Combinatorics, Probability and Computing (2006) 15, 449–461.
doi:10.1017/S0963548305007285 Printed in the United Kingdom

Evaluating the Rank Generating Function of a Graphic 2-Polymatroid

S. D. NOBLE

Department of Mathematical Sciences, Brunel University, Kingston Lane, Uxbridge UB8 3PH, UK
(e-mail: mastsdn@brunel.ac.uk)

Received 15 July 2002; revised 22 June 2004

We consider the complexity of the two-variable rank generating function, S, of a graphic 2-polymatroid. For a graph G, S is the generating function for the number of subsets of edges of G having a particular size and incident with a particular number of vertices of G. We show that for any $x, y \in \mathbb{Q}$ with $xy \neq 1$, it is #P-hard to evaluate S at (x, y). We also consider the k-thickening of a graph and computing S for the k-thickening of a graph.

1. Introduction

We consider the complexity of a two-variable graph polynomial $S(G; x, y)$ that is closely related to the Tutte polynomial and was introduced in [11]. Like the Tutte polynomial, S contains a large variety of well-studied specializations, for instance the number of perfect matchings of a graph. A less well-studied specialization is the probability that deleting the edges of a graph independently with probability $1 - p$ does not introduce any isolated vertices.

The Tutte polynomial can be viewed as a generating function for the number of subsets of edges with a particular rank and cardinality. S can be viewed in a similar way as a generating function for the number of subsets of edges incident with a particular number of vertices and a particular cardinality.

Following [3] we define the complexity class #P to consist of those enumeration problems, π, for which there is a nondeterministic algorithm $\mathscr{A}$ and a polynomial p such that:

(1) for any instance I of π, the number of distinct accepting computations of $\mathscr{A}$ with input I is equal to the solution of π on input I;
(2) the length of the longest accepting computation is bounded by $p(|I|)$.

Given two enumeration problems π_1 and π_2, we say that π_1 is Turing-reducible to π_2, which we denote by $\pi_1 \propto \pi_2$, if there is a Turing machine which can solve π_1 in polynomial time given an oracle for π_2. We now define an enumeration problem π to be #P-hard if for every $\pi' \in$ #P, $\pi \propto \pi'$. For background information on complexity see [3, 15].

In [5] the Tutte polynomial is shown to be #P-hard to evaluate at every rational point, except for those lying on one special curve and for 5 additional special points.

We show that the complexity of S is very similar to that of the Tutte polynomial in that it is #P-hard to evaluate at every rational point except for those lying on one special curve.

To ensure that the definitions that we make seem natural and that the ideas behind the proof are as transparent as possible, we make use of the language and terminology of polymatroids. The reader does not need any previous knowledge of polymatroids, and if desired may skip much of the sections on polymatroids, since all the important ideas are explained in purely graph-theoretic terms.

We begin with the definition of an integer polymatroid and move on to describe the particular class of polymatroids in which we will be interested, namely graphic 2-polymatroids. Section 3 describes the rank generating function S, originally introduced by Oxley and Whittle in [11], and gives some of the invariants that appear as specializations of S. In Section 4 we introduce the thickening operation, which plays a crucial role in the proof of the main result. The k-thickening, G^k, of a graph, G, is obtained by replacing each edge by k parallel edges. We give a formula relating $S(G^k)$ and $S(G)$. Finally, in Section 5 we formally state and prove the main result concerning the complexity of S.

Our graph-theoretical notation is fairly standard. Note, however, that all our graphs are allowed to have loops and multiple edges and, for reasons made clear later, do not have isolated vertices. We use $G \setminus A$ and G/A to denote, respectively, the graphs obtained from G by deleting the edges in A and contracting the edges in A. Given a graph G with edge set E, the graph $G|A$ is $G \setminus (E \setminus A)$. For a graph G and a set A of edges of G, $G : A$ denotes the graph formed from $G|A$ by deleting all isolated vertices.

2. Integer polymatroids

An *integer polymatroid* (E, f) consists of a finite edge set E and an integer-valued rank function f, defined on all subsets of E and satisfying

(1) $f(\emptyset) = 0$,
(2) if $X \subseteq Y$ then $f(X) \leqslant f(Y)$,
(3) if $X, Y \subseteq E$ then $f(X) + f(Y) \geqslant f(X \cup Y) + f(X \cap Y)$.

A k-polymatroid is a polymatroid (E, f) such that for all $e \in E$, $f(e) \leqslant k$. Polymatroids are a natural generalization of the well-studied class of matroids, which correspond to 1-polymatroids; see, for instance, [14] for an introduction to polymatroids or [10] for information on matroids. In this paper we will only be concerned with 2-polymatroids.

Any graph gives rise to a 2-polymatroid (E, f_G) by taking $E = E(G)$ and for any $A \subseteq E$ setting $f_G(A) = |V(G : A)|$. It is easy to check that this satisfies the definition of a 2-polymatroid. Moreover it is noted in [11] that such a 2-polymatroid (E, f_G) uniquely determines G up to the addition of isolated vertices. From now on we will assume that all our graphs do not have isolated vertices. We say (E, f) is *induced* by G if it is isomorphic to (E, f_G). This type of polymatroid derived from graphs is the one that will interest us in the rest of this paper. Although we only need to deal with 2-polymatroids derived from graphs in this way, many of the concepts that we introduce can be defined much more generally and we do this whenever it seems natural. The reader who wishes to avoid becoming involved with the theory of polymatroids will lose very little by just thinking of a graph G and the pair (E, f_G) defined above.

We need to consider two operations on a 2-polymatroid (E, f) which are defined in [11]. The *deletion* of a set A of edges, denoted by $(E, f)\backslash A$, is the 2-polymatroid $(E\backslash A, f'_A)$ where, for any $X \subseteq E\backslash A$,

$$f'_A(X) = f(X).$$

The *contraction* of a set A of edges, denoted by $(E, f)/A$ is the 2-polymatroid $(E\backslash A, f''_A)$ where, for any $X \subseteq E\backslash A$,

$$f''_A(X) = f(X \cup A) - f(A). \tag{2.1}$$

It is straightforward to check that, with these definitions, the two operations do actually produce 2-polymatroids. We will often just write $f\backslash A$ and f/A instead of $(E, f)\backslash A$ and $(E, f)/A$ respectively.

We consider the effect of deletion and contraction on graphic 2-polymatroids later, but for the moment note that contracting an edge in a polymatroid may create an edge with rank zero. Consequently it is convenient to consider a slightly larger class of polymatroids than just the ones that are induced by graphs, because later we will need to work with a class of polymatroids that contains those induced by graphs and which is also closed under both deletion and contraction. Clearly edges with rank zero do not occur in graphs because they would correspond to edges with no endpoints. We call such an edge a *circle*, and say that a polymatroid is *graphic* if it is of the form $M = (E_1 \cup E_2, f)$, where $M\backslash E_2$ is induced by some graph G and for any $e \in E_2$, $f(e) = 0$, in other words M is induced by G except for the addition of some circles, that is, special edges with no endpoints. In some places we go a little further and abuse our notation by allowing graphs, rather than just graphic polymatroids, to have circles.

A set X of edges is a *separator* for a 2-polymatroid (E, f) if $f(X) + f(E\backslash X) = f(E)$. In terms of graphs, a set X of edges is a separator in a graph G if and only if the set of endpoints of edges in X and the set of endpoints of edges in $E\backslash X$ are disjoint.

Single-element separators can have rank zero, one or two, and for a graphic polymatroid these correspond to circles, a loop on a vertex that is incident with no other edges and an edge joining two vertices that are incident with no other edges. The 2-polymatroids $U_{0,1}$, $U_{1,1}$ and $U_{2,1}$ are the graphic polymatroids with precisely one edge e, which is respectively a circle, loop or edge between two vertices.

If e is not a separator of f then it is noted in [11] that one of the following must occur:

(1) $f(E\backslash e) = f(E)$ and $f(e) = 1$,
(2) $f(E\backslash e) = f(E) - 1$ and $f(e) = 2$,
(3) $f(E\backslash e) = f(E)$ and $f(e) = 2$.

For graphic 2-polymatroids the first case corresponds to e being a loop on a vertex that is an endpoint of some other edge, the second to e being a non-loop edge with precisely one of its endpoints having degree one, that is a pendant edge, and the third to any non-loop edge for which both endpoints have degree at least two.

It is worthwhile noting the effect of the operations of deletion and contraction on a graphic 2-polymatroid. Deletion is easy,

$$(E, f_G)\backslash e = (E\backslash e, f_{G\backslash e}),$$

where $f_{G\backslash e}$ denotes the restriction of the rank function to $E\backslash e$ and so it just corresponds to normal deletion in the graph (including the deletion of any isolated vertices that are formed). Equation (2.1) shows that contracting a separator is equivalent to deleting it, but generally contraction is more difficult. Suppose we contract the non-separating edge uv where we allow $v = u$. Then

$$(E, f_G)/uv = (E\backslash e, f_{G\sim e}),$$

where $G \sim e$ is formed from G by deleting uv, replacing any loop attached at u or v or edge parallel to uv by a circle and replacing any edge uw (vw) for $w \neq u$ (v) by a loop at w and $f_{G\sim e}$ is the graphic 2-polymatroid induced by $G \sim e$. The definition of contraction is the main point that seems much more natural when phrased in terms of polymatroids rather than graphs.

3. Rank generating function

The 2-polymatroid rank generating function was introduced in [11] and is the two-variable polynomial associated with any polymatroid f, defined by

$$S(f; x, y) = \sum_{A \subseteq E} x^{f(E)-f(A)} y^{2|A|-f(A)}.$$

The reader may just regard S as being a polynomial defined only on graphs. When we consider the rank generating function of the 2-polymatroid derived from a graph G, we will usually write $S(G; x, y)$. It is easy to see that adding isolated vertices to G will not affect S.

The following specializations of S are stated in [11].

- $S(G; 1, 0)$ is the number of matchings of G.
- If G has no isolated vertices then $S(G; 0, 0)$ is the number of perfect matchings of G and $S(G; 0, 1)$ is the number of subsets of E spanning every vertex of G.
- If $x \neq 0$ then $x^{f_G(E)/2} S(G; x^{-1/2}, 0)$ is the polynomial $\sum_{k \geqslant 0} m_k x^k$ where m_k is the number of matchings of size k in G.
- $S(f; -x, -y) = (-1)^{f(E)} S(f; x, y)$.
- $S(f; \frac{1}{x}, x) = (1 + x^2)^{|E|} x^{-f(E)}$ for $x \neq 0$.
- For a graph G with no isolated vertices and $0 \leqslant p < 1$,

  $$(1-p)^{(|E|-f_G(E)/2)}\, p^{(f_G(E)/2)}\, S(G; 0, p^{1/2}(1-p)^{-1/2})$$

  is the probability that G_p has no isolated vertices; where G_p is the random graph formed by deleting all the edges of G independently with probability $1 - p$.

- Providing $x \neq 0$, $x^{f(E)}S(G; 1/x, 1) = \sum_{k \geqslant 0} r_k x^k$ where r_k is the number of subsets of E spanning k vertices. This polynomial can be thought of as a one-variable rank generating function.

Note that none of the specializations discussed above can be obtained from the Tutte polynomial.

4. Thickenings

The proof of the hardness result for the Tutte polynomial makes use of the *tensor product* construction of Brylawski [1]. The tensor product of a matroid M with a pointed matroid N, that is, a matroid with a distinguished element e, is formed by taking the 2-sum of M and N about each point of M. In [5] the matroid $U_{1,k+1}$ which is the graphic matroid induced by the graph consisting of k parallel edges was used in the role of N in order to prove the complexity results. This particular tensor product is known as a k-thickening because each edge of the graph is replaced by k parallel edges. We have not constructed a general 2-sum for a polymatroid but we define the *k-thickening* of a graphic 2-polymatroid induced by G with l circles to be the 2-polymatroid induced by G^k together with kl circles where G^k is formed by replacing each edge in G, including loops, by k parallel edges.

Following [11], if $\mathscr{M}$ denotes the class of all graphic 2-polymatroids, then $\phi : \mathscr{M} \to \mathbb{C}$ is said to be a *generalized Tutte invariant* (*for graphic* 2-*polymatroids*) if there exist constants a, b, c, d, m, n, r, s and $t \in \mathbb{C}$ such that

$$\begin{aligned}\phi(U_{2,1}) &= r,\\ \phi(U_{0,1}) &= s,\\ \phi(U_{1,1}) &= t,\end{aligned}$$

and for any graphic 2-polymatroid (E, f),

$$\phi(f) = \phi(f\backslash(E\backslash e))\phi(f\backslash e) \qquad \text{if } e \text{ is a separator of } f;$$

and if e is not a separator,

$$\phi(f) = \begin{array}{ll} a\phi(f\backslash e) + b\phi(f/e) & \text{if } f(E\backslash e) = f(E) \text{ and } f(e) = 1,\\ c\phi(f\backslash e) + d\phi(f/e) & \text{if } f(E\backslash e) = f(E) - 1 \text{ and } f(e) = 2,\\ m\phi(f\backslash e) + n\phi(f/e) & \text{if } f(E\backslash e) = f(E) \text{ and } f(e) = 2.\end{array}$$

The following theorem is from [11].

Theorem 4.1. *Let ϕ be a generalized Tutte invariant on graphic 2-polymatroids and suppose that at most two of r, s, t, a, b, c, m and n are zero. Then one of the following occurs:*

(1) $a = m;\ d = n;\ mr = mn + c^2;\ ns = mn + b^2;\ t = b + c;\ m \neq 0;\ n \neq 0;$ *and for all* 2-*polymatroids* f,

$$\phi(f) = m^{|E|-f(E)/2} n^{f(E)/2} S\left(f; \frac{c}{(mn)^{1/2}}, \frac{b}{(mn)^{1/2}}\right);$$

(2) $t^2 = rs = ar + bt = ct + ds = mr + ns$; $st = at + bs$; $rt = cr + dt$; *and, for all 2-polymatroids* f, $\phi(f) = Q(f)$ *where*

$$Q(f) = \begin{cases} s^{(|E|-f(E))}t^{f(E)} & \text{if } f(E) \leqslant |E|; \\ r^{(f(E)-|E|)}t^{(2|E|-f(E))} & \text{otherwise.} \end{cases}$$

It is easy to show that S is a generalized Tutte invariant on 2-polymatroids with

$$\begin{aligned} &r = 1 + x^2, \quad s = 1 + y^2, \quad t = x + y, \quad m = n = 1, \\ &a = d = 1, \quad b = y \text{ and } c = x. \end{aligned} \tag{4.1}$$

Our first result relates the rank generating function of a graphic polymatroid with that of its k-thickening.

Proposition 4.2. *If* $y \neq 0$ *then*

$$S\big(G^k; x, y\big) = \left(\frac{(1+y^2)^k - 1}{y^2}\right)^{f(E)/2} \cdot S\left(G; \frac{xy}{\sqrt{(1+y^2)^k - 1}}, \sqrt{(1+y^2)^k - 1}\right).$$

If $y = 0$ *then*

$$S\big(G^k; x, 0\big) = k^{f(E)/2} S\left(G; \frac{x}{\sqrt{k}}, 0\right).$$

To shorten the proof of this proposition we first prove the following lemma. We let R_k be the graph consisting of just k circles, L_k be the graph with just one vertex and k loops and M_k be the graph with 2 vertices and k edges between them.

Lemma 4.3. *If* $k \geqslant 2$,

$$\begin{aligned} S(R_k; x, y) &= (1+y^2)^k, \\ S(L_k; x, y) &= y(1+y^2)^{k-1} + \cdots + y(1+y^2) + x + y, \\ S(M_k; u, v) &= (1+y^2)^{k-1} + \cdots + (1+y^2) + 1 + x^2. \end{aligned}$$

Proof. The first equation is simple to check because each circle is a separator and $S(R_1; x, y) = 1 + y^2$. We prove the second by induction. If $k = 2$ then $S(L_k; x, y) = y(1 + y^2) + x + y$. Otherwise, using induction,

$$\begin{aligned} S(L_k; x, y) &= S(L_{k-1}; x, y) + yS(R_{k-1}; x, y) \\ &= y(1+y^2)^{k-2} + \cdots + y(1+y^2) + x + y + y(1+y^2)^{k-1}. \end{aligned}$$

The third is also proved using induction. If $k = 1$ then $S(M_k; x, y) = 1 + x^2$. Otherwise, using induction,

$$\begin{aligned} S(M_k; x, y) &= S(M_{k-1}; x, y) + S(R_{k-1}; x, y) \\ &= (1+y^2)^{k-2} + \cdots + (1+y^2) + 1 + x^2 + (1+y^2)^{k-1}. \end{aligned}$$

□

Proof of Proposition 4.2. We let $\phi(G; x, y) = S(G^k; x, y)$. To prove this result it is just necessary to show that for each value of x and y, ϕ is a generalized Tutte invariant on 2-polymatroids satisfying the conditions from the first part of Theorem 4.1. Let e be an edge of G. There are three cases to consider where e is a separator and three cases where e is not. In the following we make repeated use of equation (4.1).

In each of the cases when e is a separator, e will be replaced by k edges in G^k which together form a separator of G^k. Consequently

$$\phi(G; x, y) = \phi(G\backslash e; x, y)\phi(G|e; x, y).$$

So when e is a circle

$$\phi(G; x, y) = (1 + y^2)^k \phi(G\backslash e; x, y),$$

when e is an isolated loop

$$\phi(G; x, y) = \big(y(1 + y^2)^{k-1} + \cdots + y(1 + y^2) + x + y\big)\phi(G\backslash e; u, v),$$

and when e is an isolated (non-loop) edge

$$\phi(G; x, y) = \big((1 + y^2)^{k-1} + \cdots + (1 + y^2) + 1 + x^2\big)\phi(G\backslash e; x, y).$$

We now consider the cases where e is not a separator in G. All three cases are quite similar. We use equation (4.1), and contract and delete one of the edges replacing e in G^k, to leave respectively $(G \sim e)^k$ with $k-1$ circles and the k-thickening of G but with only $k-1$ edges replacing e. We leave the first graph and repeat the procedure with one of the $k-1$ remaining edges which replace e. We keep doing this until we have deleted or contracted all the edges which replace e. First suppose that e is a non-isolated loop of G. Then

$$\begin{aligned}\phi(G; x, y) &= yS(R_{k-1}; x, y)S\big((G \sim e)^k; x, y)\big) + \cdots + yS(R_1; x, y)S\big((G \sim e)^k; x, y\big)\\ &\quad + yS\big((G \sim e)^k; x, y\big) + S\big((G\backslash e)^k; x, y\big)\\ &= (y(1 + y^2)^{k-1} + \cdots + y(1 + y^2) + y)S\big((G \sim e)^k; x, y\big) + S\big((G\backslash e)^k; x, y\big).\end{aligned}$$

Secondly, if e is a pendant edge of G then

$$\begin{aligned}\phi(G; x, y) &= S(R_{k-1}; x, y)S\big((G \sim e)^k; x, y\big) + \cdots + S(R_1; x, y)S\big((G \sim e)^k; x, y\big)\\ &\quad + S\big((G \sim e)^k; x, y\big) + xS\big((G\backslash e)^k; x, y\big)\\ &= ((1 + y^2)^{k-1} + \cdots + (1 + y^2) + 1)S\big((G \sim e)^k; x, y\big) + xS\big((G\backslash e)^k; x, y\big).\end{aligned}$$

Finally, if e is an edge with both endpoints having degree at least two then

$$\begin{aligned}\phi(G; x, y) &= S(R_{k-1}; x, y)S\big((G \sim e)^k; x, y\big) + \cdots + S(R_1; x, y)S\big((G \sim e)^k; x, y\big)\\ &\quad + S\big((G \sim e)^k; x, y\big) + S\big((G\backslash e)^k; x, y\big)\\ &= ((1 + y^2)^{k-1} + \cdots + (1 + y^2) + 1)S\big((G \sim e)^k; x, y\big) + S\big((G\backslash e)^k; x, y\big).\end{aligned}$$

If x, y are not both zero then it is easy to check that the first case of Theorem 4.1 applies, and so

$$\begin{aligned}\phi(G;x,y) &= S(G^k;x,y)\\ &= \left((1+y^2)^{k-1}+\cdots+(1+y^2)+1\right)^{f(E)/2}\\ &\quad\cdot S\Bigg(G;\frac{x}{\sqrt{(1+y^2)^{k-1}+\cdots+(1+y^2)+1}},\\ &\qquad\frac{y(1+y^2)^{k-1}+\cdots+y(1+y^2)+y}{\sqrt{(1+y^2)^{k-1}+\cdots+(1+y^2)+1}}\Bigg).\end{aligned}$$

If $y=0$ this simplifies to

$$\phi(G;x,0)=k^{f(E)/2}S\left(G;\frac{x}{\sqrt{k}},0\right),$$

and otherwise we have

$$\phi(G;x,y)=\left(\frac{(1+y^2)^k-1}{y^2}\right)^{f(E)/2}\cdot S\left(G;\frac{xy}{\sqrt{(1+y^2)^k-1}},\sqrt{(1+y^2)^k-1}\right).$$

The case when $x=y=0$ is straightforward, since $\phi(G;x,y)$ is the number of perfect matchings of G^k. Thus

$$\phi(G;0,0)=k^{f(E)/2}S(G;0,0). \qquad \square$$

5. Main result

We begin with a formal statement of the problem which we are considering.

Problem. $\pi_1(x,y)$: RANK GENERATING FUNCTION EVALUATION AT (x,y)
Input: A graph G.
Output: The evaluation at (x,y) of the rank generating function of the 2-polymatroid, induced by G.

Our main result is as follows.

Theorem 5.1. *For $x,y\in\mathbb{Q}$ satisfying $xy\neq 1$, the problem $\pi_1(x,y)$ is #P-hard to compute; when $x,y\in\mathbb{Q}$ and $xy=1$, there is a polynomial time algorithm.*

The theorem shows that $\pi_1(x,y)$ is #P-hard for all rational values of (x,y) except for those lying on one special curve. This behaviour is very similar to the complexity of the Tutte polynomial, which is #P-hard to evaluate at almost all points in the plane [5], a result which also remains true if we restrict the input to bipartite planar graphs [13].

The rest of this section is devoted to a proof of Theorem 5.1. The case when $xy=1$ is easy because if $x\neq 0$ then

$$S\left(f;\frac{1}{x},x\right)=(1+x^2)^{|E|}x^{-f(E)},$$

and obviously this can be evaluated very quickly.

All the hardness proofs rely on the following, which is a restatement of a result of Valiant [12].

Theorem 5.2. *Computing $\pi_1(0,0)$ is #P-hard.*

Proof. If G has no isolated vertices then $S(G;0,0)$ is the number of perfect matchings of G, a quantity that is #P-hard to compute, [12]. □

The family of hyperbolae H_α defined by

$$H_\alpha = \{(x,y) : xy = \alpha\}$$

seems to play an important role in the theory. If $\alpha \neq 0$, then along the hyperbola H_α we can write

$$S(G;x,y) = S_\alpha(G;y) = \sum_{i=0}^{|E|} c_i y^{2i-f(E)}$$

for certain coefficients c_i depending on α. It is convenient to consider H_0 in two separate parts corresponding to the x and y axes, which we denote respectively by H_0^x and H_0^y, but in either case it is obvious that the restriction of S to either part is a one-variable polynomial. All this motivates the following problem, which has a crucial part in the proof of hardness of π_1.

Problem. $\pi_2(\alpha)$: H_α RANK GENERATING FUNCTION
Input: A graph G.
Output: The coefficients of $S_\alpha(G;y)$.

We use $\pi_2(0^x)$ and $\pi_2(0^y)$ in the obvious way, to denote the problem of computing the coefficients of the restriction of S to H_0^x and H_0^y respectively.

It is clear that for any x and y,

$$\pi_1(x,y) \propto \pi_2(xy).$$

We now give a result that is halfway to the main theorem of this section.

Theorem 5.3. *If $\alpha \notin \{0,1\}$ then $\pi_2(\alpha) \propto \pi_1(x,y)$ for any $x,y \in \mathbb{Q}$ such that $xy = \alpha$. Furthermore $\pi_2(0^x) \propto \pi_1(x,0)$ for any $x \in \mathbb{Q}\backslash\{0\}$ and $\pi_2(0^y) \propto \pi_1(0,y)$ for any $y \in \mathbb{Q}\backslash\{0\}$.*

Proof. We begin by proving the result in the case when $\alpha \neq 0$. The idea is as follows. Assume we have an oracle to compute $S(G;x,y)$ for any graph G. Hence for a given graph G and for any positive integer k, we can find $S(G^k;x,y)$, and so using Proposition 4.2 we can compute

$$S\left(G; \frac{xy}{\sqrt{(1+y^2)^k-1}}, \sqrt{(1+y^2)^k-1}\right).$$

All these points lie on H_α so if we do this for enough values of k we can compute the polynomial $S_\alpha(G;y)$ using Lagrange interpolation.

More precisely, suppose we input a graph G and that for some $x, y \in \mathbb{Q}$ with $xy = \alpha \neq 0$ we have an oracle to compute $S(H; x, y)$ for every graph H. We write $S_\alpha(G; y) = \sum_{i=0}^{|E|} c_i y^{2i-f(E)}$. For each k such that $1 \leqslant k \leqslant |E|+1$, we compute $S(G^k; x, y)$, and then

$$\sum_{i=0}^{|E|} c_i \left(\sqrt{(1+y^2)^k - 1}\right)^{2i-f(E)} = S\left(G; \frac{xy}{\sqrt{(1+y^2)^k-1}}, \sqrt{(1+y^2)^k-1}\right)$$
$$= \left(\frac{y}{\sqrt{(1+y^2)^k-1}}\right)^{f(E)} S(G^k; x, y).$$

Rearranging all this gives

$$\sum_{i=0}^{|E|} c_i \left((1+y^2)^k - 1\right)^i = y^{f(E)} S(G^k; x, y).$$

The square roots have been eliminated so this is an expression containing only rationals. To compute c_i for $0 \leqslant i \leqslant |E|$ we solve the $|E|+1$ equations resulting from substituting $k = 1, 2 \ldots, |E|+1$, using Gaussian elimination. To see that this gives a polynomial time algorithm to compute each c_i we need to note two facts. Firstly the $|E|+1$ equations are linearly independent. This is because the determinant of the coefficients of the equations is a Vandermonde determinant, which is well known to be nonzero. Secondly each of the coefficients is polynomially bounded in terms of the input size, that is, the length of the description of G. Gaussian elimination on an $n \times n$ matrix requires $O(n^3)$ arithmetical operations and can be done in such a way that the length of the description of the entries of the matrix remains polynomially bounded in terms of the original length of the description of the entries. See [2, 4] for a discussion of this. Thus we can recover the coefficients c_i in polynomial time.

Along H_0^y we have

$$S(G; x, y) = \sum_{A \subseteq E : f(A) = f(E)} y^{2|A| - f(E)} = \sum_{i=0}^{|E|} c_i y^{2i-f(E)},$$

and so we can use exactly the same procedure as above to show that $\pi_2(0^y) \propto \pi_1(0, y)$ for any $y \in \mathbb{Q} \backslash \{0\}$.

The final case is slightly different, just because the form of the tensor product is different when $y = 0$. Suppose we have an oracle to evaluate $S(G; x, 0)$ for any $x \in \mathbb{Q} \backslash \{0\}$. We write $S(G; x, 0) = \sum_{i=0}^{f(E)} c_i x^{f(E)-i}$. Since

$$S(G; x, 0) = \sum_{A \subseteq E : f(A) = 2|A|} x^{f(E)-f(A)}$$

we have that c_i is zero unless i is even.

Using Proposition 4.2, we have for $k \geqslant 1$

$$S\left(G; \frac{x}{\sqrt{k}}, 0\right) = \left(\frac{1}{\sqrt{k}}\right)^{f(E)} S(G^k; x, 0),$$

and so

$$\sum_{i=0}^{f(E)} c_i k^{i/2} x^{-i} = \left(\frac{1}{x}\right)^{f(E)} S(G^k; x, 0).$$

This is a rational expression because c_i is zero unless i is even. Hence we can solve for $c_0, \ldots, c_{f(E)}$ by computing the values $S(G^1; x, 0), \ldots, S(G^{f(E)+1}; x, 0)$ and solving for $c_0, \ldots, c_{f(E)}$ which is possible because the determinant of the matrix of coefficients is nonzero since it has the form of a Vandermonde determinant. □

We now move on to consider another specialization of S which plays an important role in the proof of hardness. Let $s(G; x)$ be the one-variable polynomial given by

$$s(G; x) = S(G; x, 1) = \sum_{A \subseteq E} x^{f(E)-f(A)}.$$

This polynomial seems to be a fairly natural object to consider as it is just a one-variable rank generating function. The second half of the proof of our main result is contained in the following theorem.

Theorem 5.4. *For any $x \in \mathbb{Q} \backslash \{1\}$, $\pi_1(0, 1) \propto \pi_1(x, 1)$.*

Proof. We can obviously assume that $x \neq 0$. We write

$$s(G; x) = \sum_{i=0}^{f(E)} r_i x^{f(E)-i},$$

where r_i is the number of subsets of $E(G)$ which are incident with i vertices. Let G_k be the graph formed from G by adding k loops at each non-isolated vertex. We have

$$r_i(G_k) = \sum_{j=0}^{i} r_j \binom{f(E)-j}{i-j} 2^{jk} (2^k - 1)^{i-j}. \tag{5.1}$$

This follows because each set A of edges that is incident with j vertices in G can be extended to give a set incident with i vertices by adding any number of loops at $i - j$ vertices which were isolated in $G|A$ and possibly adding loops at the vertices of $G : A$.

The idea is to use an oracle for $\pi_1(x, 1)$ to calculate $s(G; x), s(G_1; x), \ldots, s(G_{f(E)}; x)$ and then solve for the r_i. Using equation (5.1) gives for $k \geqslant 1$

$$\begin{aligned} s(G_k; x) &= \sum_{i=0}^{f(E)} \sum_{j=0}^{i} r_j \binom{f(E)-j}{i-j} 2^{jk} (2^k - 1)^{i-j} x^{f(E)-i} \\ &= \sum_{j=0}^{f(E)} \sum_{i=j}^{f(E)} r_j \binom{f(E)-j}{i-j} 2^{jk} (2^k - 1)^{i-j} x^{f(E)-i} \end{aligned}$$

$$= \sum_{j=0}^{f(E)} \sum_{i=0}^{f(E)-j} r_j \binom{f(E)-j}{i} 2^{jk}(2^k-1)^i x^{f(E)-i-j}$$
$$= \sum_{j=0}^{f(E)} r_j 2^{jk} \left(x + (2^k - 1)\right)^{f(E)-j}.$$

This means that we can solve for $r_0, \ldots, r_{f(E)}$ using the values $s(G;x), s(G_1;x), \ldots, s(G_{f(E)};x)$ because the determinant of the matrix of coefficients has the form of a Vandermonde determinant and is nonzero. The sizes of the entries of the matrix are polynomially bounded in terms of the size of the input graph G and so solving for $r_0, \ldots, r_{f(E)}$ can be done in polynomial time, as shown in [2, 4]. □

We can now prove the main theorem.

Proof of Theorem 5.1. We deal first with the case where $xy = 0$. We may clearly assume that one of x and y is nonzero. We noted that $S(G;0,0)$ is #P-hard in Theorem 5.2, and since $\pi_1(0,0) \propto \pi_2(0^x)$ and $\pi_1(0,0) \propto \pi_2(0^y)$, we see that both $\pi_2(0^x)$ and $\pi_2(0^y)$ are #P-hard. Using Theorem 5.3, if $x \neq 0$ then $\pi_2(0^x) \propto \pi_1(x,y)$ and if $y \neq 0$ then $\pi_2(0^y) \propto \pi_1(x,y)$. The result follows.

The second case is when $x, y \in \mathbb{Q}$ are such that that $xy \neq 0$ and $xy \neq 1$. The #P-hardness of $\pi_1(x,y)$ follows because

$$\pi_1(0,0) \propto \pi_2(0^y) \propto \pi_1(0,1) \propto \pi_1(xy,1) \propto \pi_2(xy) \propto \pi_1(x,y),$$

and we noted that $S(G;0,0)$ is #P-hard in Theorem 5.2. □

6. Conclusion

We have shown that the rank generating function of a graphic polymatroid is #P-hard to evaluate at any point xy for which $xy \neq 1$. Thus the complexity of the polynomial is quite similar to the Tutte polynomial.

The most interesting graph invariant which is an evaluation of S and has not been studied before is $S(G;0,1)$, which is the number of subsets of edges that are incident with every vertex. Our result implies that this is #P-complete.

For the Tutte polynomial, the hardness results remain true when the input graph is restricted to being bipartite and planar [13], although there is one additional curve along which evaluation only requires polynomial time. Similar results may hold for S but we have not made any attempt to investigate them. Since one of the reductions in the main proof involves adding loops, showing that the input may be restricted to bipartite graphs with the problem remaining #P-hard may not be straightforward.

The alternative problem to consider is to find large classes of graphs for which S may be evaluated in polynomial time. It is easy to see that using dynamic programming S may be evaluated in polynomial time for trees. In [6] it is shown that the Tutte polynomial can be evaluated at any point in polynomial time for graphs of bounded tree-width. The same method can easily be modified to show a corresponding result for S [7].

Both S and the Tutte polynomial are specializations of a very general graph polynomial U introduced in [9]. For a graph on n vertices, U is a polynomial in $n+1$ variables. Evaluating U is known to be #P-hard since the Tutte polynomial is a specialization; however, a more precise description of the points where evaluating U is #P-hard is not known. In [8], it is shown that U may be evaluated at any point in polynomial time when the input graph is restricted to having bounded tree-width, albeit much more slowly than using the method of [6].

References

[1] Brylawski, T. H. (1982) The Tutte polynomial. In *Matroid Theory and its Applications* (A. Barlotti, ed.), Liguori editore, Naples, pp. 125–275.

[2] Edmonds, J. (1967) Systems of distinct representatives and linear algebra. *J. Res. Nat. Bur. Standards Sect. B* **71B** 241–245.

[3] Garey, M. R. and Johnson, D. S. (1979) *Computers and Intractability*, W. H. Freeman, New York.

[4] Grötschel, M., Lovász, L. and Schrijver, A. (1980) *Geometric Algorithms and Combinatorial Optimization*, Springer, Berlin.

[5] Jaeger, F., Vertigan, D. L. and Welsh, D. J. A. (1990) On the computational complexity of the Jones and Tutte polynomials. *Math. Proc. Cambridge Philos. Soc.* **108** 35–53.

[6] Noble, S. D. (1998) Evaluating the Tutte polynomial for graphs of bounded tree-width. *Combin. Probab. Comput.* **7** 307–321.

[7] Noble, S. D. (1997) Complexity of graph polynomials. DPhil Thesis, University of Oxford.

[8] Noble, S. D. (2004) Evaluating a weighted graph polynomial for graphs of bounded tree-width. Submitted.

[9] Noble, S. D. and Welsh, D. J. A. (1999) A weighted graph polynomial from chromatic invariants of knots. *Ann. Inst. Fourier* **49** 1057–1087.

[10] Oxley, J. G. (1992) *Matroid Theory*, Oxford University Press, Oxford.

[11] Oxley, J. G. and Whittle, G. P. (1993) A chararcterisation of Tutte invariants for 2-polymatroids. *J. Combin. Theory Ser. B* **59** 210–244.

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

[13] Vertigan, D. L. and Welsh, D. J. A. (1992) The computational complexity of the Tutte plane: the bipartite case. *Combin. Probab. Comput.* **1** 181–187.

[14] Welsh, D. J. A. (1976) *Matroid Theory*, Academic Press, London.

[15] Welsh, D. J. A. (1993) *Complexity: Knots, Colourings, and Counting*, Cambridge University Press, Cambridge.