glass" models of statistical physics such as the Ising model or the q-state Potts model.

Key words and phrases: Computational complexity.

Research of Leslie Goldberg and Mark Jerrum was partly funded by the EPSRC grant "The complexity of counting in constraint satisfaction problems".
Marc Thurley was supported by the Deutsche Forschungsgemeinschaft within the research training group 'Methods for Discrete Structures' (GRK 1408).

STACS 2009
Symposium on Theoretical Aspects of Computer Science 2009 (Freiburg), pp. 493-504
http://drops.dagstuhl.de/opus/volltexte/2009/1821

Let $A \in \mathbb{R}^{m\times m}$ be a symmetric real matrix with entries $A_{i,j}$. The *partition function* Z_A associates with every graph $G = (V, E)$ the real number

$$Z_A(G) = \sum_{\xi:V\to[m]} \prod_{\{u,v\}\in E} A_{\xi(u),\xi(v)}.$$

We refer to the row and column indices of the matrix, which are elements of $[m] := \{1, \dots, m\}$, as *spins*. We use the term *configuration* to refer to a mapping $\xi : V \to [m]$ assigning a spin to each vertex of the graph. To avoid difficulties with models of real number computation, throughtout this paper we restrict our attention to algebraic numbers. Let $\mathbb{R}_{\mathbb{A}}$ denote the set of algebraic real numbers.[1]

Our main result is a dichotomy theorem stating that for every symmetric matrix $A \in \mathbb{R}_{\mathbb{A}}^{m\times m}$ the partition function Z_A is either computable in polynomial time or #P-hard. This extends earlier results by Dyer and Greenhill [7], who proved the dichotomy for 0-1-matrices, and Bulatov and Grohe [6], who proved it for nonnegative matrices. Therefore, in this paper we are mainly interested in matrices with negative entries.

Examples

In the following, let $G = (V, E)$ be a graph with N vertices. Consider the matrices

$$S = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix} \quad \text{and} \quad C_3 = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}.$$

It is not hard to see that $Z_S(G)$ is the number of independent sets of a graph G and $Z_{C_3}(G)$ is the number of 3-colourings of G. More generally, if A is the adjacency matrix of a graph H then $Z_A(G)$ is the number of homomorphisms from G to H. Here we allow H to have loops and parallel edges; the entry $A_{i,j}$ in the adjacency matrix is the number of edges from vertex i to vertex j.

Let us turn to matrices with negative entries. Consider

$$H_2 = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}. \tag{1.1}$$

Then $\frac{1}{2}Z_{H_2}(G)+2^{N-1}$ is the number of induced subgraphs of G with an even number of edges. Hence up to a simple transformation, Z_{H_2} counts induced subgraphs with an even number of edges. To see this, observe that for every configuration $\xi : V \to [2]$ the term $\prod_{\{u,v\}\in E} A_{\xi(u),\xi(v)}$ is 1 if the subgraph of G induced by $\xi^{-1}(2)$ has an even number of edges and -1 otherwise. Note that H_2 is the simplest nontrivial Hadamard matrix. Hadamard matrices will play a central role in this paper. Another simple example is the matrix

$$U = \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}.$$

It is a nice exercise to verify that for connected G the number $Z_U(G)$ is 2^N if G is Eulerian and 0 otherwise.

A less obvious example of a counting function that can be expressed in terms of a partition function is the number of nowhere-zero k-flows of a graph. It can be shown that the number of

[1] There is a problem with the treatment of real numbers in [6], but all results stated in [6] are valid for algebraic real numbers. We use a standard representation of algebraic numbers by polynomials and standard Turing machines as our underlying model of computation.

nowhere-zero k-flows of a graph G with N vertices is $k^{-N} \cdot Z_{F_k}(G)$, where F_k is the $k \times k$ matrix with $(k-1)$s on the diagonal and -1s everywhere else. This is a special case of a more general connection between partition functions for matrices A with diagonal entries d and off diagonal entries c and certain values of the Tutte polynomial. This well-known connection can be derived by establishing certain contraction-deletion identities for the partition functions. For example, it follows from [20, Equations (3.5.4)] and [19, Equation (2.26) and (2.9)]

Complexity

Like the complexity of graph polynomials [2, 12, 14, 16] and constraint satisfaction problems [1, 3, 4, 5, 8, 11, 13], which are both closely related to our partition functions, the complexity of partition functions has already received quite a bit of a attention. Dyer and Greenhill [7] studied the complexity of counting homomorphisms from a given graph G to a fixed graph H without parallel edges. (Homomorphisms from G to H are also known as *H-colourings* of G.) They proved that the problem is in polynomial time if every connected component of H is either a complete graph with a loop at every vertex or a complete bipartite graph, and the problem is #P-hard otherwise. Note that, in particular, this gives a complete classification of the complexity of computing Z_A for symmetric 0-1-matrices A. Bulatov and Grohe [6] extended this to symmetric nonnegative matrices. To state the result, it is convenient to introduce the notion of a *block* of a matrix A. To define the blocks of A, it is best to view A as the adjacency matrix of a graph with weighted edges; then each non-bipartite connected component of this graph corresponds to one block and each bipartite connected component corresponds to two blocks. A formal definition will be given below. Bulatov and Grohe [6] proved that computing the function Z_A is in polynomial time if the row rank of every block of A is 1 and $\#P$-hard otherwise. The problem for matrices with negative entries was left open. In particular, Bulatov and Grohe asked for the complexity of the partition function Z_{H_2} for the matrix H_2 introduced in (1.1). Note that H_2 is a matrix with one block of row rank 2. As we shall see, Z_{H_2} is computable in polynomial time. Hence the complexity classification of Bulatov and Grohe does not extend to matrices with negative entries. Nevertheless, we obtain a dichotomy, and this is our main result.

Results and outline of the proofs

Theorem 1.1 (Dichotomy Theorem). *Let $A \in \mathbb{R}_{\mathbb{A}}^{m \times m}$ be a symmetric matrix. Then the function Z_A either can be computed in polynomial time or is #P-hard.*

Furthermore, there is a polynomial time algorithm that, given the matrix A, decides whether Z_A is in polynomial time or #P-hard.

Let us call a matrix A *tractable* if Z_A can be computed in polynomial time and *hard* if computing Z_A is #P-hard. Then the Dichotomy Theorem states that every symmetric matrix with entries in $\mathbb{R}_{\mathbb{A}}$ is either tractable or hard. The classification of matrices into tractable and hard ones can be made explicit, but is very complicated and does not give any real insights. Very roughly, a matrix A is tractable if each of its blocks can be written as a tensor product of a positive matrix of row rank 1 and a tractable Hadamard matrix. Unfortunately, the real classification is not that simple, but for now let us focus on tractable Hadamard matrices. Recall that a Hadamard matrix is a square matrix H with entries from $\{-1, 1\}$ such that $H \cdot H^T$ is a diagonal matrix. Let $H \in \{-1, 1\}^{n \times n}$ be a symmetric $n \times n$ Hadamard matrix with $n = 2^k$. Let $\rho : \mathbb{F}_2^k \to [n]$ be a bijective mapping, which we call an *index mapping*. We say that a multivariate polynomial $h(X_1,\dots,X_k,Y_1,\dots,Y_k)$ over $\mathbb{F}_2$ *symmetrically represents H with respect to ρ* if, for all $\mathbf{x}=(x_1,\dots,x_k),\mathbf{y}=(y_1,\dots,y_k)\in\mathbb{F}_2^k$, it holds that

$$h(x_1,\dots,x_k,y_1,\dots,y_k)=1 \iff H_{\rho(\mathbf{x}),\rho(\mathbf{y})}=-1.$$

For example, the $\mathbb{F}_2$-polynomial $h_2(X_1,Y_1)=X_1\cdot Y_1$ symmetrically represents the matrix H_2 with respect to the index mapping $\rho(x_1)=x_1+1$. The $\mathbb{F}_2$-polynomial $h_4(X_1,X_2,Y_1,Y_2)=X_1\cdot Y_2\oplus X_2\cdot Y_1$ symmetrically represents the matrix

$$H_4=\begin{pmatrix}1&1&1&1\\1&1&-1&-1\\1&-1&1&-1\\1&-1&-1&1\end{pmatrix}$$

with respect to the index mapping $\rho(x_1,x_2)=2\cdot x_1+x_2+1$. The qualifier "symmetrically" in "symmetrically represents" indicates that the same index mapping is applied to both $\mathbf{x}$ and $\mathbf{y}$. We will need to consider asymmetric representations later. Note that we can only represent a matrix $H\in\{-1,1\}^{n\times n}$ by an $\mathbb{F}_2$-polynomial in this way if n is a power of 2. In this case, for every index mapping ρ there is a unique $\mathbb{F}_2$-polynomial symmetrically representing h with respect to ρ. We say that H has a *quadratic representation* if there is an index mapping ρ and an $\mathbb{F}_2$-polynomial h of degree at most 2 that symmetrically represents H with respect to ρ.

Theorem 1.2 (Complexity Classification for Hadamard Matrices)**.** *A symmetric Hadamard matrix H is tractable if it has a quadratic representation and hard otherwise.*

Hence, in particular, the matrices H_2 and H_4 are tractable. The tractability part of Theorem 1.2 is an easy consequence of the fact that counting the number of solutions of a quadratic equation over $\mathbb{F}_2$ (or any other finite field) is in polynomial time (see [9, 15]). The difficulty in proving the hardness part is that the degree of a polynomial representing a Hadamard matrix is not invariant under the choice of the index mapping ρ. However, for *normalised* Hadamard matrices, that is, Hadamard matrices whose first row and column consists entirely of +1s, we can show that either they are hard or they can be written as an iterated tensor product of the two simple Hadamard matrices H_2 and H_4. This gives us a canonical index mapping and hence a canonical representation by a quadratic $\mathbb{F}_2$-polynomial. Unfortunately, we could not find a direct reduction from arbitrary to normalised Hadamard matrices. To get a reduction, we first need to work with a generalisation of partition functions. If we view the matrix A defining a partition function as an edge-weighted graph, then this is the natural generalisation to graphs with edge and vertex weights. Let $A\in\mathbb{R}_{\mathbb{A}}^{m\times m}$ be a symmetric matrix and $D\in\mathbb{R}_{\mathbb{A}}^{m\times m}$ a diagonal matrix, which may be viewed as assigning the weight $D_{i,i}$ to each vertex i. We define the *partition function* $Z_{A,D}$ by

$$Z_{A,D}(G)=\sum_{\xi:V\to[m]}\prod_{\{u,v\}\in E}A_{\xi(u),\xi(v)}\cdot\prod_{v\in V}D_{\xi(v),\xi(v)},$$

for every graph $G=(V,E)$. As a matter of fact, we need a further generalisation that takes into account that vertices of even and odd degree behave differently when it comes to negative edge weights. For a symmetric matrix $A\in\mathbb{R}_{\mathbb{A}}^{m\times m}$ and two diagonal matrices $D,O\in\mathbb{R}_{\mathbb{A}}^{m\times m}$ we let

$$Z_{A,D,O}(G)=\sum_{\xi:V\to[m]}\prod_{\{u,v\}\in E}A_{\xi(u),\xi(v)}\cdot\prod_{\substack{v\in V\\ \deg(v)\text{ is even}}}D_{\xi(v),\xi(v)}\cdot\prod_{\substack{v\in V\\ \deg(v)\text{ is odd}}}O_{\xi(v),\xi(v)},$$

for every graph $G = (V, E)$. We call $Z_{A,D,O}$ the *parity-distinguishing partition function* (pdpf) defined by A, D, O. We show that the problem of computing $Z_{A,D,O}(G)$ is always either polynomial-time solvable or #P-hard, and we call a triple (A, D, O) *tractable* or *hard* accordingly. Obviously, if $D = O = I_m$ are identity matrices, then we have $Z_A = Z_{A,D} = Z_{A,D,O}$.

Returning to the proof of Theorem 1.2, we can show that, for every Hadamard matrix H, either H is hard or there is a normalised Hadamard matrix H' and diagonal matrices D', O' such that computing Z_H is polynomial time equivalent to computing $Z_{H',D',O'}$. Actually, we may assume D' to be an identity matrix and O' to be a diagonal matrix with entries $0, 1$ only. For the normalised matrix H' we have a canonical index mapping, and we can use this to represent the matrices D' and O' over $\mathbb{F}_2$. Then we obtain a tractability criterion that essentially says that (H', D', O') is tractable if the representation of H' is quadratic and that of O' is linear (remember that D' is an identity matrix, which we do not have to worry about).

For the proof of the Dichotomy Theorem 1.1, we actually need an extension of Theorem 1.2 that states a dichotomy for parity-distinguishing partition functions $Z_{A,D,O}$, where A is a "bipartisation" of a Hadamard matrix (this notion will be defined later). The proof sketched above can be generalised to give this extension. Then to prove the Dichotomy Theorem, we first reduce the problem of computing Z_A to the problem of computing Z_C for the connected components C of A. The next step is to eliminate duplicate rows and columns in the matrix, which can be done at the price of introducing vertex weights. Using the classification theorem for nonnegative matrices and some gadgetry, from there we get the desired reduction to parity-distinguishing partition functions for bipartisations of Hadamard matrices.

Let us finally mention that our proof shows that the Dichotomy Theorem not only holds for simple partition functions Z_A, but also for vertex-weighted and parity-distinguishing partition functions.

Preliminaries

Let $A \in \mathbb{R}_{\mathbb{A}}^{m\times n}$ be an $(m \times n)$-matrix. The entries of A are denoted by $A_{i,j}$. The ith row of A is denoted by $A_{i,*}$, and the jth column by $A_{*,j}$. By $\mathrm{abs}(A)$ we denote the matrix obtained from A by taking the absolute value of each entry in A.

Let I_m be the $m \times m$ identity matrix and let $I_{m;\Lambda}$ be the $m \times m$ matrix that is all zero except that $I_{j,j} = 1$ for $j \in \Lambda$.

The *Hadamard* product C of two $m \times n$ matrices A and B, written $C = A \circ B$, is the $m \times n$ component-wise product in which $C_{i,j} = A_{i,j}B_{i,j}$. $-A$ denotes the Hadamard product of A and the matrix in which every entry is -1.

We write $\langle u, v\rangle$ to denote the inner product (or dot product) of two vectors in $\mathbb{R}_{\mathbb{A}}^n$.

Recall that the *tensor product* (or *Kronecker product*) of an $r \times s$ matrix B and an $t \times u$ matrix C is an $rt \times su$ matrix $B \otimes C$. For $k \in [r]$, $i \in [t]$, $\ell \in [s]$ and $j \in [u]$, we have $(B \otimes C)_{(k-1)t+i,(\ell-1)u+j} = B_{k,\ell}C_{i,j}$. It is sometimes useful to think of the product in terms of rs "blocks" or "tiles" of size $t \times u$.

$$B \otimes C = \begin{pmatrix} B_{11}C & \dots & B_{1s}C \\ \vdots & \ddots & \vdots \\ B_{r1}C & \dots & B_{rs}C \end{pmatrix}$$

For index sets $I \subseteq [m], J \subseteq [n]$, we let $A_{I,J}$ be the $(|I|\times|J|)$-*submatrix* with entries $A_{i,j}$ for $i \in I$, $j \in J$. The matrix A is *indecomposable* if there are no index sets $I \subseteq [m], J \subseteq [n]$ such that $(I,J) \neq (\emptyset,\emptyset)$, $(I,J) \neq ([m],[n])$ and $A_{i,j} = 0$ for all $(i,j) \in \big(([m]\setminus I)\times J\big) \cup \big(I \times ([n]\setminus J)\big)$. Note that, in particular, an indecomposable matrix has at least one nonzero entry. The *blocks* of a matrix are the maximal indecomposable submatrices. For every symmetric matrix $A \in \mathbb{R}^{n\times n}$ we can define a graph G with vertex set $[n]$ and edge set $\big\{\{i,j\} \bigm| A_{i,j} \neq 0\big\}$. We call the matrix A *bipartite* if the graph G is bipartite. We call A *connected* if the graph G is connected. The *connected components* of A are the maximal submatrices $A_{C,C}$ such that $G[C]$, the subgraph of G induced by $C \subseteq [n]$, is a connected component. If the connected component $G[C]$ is not bipartite then $A_{C,C}$ is a block of A. If the connected component $G[C]$ is bipartite and contains an edge then $A_{C,C}$ has the form $\begin{pmatrix} 0 & B \\ B^T & 0 \end{pmatrix}$, where B is a block of A. Furthermore, all blocks of A arise from connected components in this way.

For two Counting Problems f and g, we write $f \leq g$ if there is a polynomial time Turing reduction from f to g. If $f \leq g$ and $g \leq f$ holds, we write $f \equiv g$. For a symmetric matrix A and diagonal matrices D, O of the same size, EVAL(A,D,O) (EVAL(A,D), EVAL(A)) denotes the problem of computing $Z_{A,D,O}(G)$ ($Z_{A,D}(G)$, $Z_A(G)$, respectively) for an input graph G (which need not be a simple graph - it may have loops and/or multi-edges).

2. Hadamard matrices

The main focus of this section is to prove Theorem 2.2 below which is a strengthened version of Theorem 1.2. Suppose that H is an $n \times n$ Hadamard matrix and that Λ^R and Λ^C are subsets of $[n]$. It will be useful to work with the *bipartisation* M, Λ of H, Λ^R and Λ^C which we define as follows. Let $m = 2n$ and let M be the $m \times m$ matrix defined by the following equations for $i,j \in [n]$: $M_{i,j} = 0$, $M_{i,n+j} = H_{i,j}$, $M_{n+i,j} = H_{j,i}$, and $M_{n+i,n+j} = 0$. The matrix M can be broken into four "tiles" as follows.

$$M = \begin{pmatrix} 0 & H \\ H^T & 0 \end{pmatrix}.$$

Let $\Lambda = \Lambda^R \cup \{n + j \mid j \in \Lambda^C\}$. Note that the matrix $I_{m;\Lambda}$ can be decomposed naturally in terms of the tiles $I_{n;\Lambda^R}$ and $I_{n;\Lambda^C}$.

$$I_{m;\Lambda} = \begin{pmatrix} I_{n;\Lambda^R} & 0 \\ 0 & I_{n;\Lambda^C} \end{pmatrix}.$$

We identify a set of conditions on H, Λ^R and Λ^C that determine whether or not the problem EVAL$(M, I_m, I_{m;\Lambda})$ can be computed in polynomial time. We will see how this implies Theorem 1.2.

The Group Condition. For an $n \times n$ matrix H and a row index $l \in [n]$, let

$$G(H,l) := \{H_{i,*} \circ H_{l,*} \mid i \in [n]\} \cup \{-H_{i,*} \circ H_{l,*} \mid i \in [n]\}.$$

The *group condition for H* is:

(GC) For all $l \in [n]$, both $G(H,l) = G(H,1)$ and $G(H^T,l) = G(H^T,1)$.

The group condition gets its name from the fact that the condition implies that $G(H,l)$ is an Abelian group . As all elements of this group have order 2, the group condition gives us some information about the order of such matrices:

Lemma 2.1. *Let H be an $n \times n$ Hadamard matrix. If H satisfies* **(GC)** *then $n = 2^k$ for some integer k.*

The Representability Conditions. We describe Hadamard matrices H satisfying **(GC)** by $\mathbb{F}_2$-polynomials. By Lemma 2.1 these matrices have order $n = 2^k$. We extend our notion of "symmetric representation": Let $\rho^R : \mathbb{F}_2^k \to [n]$ and $\rho^C : \mathbb{F}_2^k \to [n]$ be index mappings (i.e. bijective mappings) and $X = (X_1, \dots, X_k)$ and $Y = (Y_1, \dots, Y_k)$. A polynomial $h(X, Y)$ over $\mathbb{F}_2$ *represents H with respect to ρ^R and ρ^C* if for all $\mathbf{x}, \mathbf{y} \in \mathbb{F}_2^k$ it holds that

$$h(\mathbf{x}, \mathbf{y}) = 1 \iff H_{\rho^R(\mathbf{x}), \rho^C(\mathbf{y})} = -1.$$

So a symmetric representation is just a representation with $\rho^R = \rho^C$. We say that the set Λ^R is *linear with respect to* ρ^R if there is a linear subvectorspace $L^R \subseteq \mathbb{F}_2^k$ a such that $\rho^R(L^R) = \Lambda^R$. Note that, if Λ^R is linear, then $|\Lambda^R| = 2^l$ for some $l \leq k$. We may therefore define a *coordinatisation of* Λ^R *(with respect to* ρ^R*)* as a linear map $\phi^R : \mathbb{F}_2^l \to \mathbb{F}_2^k$ such that $\phi^R(\mathbb{F}_2^l) = L^R$, that is Λ^R is just the image of the concatenated mapping $\rho^R \circ \phi^R$. We define the notion of linearity of Λ^C with respect to ρ^C and the coordinatisation of Λ^C with respect to ρ^C similarly. For a permutation $\pi \in S_k$ we use the shorthand $X_\pi \cdot Y := \bigoplus_{i=1}^k X_{\pi(i)} \cdot Y_i$.

The following conditions stipulate the representability **(R)** of H by $\mathbb{F}_2$-polynomials, the linearity **(L)** of the sets Λ^R and Λ^C, and the appropriate degree restrictions on the associated polynomials **(D)**.

(R) There are index mappings $\rho^R : \mathbb{F}_2^k \to [n]$ and $\rho^C : \mathbb{F}_2^k \to [n]$ and a permutation $\pi \in S_k$ such that (w.r.t. ρ^R and ρ^C) the matrix H is represented by a polynomial of the form

$$h(X, Y) = X_\pi \cdot Y \oplus g^R(X) \oplus g^C(Y). \tag{2.1}$$

Moreover, if Λ^R is non-empty, then $\rho^R(0) \in \Lambda^R$. Similarly, if Λ^C is non-empty, then $\rho^C(0) \in \Lambda^C$.

Finally, if H is symmetric and $\Lambda^R = \Lambda^C$, then $g^R = g^C$ and $\rho^R = \rho^C$.

(L) Λ^R and Λ^C are linear with respect to ρ^R and ρ^C respectively.

(D) Either Λ^R is empty or there is a coordinatisation ϕ^R of Λ^R w.r.t ρ^R such that the polynomial $g^R \circ \phi^R$ has degree at most 2. Similarly, either Λ^C is empty or there is a coordinatisation ϕ^C of Λ^C w.r.t ρ^C such that the polynomial $g^C \circ \phi^C$ has degree at most 2. Finally, if H is symmetric and $\Lambda^R = \Lambda^C$ is nonempty then $\phi^R = \phi^C$.

Actually, it turns out that condition (D) is invariant under the choice of the coordinatisations ϕ^R, ϕ^C. However, the conditions are not invariant under the choice of the representation ρ^R, ρ^C, and this is a major source of technical problems.

Before we can apply the conditions **(R)**, **(L)** and **(D)** we deal with one technical issue. Let H be an $n \times n$ Hadamard matrix and let $\Lambda^R, \Lambda^C \subseteq [n]$ be subsets of indices. Let M, Λ be the bipartisation of H, Λ^R and Λ^C. We say that H is *positive* for Λ^R and Λ^C if there is an entry $H_{i,j} = +1$ such that (1) $i \in \Lambda^R$ or $\Lambda^R = \emptyset$, (2) $j \in \Lambda^C$ or $\Lambda^C = \emptyset$, and (3) If H is symmetric and $\Lambda^R = \Lambda^C$ then $i = j$. Otherwise, note that $-H$ is positive for Λ^R and Λ^C. Since $Z_{M,I_m,I_{m;\Lambda}}(G) = (-1)^{|E(G)|} Z_{-M,I_m,I_{m;\Lambda}}(G)$, the problems $\mathrm{EVAL}(M, I_m, I_{m;\Lambda})$ and $\mathrm{EVAL}(-M, I_m, I_{m;\Lambda})$ have equivalent complexity, so we lose no generality by restricting attention to the positive case, which is helpful for a technical reason.

Theorem 2.2. *Let H be an $n \times n$ Hadamard matrix and let $\Lambda^R, \Lambda^C \subseteq [n]$ be subsets of indices. Let M, Λ be the bipartisation of H, Λ^R and Λ^C and let $m = 2n$. If H is positive for Λ^R and Λ^C then* $\mathrm{EVAL}(M, I_m, I_{m;\Lambda})$ *is polynomial-time computable if, and only if, H Λ^R and Λ^C satisfy the group condition* **(GC)** *and conditions* **(R)***,* **(L)***, and* **(D)***. Otherwise* $\mathrm{EVAL}(M, I_m, I_{m;\Lambda})$ *is* #P*-hard. If H is not positive for Λ^R and Λ^C then* $\mathrm{EVAL}(M, I_m, I_{m;\Lambda})$ *is polynomial-time* *computable if, and only if,* $-H$ Λ^R *and* Λ^C *satisfy the group condition* **(GC)** *and conditions* **(R)***,* **(L)***, and* **(D)***. Otherwise* $\mathrm{EVAL}(M, I_m, I_{m;\Lambda})$ *is* #P*-hard. There is a polynomial-time algorithm that takes input* H*,* Λ^R *and* Λ^C *and decides whether* $\mathrm{EVAL}(M, I_m, I_{m;\Lambda})$ *is polynomial-time computable or* #P*-hard.*

The theorem is proved using a sequence of lemmas.

Lemma 2.3 (Group Condition Lemma)**.** *Let* H *be an* $n \times n$ *Hadamard matrix and let* $\Lambda^R, \Lambda^C \subseteq [n]$ *be subsets of indices. Let* M, Λ *be the bipartisation of* H*,* Λ^R *and* Λ^C *and let* $m = 2n$*. If* H *does not satisfy* **(GC)** *then* $\mathrm{EVAL}(M, I_m, I_{m;\Lambda})$ *is* #P*-hard. There is a polynomial-time algorithm that takes determines whether* H *satisfies* **(GC)***.*

Proof sketch. For any integer p and a symmetric non-negative matrix $C^{[p]}$, which depends upon H, the proof uses gadgetry to transform an input to $\mathrm{EVAL}(C^{[p]})$ into an input to $\mathrm{EVAL}(M, I_m, I_{m;\Lambda})$. The fact that H does not satisfy **(GC)** is used to show that, as long as p is sufficiently large with respect to M, then $C^{[p]}$ has a block of rank greater than one. By a result of Bulatov and Grohe, $\mathrm{EVAL}(C^{[p]})$ is #P-hard, so $\mathrm{EVAL}(M, I_m, I_{m;\Lambda})$ is #P-hard.

Lemma 2.4 (Polynomial Representation Lemma)**.** *Let* H *be an* $n \times n$ *Hadamard matrix and* $\Lambda^R, \Lambda^C \subseteq [n]$ *subsets of indices. Suppose that* H *satisfies* **(GC)** *and that* H *is positive for* Λ^R *and* Λ^C*. Then the Representability Condition* **(R)** *is satisfied. There is a polynomial-time algorithm that computes the representation.*

Proof sketch. The representation is constructed inductively. First, permutations are used to transform H into a normalised matrix $\hat{H}$, that is, a Hadamard matrix $\hat{H}$ whose first row and column consist entirely of +1s, which still satisfies **(GC)**. We then show that there is a permutation of $\hat{H}$ which can be expressed as the tensor product of a simple Hadamard matrix (either H_2 or H_4) and a smaller normalised symmetric Hadamard matrix H'. By induction, we construct a representation for H' and use this to construct a representation for the normalised matrix $\hat{H}$ of the form $X_\pi \cdot Y$ for a permutation $\pi \in S_k$. We use this to construct a representation for H.

Lemma 2.5 (Linearity Lemma)**.** *Let* H *be an* $n \times n$ *Hadamard matrix and* $\Lambda^R, \Lambda^C \subseteq [n]$ *subsets of indices. Let* M, Λ *be the bipartisation of* H*,* Λ^R *and* Λ^C *and let* $m = 2n$*. Suppose that* **(GC)** *and* **(R)** *are satisfied. Then the problem* $\mathrm{EVAL}(M, I_m, I_{m;\Lambda})$ *is* #P*-hard unless the Linearity condition* **(L)** *holds. There is a polynomial-time algorithm that determines whether* **(L)** *holds.*

Proof sketch. For a symmetric non-negative matrix C, which depends upon H, the proof uses gadgetry to transform an input to $\mathrm{EVAL}(C, I_m, I_{m;\Lambda})$ to an input of $\mathrm{EVAL}(M, I_m, I_{m;\Lambda})$. By **(R)**, there are bijective index mappings $\rho^R : \mathbb{F}_2^k \to [n]$ and $\rho^C : \mathbb{F}_2^k \to [n]$ and a permutation $\pi \in S_k$ such that (w.r.t. ρ^R and ρ^C) the matrix H is represented by a polynomial of the appropriate form. Let τ^R be the inverse of ρ^R and τ^C be the inverse of ρ^C. Let $L^C = \tau^C(\Lambda^C)$ and $L^R = \tau^R(\Lambda^R)$. We show that either $\mathrm{EVAL}(C, I_m, I_{m;\Lambda})$ is #P-hard or **(L)** is satisfied. In particular, the assumption that $\mathrm{EVAL}(C, I_m, I_{m;\Lambda})$ is not #P-hard means that its blocks all have rank 1 by the result of Bulatov and Grohe. We use this fact to show that L^R and L^C are linear subspaces of $\mathbb{F}_2^k$. To show that L^R is a linear space of $\mathbb{F}_2^k$, we use L^R to construct an appropriate linear subspace and compare Fourier coefficients to see that it is in fact L^R itself.

Lemma 2.6 (Degree Lemma)**.** *Let H be an $n \times n$ Hadamard matrix and $\Lambda^R, \Lambda^C \subseteq [n]$ subsets of indices. Let M, Λ be the bipartisation of H, Λ^R and Λ^C and let $m = 2n$. Suppose that* **(GC)**,**(R)** *and* **(L)** *are satisfied. Then* $\text{EVAL}(M, I_m, I_{m;\Lambda})$ *is* #P*-hard unless the Degree Condition* **(D)** *holds. There is a polynomial-time algorithm that determines whether* **(D)** *holds.*

Proof sketch. For any (even) integer p and a symmetric non-negative matrix $C^{[p]}$, which depends upon H, the proof uses gadgetry to transform an input to $\text{EVAL}(C^{[p]})$ into an input to $\text{EVAL}(M, I_m, I_{m;\Lambda})$. Using the representation of H, a coordinatisation ϕ^R with respect to Λ^R, and a coordinatisation ϕ^C with respect to Λ^C, some of the entries $C^{[p]}_{a,b}$ of the matrix $C^{[p]}$ may be expressed as sums, over elements in $\mathbb{F}_2^\ell$, for some ℓ, of appropriate powers of -1. We study properties of polynomials $g(X_1, \ldots, X_k) \in \mathbb{F}_2[X_1, \ldots, X_k]$, discovering that the number of roots of a certain polynomial $g_{\alpha,\beta,\gamma}(X_1, \ldots, X_k)$, which is derived from $g(X_1, \ldots, X_k)$, depends upon the degree of g. From this we can show that if **(D)** does not hold then there is an even p such that $\text{EVAL}(C^{[p]})$ is #P-hard.

Proof of Theorem 2.2. By the equivalence of the problems $\text{EVAL}(M, I_m, I_{m;\Lambda})$ and $\text{EVAL}(-M, I_m, I_{m;\Lambda})$ we can assume that H is positive for Λ^R and Λ^C. The hardness part follows directly from the Lemmas above. We shall give the proof for the tractability part. Given H, Λ^R and Λ^C satisfying **(GC)**, **(R)**, **(L)** and **(D)**, we shall show how to compute $Z_{M,I_m,I_{m;\Lambda}}(G)$ for an input graph G in polynomial time.

Note first that $Z_{M,I_m,I_{m;\Lambda}}(G) = 0$ unless G is bipartite. If G has connected components $G_1, \ldots G_c$, then

$$Z_{M,I_m,I_{m;\Lambda}}(G) = \prod_{i=1}^{c} Z_{M,I_m,I_{m;\Lambda}}(G_i).$$

Therefore, it suffices to give the proof for connected bipartite graphs. Let $G = (V, E)$ be such a graph with vertex bipartition $U \dot{\cup} W = V$. Let $V_o \subseteq V$ be the set of odd-degree vertices in G and let $U_o = W \cap V_o$ and $W_o = W \cap V_o$ be the corresponding subsets of U and W. Let $U_e = U \setminus U_o$ and $W_e = W \setminus W_o$. We have

$$Z_{M,I_m,I_{m;\Lambda}}(G) \;=\; \sum_{\xi:V\to[m]} \prod_{\{u,w\}\in E} M_{\xi(u),\xi(w)} \prod_{v\in V_o} (I_{m;\Lambda})_{\xi(v),\xi(v)} = \sum_{\substack{\xi:V\to[m]\\ \xi(V_o)\subseteq\Lambda}} \prod_{\{u,w\}\in E} M_{\xi(u),\xi(w)}.$$

As G is bipartite and connected this sum splits into $Z_{M,I_m,I_{m;\Lambda}}(G) = Z^{\to} + Z^{\leftarrow}$ for values

$$Z^{\to} = \sum_{\substack{\xi:U\to[n]\\ \xi(U_o)\subseteq\Lambda^R}} \sum_{\substack{\zeta:W\to[n]\\ \zeta(W_o)\subseteq\Lambda^C}} \prod_{\substack{\{u,w\}\in E\\ u\in U}} H_{\xi(u),\zeta(w)} \quad \text{and} \quad Z^{\leftarrow} = \sum_{\substack{\xi:U\to[n]\\ \xi(U_o)\subseteq\Lambda^C}} \sum_{\substack{\zeta:W\to[n]\\ \zeta(W_o)\subseteq\Lambda^R}} \prod_{\substack{\{u,w\}\in E\\ u\in U}} H_{\zeta(w),\xi(u)}$$

We will show how to compute $Z^{\to}$. The computation of the value $Z^{\leftarrow}$ is similar.

Fix configurations $\xi : U \to [n]$ and $\zeta : W \to [n]$ and let ρ^R, ρ^C be the index mappings and h the $\mathbb{F}_2$-polynomial representing H as given in condition **(R)**. Let τ^R be the inverse of ρ^R and let τ^C be the inverse of ρ^C. Let $L^R = \tau^R(\Lambda^R)$ and $L^C = \tau^C(\Lambda^C)$. Then ξ and ζ induce a configuration $\varsigma : V \to \mathbb{F}_2^k$ defined by

$$\varsigma(v) := \begin{cases} \tau^R(\xi(v)) & , \text{ if } v \in U \\ \tau^C(\zeta(v)) & , \text{ if } v \in W \end{cases}$$

which implies, for all $u \in U, w \in W$ that $h(\varsigma(u), \varsigma(w)) = 1$ iff $H_{\xi(u),\zeta(w)} = -1$. Let ϕ^R and ϕ^C be coordinatisations of Λ^R and Λ^C w.r.t. ρ^R and ρ^C satisfying **(L)** and **(D)**. We can simplify

$$\begin{aligned} Z^{\rightarrow} &= \sum_{\substack{\xi:U\to[n]\\ \xi(U_o)\subseteq\Lambda^R}} \sum_{\substack{\zeta:W\to[n]\\ \zeta(W_o)\subseteq\Lambda^C}} \prod_{\substack{\{u,w\}\in E\\ u\in U}} (-1)^{h(\tau^R(\xi(u)),\tau^C(\zeta(w)))} \\ &= \sum_{\substack{\varsigma:V\to\mathbb{F}_2^k\\ \varsigma(U_o)\subseteq L^R\\ \varsigma(W_o)\subseteq L^C}} (-1)^{\bigoplus_{\{u,w\}\in E:u\in U} h(\varsigma(u),\varsigma(w))} \end{aligned}$$

Define, for $a \in \mathbb{F}_2$, sets

$$s_a := \left| \left\{ \varsigma : V \to \mathbb{F}_2^k \mid \varsigma(U_o) \subseteq L^R,\ \varsigma(W_o) \subseteq L^C,\ \bigoplus_{\substack{\{u,w\}\in E\\ u\in U}} h(\varsigma(u), \varsigma(w)) = a \right\} \right|. \tag{2.2}$$

Then $Z^{\rightarrow} = s_0 - s_1$. Therefore, it remains to show how to compute the values s_a. Define, for each $v \in V$, a tuple $X^v = (X_1^v, \ldots, X_k^v)$ and let h_G be the $\mathbb{F}_2$-polynomial

$$h_G := \bigoplus_{\substack{\{u,w\}\in E\\ u\in U}} h(X^u, X^w) = \bigoplus_{\substack{\{u,w\}\in E\\ u\in U}} (X^u)_\pi \cdot X^w \oplus \bigoplus_{u\in U_o} g^R(X^u) \oplus \bigoplus_{w\in W_o} g^C(X^w). \tag{2.3}$$

Here the second equality follows from the definition of the polynomial h given in condition **(R)** and the fact that the terms $g^R(X^u)$ and $g^C(X^w)$ in the definition of h appear exactly $\deg(u)$ and $\deg(w)$ many times in h_G. Therefore, these terms cancel for all even degree vertices.

Let $\mathrm{var}(h_G)$ denote the set of variables in h_G and for mappings $\chi : \mathrm{var}(h_G) \to \mathbb{F}_2$ we use the expression $\chi(X^v) := (\chi(X_1^v), \ldots, \chi(X_k^v))$ as a shorthand and define the $\mathbb{F}_2$-sum $h_G(\chi) := \bigoplus_{\{u,w\}\in E:u\in U} h(\chi(X^u), \chi(X^w))$. We find that s_a can be expressed by

$$s_a = \left| \left\{ \chi : \mathrm{var}(h_G) \to \mathbb{F}_2 \mid \begin{array}{ll} \chi(X^u) \in L^R & \text{for all } u \in U_o, \\ \chi(X^w) \in L^C & \text{for all } w \in W_o, \end{array} h(\chi) = a) \right\} \right| \tag{2.4}$$

By equation (2.4) we are interested only in those assignments χ of the variables of h_G which satisfy $\chi(X^u) \in L^R$ and $\chi(X^w) \in L^C$ for all $u \in U_o$ and $w \in W_o$. With $|\Lambda^R| = 2^{\ell^R}$ and $|\Lambda^C| = 2^{\ell^C}$ for some appropriate ℓ^R, ℓ^C, we introduce variable vectors $Y^u = (Y_1^u, \ldots, Y_{\ell^R}^u)$ and $Z^w = (Z_1^w, \ldots, Z_{\ell^C}^w)$ for all $u \in U_o$ and $w \in W_o$. If $u \in U_o$ or $w \in W_o$ then we can express the term $(X^u)_\pi \cdot X^w$ in h_G in terms of these new variables. In particular, let

$$\begin{aligned} h''_G \;=\; & \bigoplus_{\substack{\{u,w\}\in E\\ u\in U_o,w\in W_o}} (\phi^R(Y^u))_\pi \cdot \phi^C(Z^w) \oplus \bigoplus_{\substack{\{u,w\}\in E\\ u\in U_e,w\in W_e}} (X^u)_\pi \cdot X^w \\ & \oplus \bigoplus_{\substack{\{u,w\}\in E\\ u\in U_e,w\in W_o}} (X^u)_\pi \cdot \phi^C(Z^w) \oplus \bigoplus_{\substack{\{u,w\}\in E\\ u\in U_o,w\in W_e}} (\phi^R(Y^u))_\pi \cdot X^w. \end{aligned}$$

Let

$$h'_G = h''_G \oplus \bigoplus_{u\in U_o} g^R(\phi^R(Y^u)) \oplus \bigoplus_{w\in W_o} g^C(\phi^C(Z^w)) \tag{2.5}$$

We therefore have

$$s_a = \left| \left\{ \chi : \mathrm{var}(h'_G) \to \mathbb{F}_2 \mid h'_G(\chi) = a) \right\} \right|. \tag{2.6}$$

By condition **(D)**, the polynomials $g^R \circ \phi^R$ and $g^C \circ \phi^C$ are of degree at most 2 and therefore h'_G is a polynomial of degree at most 2. Furthermore, we have expressed s_a as the number of solutions to a polynomial equation over $\mathbb{F}_2$. Therefore, the proof now follows by the following well-known fact.

Fact 2.7. *The number of solutions to polynomial equations of degree at most* 2 *over* $\mathbb{F}_2$ *can be computed in polynomial time.*

This is a direct consequence of Theorems 6.30 and 6.32 in [15] (see also [9]). ∎

3. The General Case

In this section we will prove Theorem 1.1. Before we can give the proof some further results have to be derived, which then enable us to extend Theorems 1.2 and 2.2. It will be convenient to focus on connected components. This is expressed by the following Lemma.

Lemma 3.1. *Let A be a symmetric matrix with entries in $\mathbb{R}_{\mathbb{A}}$ and let $A_1, \dots, A_c$ denote its components. Then the following holds*

(1) *If* $\text{EVAL}(A_i)$ *is* #P*-hard for some* $i \in [c]$ *then* $\text{EVAL}(A)$ *is* #P*-hard.*
(2) *If* $\text{EVAL}(A_i)$ *is* PTIME *computable for all* $i \in [c]$ *then* $\text{EVAL}(A)$ *is* PTIME *computable.*

Recall that for each connected symmetric matrix A there is a block B such that either $A = B$ or, up to permutation of the rows and columns, $A = \begin{pmatrix} 0 & B \\ B^T & 0 \end{pmatrix}$. We call B the block *underlying* A. For such connected A we furthermore see that the evaluation problem is either #P-hard or we can reduce it to the evaluation problem on bipartisations of Hadamard matrices.

Lemma 3.2. *Suppose that A is a symmetric connected matrix.*
Then either $\text{EVAL}(A)$ *is* #P*-hard or the following holds.*

(1) *If A is not bipartite there is a symmetric $r \times r$ Hadamard matrix H and a set $\Lambda^R \subseteq [r]$ such that*
$$\text{EVAL}(A) \equiv \text{EVAL}(H, I_r, I_{r;\Lambda^R}).$$
(2) *If A is bipartite then there is an $r \times r$ Hadamard matrix H, sets $\Lambda^R, \Lambda^C \subseteq [r]$ and a bipartisation M, Λ of H, Λ^R and Λ^C such that*
$$\text{EVAL}(A) \equiv \text{EVAL}(M, I_{2r}, I_{2r;\Lambda}).$$

Furthermore it can be decided in time polynomial in the size of A which of the three alternatives (#P-hardness, (1), or (2)) holds.

We are now able to prove the main Theorem.

Proof of Theorem 1.1. Given a symmetric matrix $A \in \mathbb{R}_{\mathbb{A}}^{m \times m}$. By Lemma 3.1 we may assume that the matrix A is connected. By Lemma 3.2, Theorem 2.2 the problem $\text{EVAL}(A)$ is either polynomial time computable or #P-hard. The existence of a polynomial time algorithm for deciding which of the two possibilities holds, given a matrix A, follows directly by these results. ∎

References

[1] L. Barto, M. Kozik, and T. Niven. Graphs, polymorphisms and the complexity of homomorphism problems. In *Proceedings of the 40th ACM Symposium on Theory of Computing*, 2008. To appear.

[2] Markus Bläser and Holger Dell. Complexity of the cover polynomial. In L. Arge, Ch. Cachin, T. Jurdzinski, and A. Tarlecki, editors, *of the 34th International Colloquium on Automata, Languages and Programming*, volume 4596 of *Lecture Notes in Computer Science*, pages 801–812. Springer Verlag, 2007.

[3] A. Bulatov. A dichotomy theorem for constraint satisfaction problems on a 3-element set. *Journal of the ACM*, 53:66–120, 2006.

[4] A. Bulatov. The complexity of the counting constraint satisfaction problem. In *Proceedings of the 35th International Colloquium on Automata, Languages and Programming*, Lecture Notes in Computer Science. Springer Verlag, 2008. To appear.

[5] A. Bulatov and V. Dalmau. Towards a dichotomy theorem for the counting constraint satisfaction problem. In *Proceedings of the 43rd Annual IEEE Symposium on Foundations of Computer Science*, pages 562–571, 2003.

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

[7] M. Dyer and C. Greenhill. The complexity of counting graph homomorphisms. *Random Structures and Algorithms*, 17(3–4):260–289, 2000.

[8] M.E. Dyer, L.A. Goldberg, and M. Paterson. On counting homomorphisms to directed acyclic graphs. *Journal of the ACM*, 54(6), 2007.

[9] Andrzej Ehrenfeucht and Marek Karpinski. The computational complexity of (*xor, and*)-counting problems. Technical Report 8543-CS, 1990. Available at `http://citeseer.ist.psu.edu/ehrenfeucht90computational.html`.

[10] M. Freedman, L. Lovász, and A. Schrijver. Reflection positivity, rank connectivity, and homomorphism of graphs. *Journal of the American Mathematical Society*, 20:37–51, 2007.

[11] L. A. Goldberg, S. Kelk, and M. Paterson. The complexity of choosing an H-colouring (nearly) uniformly at random. In *Proceedings of the 34th ACM Symposium on Theory of Computing*, pages 53–62, 2002.

[12] L.A. Goldberg and M. Jerrum. Inapproximability of the tutte polynomial. In *Proceedings of the 39th Annual ACM Symposium on Theory of Computing*, pages 459–468, 2007.

[13] P. Hell and J. Nešetřil. On the complexity of H-coloring. *Journal of Combinatorial Theory, Series B*, 48:92–110, 1990.

[14] F. Jaeger, D. L. Vertigan, and D. J. A. Welsh. On the computational complexity of the Jones and Tutte polynomials. *Mathematical Proceedings of the Cambridge Philosophical Society*, 108:35–53, 1990.

[15] Rudolf Lidl and Harald Niederreiter. *Finite fields*, volume 20 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, Cambridge, 2nd edition, 1997.

[16] M. Lotz and J.A. Makowsky. On the algebraic complexity of some families of coloured tutte polynomials. *Advances in Applied Mathematics*, 32:327–349, 2004.

[17] L. Lovász. The rank of connection matrices and the dimension of graph algebras. *European Journal of Combinatorics*, 27:962–970, 2006.

[18] L. Lovász and A. Schrijver. Graph parameters and semigroup functions. *European Journal of Combinatorics*. To appear.

[19] Alan Sokal. The multivariate Tutte polynomial. In *Surveys in Combinatorics*. Cambridge University Press, 2005.

[20] D. J. A. Welsh. *Complexity: Knots, Colourings and Counting*, volume 186 of *London Mathematical Society Lecture Note Series*. Cambridge University Press, 1993.