Approximate Counting via Random Optimization

Alexander Barvinok[1]

[1]*Department of Mathematics, University of Michigan, Ann Arbor, MI 48109-1109*

Received 11 November 1996; accepted 13 February 1997

ABSTRACT: Let $\mathscr{F}_n$ be a family of subsets of $\{1,\dots,n\}$. We propose a simple randomized algorithm to estimate the cardinality of $\mathscr{F}_n$ from the maximum weight of a subset $X \in \mathscr{F}_n$ in a random weighting of $\{1,\dots,n\}$. The examples include enumeration of perfect matchings in graphs, bases in matroids, and Hamiltonian cycles in graphs. © 1997 John Wiley & Sons, Inc. *Random Struct. Alg.*, **11**, 187–198 (1997)

Key Words: approximate counting; average case in optimization; concentration of measure

1. INTRODUCTION AND MAIN RESULTS

Let $\mathscr{F}_n$ be a family of subsets of the set $\{1,\dots,n\}$. In this paper we describe a very simple method which allows us to estimate the cardinality $|\mathscr{F}_n|$ by solving a randomly generated optimization problem on $\mathscr{F}_n$. The method produces a very crude estimate, but catches some general information about $|\mathscr{F}_n|$ for large n. For example, it allows us to determine whether $|\mathscr{F}_n|$ is exponentially large in n.

The known methods of approximate counting based on the Markov chain approach are much more precise. Their application, however, relies on the ability to generate a rapidly mixing Markov chain on $\mathscr{F}_n$. In many important cases the existence of such rapidly mixing Markov chains is not known. Our method does not require any special structural properties of $\mathscr{F}_n$ but, as we mentioned above, it is not nearly as precise.

Contract grant sponsor: Alfred P. Sloan Fellowship
Contract grant sponsor: NSF; contract grant number DMS 95-01129

As a general reference in the area of computational complexity and algorithms, we use [10].

Let us assume that the family $\mathscr{F}_n$ is given by its *optimization oracle*.

1.1. Optimization Oracle

We suppose that we have a "black box" which, for any given weighting $c_i \in \{-1, 1\}$, $i = 1, \ldots, n$, outputs the maximum weight of a set from $\mathscr{F}_n$, that is, the number

$$w(\mathscr{F}_n, c) = \max\left\{ \sum_{i \in X} c_i \colon X \in \mathscr{F}_n \right\}.$$

Later in this section we describe several important examples of families $\mathscr{F}_n$, where the optimization oracle can be efficiently constructed. Now we present our main algorithm. Our goal is to estimate the coefficient α in the expression $|\mathscr{F}_n| = e^{\alpha n}$.

1.2. The Algorithm

Input: A family $\mathscr{F}_n$ of subsets of $\{1, \ldots, n\}$, given by its optimization oracle.

Output: A number γ approximating $\alpha = (\ln |\mathscr{F}_n|)/n$.

The algorithm.

Step 1. Sample independently n random variables $c_1, \ldots, c_n$, where

$$c_i = \begin{cases} 1, & \text{with probability } 1/2, \\ -1, & \text{with probability } 1/2. \end{cases}$$

Step 2. Apply the optimization oracle with the input $c = (c_1, \ldots, c_n)$. Let $w(\mathscr{F}_n, c)$ be the output [see (1.1)]. Compute

$$\gamma(\mathscr{F}_n, c) = \frac{2}{n} w(\mathscr{F}_n, c) - \frac{1}{n} \sum_{i=1}^{n} c_i.$$

Step 3. Output $\gamma = \gamma(\mathscr{F}_n, c)$.

The following result provides some bounds for γ in terms of α.

Theorem 1.3. *With probability at least* 0.9 *the output* γ *of the algorithm satisfies the inequalities*

$$\frac{\alpha}{2 \ln(3/\alpha)} - \frac{7}{\sqrt{n}} \leq \gamma \leq 4\sqrt{\alpha} + \frac{7}{\sqrt{n}}$$

with $\alpha = (\ln |\mathscr{F}_n|)/n$.

In other words, some information about the cardinality of the family can be read off from the discrepancy of a random coloring (see, for example, Chapter 12 of [1] for a general discussion of the discrepancy). Suppose that $\{\mathcal{F}_n: n \geq 1\}$ is a sequence of families. Theorem 1.3 implies that the cardinality $|\mathcal{F}_n|$ grows exponentially in n (that is, $|\mathcal{F}_n| \geq e^{\alpha n}$ for some $\alpha > 0$ and all sufficiently large n) if and only if the discrepancy of a random coloring grows linearly in n.

One can obtain slightly better bounds. Two remarks are in order.

1.4. Remarks

1. Let us apply Algorithm 1.2 independently k times and let $\bar{\gamma} = (\gamma_1 + \cdots + \gamma_k)/k$ be the average of the outputs. Then with probability at least 0.9 we have

$$\frac{\alpha}{2\ln(3/\alpha)} - \frac{7}{\sqrt{kn}} \leq \bar{\gamma} \leq 4\sqrt{\alpha} + \frac{7}{\sqrt{kn}}.$$

Indeed, for $X \subset \{1, \dots, n\}$ and an integer m let $X + m = \{x + m: x \in X\}$ denote the shift of X. Now we apply Theorem 1.3 to the family $\mathcal{F}_n^k \subset \{1, \dots, kn\}$ defined as

$$\mathcal{F}_n^k = \left\{X_0 \cup \cdots \cup X_{k-1}: X_j - jn \in \mathcal{F}_n \text{ for } j = 0, \dots, k-1\right\}.$$

We observe that $|\mathcal{F}_n^k| = e^{\alpha k n}$ and that $\bar{\gamma}$ is the output of Algorithm 1.2 applied to $\mathcal{F}_n^k$.

2. If we want the output γ to satisfy the inequalities with a higher probability $1 - \epsilon$, we can run Algorithm 1.2 independently $O(\ln(\epsilon^{-1}))$ times and then take the median of the computed γs.

Let us call a γ that satisfies the inequalities of Theorem 1.3 *typical.*

Algorithm 1.2 allows us to test probabilistically whether the cardinality $|\mathcal{F}_n|$ is exponentially large in n in the following sense. Let us fix an $\alpha > 0$. Then there exists $0 < \beta \leq \alpha$ such that for all sufficiently large n, whenever $|\mathcal{F}_n| \geq e^{\alpha n}$, a typical output γ provides a certificate that $|\mathcal{F}_n| \geq e^{\beta n}$. Indeed, one can choose any

$$0 < \beta < \frac{\alpha^2}{64\ln^2(3/\alpha)},$$

since by Theorem 1.3 for a typical γ we have

$$\frac{1}{n}\ln|\mathcal{F}_n| \geq \left(\frac{\gamma}{4} - \frac{7}{4\sqrt{n}}\right)^2 \geq \left(\frac{\alpha}{8\ln(3/\alpha)} - \frac{7}{2\sqrt{n}}\right)^2 > \beta$$

for all sufficiently large n.

There are several important examples of families $\mathcal{F}_n$ for which the optimization oracle can be efficiently constructed, but the counting problem seems to be really difficult. Here are some of them.

1.5. Examples

1.5.1. Perfect Matchings in Graphs. Let G_n be a simple undirected graph with n edges, which we number $1,\ldots,n$. A collection $X \subset \{1,\ldots,n\}$ of edges is called a *perfect matching* iff every vertex from G_n is incident to precisely one edge from X. Let $\mathscr{F}(G_n)$ be the set of all perfect matchings in a given graph G_n. The problem of finding the maximum weight of a matching from $\mathscr{F}(G_n)$ admits an algorithm of $O(m^3)$ complexity, where m is the number of vertices of G_n (see, for example, Section 9.2 of [8]). So, it is easy to design the optimization oracle for this family. On the other hand, to compute the number $|\mathscr{F}(G_n)|$ of perfect matchings in a given graph is a #P-complete problem, and even to estimate this number seems to be really difficult. The Markov chain approach for estimating the number of perfect matchings in a given graph resulted in several remarkable successes: polynomial time approximation schemes were found for "dense graphs," "almost all graphs," and some important special graphs, like lattice graphs, etc. (see, for example, [6] for a survey). However, the author is unaware of a polynomial time algorithm which would give a nontrivial estimate of the number of perfect matchings for *any* given graph. (We note in passing that a recent randomized polynomial time algorithm of the author [3] estimates the permanent of any given $n \times n$ nonnegative matrix within a factor $2^{O(n)}$, thus providing an approximate algorithm for estimating the number of perfect matchings in a bipartite graph.) Our method provides a randomized polynomial time algorithm for checking whether the number of perfect matchings in a graph is exponentially large with respect to the number of edges in the graph.

1.5.2. Bases in Matroids. A *matroid* on a finite set G is a collection $\mathscr{F}_n$ of subsets of G, called *bases*, such that for any two bases $X, Y \in \mathscr{F}_n$ and for any $x \in X \setminus Y$ there exists a $y \in Y \setminus X$ such that $X \setminus \{x\} \cup \{y\}$ is a base (see [11]). A subset of a base is called an *independent set*. For example, the spanning trees of a connected graph constitute a matroid on the set of edges of the graph, called a *graphic matroid*. The subsets consisting of linearly independent vectors from a given finite set in a vector space are the independent sets of a matroid, called a *linear matroid*. The problem of finding the largest weight of a base or the largest weight of an independent set in a given matroid on the set with n elements can be solved by a greedy algorithm in $O(n)$ time, provided we are able to test independence (see, for example, [7]). Hence, it is easy to design the optimization oracles for the family of the bases of a matroid and for the family of independent sets of a matroid. However, to compute the number of bases or the number of independent sets in a given matroid is a #P-hard problem. Efficient counting algorithms are known for some particular cases: the number of spanning trees can be computed in polynomial time and the Markov chain approach works well in some cases (see [6]). However, the author is unaware of any polynomial time approximation algorithm for counting bases in general matroids (the problem is provably hard for deterministic algorithms; see [2]). Examples of particularly interesting and difficult problems include counting forests in a given graph and counting subsets of linearly independent vectors in a given set in a vector space over GF(2). We also note that there is a polynomial time algorithm for finding the largest weight of a common base of two given matroids on the same ground set (see [5]), so one can construct the optimization oracle for the intersection of two given matroids. For example, the computation of the permanent of a given 0–1 matrix reduces to counting bases in the intersection of two matroids. Hence our method provides a randomized polynomial time algorithm for checking whether the number of bases in the intersection of two matroids is exponentially large with respect to the size of their ground set.

Finally, we discuss the situation where the optimization oracle is not available and the family $\mathscr{F}_n$ is given by a much less powerful *membership oracle*.

1.6. Membership Oracle

We suppose that we have a black box which, for any given subset $X \subset \{1, \ldots, n\}$, outputs the answer "yes" if $X \in \mathscr{F}_n$ and "no" if $X \notin \mathscr{F}_n$.

Now our tools are severely limited, but still we can say something about the complexity status of the problem of estimating the cardinality of $\mathscr{F}_n$. Although we cannot find the maximum weight $w(\mathscr{F}_n, c)$ of a subset $X \in \mathscr{F}_n$ efficiently, we can provide a lower bound for $w(\mathscr{F}_n, c)$ by *guessing* an appropriate $X \in \mathscr{F}_n$ and computing its weight. This would give us a lower bound for $|\mathscr{F}_n|$. Suppose that $|\mathscr{F}_n| \geq e^{\alpha n}$. Then there exists a *polynomial size probabilistic certificate* that $|\mathscr{F}_n| \geq e^{\beta n}$ with

$$\beta = \left(\frac{\alpha}{8 \ln(3/\alpha)} - \frac{7}{2\sqrt{n}} \right)^2.$$

Such a certificate consists of choosing a random weighting $c_i \in \{-1, 1\}$, demonstrating a subset $X \subset \{1, \ldots, n\}$, such that

$$\frac{2}{n} \sum_{i \in X} c_i - \frac{1}{n} \sum_{i=1}^{n} c_i \geq 4\sqrt{\beta} + \frac{7}{\sqrt{n}} = \frac{\alpha}{2 \ln(3/\alpha)} - \frac{7}{\sqrt{n}}, \tag{1.7}$$

and the membership oracle verification that $X \in \mathscr{F}_n$. Indeed, Theorem 1.3 implies that the probability that either

(a) $|\mathscr{F}_n| \geq e^{\alpha n}$ but no set $X \in \mathscr{F}_n$ satisfying (1.7) exists;
or
(b) $|\mathscr{F}_n| < e^{\beta n}$ but a set $X \in \mathscr{F}_n$ satisfying (1.7) exists

does not exceed 0.1.

By choosing several random weighting c independently and checking whether we can demonstrate X in the majority of instances, we can make the probability of error arbitrarily small.

Of course, to find such a certificate (the set X) may be difficult, but its very existence seems to be of interest. In other words, the exponential size has a polynomial size probabilistic certificate. For any $\alpha > 0$ there exists a $\beta > 0$, such that for all sufficiently large n and for any family $\mathscr{F}_n$, given by its membership oracle and such that $|\mathscr{F}_n| \geq e^{\alpha n}$, the inequality $|\mathscr{F}_n| \geq e^{\beta n}$ has a probabilistic certificate whose size is polynomial in n.

Example 1.8. Let G_n be a simple undirected graph with n edges, which we number $\{1, \dots, n\}$. A collection $X \subset \{1, \dots, n\}$ of edges called a *Hamiltonian cycle* if every vertex of G_n is incident to precisely two edges from X and the edges from X constitute one cycle. Let $\mathscr{F}(G_n)$ be the set of all Hamiltonian cycles in G_n. It is easy to design the membership oracle for this family whose complexity is $O(n)$, whereas the corresponding optimization problem is NP-hard. Thus our construction implies that one can furnish a polynomial size probabilistic certificate that the number of Hamiltonian cycles in a graph is exponentially large with respect to the number of edges in the graph.

2. PROOFS

The idea of the proof of Theorem 1.3 is geometric. Let $I_n = \{-1, 1\}^n$ be the set of all n-vectors $x = (x_1, \dots, x_n)$ whose coordinates are -1 or 1. For $x, y \in I_n$ we let

$$\langle x, y \rangle = \sum_{i=1}^{n} x_i y_i.$$

A subset $X \subset \{1, \dots, n\}$ we represent by a point $x \in I_n$ such that

$$x_i = \begin{cases} 1, & \text{if } i \in X, \\ -1, & \text{if } i \notin X. \end{cases}$$

Hence we represent $\mathscr{F}_n$ by a subset $F_n \subset I_n$, $F_n = \{x \colon X \in \mathscr{F}_n\}$. Then, for any $c \in I_n$, the value $\gamma(\mathscr{F}_n, c)$, computed in Step 2 of Algorithm 1.2, can be written as

$$\gamma(\mathscr{F}_n, c) = \frac{1}{n} \max\{\langle c, x \rangle \colon x \in F_n\}. \tag{2.1}$$

So, we have to prove that if a set $F_n \subset I_n$ is "small," then for a "typical" c, all the values of $\langle c, x \rangle$ for $x \in F_n$ are small and, conversely, if F_n is "large," then there is an $x \in F_n$ with a large value of $\langle c, x \rangle$. The first assertion is intuitively clear, since if a set $F_n \subset \mathbb{R}^n$ contains few vectors, then a "random" vector $c \in \mathbb{R}^n$ will be nearly orthogonal to each $x \in F_n$. The second assertion is a bit more tricky, because no upper bound on the cardinality of a set $F_n \subset \mathbb{R}^n$ follows from the fact that the values $\langle c, x \rangle \colon x \in F_n$ are all small for a typical c (the points of F_n may cluster around some point in $\mathbb{R}^n$). However, since the vectors from F_n are vertices of the cube I_n, they do not have much room to cluster, so some kind of an upper bound on $|F_n|$ should follow. We also note that we could have sampled c from any sufficiently symmetric distribution in $\mathbb{R}^n$ instead of the uniform distribution on I_n.

Let us consider I_n as a probability space with the normalized counting measure μ and the l^1 metric:

$$d(x, y) = \sum_{i=1}^{n} |x_i - y_i|.$$

For a function $f \colon I_n \to \mathbb{R}$ we define its expectation $\mathbf{E}(f)$ by the formula

$$\mathbf{E}(f) = \int_{I_n} f \, d\mu = \frac{1}{2^n} \sum_{x \in I_n} f(x).$$

We need the following powerful result on the "measure concentration" for I_n.

Proposition 2.2. *Let $f\colon I_n \to \mathbb{R}$ be a function such that $|f(x)-f(y)| \le d(x,y)$ for any two $x, y \in I_n$. Then for any $\epsilon > 0$ one has*

$$\mu\{x \in I_n \colon f(x) - \mathbf{E}(f) \ge \epsilon\} \le \exp\left\{-\frac{\epsilon^2}{16n}\right\}$$

and

$$\mu\{x \in I_n \colon f(x) - \mathbf{E}(f) \le -\epsilon\} \le \exp\left\{-\frac{\epsilon^2}{16n}\right\}.$$

Proof. See Theorem 7.8 and Example 7.9 of [9]. ■

Definition 2.3. *For a subset $F_n \subset I_n$ and a $c \in I_n$ let*

$$\delta(F_n, c) = \max\{\langle c, x\rangle \colon x \in F_n\}$$

and

$$\Delta(F_n) = \mathbf{E}(\delta(F_n, \cdot)) = \frac{1}{2^n} \sum_{c \in I_n} \delta(F_n, c).$$

First, we prove an upper found for $\delta(F_n, c)$. The proof is based on the same idea as the proof of the general bound for the discrepancy of a set (see Theorem 1.1 in Chapter 2 of [1]).

Lemma 2.4. *Suppose that $|F_n| \le e^{\alpha n}$ for some $\alpha > 0$. Then*

$$\delta(F_n, c) \le 4\sqrt{\alpha}\, n + 7\sqrt{n}$$

with probability at least 0.95.

Proof. Let us choose a $w \in F_n \subset I_n$ and let $f_w(x) = \langle x, w\rangle$ for $x \in I_n$. Then $\mathbf{E}(f_w) = 0$ and $|f_w(x) - f_w(y)| \le d(x,y)$ for any two $x, y \in I_n$. Let

$$A_w = \left\{c \in I_n \colon f_w(c) \ge 4\sqrt{\alpha}\, n + 7\sqrt{n}\right\}.$$

Proposition 2.2 implies that $\mu(A_w) \le \exp\{-\alpha n - 3\}$. Therefore,

$$\mu\left(\bigcup_{w \in F_n} A_w\right) \le \sum_{w \in F_n} \mu(A_w) \le e^{-3} < 0.05.$$

Therefore, the probability that a randomly chosen $c \in I_n$ does not belong to the union $\bigcup_{w \in I_n} A_w$ is at least 0.95. Since for any $c \in I_n \setminus \bigcup_{w \in I_n} A_w$ one has $\delta(F_n, c) \le 4\sqrt{\alpha}\, n + 7\sqrt{n}$, the proof follows. ■

To prove the lower bound, we apply an inductive argument to $\Delta(F_n)$ and then use Proposition 2.2. For $x \in I_{n-1}$ let us define $x^+, x^- \in I_n$ by the formulas

$$x^+ = (x_1, \dots, x_{n-1}, 1) \quad \text{and} \quad x^- = (x_1, \dots, x_{n-1}, -1).$$

We also agree that $I_0 = \{0\}$ and that $0^+ = 1$ and $0^- = -1$. For a subset $F_n \subset I_n$ let us define $F^+_{n-1}, F^-_{n-1} \subset I_{n-1}$ as

$$F^+_{n-1} = \{x \in I_{n-1}\colon x^+ \in F_n\} \quad \text{and} \quad F^-_{n-1} = \{x \in I_{n-1}\colon x^- \in F_n\}.$$

We need the following technical lemma.

Lemma 2.5. *For $n \geq 1$ one has*

$$\Delta(F^+_{n-1}), \Delta(F^-_{n-1}) \leq \Delta(F_n) \tag{2.5.1}$$

and

$$\frac{\Delta(F^+_{n-1}) + \Delta(F^-_{n-1})}{2} \leq \Delta(F_n) - 1. \tag{2.5.2}$$

Proof. For any $x \in F^+_{n-1}$ we have $x^+ \in F_n$, and for any $c \in I_{n-1}$ we have

$$\langle c, x \rangle = \frac{\langle c^+, x^+ \rangle + \langle c^-, x^+ \rangle}{2} \leq \frac{\delta(F_n, c^+) + \delta(F_n, c^-)}{2}.$$

Therefore, by taking the maximum over $x \in F^+_{n-1}$,

$$\Delta(F^+_{n-1}, c) \leq \frac{\delta(F_n, c^+) + \delta(F_n, c^-)}{2}.$$

Averaging over $c \in I_{n-1}$, we get

$$\Delta(F^+_{n-1}) = \frac{1}{2^{n-1}} \sum_{c \in I_{n-1}} \delta(F^+_{n-1}, c) \leq \frac{1}{2^n} \sum_{c \in I_{n-1}} (\delta(F_n, c^+) + \delta(F_n, c^-)) = \Delta(F_n).$$

The inequality $\Delta(F^-_{n-1}) \leq \Delta(F_n)$ is proven similarly.

Let us choose a $c \in I_{n-1}$. Then for any $x \in F^+_{n-1}$ we have $x^+ \in F_n$ and

$$\langle c, x \rangle = \langle c^+, x^+ \rangle - 1 \leq \delta(F_n, c^+) - 1.$$

Maximizing over $x \in F^+_{n-1}$, we get

$$\delta(F^+_{n-1}, c) \leq \delta(F_n, c^+) - 1.$$

Similarly,

$$\delta(F^-_{n-1}, c) < \delta(F_n, c^-) - 1.$$

Therefore,

$$\frac{\delta(F^+_{n-1}, c) + \delta(F^-_{n-1}, c)}{2} \leq \frac{\delta(F_n, c^+) + \delta(F_n, c^-)}{2} - 1.$$

Averaging over $c \in I_{n-1}$, we get

$$\frac{\Delta(F_{n-1}^{+}) + \Delta(F_{n-1}^{-})}{2} \leq \frac{1}{2^n} \sum_{c \in I_{n-1}} \left(\delta(F_n, c^+) + \delta(F_n, c^-)\right) - 1 = \Delta(F_n) - 1,$$

so (2.5.2) is proven. ■

The following result is crucial for our considerations.

Lemma 2.6. *Suppose that* $|F_n| \geq e^{\alpha n}$ *for some* $\alpha > 0$. *Then*

$$\Delta(F_n) \geq \frac{\alpha n}{2 \ln(3/\alpha)}.$$

Proof. Let us construct a sequence $F_n, F_{n-1}, \dots, F_1, F_0$, where $F_k \subset I_k$. Each time we choose F_{k-1} to be either F_{k-1}^{+} or F_{k-1}^{-}, following the rules:

Rule 1. If $\min\{|F_{k-1}^{+}|, |F_{k-1}^{-}|\} < (\alpha/3)|F_k|$, we let F_{k-1} be the set of the larger cardinality (either of the two if $|F_{k-1}^{-}| = |F_{k-1}^{+}|$).

Rule 2. If $\min\{|F_{k-1}^{+}|, |F_{k-1}^{-}|\} \geq (\alpha/3)|F_k|$, we let F_{k-1} to be the set with the smaller value of $\Delta(\cdot)$ [either of the two if $\Delta(F_{k-1}^{+}) = \Delta(F_{k-1}^{-})$].

Rule 1 makes sure that $F_{n-1}, \dots, F_1, F_0$ are nonempty, so $\Delta(F_0) = 0$. By (2.5.1) it follows that $\Delta(F_k) \geq \Delta(F_{k-1})$ for $k = n, n-1, \dots, 1$ and by (2.5.2) it follows that $\Delta(F_k) \geq \Delta(F_{k-1}) + 1$ if we have applied Rule 2.

Let m be the number of times we used Rule 2. Then

$$\Delta(F_n) \geq m + \Delta(F_0) = m.$$

Our goal is to prove a lower bound for m. We observe that $|F_{k-1}^{+}| + |F_{k-1}^{-}| = |F_k|$. Therefore, if we have applied Rule 1 to produce F_{k-1} from F_k, we must have $|F_{k-1}| \geq (1 - \alpha/3)|F_k|$. On the other hand, if we have applied Rule 2 to produce F_{k-1} from F_k, we must have $|F_{k-1}| \geq (\alpha/3)|F_k|$. Therefore,

$$1 = |F_0| \geq \left(1 - \frac{\alpha}{3}\right)^{n-m} \left(\frac{\alpha}{3}\right)^m |F_n| \geq \left(1 - \frac{\alpha}{3}\right)^{n-m} \left(\frac{\alpha}{3}\right)^m e^{\alpha n}.$$

Since $0 \leq \alpha < 1$ we have that

$$\left(1 - \frac{\alpha}{3}\right)^{n-m} \geq \left(1 - \frac{\alpha}{3}\right)^n = \exp\left\{\ln\left(1 - \frac{\alpha}{3}\right) n\right\} \geq \exp\left\{-\frac{\alpha n}{2}\right\}.$$

Therefore, $(\alpha/3)^m \leq \exp\{-\alpha n/2\}$ and we have that

$$m \geq \frac{\alpha n}{2 \ln(3/\alpha)},$$

so the proof follows. ■

Corollary 2.7. *Suppose that* $|F_n| \geq e^{\alpha n}$ *for some* $\alpha > 0$. *Then*

$$\delta(F_n, c) \geq \frac{\alpha n}{2 \ln(3/\alpha)} - 7\sqrt{n}$$

with probability at least 0.95.

Proof. We apply Proposition 2.2 and Lemma 2.6. It is easy to see that $|\delta(F_n, x) - \delta(F_n, y)| \leq d(x, y)$ for any two x and y. Therefore, the probability that $\delta(F, c) \leq \Delta(F_n) - 7\sqrt{n}$ does not exceed $e^{-3} < 0.05$ and the proof follows. ■

Now, we are ready to prove our main result, Theorem 1.3.

Proof of Theorem 1.3. The output $\gamma(\mathscr{F}_n, c)$ of Algorithm 1.2 is represented as $\gamma(\mathscr{F}_n, c) = (1/n)\delta(F_n, c)$; cf. (2.1). Now we apply Lemma 2.4 and Corollary 2.7. ■

3. POSSIBLE RAMIFICATIONS

Different versions of the method of approximate counting via random optimization can be obtained by using different ways of random weighting. Indeed, a particular way of weighting may appear more powerful for a particular family $\mathscr{F}_n$ and a particular asymptotic question that we want to solve. Here is an example which concerns approximate counting of permutations.

Let S_n be the symmetric group of all permutations of the set $\{1, \dots, n\}$ and let $\mathscr{F}_n \subset S_n$. Let us choose an $n \times n$ matrix $c = (c_{ij})$ from the uniform distribution on the unit sphere

$$S^{n^2-1} = \left\{ (c_{ij}) \colon \sum_{i,j=1,\dots,n} c_{ij}^2 = 1 \right\}$$

in the Euclidean n^2-dimensional space of all $n \times n$ matrices.

Theorem 3.1. *For* $c \in S^{n^2-1}$ *and* $\mathscr{F}_n \subset S_n$ *let*

$$w(\mathscr{F}_n, c) = \max\left\{ \sum_{i=1}^{n} c_{i\sigma(i)} \colon \sigma \in \mathscr{F}_n \right\} \quad \textit{and} \quad \alpha(\mathscr{F}_n) = 1 - \frac{\ln|\mathscr{F}_n|}{n \ln n}.$$

Let us choose $c \in S^{n^2-1}$ *at random from the uniform distribution. Then the probability that*

$$1 - \sqrt{\alpha(\mathscr{F}_n)} + o(1) \leq \frac{w(\mathscr{F}_n, c)}{\sqrt{2 \ln n}} \leq \sqrt{1 - \alpha(\mathscr{F}_n)} + o(1)$$

tends to 1 *as n grows to infinity. Here* $o(1)$ *stands for a function depending on n alone which tends to* 0 *as n grows to infinity.*

Proof. See [4]. ■

So, Theorem 3.1 allows us to estimate the coefficient α in the expression $|\mathscr{F}_n| = \exp\{(1-\alpha)n \ln n\}$ from the maximum weight of a permutation from $\mathscr{F}_n$ in a random weighting.

3.2. Examples

3.2.1. Cycle Covers. Let G_n be a simple directed graph with n vertices, $1,\ldots,n$. Let $\mathscr{F}(G_n)$ be the set of the permutations $\sigma \in S_n$ such that $(i,\sigma(i))$ is an edge of G_n for $i=1,\ldots,n$. Hence the permutations $\sigma \in \mathscr{F}(G_n)$ correspond to *cycle covers* of G_n (cf. Example 1.5.1). Note, that if $A=(a_{ij})$ is a 0−1 matrix interpreted as the adjacency matrix of G_n [that is, $a_{ij}=1$ if and only if (i,j) is an edge of G_n], then $|\mathscr{F}(G_n)|$ is equal to the permanent of A. Suppose that we assign a weight c_{ij} to each edge (i,j) of G_n. The problem of computing $w(\mathscr{F}(G_n),c)$, that is the maximum weight of a cycle cover in a given graph, admits a polynomial time algorithm (see, for example, [8]). Hence Theorem 3.1 provides a randomized polynomial time algorithm for approximating the number $|\mathscr{F}(G_n)|$ of cycle covers in a given graph, or, equivalently, the permanent of a given 0−1 matrix. Roughly speaking, it allows us to test whether the permanent of an $n \times n$ matrix grows as fast as $(n!)^{\delta}$ for some $\delta > 0$. We note, however, that the algorithm from [3] gives us a better estimate.

3.2.2. Hamiltonian Circuits. Let G_n be a simple directed graph with n vertices, $1,\ldots,n$. Let $\mathscr{F}(G_n)$ be the set of the permutations $\sigma \in S_n$ which consist of one cycle $i_1 \to i_2 \to \cdots \to i_n \to i_1$ and such that (i_k, i_{k+1}) for $k=1,\ldots,n-1$ and (i_n,i_1) are edges of G_n. The permutation from $\mathscr{F}(G_n)$ correspond to Hamiltonian circuits in G_n (cf. Example 1.8). So $|\mathscr{F}(G_n)|$ is the number of Hamiltonian circuits in G_n. Furthermore, if we assign a weight c_{ij} to the edge (i,j), then $w(\mathscr{F}_n,c)$ is the largest weight of a Hamiltonian circuit in G_n with this weighting. To compute the largest weight of a Hamiltonian circuit in a given weighted graph is an NP-hard problem. Nevertheless, we can use Theorem 3.1 to *certify* a lower bound for the number of Hamiltonian circuits. Suppose that $|\mathscr{F}(G_n)| \geq \exp\{(1-\alpha)n \ln n\}$ for some $0 \leq \alpha < 1$. We claim that for all sufficiently large n there exists a polynomial size probabilistic certificate that $|\mathscr{F}(G_n)| \geq \exp\{\beta n \ln n\}$ with $\beta = (1-\sqrt{\alpha}+o(1))^2$. Such a certificate consists of choosing random weights $(c_{ij}) \in S^{n^2-1}$ and demonstrating a Hamiltonian circuit in G_n whose weight is at least $\sqrt{2 \ln n}\,(1-\sqrt{\alpha}+o(1))$. Indeed, the left hand side inequality of Theorem 3.1 implies such a circuit exists with probability which tends to 1 as n grows to infinity. Then the right hand side inequality of Theorem 3.1 implies that the desired lower bound for $|\mathscr{F}(G_n)|$ follows with probability which tends to 1 as n grows to infinity. In other words, there exists a polynomial size probabilistic certificate that the number of Hamiltonian circuits in a graph with n vertices grows at least as fast as $(n!)^{\delta}$ for some $\delta > 0$.

An interesting problem is to sharpen the inequalities in Theorems 1.3 and 3.1 when $\mathscr{F}_n$ has some special structure (for example, when $\mathscr{F}_n$ is the set of perfect matchings in a graph with n edges in Theorem 1.3 or when $\mathscr{F}_n$ is the set of Hamiltonian circuits in a graph with n vertices in Theorem 3.1).

REFERENCES

[1] N. Alon, J. H. Spencer, and P. Erdös, *The Probabilistic Method*, Wiley-Interscience, New York, 1992.

[2] Y. Azar, A. Z. Broder, and A. N. Frieze, On the problem of approximating the number of bases of a matroid, *Inform. Process. Lett.* **50**, 9–11 (1994).

[3] A. Barvinok, Computing mixed discriminants, mixed volumes and permanents, *Discrete Comput. Geom.*, to appear.

[4] A. Barvinok and R. Robb, On the mean radius of permutation polytopes, unpublished.

[5] M. Grötschel, L. Lovász, and A. Schrijver, *Geometric Algorithms and Combinatorial Optimization*, Springer-Verlag, Berlin, New York, 1993.

[6] M. Jerrum and A. Sinclair, The Markov chain Monte Carlo method: an approach to approximate counting and integration, in *Approximation Algorithms for NP-hard Problems*, D. S. Hochbaum, Ed., PWS Publishing, Boston, 1996, pp. 482–520.

[7] E. L. Lawler, *Combinatorial Optimization: Networks and Matroids*, Holt, Rinehart and Winston, New York, 1976.

[8] L. Lovász and M. D. Plummer, *Matching Theory*, Akademiai Kiado, Budapest, 1986.

[9] V. D. Milman and G. Schechtman, *Asymptotic Theory of Finite Dimensional Normed Spaces*, Lecture Notes in Mathematics, Vol. 1200, Springer-Verlag, Berlin, 1986.

[10] C. H. Papadimitriou, *Computational Complexity*, Addison-Wesley, Reading, MA, 1994.

[11] D. J. A. Welsh, *Matroid Theory*, Academic Press, London, 1976.