INFORMATION AND COMPUTATION 105, 268–288 (1993)

Automating Pólya Theory: The Computational Complexity of the Cycle Index Polynomial

LESLIE ANN GOLDBERG*

Department of Algorithms and Discrete Mathematics, Department 1423, Sandia National Laboratories, P.O. Box 5800, Albuquerque, New Mexico 87185

In this paper we investigate the computational difficulty of evaluating and approximately evaluating Pólya's *cycle index polynomial.* We start by investigating the difficulty of determining a particular coefficient of the cycle index polynomial. In particular, we consider the following problem, in which i is taken to be a fixed positive integer: Given a set of generators for a permutation group G whose degree, n, is a multiple of i, determine the coefficient of $x_i^{n/i}$ in the cycle index polynomial of G. We show that this problem is #P-hard for every fixed $i > 1$. Next, we consider the evaluation problem. Let $y_1, y_2, \ldots$ stand for an arbitrary fixed sequence of non-negative real numbers. The cycle index evaluation problem that is associated with this sequence is the following: Given a set of generators for a degree n permutation group G, evaluate the cycle index polynomial of G at the point $(y_1, \ldots, y_n)$. We show that if there exists an i such that $y_i \neq y_1^i$ and $y_i \neq 0$ then the evaluation problem associated with $y_1, y_2, \ldots$, is #P-hard. We observe that the evaluation problem is solvable in polynomial time if $y_j = y_1^j$ for every positive integer j and that it is solvable in polynomial time if $y_j = 0$ for every integer $j > 1$. Finally, we consider the approximate evaluation problem. We show that it is NP-hard to *approximately* solve the evaluation problem if there exists an i such that $y_i > y_1^i$. Furthermore, we show that it is NP-hard to approximately solve the evaluation problem if $y_1 = y_2 = \cdots = y$ for some positive non-integer y. We derive some corollaries of our results which deal with the computational difficulty of counting equivalence classes of combinatorial structures.

1. INTRODUCTION

In this paper, we investigate the computational difficulty of evaluating and approximately evaluating Pólya's *cycle index polynomial.* This polynomial is important in the field of combinatorial enumeration because it provides an elegant method for counting equivalence classes of

* This material is based upon work which was performed at the University of Edinburgh and was supported under a National Science Foundation Graduate Fellowship and a Marshall Scholarship. Preprints of this paper appeared under the author's maiden name, Henderson.

0890-5401/93 $5.00

combinatorial structures. In particular, the cycle index polynomial of a group of symmetries can be used to count structures up to isomorphism under the symmetries in the group.

Before describing the computational problems that we study, we provide the necessary definitions. Suppose that G is a group of permutations of $\{1, ..., n\}$. It is well known that each permutation $g \in G$ decomposes the set $\{1, ..., n\}$ into a collection of *cycles*, which we will call the cycles of g. We use the notation $c(g)$ to denote the number of cycles in this decomposition and the notation $c_i(g)$ to denote the number of cycles of length i. The *cycle index polynomial* of G is the n-variable polynomial $P_G(x_1, ..., x_n) = (1/|G|) \sum_{g \in G} x_1^{c_1(g)} \cdots x_n^{c_n(g)}$.

The first computational problem that we discuss is the generic cycle index evaluation problem:

Generic Cycle Index Evaluation

Input: A set of generators for a degree n permutation group G and n non-negative real numbers $y_1, ..., y_n$.

Output: $P_G(y_1, ..., y_n)$.

It is easy to see that we could implement an algorithm that solves the generic cycle index evaluation problem by summing over the permutations in the group G. However, the size of a permutation group can be exponential in the size of its smallest generating set,[1] so this method is infeasible computationally. In fact, no feasible method for solving this problem is known to exist. Furthermore, the construction from Lubiw's #P-hardness proof for *#Fixed-Point-Free Automorphism* [Lub 81] can be used to show that the generic cycle index evaluation problem is #P-hard.

Although the cycle index polynomial can be used to solve various combinatorial counting problems, a proof that the generic cycle index evaluation problem is #P-hard does not necessarily imply that the counting problems are #P-hard. On the contrary, the counting problems that are associated with the cycle index polynomial correspond to *special cases* of the generic cycle index evaluation problem. In particular, each of the counting problems that we discuss in Section 2 can be associated with a specific sequence $y_1, y_2, ...$ of non-negative real numbers in such a way that solving the counting problem for a degree n permutation group G is equivalent to evaluating the cycle index polynomial of G at the point $(y_1, ..., y_n)$.

In order to obtain interesting results about the computational difficulty of combinatorial counting problems and in order to obtain the strongest

[1] A degree n permutation group can contain up to $n!$ permutations. However, every degree n permutation group has a generating set of size at most $n-1$, as [Jer 86] demonstrates.

possible results about the difficulty of evaluating the cycle index polynomial we let $y_1, y_2, \ldots$ stand for an arbitrary fixed sequence of non-negative real numbers and we study the computational difficulty of the following cycle index evaluation problem:

Cycle Index Evaluation$(y_1, y_2, \ldots)$

Input: A set of generators for a degree n permutation group G.

Output: $P_G(y_1, \ldots, y_n)$.

In order to show that the cycle index evaluation problem is #P-hard we consider the difficulty of determining a particular coefficient of the cycle index polynomial. In particular, we consider the following problem in which i is taken to be a fixed positive integer.

Cycle Index Coefficient(i)

Input: A set of generators for a permutation group G whose degree, n, is a multiple of i.

Output: The coefficient of $x_i^{n/i}$ in the cycle index polynomial of G.

We obtain the following result:

THEOREM 1. *Let $i > 1$ be a fixed positive integer.* Cycle Index Coefficient(i) is *#P-hard.*

The coefficient of $x_i^{n/i}$ in P_G is $1/|G|$ times the number of permutations in G that have n/i cycles of length i. Therefore Theorem 1 implies that it is #P-hard to determine how many permutations in a group have a given cycle structure.

As well as being interesting in its own right, Theorem 1 is the main tool which we use to establish the computational difficulty of cycle index evaluation. Using Theorem 1 we obtain the following result:

THEOREM 2. *If $y_1, y_2, \ldots$ is a sequence of non-negative real numbers and there exists an i such that $y_i \neq y_1^i$ and $y_i \neq 0$ then* Cycle Index Evaluation$(y_1, y_2, \ldots)$ *is #P-hard.*

Theorem 2 has some interesting corollaries which describe the computational difficulty of solving certain counting problems. The corollaries are discussed in Section 2.

It would be interesting to determine the computational difficulty of *Cycle Index Evaluation*$(y_1, y_2, \ldots)$ when $y_1, y_2, \ldots$, is a sequence for which the condition in Theorem 2 is false. We have not solved this problem in this work, although we make the following observations:

Observation 1. Let $y_1, y_2, \ldots$ be a fixed sequence of non-negative real numbers such that for every positive integer j we have $y_j = y_1^j$. Then *Cycle Index Evaluation*$(y_1, y_2, \ldots)$ can be solved in polynomial time.

Observation 2. Let $y_1, y_2, \ldots$ be a fixed sequence of non-negative real numbers such that for every integer $j > 1$ we have $y_j = 0$. Then *Cycle Index Evaluation* $(y_1, y_2, \ldots)$ can be solved in polynomial time.

We conjecture that *Cycle Index Evaluation*$(y_1, y_2, \ldots)$ is #P-hard for every sequence $y_1, y_2, \ldots$ which fails to satisfy the conditions in Theorem 2, Observation 1, and Observation 2. The techniques that we use to prove Theorem 4 can be adapted to establish the #P-hardness of *Cycle Index Evaluation*$(y_1, y_2, \ldots)$ for many such sequences.

Since *Cycle Index Evaluation*$(y_1, y_2, \ldots)$ is almost always #P-hard we are interested in determining the computational difficulty of *approximately* solving the cycle index evaluation problem. In particular, suppose that q is a function from $\mathbb{N}$ to $\mathbb{N}$ and consider the following approximation problem:

Cycle Index Approximation$(q, y_1, y_2, \ldots)$

Input: A set of generators for a degree n permutation group G.

Otput: A quantity $z \in \mathbb{R}$ such that $(1/q(n))\, P_G(y_1, \ldots, y_n) \leqslant z \leqslant q(n)\, P_G(y_1, \ldots, y_n)$.

We obtain the following result concerning the computational difficulty of *Cycle Index Approximation*$(q, y_1, y_2, \ldots)$.

THEOREM 3. *If $y_1, y_2, \ldots$ is a sequence of non-negative real numbers and there exists an i such that $y_i > y_1^i$ then* Cycle Index Approximation$(q, y_1, y_2, \ldots)$ *is* NP-*hard for every polynomial q.*

As one would expect, we will be able to use Theorem 3 to derive corollaries about the computational difficulty of approximately solving certain combinatorial counting problems.

It seems to be difficult to determine the computational complexity of *Cycle Index Approximation*$(q, y_1, y_2, \ldots)$ when $y_1, y_2, \ldots$ is a fixed sequence such that the conditions in Observation 1, Observation 2, and Theorem 3 are false.

We consider the special case in which $y_1 = y_2 = \cdots = y$ for some positive real number y and we obtain the following theorem.

THEOREM 4 (Goldberg, Jerrum). *If y is a positive real number that is not an integer then* Cycle Index Approximation$(q, y, y, \ldots)$ *is* NP-*hard for every polynomial q.*

It will be clear from the proof of Theorem 4 that our technique does not say anything about the difficulty of *Cycle Index Approximation*$(q, y, y, ...)$ when y is an integer. The condition that y be a non-integer seems rather odd at first but we will see in Section 2 that it is precisely the *integer* values of y for which $P_G(y, y, ...)$ has a combinatorial meaning. Therefore, our theorem leaves open the possibility that the combinatorial interpretation of $P_G(y, ..., y)$ in the integer case could be exploited to provide a fast algorithm.

The structure of this paper is the following: Section 2 describes the combinatorial significance of the cycle index polynomial and, therefore, the significance of our results. In that section, we derive some corollaries of Theorems 2 and 3 which relate to the difficulty of counting equivalence classes of combinatorial structures. Section 3 discusses the computational difficulty of evaluating the cycle index polynomial. It contains a proof of Theorems 1 and 2. Finally, Section 4 discusses the difficulty of approximately evaluating the cycle index polynomial. It contains the proofs of Theorems 3 and 4.

Before considering the combinatorial significance of the cycle index polynomial, we state two definitions which are used throughout the paper.

1. The *cycle bound* of a permutation group is the length of the longest cycle of a permutation in the group. That is, the cycle bound of G is the maximum over all permutations $g \in G$ of the maximum i such that $c_i(g) > 0$.

2. Let ID_j denote the trivial group of permutations of $\{1, ..., j\}$. (That is, let ID_j consist of the identity permutation on $\{1, ..., j\}$.) Let G be any group of permutations of $\{1, ..., n\}$. The *Kranz Group* $G[ID_j]$ [DeB 64] is the group of permutations of $\{\langle a, b\rangle \mid 1 \leqslant a \leqslant n,\ 1 \leqslant b \leqslant j\}$ with the following description. Each permutation $g \in G$ corresponds to exactly one permutation $g[ID_j] \in G[ID_j]$. If g maps the object o_1 to o_2 then $g[ID_j]$ maps $\langle o_1, l\rangle$ to $\langle o_2, l\rangle$ for $1 \leqslant l \leqslant j$. We will use the fact that $P_G(y_1^j, ..., y_n^j) = P_{G[ID_j]}(y_1, ..., y_n)$.

Having stated these definitions, we proceed to consider the combinatorial significance of the cycle index polynomial.

2. The Combinatorial Significance of the Cycle Index Polynomial

In order to explain the combinatorial significance of our results, we must attach a combinatorial meaning to the evaluation of the cycle index polynomial. There are numerous papers and books that discuss the significance of this polynomial (see, for example, [Rea 87]). Therefore, we provide only

a brief discussion of the relevant counting problems here. The material in this section is based on De Bruijn's generalization of Pólya's theory of counting and is taken from [DeB 63, DeB 64].

The basic framework is the following. Suppose that we have an n-object set $N = \{1, ..., n\}$ and an m-object set $M = \{1, ..., m\}$. Let $\mathcal{F}$ denote the set of functions from N to M and let G be a group of permutations of N. We say that two functions f_1 and f_2 are equivalent whenever there exists $g \in G$ such that $f_1 g = f_2$. Let fG denote the equivalence class of f.

Pólya's theorem shows that the number of equivalence classes in $\mathcal{F}$ is $P_G(m, ..., m)$. So the problem of counting the equivalence classes in $\mathcal{F}$ is the same as the problem of evaluating the cycle index polynomial of G at the point $(m, ..., m)$. Using Pólya's theorem, we derive the following corollary of theorem 2.

COROLLARY 1. *Let $m > 1$ be a fixed integer. The following problem is # P-hard.*

Input: *A set of generators for a degree n permutation group G*

Output: *The number of equivalence classes (under G) in the set of functions*

$$\mathcal{F}: \{1, ..., n\} \to \{1, ..., m\}.$$

In order to attach combinatorial meanings to the cycle index evaluation problems that are associated with sequences other than $m, m, ...$, we consider a generalization of the basic framework. Suppose that h is a permutation of M. We can define an induced permutation on the equivalence classes of $\mathcal{F}$ that maps fG to hfG. We say that an equivalence class fG is *invariant* with respect to h if $fG = hfG$. The equivalence class of f is invariant with respect to h whenever it is the case that f is equivalent to hf.

De Bruijn shows that the number of equivalence classes in $\mathcal{F}$ that are invariant with respect to h is $P_G(y_1, ..., y_n)$, where y_j denotes the number of objects $k \in M$ such that $h^j(k) = k$. Therefore, we can compute the number of equivalence classes in $\mathcal{F}$ that are invariant with respect to h by evaluating a cycle index polynomial. Using De Bruijn's theorem, we derive the following corollary of Theorem 2.

COROLLARY 2. *Let $m > 1$ be a fixed integer and let h be any fixed permutation of $\{1, ..., m\}$. The following problem is # P-hard*:

Input: *A set of generators for a degree n permutation group G*

Output: *The number of equivalence classes in the set of functions $\mathcal{F}: \{1, ..., n\} \to \{1, ..., m\}$ that are invariant with respect to h.*

Proof. Let y_j denote the number of objects $k \in M$ such that $h^j(k) = k$. By De Bruijn's theorem, the number of equivalence classes in $\mathcal{F}$ that are invariant with respect to h is $P_G(y_1, ..., y_n)$. Suppose that y_1 is zero or one. Let i be the order of h. Then $y_i = m$ so $y_i \neq y_1^i$ and $y_i \neq 0$. The corollary follows from Theorem 2. So, suppose that $y_1 > 1$. Let p be a prime number that is larger than m. It is easy to see that $y_p = y_1$. We conclude that $y_p \neq y_1^p$ and that $y_p \neq 0$. The corollary follows from Theorem 2. ∎

In addition, we can derive a corollary of Theorem 3.

COROLLARY 3. *Let $m > 1$ be a fixed integer and let h be any fixed permutation of $\{1, ..., m\}$. Let y_j denote the number of elements in $\{1, ..., m\}$ that are fixed by h^j. If there exists some i such that $y_i > y_1^i$ then the following problem is* NP*-hard for any polynomial q:*

Input: A set of generators for a degree n permutation group G

Output: A quantity z that is within a factor of $q(n)$ of the number of equivalence classes in the set of functions $\mathcal{F}: \{1, ..., n\} \to \{1, ..., m\}$ that are invariant with respect to h.

The condition that there exists an i such that $y_i > y_1^i$ restricts the values of m and h to which the NP-hardness result applies. This restriction makes Corollary 3 more difficult to appreciate than Corollaries 1 and 2, so it is worth considering a special case. Suppose that G is a group of permutations of $\{1, ..., n\}$ and that h is the permutation $(1, 2)$ acting on the set $\{1, 2\}$. Let $\mathcal{F}$ be the set of functions from $\{1, ..., n\}$ to $\{1, 2\}$. We say that a function $f \in \mathcal{F}$ is *self-complementary* if and only if f is equivalent to hf. We say that the equivalence class fG is self-complementary whenever its members are. We can apply Theorem 3 directly to the problem of counting self-complementary equivalence classes, obtaining the following corollary:

COROLLARY 4. *The following problem is* NP*-hard for any polynomial q:*

Input: A set of generators for a degree n permutation group G.

Output: A quantity z that is within a factor of $q(n)$ of the number of self-complementary equivalence classes in the set of functions $\mathcal{F}: \{1, ..., n\} \to \{1, 2\}$.

Proof. By De Bruijn's theorem, the number of self-complementary equivalence classes is equal to $P_G(y_1, y_2, ...)$, where y_j denotes the number of objects $k \in \{1, 2\}$ that are fixed by $(1, 2)^j$. It is easy to see that $y_j = 0$ if j is an odd number and that $y_j = 2$ otherwise. Therefore, the number of self-complementary equivalence classes is $P_G(0, 2, 0, 2, ...)$. ∎

Corollaries 1–4 relate the problem of evaluating the cycle index polynomial to the problem of counting equivalence classes of combinatorial

structures. In the remainder of this paper we will leave aside the equivalence classes and we will focus on the cycle index polynomial. In order to make the material from this section more concrete, however, we conclude the section by giving an example. We show how to use Pólya's theorem and De Bruijn's theorem to count unlabeled graphs and self-complementary graphs.

First, observe that we can encode an undirected graph with v vertices as a function from the $\binom{v}{2}$ unordered pairs of vertices to the set $\{1, 2\}$. (A given pair of vertices is mapped to 2 if it is an edge and to 1 otherwise.) If we let G_v be the group of all permutations of unordered *pairs* of vertices that can be produced by permuting the v vertices of a graph then (by Pólya's theorem) the number of isomorphism classes of v-vertex graphs is $P_{G_v}(2, ..., 2)$. Therefore, we can compute the number of unlabeled v-vertex graphs by evaluating the cycle index polynomial of the appropriate group at the point $(2, ..., 2)$.

A graph is *self-complementary* if it is isomorphic to the graph that is obtained by turning all of its edges into non-edges and its non-edges into edges. (For example, the graph with vertices v_1, v_2, v_3, and v_4 and edges (v_1, v_2), (v_2, v_3), and (v_3, v_4) is self-complementary.) The number of isomorphism classes of graphs that are self-complementary is simply the number of equivalence class of encoded graphs that are invariant with respect to the permutation $(1, 2)$. Therefore, by De Bruijn's theorem, the number of unlabed v-vertex graphs that are self-complementary is $P_{G_v}(0, 2, 0, 2, ...)$.

Now that we have given a combinatorial meaning to the evaluation of the cycle index polynomial, we proceed to consider the computational difficulty of performing the evaluation.

3. The Difficulty of Evaluating the Cycle Index Polynomial

In this section we focus on the computational difficulty of the problem *Cycle Index Evaluation*$(y_1, y_2, ...)$. We start with some observations that place upper bounds on the difficulty of the problem.

Observation 1. Let $y_1, y_2, ...$ be a fixed sequence of non-negative real numbers such that for every positive integer j we have $y_j = y_1^j$. Then $P_G(y_1, ..., y_n) = y_1^n$. Therefore, *Cycle Index Evaluation*$(y_1, y_2, ...)$ can be solved in polynomial time.

Observation 2. Let $y_1, y_2, ...$ be a fixed sequence of non-negative real numbers such that for every integer $j > 1$ we have $y_j = 0$. Then $P_G(y_1, ..., y_n) = y_1^n / |G|$. Therefore, *Cycle Index Evaluation*$(y_1, y_2, ...)$ can be solved in polynomial time.

Observation 3. Let $y_1, y_2, \ldots$ be a fixed sequence of non-negative integers. The following problem is in $\#\mathrm{P}$.

Input: A set of generators for a degree n permutation group G.

Output: $\sum_{g \in G} y_1^{c_1(g)} \cdots y_n^{c_n(g)}$.

The main results of this section are Theorems 1 and 2. We begin our presentation of these results by setting up the framework for the proof of Theorem 1. Then we prove a slightly stronger version of Theorem 1 than the version stated in the introduction. Finally, we use the strengthened version of the theorem to prove Theorem 2.

Suppose that p is a prime number, that k is a positive integer, and that i is an integer such that $i \not\equiv 0 \pmod{p}$. We use the notation $\Upsilon(i, p, k)$ to stand for the size of the set $\{m \mid (0 \leqslant m < pk)$ and $(m \equiv i \pmod{p})$ and $(\gcd(m, pk) = 1)\}$. We will use the following fact in the proof of Theorem 1.

FACT 1. *Let p be a prime number. Let k be a positive integer and let i and j be integers such that $i \not\equiv 0 \pmod{p}$ and $j \not\equiv 0 \pmod{p}$. Then $\Upsilon(i, p, k) = \Upsilon(j, p, k) \neq 0$.*

Proof. Let k' be the positive integer such that $k = p^\alpha k'$ for some $\alpha \geqslant 0$ and $\gcd(p, k') = 1$. Since the integers mod p form a field, there is a non-negative integer λ such that $0 \leqslant \lambda < p$ and $i + \lambda k' \equiv j \pmod{p}$.

We use the notation $S_{i,p,k}$ to represent the set $\{[lp + i] \bmod pk \mid l \in \mathbb{N}\}$. Using this notation, we see that $\Upsilon(i, p, k) = |\{m \in S_{i,p,k} \mid \gcd(m, pk) = 1\}|$. Furthermore, $S_{j,p,k} = \{[lp + i + \lambda k'] \bmod pk \mid l \in \mathbb{N}\}$.

The equality of $\Upsilon(i, p, k)$ and $\Upsilon(j, p, k)$ follows from the fact that $\gcd([lp + i] \bmod pk, pk) = \gcd(lp + i, pk) = \gcd(lp + i, k') = \gcd(lp + i + \lambda k', k') = \gcd([lp + i + \lambda k'] \bmod pk, pk)$.

The fact that $\Upsilon(i, p, k) \neq 0$ follows from the fact that $\Upsilon(1, p, k) \neq 0$. ∎

We have now established the fact that the value of $\Upsilon(i, p, k)$ does not depend upon i (so long as $i \not\equiv 0 \pmod{p}$). Therefore, we will drop the parameter "i," and we will refer to $\Upsilon(p, k)$. Using Fact 1, we can now prove Theorem 1:

THEOREM 1. *Let $i > 1$ be a fixed positive integer. The following problem is $\#\mathrm{P}$-hard:*

Cycle-Bounded Cycle Index Coefficient(i)

Input: A set of generators for a permutation group G whose cycle bound is i, and whose degree, n, is a multiple of i.

Output: The coefficient of $x_i^{n/i}$ in the cycle index polynomial of G.

As we pointed out in the introduction, the coefficient of $x_i^{n/i}$ in P_G is $1/|G|$ times the number of permutations in G that have n/i cycles of length i. Therefore, *Cycle-Bounded Cycle Index Coefficient*(i) is polynomially equivalent to the following problem:

Input: A set of generators for a permutation group G whose cycle bound is i, and whose degree, n, is a multiple of i.

Output: The number of permutations in G that have n/i cycles of length i.

The #P-hardness of this problem is established by considering three cases. Let p be a prime number and k be a positive integer such that $i = pk$. Lemma 1 establishes the #P-hardness of *Cycle-Bounded Cycle Index Coefficient*(i) for $p > 3$. Lemma 2 establishes the result for $p = 3$ and Lemma 3 establishes the result for $p = 2$.

LEMMA 1. *Let $p > 3$ be a fixed prime and let k be any fixed positive integer. The following problem is #P-hard:*

Input: A set of generators for a permutation group G whose cycle bound is pk, and whose degree, n, is a multiple of pk.

Output: The number of permutations in G that have n/pk cycles of length pk.

Proof. For any integer $l \geqslant 3$ the following problem is #P-hard [Edw 86]:

#Grap l-Colorability

Input: An undirected graph Γ.

Output: The number of l-colorings[2] of Γ.

We proceed by reduction from *#Graph $(p-1)$-Colorability*. Suppose that we have a graph Γ with vertex set $\{v_1, ..., v_v\}$ and edge set $\{e_1, ..., e_\mu\}$. We construct a permutation group G by using the following method.

1. For each vertex v_i, we introduce a set V_i of pk objects and a permutation g_{v_i} that cycles them.

2. For each edge e_j, we introduce a set E_j of pk objects and a permutation g_{e_j} that cycles them.

3. Let G be the group generated by the following three sets:

[2] An l-coloring of a graph is an assignment of a color from the set $\{1, ..., l\}$ to each vertex in the graph in such a way that no two adjacent vertices receive the same color.

(i) $\bigcup_i \{g_{v_i}^p\}$
(ii) $\bigcup_j \{g_{e_j}^p\}$
(iii) $\{g_i \mid g_i = g_{v_i} g_{e_\alpha} g_{e_\beta} \cdots g_{e_a}^{-1} g_{e_b}^{-1}\}$

where v_i is a vertex of Γ and it is the vertex of smaller index in edges e_α, e_β, ... and the vertex of larger index in edges e_a, e_b, ...

We claim that each $(p-1)$-coloring of Γ corresponds to a set of $\Upsilon(p,k)^{\nu+\mu}$ permutations in G, each of which has n/pk cycles of length pk. Furthermore, we claim that G has no other permutations that have n/pk cycles of length pk. We prove the claim in two steps.

1. Suppose that we have a $(p-1)$-coloring of Γ and let c_i denote the color of vertex v_i. Since $c_i \not\equiv 0 \pmod p$, we can use Fact 1 to show that there are $\Upsilon(p,k)$ members of the set $\{[c_i + pl] \bmod pk \mid l \in \mathbb{N}\}$ that are relatively prime to pk. Therefore, the set $\{g_{v_i}^{c_i} g_{v_i}^{pl} \mid l \in \mathbb{N}\}$ contains $\Upsilon(p,k)$ permutations that are cycles of length pk.

Suppose that e_j is an edge in Γ whose smaller endpoint is colored with color c and whose larger endpoint is colored with a diffeent color, d. The restriction of the permutation $g_1^{c_1} \cdots g_\nu^{c_\nu}$ to the objects in E_j is $g_{e_j}^{c-d}$. Since $c - d \not\equiv 0 \pmod p$ we can use Fact 1 to show that the set $\{g_{e_j}^{c-d} g_{e_j}^{pl} \mid l \in \mathbb{N}\}$ contains $\Upsilon(p,k)$ permutations that are cycles of length pk.

Finally, we conclude that the set $\{g_1^{c_1} \cdots g_\nu^{c_\nu} g_{v_1}^{pl} \cdots g_{v_\nu}^{pl_\nu} g_{e_1}^{pl'_1} \cdots g_{e_\mu}^{pl'_\mu} \mid l_i, l'_i \in \mathbb{N}\}$ contains $\Upsilon(p,k)^{\nu+\mu}$ permutations which have n/pk cycles of length pk.

2. Suppose that g is a permutation in G which has n/pk cycles of length pk. It is easy to see that we can rewrite g as $g_1^{c_1} \cdots g_\nu^{c_\nu} g_{v_1}^{pl_1} \cdots g_{v_\nu}^{pl_\nu} g_{e_1}^{pl'_1} \cdots g_{e_\mu}^{pl'_\mu}$, where $0 \leqslant c_1, ..., c_\nu < p$ and $l_i, l'_i \in \mathbb{N}$. Since the restriction of g to the objects in V_i is a cycle of length pk, it must be the case that $c_i \neq 0$ for all i. Consider the function that assigns color c_i to vertex v_i for each i. We must show that this function is a coloring of Γ.

Suppose that there is an edge e_j whose vertices are both assigned the same color. Then the restriction of g to the objects in E_j is $g_{e_j}^{pl'_j}$. Now $gcd(pl'_j, pk) \neq 1$. Therefore, the restriction of g to the objects in E_j is not a cycle of length pk, which is a contradiction. ∎

LEMMA 2. *Let k be any fixed positive integer. The following problem is # P-hard:*

Input: A set of generators for a permutation group G whose cycle bound is $3k$, and whose degree, n, is a multiple of $3k$.

Output: The number of permutations in G that have $n/3k$ cycles of length $3k$.

Proof. This proof is very similar to the proof of Lemma 1. We start by observing that the following problem is # P-hard:

 # Not-All-Equal 3Sat

Input: A set U of Boolean variables and a collection C of clauses over U, each of which contains three literals.

Output: The number of assignments of truth values to the variables that have the property that the number of "true" literals in any given clause is either one or two.

To see that *# Not-All-Equal 3Sat* is # P-hard, recall that the following problem is # P-hard [Val 79].

 # Monotone 2Sat

Input: A Set U of Boolean variables and a collection C of clauses over U, each of which contains two variables.

Output: The number of assignments of truth values to the variables that have the property that the number of "true" literals in any given clause is either one or two.

Let $\langle U, C \rangle$ be an input to *# Monotone 2Sat*. Let $U' = U \cup \{x\}$ for some variable x that is not in U and let $C' = \{c \cup \{x\} \mid c \in C\}$. The assignments of truth values to the variables in U that are counted in the output *# Monotone* $2Sat(U, C)$ are in one-to-one correspondence with the assignments of truth values to the variables in U' that are counted in *# Not-All-Equal* $3Sat(U', C')$ and have $x =$ "false." The result follows from the fact that $x =$ "false" in exactly half of the assignments that are counted in *# Not-All-Equal* $3Sat(U', C')$, which follows from the fact that the definition of the problem *# Not-All-Equal 3Sat* does not change if we substitute "false" for "true."

Now that we have established the # P-hardness of *# Not-All-Equal 3Sat*, we proceed by reduction from this problem. Suppose that we have a set $U = \{u_1, ..., u_v\}$ of variables and a collection $\{c_1, ..., c_\mu\}$ of clauses over U. We construct a permutation group G by using the following method.

1. For each variable u_i, we introduce a set U_i of $3k$ objects and a permutation g_{u_i} that cycles them.

2. For each clause c_j, we introduce a set C_j of $3k$ objects and a permutation g_{c_j} that cycles them.

3. Let G be the group generated by the following three sets:

(i) $\bigcup_i \{g_{u_i}^3\}$

(ii) $\bigcup_j \{g_{c_j}^3\}$

(iii) $\{g_i \mid g_i = g_{u_i} g_{c_\alpha} g_{c_\beta} \cdots g_{c_a}^{-1} g_{c_b}^{-1}\}$

where u_i is a variable in U that occurs positively in clauses c_α, c_β, ..., and negatively in clauses c_a, c_b, ...

We claim that each assignment of truth values to the variables in U that has the property that the number of "true" literals in any given clause in C is either one or two corresponds to a set of $\Upsilon(3, k)^{\nu+\mu}$ permutations in G, each of which has $n/3k$ cycles of length $3k$. Furthermore, we claim that G has no other permutations that have $n/3k$ cycles of length $3k$. We prove the claim in two steps.

1. Suppose that we have an assignment of truth values to the variables in U. Let t_i be 1 if the variable u_i is assigned the value "true" and -1 otherwise. Since $t_i \not\equiv 0 \pmod 3$ we can use fact 1 to show that the set $\{g_{u_i}^{t_i} g_{u_i}^{3l} | l \in \mathbb{N}\}$ contains $\Upsilon(3, k)$ permutations that are cycles of length $3k$.

Consider any clause c_j. If u_i or $\overline{u_i}$ is a "true" literal in c_j then the restriction of $g_i^{t_i}$ to the objects in C_j is g_{c_j}. If u_i or $\overline{u_i}$ is a "false" literal in c_j then the restriction of $g_i^{t_i}$ to the objects in C_j is $g_{c_j}^{-1}$.

Now, suppose that exactly one of the literals in c_j is "true." In this case the restriction of $g_1^{t_1} \cdots g_\nu^{t_\nu}$ to the objects in C_j is $g_{c_j}^{-1}$. Since $-1 \not\equiv 0 \pmod 3$ we can use Fact 1 to show that the set $\{g_{c_j}^{-1} g_{c_j}^{3l} | l \in \mathbb{N}\}$ contains $\Upsilon(3, k)$ permutations that are cycles of length $3k$. Alternatively, suppose that exactly two of the literals in c_j are "true." In this case the restriction of $g_1^{t_1} \cdots g_\nu^{t_\nu}$ to the objects in C_j is g_{c_j}. Since $1 \not\equiv 0 \pmod 3$ we can use Fact 1 to show that the set $\{g_{c_j} g_{c_j}^{3l} | l \in \mathbb{N}\}$ contains $\Upsilon(3, k)$ permutations that are cycles of length $3k$.

Finally, we conclude that the set $\{g_1^{t_1} \cdots g_\nu^{t_\nu} g_{u_1}^{3l_1} \cdots g_{u_\nu}^{3l_\nu} g_{c_1}^{3l'_1} \cdots g_{c_\mu}^{3l'_\mu} | l_i, l'_i \in \mathbb{N}\}$ contains $\Upsilon(3, k)^{\nu+\mu}$ permutations which have $n/3k$ cycles of length $3k$.

2. Suppose that g is a permutation in G which has $n/3k$ cycles of length $3k$. It is easy to see that we can rewrite g as $g_1^{t_1} \cdots g_\nu^{t_\nu} g_{u_1}^{3l_1} \cdots g_{u_\nu}^{3l_\nu} g_{c_1}^{3l'_1} \cdots g_{c_\mu}^{3l'_\mu}$, where $t_1, ..., t_\nu \in \{-1, 0, 1\}$ and $l_i, l'_i \in \mathbb{N}$. Since the restriction of g to the objects in U_i is a cycle of length $3k$, it must be the case that $t_i \neq 0$ for all i. Consider the truth assignment that gives u_i the value "true" if t_i is 1 and "false" otherwise. We must show that one or two literals are "true" in any given clause.

Suppose that c_j is a clause with three "true" literals. Then the restriction of g to the objects in C_j is $g_{c_j}^{3+3l'_j}$. Since $\gcd(3 + 3l'_j, 3k) \neq 1$, the restriction of g to the objects in C_j is not a cycle of length $3k$, which is a contradiction. Similarly, if C_j has three "false" literals then the restriction of g to the objects in C_j is $g_{c_j}^{-3+3l'_j}$. Since $\gcd(-3 + 3l'_j, 3k) \neq 1$, the restriction of g to the objects in C_j is not a cycle of length $3k$. Once again, we get a contradiction. ∎

LEMMA 3. *Let k be any fixed positive integer. The following problem is #P-hard:*

Input: A set of generators for a permutation group G whose cycle bound is 2k, and whose degree, n, is a multiple of 2k.

Output: The number of permutations in G that have n/2k cycles of length 2k.

Proof. Lubiw's proof that *# Fixed-Point-Free Automorphism* is *# P*-hard[3] [Lub 81] establishes the lemma for the case $k = 1$. Following Lubiw, we proceed by reduction from the following # P-hard problem [Val 79]:

 # Satisfiability

Input: A Set U of Boolean variables and a collection C of clauses over U, each of which contains three literals.

Output: The number of assignments of truth values to the variables that have the property that each clause has at least one "true" litral.

Our clause checker will be a generalization of Lubiw's, so we use the following gadget (which was used in her paper).

GADGET 1. Let H be the group of permutations of $\{1, ..., 8\}$ that is generated by

$$h[1] = (1\ 2)(3\ 4)(5\ 6)(7\ 8)$$

$$h[2] = (1\ 3)(2\ 4)(5\ 7)(6\ 8)$$

$$h[3] = (1\ 5)(2\ 6)(3\ 7)(4\ 8).$$

This group is the commutative. Therefore, $H = \{h[1]^i\, h[2]^j\, h[3]^k \mid i, j, k \in \{0, 1\}\}$. Every member of H except the identity is the product of four transpositions $(i_1\ i_2)(i_3\ i_4)(i_5\ i_6)(i_7\ i_8)$, where $i_1, ..., i_8$ is a permutation of 1, ..., 8.

In order to generalize to the case $k > 1$, we need an additional gadget.

GADGET 2. Let S_l be a set of k objects $S_l[1], ..., S_l[k]$.
Let S_m be a set of k objects $S_m[1], ..., S_m[k]$.
Let Swap(lm) represent the permutation $(S_l[1]\ S_m[1]) \cdots (S_l[k]\ S_m[k])$.
Let Cycle(l) represent the permutation $(S_l[1] \cdots S_l[k])$.
It is easy to prove the following identity:[4]

$$\begin{aligned}\text{Cycle}(m)\,\text{Swap}(lm) &= \text{Swap}(lm)\,\text{Cycle}(l)\\ &= (S_m[1]\ S_l[1] \cdots S_m[k]\ S_l[k]).\end{aligned}$$

[3] Since every permutation in Lubiw's group has cycle bound 2, her proof actually shows that *Cycle Index Evaluation*$(y_1, y_2, ...)$ is # P-hard whenever $y_1 = 0$ and $y_2 \neq 0$.

[4] Note that permutations are being composed from right to left.

Let J' be the group $\langle \text{Cycle}(l), \text{Cycle}(m) \rangle$[5] and let $J = \langle \text{Swap}(lm), \text{Cycle}(l), \text{Cycle}(m) \rangle$. Using the identity, it is easy to see that $J = J' \cup \{\text{Swap}(lm)\,\lambda \mid \lambda \in J'\}$. Clearly, J' has no cycle of length $2k$. We know from the identity that at least one member of J is a cycle of length $2k$, however, Let $\Upsilon(k)$ denote the number of members of J that are cycles of length $2k$.

We are now ready to proceed. Suppose that we have a set $U = \{u_1, ..., u_v\}$ of variables and a collection $\{c_1, ..., v_\mu\}$ of clauses over U. We construct a permutation group G by using the following method.

1. For each variable u_i, we introduce a set U_i of $2k$ objects and a permutation g_{u_i} that cycles them.

2. For each clause c_j, we introduce eight sets of objects, $C_{j1}, ..., C_{j8}$. Each set C_{jl} contains k objects, $C_{jl}[1], ..., C_{jl}[k]$. Using the notation that we defined in our description of gadget 2, we let $\text{Swap}_j(lm)$ represent the permutation $(C_{jl}[1]\, C_{jm}[1]) \cdots (C_{jl}[k]\, C_{jm}[k])$ and we let $\text{Cycle}_j(l)$ represent the permutation $(C_{jl}[1] \cdots C_{jl}[k])$. We introduce three permutations:

$$h_j[1] = \text{Swap}_j(12)\,\text{Swap}_j(34)\,\text{Swap}_j(56)\,\text{Swap}_j(78)$$

$$h_j[2] = \text{Swap}_j(13)\,\text{Swap}_j(24)\,\text{Swap}_j(57)\,\text{Swap}_j(68)$$

$$h_j[3] = \text{Swap}_j(15)\,\text{Swap}_j(26)\,\text{Swap}_j(37)\,\text{Swap}_j(48).$$

3. Let G' be the group generated by $\bigcup_j \{\text{Cycle}_j(1), ..., \text{Cycle}_j(8)\}$.

4. Let G be the group generated by the following four sets:

   (i) $\bigcup_i \{g_{u_i}^2\}$

   (ii) $\{\lambda \mid \lambda \in G'\}$

   (iii) $\{\pi_i \mid \pi_i = g_{u_i} h_j[1]\, h_k[1] \cdots h_l[2]\, h_m[2] \cdots h_y[3]\, h_z[3] \cdots\}$

where u_i occurs in position 1 in clauses $c_j, c_k, ...$ and in position 2 in clauses $c_l, c_m, ...$ and in position 3 in clauses $c_y, c_z, ...$.

   (iv) $\{\rho_i \mid \rho_i = g_{u_i} h_j[1]\, h_k[1] \cdots h_l[2]\, h_m[2] \cdots h_y[3]\, h_z[3] \cdots\}$

where $\overline{u_i}$ occurs in position 1 in clauses $c_j, c_k, ...$ and in position 2 in clauses $c_l, c_m, ...$ and in position 3 in clauses $c_y, c_z, ...$.

We claim that each assignment of truth values to the variables in U which has the property that each clause has at least one "true" literal corresponds to a set of $\Upsilon(2, k)^v\, \Upsilon(k)^{4\mu}$ permutations in G, each of which has $n/2k$ cycles of length $2k$. Furthermore, we claim that G has no other

[5] Recall that the notation $\langle g_1, g_2, ... \rangle$ represents the group generated by $g_1, g_2, ...$.

permutations that have $n/2k$ cycles of length $2k$. We prove the claim in two steps.

1. Suppose that we have an assignment of truth values to the variables in U. Let g_i denote π_i if the variable u_i is assigned the value "true" and let g_i denote ρ_i otherwise. The restriction of g_i to the objects in U_i is g_{u_i}. Since $1 \not\equiv 0 \pmod 2$ we can use fact 1 to show that the set $\{g_{u_i} g_{u_i}^{2l} \mid l \in \mathbb{N}\}$ contains $\Upsilon(2, k)$ permutations that are cycles of length $2k$.

Let g be $g_1 \cdots g_v$. Consider any clause c_j and let g_j' denote the restriction of g to the objects associated with c_j. By construction, $h_j[t]$ is a factor of g_j' if and only if the tth literal in c_j has been assigned the value "true." Suppose that at least one of the literals in c_j is "true." Our consideration of Gadget 1 shows that $g_j' = \text{Swap}_j(i_1, i_2)\ \text{Swap}_j(i_3, i_4)\ \text{Swap}_j(i_5, i_6)\ \text{Swap}_j(i_7, i_8)$ where $i_1, \ldots, i_8$ is a permutation of $1, \ldots, 8$. If we consider one of the four factors $\text{Swap}_j(i_l, i_m)$ and the permutations $\text{Cycle}_j(i_l)$ and $\text{Cycle}_j(i_m)$ then we can use our analysis of Gadget 2 to show that the set $\{\text{Swap}_j(i_l, i_m)\,\lambda \mid \lambda \in \langle \text{Cycle}_j(i_l), \text{Cycle}_j(i_m)\rangle\}$ has $\Upsilon(k)$ permutations which are cycles of length $2k$. Therefore, the set $\{g_j'\lambda \mid \lambda \in \langle \text{Cycle}_j(1), \ldots, \text{Cycle}_j(8)\rangle\}$ has $\Upsilon(k)^4$ permutations which are cycles of length $2k$.

Finally, we conclude that the set $\{g_1 \cdots g_v\, g_{u_1}^{2l_1} \cdots g_{u_v}^{2l_v} \lambda \mid l_i \in \mathbb{N},\ \lambda \in G'\}$ contains $\Upsilon(2, k)^v\, \Upsilon(k)^{4\mu}$ permutations which have $n/2k$ cycles of length $2k$.

2. Suppose that g is a permutation in G with cycles of length $2k$. It is easy to see that we can rewrite g as $\pi_1^{t_1} \rho_1^{f_1} \cdots \pi_v^{t_v} \rho_v^{f_v} g_{u_1}^{2l_1} \cdots g_{u_v}^{2l_v} \lambda$, where t_i, $f_i \in \{0, 1\}$, $l_i \in \mathbb{N}$, and $\lambda \in G'$. Since the restriction of g to the objects in U_i is a cycle of length $2k$, it must be the case that one of t_i, f_i is 1 and the other is 0 for each i. Consider the truth asignment that gives u_i the value "true" if t_i is 1 and "false" otherwise. We must show that each clause contains at least one "true" literal.

Suppose that c_j is a clause with no "true" literals. Then none of $h_j[1], \ldots, h_j[3]$ is a factor of g. Therefore, the restriction of g to the objects associated with c_j is not a cycle of length $2k$, which is a contradiction. ∎

Having completed the proof of Theorem 1, we use it to prove the following theorem.

THEOREM 2. *If $y_1, y_2, \ldots$ is a sequence of non-negative real numbers and there exists an i such that $y_i \neq y_1^i$ and $y_i \neq 0$ then* Cycle Index Evaluation$(y_1, y_2, \ldots)$ *is* #P*-hard.*

Proof. First, suppose that $y_1 = 0$. Choose the index i such that for all $j < i$ we have $y_j = 0$ and $y_i \neq 0$. If G is a permutation group whose cycle bound is i and whose degree, n, is a multiple of i then the coefficient of $x_i^{n/i}$ in P_G is $|G|\, y_i^{-n/i} P_G(y_1, \ldots, y_n)$ so the result follows from Theorem 1.

Otherwise, choose the index i such that for every $j<i$ either $y_j=0$ or $y_j=y_1^j$ but $y_i\neq 0$ and $y_i\neq y_1^i$. Let $\hat{G}$ stand for the set

$$\hat{G}=\{g\in G \mid g \text{ has no cycle whose length is a member of } \{j \mid y_j=0\}\}.$$

Let $P'_{G,i}$ be the single-variable polynomial defined by $P'_{G,i}(z)=(1/|G|)\sum_{g\in\hat{G}} z^{c_i(g)}$. If G is a permutation group whose degree, n, is a multiple of i then the coefficient of $z^{n/i}$ in $P'_{G,i}$ is the same as the coefficient of $x_i^{n/i}$ in P_G. Furthermore, we claim that if G has cycle bound i then $P_G(y_1,\ldots,y_n)=y_1^n P'_{G,i}(y_i/y_1^i)$.

Suppose that the claim is true. Suppose further that we could compute the values $P_{G[ID_1]}(y_1,\ldots,y_n)=P_G(y_1^l,\ldots,y_n^l)$ for $1\leqslant l\leqslant n+1$. Then we would be able to evaluate $P'_{G,i}$ at the $n+1$ points $z=(y_i/y_1^i)^l$ for $1\leqslant l\leqslant n+1$. (Note that $y_i/y_1^i\neq 1$ and that $y_i/y_1^i\neq 0$.) We could interpolate to get the coefficient of $z^{n/i}$ in $P'_{G,i}$ which is the coefficient of $x_i^{n/i}$ in P_G. The theorem follows from the proof of the claim (which will be given below) and from the #P-hardness of *Cycle-Bounded Cycle Index Coefficient*(i), which was established in Theorem 1.

To prove the claim we must show that $P_G(y_1,\ldots,y_n)=y_1^n P'_{G,i}(y_i/y_1^i)$. Since the cycle bound of G is i the value of $P_G(y_1,\ldots,y_n)$ can be written as

$$P_G(y_1,\ldots,y_n)=\frac{1}{|G|}\sum_{g\in G} y_i^{c_i(g)}\prod_{1\leqslant j<i} y_j^{c_j(g)}.$$

Note that we can restrict the summation to permutations $g\in\hat{G}$ since all of the terms which are eliminated by this restriction are equal to zero. Using the fact that $0^{c_j(g)}=y_1^{jc_j(g)}$ when $c_j(g)=0$ we can replace the right hand side with

$$P_G(y_1,\ldots,y_n)=\frac{1}{|G|}\sum_{g\in\hat{G}} y_i^{c_i(g)}\prod_{1\leqslant j<i} y_1^{jc_j(g)}.$$

Since $\sum_{j=1}^{i} jc_j(g)=n$ we get

$$P_G(y_1,\ldots,y_n)=\frac{1}{|G|}\sum_{g\in\hat{G}} y_i^{c_i(g)}\, y_1^{n-ic_i(g)}.$$

Simplifying the right hand side we get

$$P_G(y_1,\ldots,y_n)=\frac{y_1^n}{|G|}\sum_{g\in\hat{G}}(y_i/y_1^i)^{c_i(g)}=y_1^n P'_{G,i}(y_i/y_1^i). \quad \blacksquare$$

4. The Difficulty of Approximately Evaluating The Cycle Index Polynomial

In this section we focus on the computational difficulty of *Cycle Index Approximation*$(q, y_1, y_2, \ldots)$. We start with the following lemma.

LEMMA 4. *Let $i > 1$ be a fixed positive integer. The following problem is* NP-*hard*:

Cycles-of-Length(i)

Input: *A set of generators for a permutation group G whose cycle bound is i, and whose degree, n, is a multiple of i.*

Output: "*Yes*," *if G has a permutation that has n/i cycles of length i.* "*No*," *otherwise.*

Proof. It is known [GJ79] that it is NP-hard to decide, given an input for *# Graph l-Colorability*, *# Not-All-Equal 3Sat*, or *# Satisfiability*, whether the corresponding output is zero. Therefore, the lemma follows from the proof of Theorem 1. ∎

Using this lemma, it is easy to prove Theorem 3.

THEOREM 3. *Let $y_1, y_2, \ldots$ be a fixed sequence of non-negative real numbers. If there exists an i such that $y_i > y_1^i$ then* Cycle Index Approximation$(q, y_1, y_2, \ldots)$ *is* NP-*hard for every polynomial q.*

Proof. Choose the index i such that $\forall j < i \cdot y_j \leqslant y_1^j$ and $y_i > y_1^i$. Let G be any input to *Cycles-of-Length*(i) and let n be the degree of G. We make the following observations:

1. If G has a permutation that decomposes $\{1, \ldots, n\}$ into n/i cycles of length i then $P_G(y_1, \ldots, y_n) \geqslant |G|^{-1} y_i^{n/i}$.
2. Otherwise, $P_G(y_1, \ldots, y_n) \leqslant y_i^{n/i-1} y_1^i$.

Let r be a polynomial and recall that $P_G(y_1^{r(n)}, \ldots, y_n^{r(n)}) = P_{G[ID_{r(n)}]}(y_1, \ldots, y_n)$. Using Observations 1 and 2, we conclude:

1. If *Cycles-of-Length*$(i)(G)$ is "Yes," then $P_{G[ID_{r(n)}]}(y_1, \ldots, y_n) \geqslant |G|^{-1} y_i^{r(n) \times n/i}$.
2. Otherwise, $P_{G[ID_{r(n)}]}(y_1, \ldots, y_n) \leqslant |G|^{-1} y_i^{r(n) \times n/i} \times [y_1^i / y_i]^{r(n)} |G|$.

To establish the theorem, we need only choose the polynomial r in such a way that $[y_1^i / y_i]^{r(n)} |G|$ is exponentially small. ∎

We mentioned in the introduction to this paper that it is difficult to determine the computational complexity of *Cycle Index Approximation*$(q, y_1, y_2, \ldots)$

when $y_1, y_2, \ldots$ is a fixed sequence such that $y_j \leqslant y_1^j$ for all j and for some i it is the case that $y_i < y_1^i$. We consider the special case in which $y_1 = y_2 = \cdots = y$ for some positive real number y and we obtain the following theorem.

THEOREM 4 (Goldberg, Jerrum). *If y is a positive real number that is not an integer then* Cycle Index Approximation$(q, y, y, \ldots)$ *is* NP*-hard for every polynomial q.*

Before proving Theorem 4, we set up the framework for the proof. Let S_l stand for the symmetric group of degree l and let A_l stand for the alternating group of degree l. Define the polynomials $f_{C,l}$ and $f_{A,l}$ as follows: $f_{C,l}(x) = \sum_{g \in S_l - A_l} x^{c(g)}$ and $f_{A,l}(x) = \sum_{g \in A_l} x^{c(g)}$. We use the following fact:

FACT 2. *Suppose that y is a positive real number that is not an integer and that $l = \lceil y \rceil + 1$. Then $f_{C,l}(y) > f_{A,l}(y)$.*

Proof. Let $f_l(x) = f_{A,l}(x) - f_{C,l}(x)$. It is easy to see that the coefficient of x^l in $f_{A,l}$ is 1 and that the degree of $f_{A,l}$ is l. The degree of $f_{C,l}$ is less than l. Therefore, f_l is a degree l polynomial and for big enough values of i, $f_l(i)$ is positive. Suppose that i is an integer such that $0 \leqslant i < l$. We claim that $f_l(i) = 0$. (To see that the claim is correct, use Pólya's theorem to show that $P_{S_l}(i) = P_{A_l}(i)$ for every integer i such that $0 \leqslant i < l$. Then use the definition of the cycle index polynomial, observing that $|S_l| = 2 \times |A_l|$.) Since a degree l polynomial has at most l zeros, we conclude that $f_l(i)$ is negative in the range $l - 2 < i < l - 1$, which establishes the fact. ∎

Using Fact 2, it is not hard to prove Theorem 4.

Proof of Theorem 4. Suppose that y is a positive real number that is not an integer and let $l = \lceil y \rceil + 1$. Fact 2 shows that $f_{C,l}(y) > f_{A,l}(y)$. Let r be a polynomial such that $[f_{A,l}(y)/f_{C,l}(y)]^{r(v)} 2^v$ is exponentially small (as a function of v). We proceed by reduction from the following NP-hard problem [GJ 79]:

Simple Max Cut

Input: A connected graph Γ and a positive integer k.

Output: "Yes," if Γ has a cut-set[6] whose size is at least k. "No," otherwise.

Suppose that we have a graph Γ with vertex set $\{v_1, \ldots, v_v\}$ and edge set $\{e_1, \ldots, e_\mu\}$. We construct a permutation group G using the following method:

[6] A size k cut-set of Γ is a partition of the vertices of Γ into two disjoint (and indistinguishable) subsets such that the number of edges which span the two subsets is k.

1. For each edge e_j, we introduce $r(v)$ sets of objects, $E_j[1], \ldots, E_j[r(v)]$. Each set $E_j[\kappa]$ contains l objects. We use the notation $A_j[\kappa]$ to stand for the alternating group of degree l acting on the objects in $E_j[\kappa]$.

2. For each vertex v_i, we let g_{v_i} be the permutation which transposes the first two objects in each set $E_j[\kappa]$ such that e_j is incident on v_i and $1 \leqslant \kappa \leqslant r(v)$.

3. Let G' be the group generated by $\{\lambda \in A_j[\kappa] \mid 1 \leqslant j \leqslant \mu, 1 \leqslant \kappa \leqslant r(v)\}$.

4. Let G be the group generated by $\{g_{v_i} \mid 1 \leqslant i \leqslant v\} \cup \{\lambda \mid \lambda \in G'\}$.

Each permutation $g \in G$ corresponds to exactly one (unordered) partition (S, T) of the vertices in Γ and to one permutation $\lambda \in G'$. g can be written as $\prod_{v_i \in S} g_{v_i}\lambda$ and as $\prod_{v_i \in T} g_{v_i}\lambda$. We associate g with the cut-set (S, T). Consider an edge e_j with endpoints v_α and v_β and let $g_{j,\kappa}$ be the restriction of g to the objects in $E_j[\kappa]$. It is easy to see that $g_{j,\kappa} \in S_l - A_l$ if exactly one of v_α, v_β is in S and that $g_{j,\kappa} \in A_l$ otherwise. That is, $g_{j,\kappa} \in S_l - A_l$ if e_j spans the two subsets of the cut-set that is associated with g and $g_{j,\kappa} \in A_l$ otherwise.

Let (S, T) be a cut-set of Γ and let $G(S, T)$ stand for the set of permutations in G that are associated with the cut-set (S, T). Suppose that the size of the cut-set (S, T) is k. It is not difficult to see that $\sum_{g \in G(S,T)} x^{c(g)} = f_{C,l}(x)^{r(v)k} f_{A,l}(x)^{r(v)(\mu-k)}$.

We make the following observations:

1. If Γ has a cut-set (S, T) whose size, k', is at least k, then

$$P_G(y, \ldots, y) \geqslant |G|^{-1} f_{C,l}(y)^{r(v)k'} f_{A,l}(y)^{r(v)(\mu-k')}.$$

Fact 2 shows that $f_{C,l}(y) > f_{A,l}(y)$. Therefore,

$$P_G(y, \ldots, y) \geqslant |G|^{-1} f_{C,l}(y)^{r(v)k} f_{A,l}(y)^{r(v)(\mu-k)}.$$

2. If Γ does not have a cut-set whose size is at least k, then

$$\begin{aligned} P_G(y, \ldots, y) &\leqslant 2^v |G|^{-1} f_{C,l}(y)^{r(v)(k-1)} f_{A,l}(y)^{r(v)(\mu-k+1)} \\ &= |G|^{-1} f_{C,l}(y)^{r(v)k} f_{A,l}(y)^{r(v)(\mu-k)} \\ &\quad \times [f_{A,l}(y)/f_{C,l}(y)]^{r(v)} 2^v. \end{aligned}$$

The proof is concluded by observing that we have chosen the polynomial r in such a way that $[f_{A,l}(y)/f_{C,l}(y)]^{r(v)} 2^v$ is exponentially small. (We chose r so that the relevant quantity was exponentially small as a function of v. By construction, it is also exponentially small as a function of the degree of G.) ∎

As we pointed out in the introduction to this paper, our proof of Theorem 4 says nothing about the difficulty of *Cycle Index Approximation*$(q, y, y, ...,)$ when y is an *integer*. Furthermore, it is the *integer* values of y for which $P_G(y, ..., y)$ has a combinatorial meaning. It is an interesting open problem to determine the computational difficulty of *Cycle Index Approximation*$(q, y_1, y_2, ...)$ when $y_j \leqslant y_1^j$ for all j and there exists an i such that $y_i < y_1^i$. It would also be interesting to determine the difficulty of *Cycle Index Approximation*$(q, y, y, ...)$ for integer values of y.

ACKNOWLEDGMENTS

I am grateful to Paul Goldberg who helped with the proof of fact 1, to Alistair Sinclair who read the paper and provided useful comments, to an anonymous referee who found an error in the original proof of Theorem 2, and to my advisor, Mark Jerrum, who suggested the problems that are considered in this paper and provided many helpful suggestions. Theorem 4 is joint work with Mark Jerrum.

RECEIVED November 9, 1990; FINAL MANUSCRIPT RECEIVED June 18, 1991

REFERENCES

[DeB 63] DE BRUIJN, N. G. (1963), Enumerative combinatorial problems concerning structures, *Nieuw Arch. Wisk.* **3**, No. 11, 142–161.

[DeB 64] DE BRUIJN N. G. (1964), Pólya's theory of counting, *in* "Applied Combinatorial Mathematics" (E. F. Beckenbach, Ed.), Wiley, New York.

[Edw 86] EDWARDS, K. (1986), The complexity of colouring problems on dense graphs, *Theoret. Comput. Sci.* **43**, 337–343.

[GJ 79] GAREY, M. R., AND JOHNSON, D. S. (1979), Computers and Intractability: A Guide to the Theory of NP-Completeness," Freeman.

[Jer 86] JERRUM, M. R. (1986), A compact representation for permutation groups, *J. Algorithms* **7**, 60–78.

[Lub 81] LUBIW, A. (1981), Some NP-complete problems similar to graph isomorphism, *SIAM J. Comput.* **10**, No. 1, 11–21.

[Rea 87] READ, R. C. (1987), The legacy of Pólya's paper: Fifty years of Pólya theory, *in* "Combinatorial Enumeration of Groups, Graphs, and Chemical Compounds" (G. Pólya and R. C. Read, Eds.), Springer-Verlag, Berlin/New York.

[Val 79] VALIANT, L. G. (1979), The complexity of enumeration and reliability problems, *SIAM J. Comput.* **8**, No. 3, 410–421.