CS294 Partition Functions **Fall 2025**

Lecture Note 4

Instructor: Alistair Sinclair

4.1 Approximate Counting

We have seen that many natural counting problems, including their generalizations to computing partition functions, are #P-hard. Notwithstanding the two classical algorithms we saw in the last lecture, it turns out that almost all interesting problems of this type are hard, and this even applies to computing partition functions $Z(\lambda)$ at any *specific* value of λ (except for one or two trivial values). For the rest of the course we will assume that we have no hope of finding polynomial time algorithms for the vast majority of problems of this kind.

The next best thing we can hope for is a polynomial time *approximation algorithm*, formalized as follows:

Definition 4.1. *Let $f : \Sigma^* \to \mathbb{R}^+$ be a function. A* fully-polynomial randomized approximation scheme (fpras) *for f is an algorithm that, on input $(x, \varepsilon) \in \Sigma^* \times (0, 1]$, runs in time* $\mathrm{poly}(|x|, \varepsilon^{-1})$ *and outputs a value $A(x, \varepsilon)$ satisfying*

$$\Pr\big[(1+\varepsilon)^{-1} f(x) \le A(x,\varepsilon) \le (1+\varepsilon) f(x)\big] \ge \frac{3}{4}. \tag{4.1}$$

The variant of the above definition in which the algorithm is deterministic, so that the approximation in condition (4.1) is always satisfied, is known as a *fully-polynomial time approximation scheme (fptas)*. Note that an approximation scheme always determines whether or not $f(x) > 0$ (in the randomized case, it does this with high probability); thus if f is a counting problem in #P, the algorithm solves the associated decision problem.

Note: It is a matter of taste whether to replace $(1+\varepsilon)^{-1}$ by $(1-\varepsilon)$ in the lower bound in (4.1). Since $(1+2\varepsilon)^{-1} \le (1-\varepsilon) \le (1+\varepsilon)^{-1}$ for all $\varepsilon \in (0, 1/2]$, the two criteria are essentially equivalent, and we shall freely use whichever is more convenient in each of our applications.

The reason for the constant $\frac{3}{4}$ in (4.1) is the following.

Proposition 4.2. *The probability $\frac{3}{4}$ in* (4.1) *can be boosted to $1-\delta$ for any desired $\delta > 0$ using $O(\log \delta^{-1})$ repeated trials.*

Proof. Perform t independent trials of the fpras with the same inputs (x, ε), and take the median of the outputs. Call a number "good" if it falls in the range $[(1+\varepsilon)^{-1} f(x), (1+\varepsilon) f(x)]$. The median will be good unless at least half the trial outputs are bad. But the number of bad outputs is the number of heads in t tosses of a coin with heads probability at most $\frac{1}{4}$. By a standard Chernoff bound, the probability that at least half the tosses are heads is $\exp(-ct)$ for a constant c, so by taking $t = c^{-1} \ln \delta^{-1}$ this probability is at most δ. □

Exercise: Use your favorite form of the Chernoff bound to compute a value for c in the above proof. Also, check that the constant $\frac{3}{4}$ in (4.1) could be replaced by any constant strictly greater than $\frac{1}{2}$.

The notion of a $(1+\varepsilon)$ approximation scheme as in Definition 4.1 seems quite strong, but in fact it turns out that, for most natural problems, even a very weak approximation algorithm can be turned into an approximation scheme. We illustrate this with the example of counting q-colorings of a graph. Suppose we have an algorithm that approximates the number of q-colorings in any graph within a factor of c (where c is a large constant). For a graph G, let $f(G)$ be the number of colorings of G. Let G' be the graph consisting of m disjoint copies of G; clearly $f(G') = f(G)^m$. So if we use our algorithm to approximate $f(G')$ within a factor of c, and take the mth root of the result, we get an approximation of $f(G)$ within $c^{1/m}$. Choosing $m = \frac{\log c}{\log(1+\varepsilon)} = O(\varepsilon^{-1})$ yields an approximation within $1+\varepsilon$, as desired. This simple amplification trick can be applied to most problems of interest, implying that the notion of an fpras (or fptas) is *robust*. This should be contrasted with the situation for optimization problems, where the optimal approximation factor achievable in polynomial time is known to vary widely for different problems.

Exercise: Verify that the above trick also works when the initial approximation is within a factor $\mathrm{poly}(n)$, or even $\exp(O(n^{1-\delta}))$ for $\delta > 0$, where n is the size of the input graph. (Note that now the factor c in the above argument depends on the size of G'!) Why doesn't the trick work for an approximation factor of the form $\exp(\Theta(n))$ (and why would it be surprising if it did)?

4.2 Approximate Counting and Random Sampling

In this section we discuss the close connection between approximate counting and random sampling, as initially observed in [JVV86]. In a sense we will make precise, these two problems are polynomial time equivalent in most cases. To avoid excessive formality, our discussion will be based mainly on examples.

Consider a counting problem in #P. To each input $x \in \Sigma^*$, we can associate a set $\Omega = \Omega(x)$ of "solutions" that we want to count. (Generically these are the accepting computations of the nondeterministic Turing machine; in concrete examples, they are combinatorial structures such as matchings in a graph, satisfying assignments of a boolean formula, etc.) A *(uniform) random sampling algorithm* takes as input x and outputs a random element of $\Omega(x)$ from a uniform distribution. We often allow a small error in this distribution, measured in terms of the *(total) variation distance*:

Definition 4.3. *Let μ_1, μ_2 be two probability distributions on a common discrete probability space Ω. The* (total) variation distance *between μ_1 and μ_2 is defined by*

$$\|\mu_1 - \mu_2\|_{\mathrm{TV}} := \frac{1}{2}\sum_{z\in\Omega} |\mu_1(z) - \mu_2(z)| = \max_{S\subseteq\Omega} |\mu_1(S) - \mu_2(S)|.$$

Exercise: Check the equality in the above definition.

Thus we will speak of an *ε-approximate uniform random sampling algorithm* if the variation distance from uniform (specified as part of the input) is at most ε. We typically require the runtime of the sampling algorithm to depend polynomially on $\log \varepsilon^{-1}$.

More generally, suppose we are in the setting of partition functions as discussed in Lecture 1. Here elements σ of the associated set of combinatorial structures Ω are equipped with weights $\exp(-\beta H(\sigma))$. A *random sampling algorithm* in this context is required to output a random element of Ω according to these weights, which of course is just the Gibbs distribution. Again we allow a variation distance of ε.

4.2.1 From Sampling to Approximate Counting

We show how to reduce approximate counting to random sampling. Again, let's work with the concrete example of q-colorings of a graph G. Assume we have a black box that outputs independent random samples

from the uniform distribution over Ω, the set of q-colorings of G. Our goal is to estimate $|\Omega|$, which we can write as

$$|\Omega| = \frac{|\Omega_0|}{|\Omega_1|} \times \frac{|\Omega_1|}{|\Omega_2|} \times \cdots \times \frac{|\Omega_{n-1}|}{|\Omega_n|} \times |\Omega_n| =: \prod_{i=1}^{n} \frac{1}{Z_i}, \tag{4.2}$$

where $\Omega_0 = \Omega$, Ω_i denotes the set of colorings in which the colors of vertices $1, 2, \ldots, i$ have been "pinned" to legal values $c_1, c_2, \ldots, c_i$, and $Z_i := \frac{|\Omega_i|}{|\Omega_{i-1}|}$. Note that $|\Omega_n| = 1$. By "legal" here we mean that the partial coloring given by the pinnings can be extended to at least one proper coloring of the whole of G; equivalently, we require that each $|\Omega_i|$ be non-empty. We will show how to get a $(1+\varepsilon)$ approximation of $Z := \prod_i Z_i = \frac{1}{|\Omega|}$, which of course is equivalent to getting a $(1+\varepsilon)$ approximation of $|\Omega|$.

The key observation is that we can estimate each Z_i by random sampling from Ω_{i-1}: since $\Omega_i \subseteq \Omega_{i-1}$, Z_i is just the fraction of colorings in Ω_{i-1} that belong to Ω_i. To estimate this ratio, we can just take t independent samples from Ω_{i-1} and let $\hat{Z}_i$ be the proportion of the sample in which vertex i has color c_i. Clearly $\hat{Z}_i$ is an unbiased estimator of Z_i. To see how large the sample size t should be, we can appeal to a standard application of Chebyshev's inequality:

Proposition 4.4 (Unbiased Estimator Theorem)**.** *Let $X_1, \ldots, X_t$ be iid non-negative r.v.'s with common mean $\mu = \mathrm{E}(X_i)$ and variance $\sigma^2 = \mathrm{Var}(X_i)$, and let Y be the sample mean $\frac{1}{t}\sum_{i=1}^{t} X_i$. Then we have*

$$\Pr\big[|Y - \mu| \geq \varepsilon\mu\big] \leq \frac{1}{4}$$

provided $t \geq \frac{4}{\epsilon^2}\frac{\sigma^2}{\mu^2}$.

Proof. Note that $\mathrm{E}(Y) = \mu$ and $\mathrm{Var}(Y) = \frac{\sigma^2}{t}$. Chebyshev's inequality then gives

$$\Pr\big[|Y - \mu| \geq \varepsilon\mu\big] \leq \frac{\mathrm{Var}(Y)}{(\varepsilon\mu)^2} = \frac{\sigma^2/t}{\varepsilon^2\mu^2}. \tag{4.3}$$

□

Note that the key quantity in this analysis is the ratio $\gamma(Y) := \frac{\mathrm{Var}(Y)}{(\mathrm{E}(Y))^2}$, which we sometimes call the *critical ratio* of the estimator Y. The error estimate (4.3) requires that $\gamma(Y) \leq \frac{\varepsilon^2}{4}$. In the context of independent trials, where $Y = \frac{1}{t}\sum_{i=1}^{t} X_i$, we have $\gamma(Y) = \frac{1}{t}\gamma(X_i)$, leading to the bound $t \geq \frac{4}{\varepsilon^2}\gamma(X_i)$.

What is the critical ratio of our estimators $\hat{Z}_i$ in the colorings example above? Since $\hat{Z}_i$ is an average of t independent Bernoulli trials with mean Z_i, we have

$$\gamma(\hat{Z}_i) = \frac{\mathrm{Var}(\hat{Z}_i)}{\mathrm{E}(\hat{Z}_i)^2} \leq \frac{1}{tZ_i}. \tag{4.4}$$

Note that, if we choose c_i to be a most likely color at vertex i, then $Z_i \geq \frac{1}{q}$. Since we don't know how to find such a color, we instead choose c_i to be a color with the largest value of the *estimator* $\hat{Z}_i$, so that $\hat{Z}_i \geq \frac{1}{q}$.

For each color c, let Z_i^c denote the fraction of colorings in Ω_{i-1} in which vertex i has color c, and consider any color c for which $Z_i^c < \frac{1}{2q}$. For the associated estimator $\hat{Z}_i^c$, we have by Chebyshev

$$\Pr\big[\hat{Z}_i^c \geq \tfrac{1}{q}\big] \leq \Pr\big[\hat{Z}_i^c - \mathrm{E}(\hat{Z}_i^c) \geq \tfrac{1}{2q}\big] \leq \frac{\mathrm{Var}(\hat{Z}_i^c)}{1/4q^2} \leq \frac{Z_i^c/t}{1/4q^2} \leq \frac{2q}{t}.$$

Hence with our choice of c_i (taking a union bound over colors), with probability at least $1 - \frac{2q^2}{t}$, we have $Z_i \geq \frac{1}{2q}$ and hence by (4.4) $\gamma(\hat{Z}_i) \leq \frac{2q}{t}$. The probability that this condition is violated at any stage i is at most $\frac{2q^2 n}{t} \leq \frac{1}{8}$ if we take $t \geq 16q^2 n$.

At this point we could use Proposition 4.4 to analyze each estimator $\hat{Z}_i$ separately, taking care to replace the approximation ratio $1+\varepsilon$ by $1+\varepsilon/n$ and the error probability $\frac{1}{4}$ by $\frac{1}{4n}$ in each $\hat{Z}_i$, since we are taking the product over n such estimators. However, we can do better by analyzing the product estimator $\hat{Z} := \prod_{i=1}^n \hat{Z}_i$ directly. Note that, since the $\hat{Z}_i$ are independent, $\mathrm{E}(\hat{Z}) = \prod_{i=1}^n \mathrm{E}(\hat{Z}_i) = \frac{1}{|\Omega|}$. For the critical ratio, we have

$$\gamma(\hat{Z}) = \frac{\mathrm{Var}(\hat{Z})}{\mathrm{E}(\hat{Z})^2} = \frac{\mathrm{E}(\hat{Z}^2)}{\mathrm{E}(\hat{Z})^2} - 1 = \prod_i \frac{\mathrm{E}(\hat{Z}_i^2)}{\mathrm{E}(\hat{Z}_i)^2} - 1 = \prod_i \big(\gamma(\hat{Z}_i) + 1\big) - 1 \leq \left(1 + \frac{2q}{t}\right)^n - 1 \leq \frac{4qn}{t},$$

where we have used our assumptions that $\gamma(\hat{Z}_i) \leq \frac{2q}{t}$ for all i and that $\frac{2qn}{t}$ is small.

Finally, choosing $t \geq \frac{32qn}{\varepsilon^2}$ and applying Chebyshev again ensures that

$$\Pr\big[|\hat{Z} - Z| \geq \varepsilon Z\big] \leq \frac{1}{8}. \tag{4.5}$$

Recall that this analysis assumed that $\gamma(\hat{Z}_i) \geq \frac{1}{2q}$ for all i, which happens with probability at least $\frac{7}{8}$ if we also ensure that $t \geq 16q^2 n$. Hence the probability bound in (4.5) is at most $\frac{1}{4}$ with no assumptions, so $\hat{Z}$ satisfies the accuracy requirement of an fpras for counting q-colorings. The runtime of this estimator is dominated by the number of samples (calls to the black box), which is $nt = O(n^2 \varepsilon^{-2})$, as required. ■

It should be clear that the same analysis would still work, with minor modifications, if the sampling distribution for colorings were only ε'-uniform, for suitably small ε'. This introduces a small additional error everywhere, which can be absorbed into the $1+\varepsilon$ approximation ratio.

The above analysis shows that, if we can sample uniformly in polynomial time from a slightly generalized version of q-colorings, then we get an fpras for counting them. The generalization we need allows the colors at certain vertices to be pinned (fixed)—equivalently, we can remove these vertices from the graph and instead specify a *list* of allowable colors at each vertex; the resulting problem is known as "list-coloring."

The above reduction is quite generic and applies to the partition function of essentially any spin system. In many cases it is possible to dispense with the generalization to pinned instances: this applies when the pinning of a variable results in another instance of the same ("unpinned") problem. Examples include satisfiability (pinning a variable to T/F result in a simplified formula), independent sets (pinning a vertex v to be occupied/unoccupied leads to the same problem in graphs $G \setminus N(v)$, $G \setminus \{v\}$ respectively, where $N(v)$ denotes the neighborhood of v including v itself), etc. Such problems are called "self-reducible."

4.2.2 Other approaches

We briefly sketch two alternative techniques for reducing approximate counting to random sampling. The first approach is suited to problems with hard constraints, and involves introducing the constraints one by one. We illustrate it again with the example of q-colorings. For this analysis, we will assume that $q \geq \Delta + 1$, where Δ is the maximum degree of G; this is essentially w.l.o.g. because for smaller values of q it is non-trivial to even decide whether a q-coloring of G exists [Bro41], and becomes NP-complete at least for $q \leq \frac{\Delta}{2} + 1$, and approximate counting has been shown to be NP-hard already for $q < \Delta$ [GŠV15]. For $q \geq \Delta + 1$ a q-coloring always exists [**Exercise:** Why?], but counting them is a non-trivial task.

The idea is again to express $|\Omega|$ as a cascading product of ratios, as in (4.2):

$$|\Omega| = \frac{|\Omega_m|}{|\Omega_{m-1}|} \times \frac{|\Omega_{m-1}|}{|\Omega_{m-2}|} \times \cdots \times \frac{|\Omega_1|}{|\Omega_0|} \times |\Omega_0| =: \prod_{i=1}^{m} Y_i \times q^n,$$

where m is the number of edges in the graph G and Ω_i is the set of q-colorings of the graph with the same vertex set as G but only the first i edges of G. Thus $\Omega_m = \Omega$, and $|\Omega_0| = q^n$ (the number of colorings of the empty graph).

Again each ratio $Y_i = \frac{|\Omega_i|}{|\Omega_{i-1}|}$ can be estimated by random sampling from Ω_{i-1}: just count the proportion of colorings in Ω_{i-1} that assign different colors to the two endpoints of edge $i = \{u, v\}$. The key observation for the analysis is that the ratio $Y_i = \frac{|\Omega_i|}{|\Omega_{i-1}|} \geq \frac{1}{2}$. To see this, note that any coloring $\sigma \in \Omega_{i-1} \setminus \Omega_i$ can be mapped to a coloring in Ω_i simply by changing the color of u to some color different from that of v and any of u's other neighbors—this is always possible assuming that $q \geq \Delta + 1$. Moreover, this mapping is injective since the original coloring can always be recovered by simply recoloring u to the same color as v. The fact that all the ratios Y_i are bounded below by a constant means that essentially the same analysis as before goes through (in that case, the crucial ingredient was the lower bound $Z_i \geq \frac{1}{2q}$), and indeed is a little simpler as we don't need to worry about how to choose the color to branch on at each step.

Exercise: Fill in the details of the above analysis, by pattern-matching with the analysis from the previous section.

We mention one more approach to approximate counting via random sampling that applies specifically to Gibbs distributions with soft constraints, where there are $|\Omega| = q^n$ possible configurations σ (corresponding to all possible spin assignments) and the partition function is of the form $Z(\beta) = \sum_\sigma \exp(-\beta H(\sigma))$. We assume that $H(\sigma)$ is integer-valued and $|H(\sigma)| \leq n^c$ for some constant c; note that this is always the case in any spin system, since $H(\sigma)$ is a sum over fixed vertex and edge (or possibly larger clique) potentials. The idea is to bootstrap the computation of $Z(0)$, which is trivial (being equal to $|\Omega| = q^n$), to $Z(\beta)$ for any desired $\beta > 0$. To do this, we define a sequence

$$0 = \beta_0 < \beta_1 < \ldots < \beta_t = \beta,$$

where $\beta_i = \beta_{i-1} + \frac{1}{n^c}$, and write

$$Z(\beta) = \prod_{i=1}^{t} \frac{Z(\beta_i)}{Z(\beta_{i-1})} \times Z(\beta_0). \tag{4.6}$$

Now we may assume w.l.o.g. that $\beta \leq n^2$, since approximating $Z(\beta)$ for larger values of β reduces to this case. To see this, note that $Z(\beta) = k \exp(\beta M) + Y$, where $-M = \min_\sigma H(\sigma)$ and $k \geq 1$ is the number of configurations achieving this minimum value (the so-called "ground states"). The residual term Y is the total weight of all remaining configurations, which is bounded above by $q^n \exp(\beta(M-1)) = q^n \exp(-\beta)\exp(\beta M)$. Hence for any $\beta \geq n^2$ we have $Z(\beta) = k \exp(\beta M)(1 + \exp(-\Theta(n^2))$. Thus from a $(1+\varepsilon)$ estimate for $Z(n^2)$ we can easily compute a $(1+\varepsilon)$ estimate for $Z(\beta)$ for any larger value of β [**Exercise**: How?]. This ensures that the number of values β_i in our sequence is at most n^{2+c}, which is polynomial in n.

It remains to discuss how to compute the successive ratios in (4.6). The first key point is that

$$\begin{aligned}
\frac{Z(\beta_i)}{Z(\beta_{i-1})} &= \frac{1}{Z(\beta_{i-1})} \sum_\sigma \exp\big(-\beta_i H(\sigma)\big) \\
&= \frac{1}{Z(\beta_{i-1})} \sum_\sigma \exp\big(-\beta_{i-1} H(\sigma)\big) \exp\big((\beta_{i-1} - \beta_i) H(\sigma)\big) \\
&= \mathrm{E}_{\beta_{i-1}}\Big(\exp\big((\beta_{i-1} - \beta_i) H(\sigma)\big)\Big),
\end{aligned}$$

where $\mathrm{E}_{\beta_{i-1}}$ denotes expectation w.r.t. the Gibbs distribution with parameter β_{i-1}. Thus the ratio can be estimated by computing the sample mean of the r.v. $X_i := \exp\big((\beta_{i-1} - \beta_i)H(\sigma)\big)$ under this distribution. As usual, the sample size required for a good estimate is determined by the critical ratio of the r.v. But by our choice $\beta_i - \beta_{i-1} = \frac{1}{n^c}$, and the assumption $|H(\sigma)| \le n^c$, the r.v. X_i takes values in the bounded range $[e^{-1}, e]$, so its critical ratio is bounded by a constant. Hence, following the same argument as in our earlier analysis, a sample size of $O(n^{2+c}\varepsilon^{-2})$ suffices for each ratio in order to ensure a final estimate within $1+\varepsilon$.

A caveat is in order for this last scheme: note that it assumes the ability to sample at $\beta \approx n^2$, which corresponds to very low temperature. In particular, this temperature is low enough that the Gibbs weight of any ground state dominates that of all non-ground states combined. This is sometimes possible, but often not. When not, the same scheme will allow us to approximate $Z(\beta)$ up to whatever value of β we can efficiently perform random sampling.

For an optimized version of this procedure, which uses a carefully chosen set of intermediate values β_i, see [ŠVV09].

4.2.3 From Approximate Counting to Random Sampling

The reverse reduction, from sampling to counting, is less relevant for us, but we briefly sketch it for completeness. Consider again the recursion tree implicit in the first reduction from counting to sampling in Section 4.2.1, where we successively pin the colors assigned to vertices of the graph. Now suppose that, instead of a random sampler, we have a black box that returns an approximate count of the number of configurations with each of the q possible colors at the next vertex. We can use these counts to pick a color for that vertex with the right marginal probabilities, then pin that color and recurse.

If our counts are exact, we will eventually output each complete coloring with exactly the same probability. If our counts are only approximate, within ratio $(1+\frac{1}{n})$ say, then these probabilities could be biased by at most a factor $(1+\frac{1}{n})^n \le e$ (a constant). Moreover, before outputting any coloring σ, we know *a posteriori* the probability p_σ that we arrived at it (by keeping track of the branching probabilities at each step of the recursion). We then output σ with probability $(2e\hat{N}p_\sigma)^{-1}$, where $\hat{N}$ is an estimate of N, the total number of colorings (also obtained from our approximate counter); otherwise we reject σ and restart. This final step, known as "rejection sampling", ensures that all colorings are output with the same probability [**Exercise:** Why?].

The crucial point here is that the output probability $(2e\hat{N}p_\sigma)^{-1}$ is less than 1, but not too small (which would result in many restarts). This follows from the facts that $p_\sigma \in [\frac{1}{eN}, \frac{e}{N}]$ (by our observation in the previous paragraph) and that (more crudely) $\hat{N} \in [\frac{N}{2}, 2N]$. Together these imply

$$2e\hat{N}p_\sigma \;\ge\; 2e \times \tfrac{N}{2} \times \tfrac{1}{eN} = 1$$

and

$$2e\hat{N}p_\sigma \;\le\; 2e \times 2N \times \tfrac{e}{N} = 4e^2;$$

the second inequality means that we get an actual output with at least constant probability, so we expect to perform only a constant number of trials before we get a sample. Clearly the running time per trial is polynomial in n.

Note that the above analysis requires accuracy $1 + O(\frac{1}{n})$ from the counter [**Exercise:** Why?]. Also, it assumes that the approximate counting estimates are always within the desired accuracy; if they are good only with high probability, we can use Proposition 4.2 to boost this probability so that it is very close to 1, and can thus be absorbed into a small variation distance in the final output distribution. (By contrast, the above scheme is perfectly uniform.) For a more sophisticated algorithm, based on a suitable weighted random walk on the tree, that can handle much larger (polynomial) errors in the approximate counter, see [SJ89].

4.3 Worst Case Complexity of Approximate Counting

We conclude this topic by locating approximate counting for arbitrary #P problems within the polynomial hierarchy. Recall that #P itself (i.e., exact counting) is powerful enough to contain the entire hierarchy, so it is perhaps surprising to learn that approximate counting lives in only the second level of the hierarchy, namely $\mathrm{RP}^{\mathrm{NP}}$. This result is implicit in papers of Sipser [Sip83] and Stockmeyer [Sto83]; we give a proof based on a random bisection technique of Valiant and Vazirani [VV86].

Theorem 4.5. *For any function $f \in \#\mathrm{P}$, there is a probabilistic algorithm with an NP oracle that, on input (x, ε), runs in time polynomial in $|x|$ and ε^{-1} and outputs a number $A(x, \varepsilon)$ such that $\Pr[(1+\varepsilon)^{-1} f(x) \leq A(x, \varepsilon) \leq (1+\varepsilon) f(x)] \geq \frac{3}{4}$.*

Proof. For any input $x \in \Sigma^*$, let $\Omega = \Omega(x)$ denote the set of accepting computations of the polynomial time nondeterministic TM associated with f. Our task is to estimate $|\Omega| = f(x)$. To do this we use repeated random bisections: we repeatedly intersect Ω with a random hyperplane that, on average, cuts it in half, until after some number t of cuts the resulting set is empty; then 2^t is a good estimate of $|\Omega|$. The emptiness test can be handled by the NP oracle.

We may assume that $\Omega \subseteq \{0,1\}^m$, where each string in Ω encodes an accepting computation of the TM, and $m = \mathrm{poly}(|x|)$. Let $N = |\Omega|$, and suppose that $2^{k-1} < N \leq 2^k$. Pick independent random vectors $y_1, y_2, \ldots$, where each $y_i \in \{0,1\}^m$. Define

$$S_t = \{z \in \Omega : z \cdot y_i = 0 \pmod 2 \text{ for } 1 \leq i \leq t\}.$$

Let $t^* = \min\{t : |S_t| = 0\}$. Thus t^* is the number of random bisections needed to reduce the size of the remaining set to 0. Then we take $\hat{N} = 2^{t^*}$ as our estimator of N. Note that S_t belongs to NP [**Exercise:** Why?], so we can effectively compute the value of t^* using the NP oracle, as claimed above.

Note that $|S_t|$ is the sum of N indicator r.v.'s with expectation $p_t := 2^{-t}$, and that these r.v.'s are *pairwise* independent (but not mutually independent) [**Exercise:** Why?]. Thus $\mathrm{E}(|S_t|) = Np_t$, and $\mathrm{Var}(|S_t|) = Np_t(1-p_t) \leq Np_t = \mathrm{E}(|S_t|)$. We may now claim:

- If $t \geq k+3$ then $\Pr[|S_t| > 0] \leq \frac{1}{8}$. To see this, note that $\mathrm{E}(|S_t|) = Np_t \leq 2^k 2^{-(k+3)} = \frac{1}{8}$ and apply Markov's inequality.
- If $t \leq k-4$ then $\Pr[|S_t| = 0] \leq \frac{1}{8}$. To see this, note that $\mathrm{E}(|S_t|) = Np_t \geq 2^{k-1}2^{-(k-4)} = 8$. Then by Chebyshev: $\Pr\big[|S_t| = 0\big] \leq \Pr\big[\big||S_t| - \mathrm{E}(|S_t|)\big| \geq \mathrm{E}(|S_t|)\big] \leq \frac{\mathrm{Var}(|S_t|)}{\mathrm{E}(|S_t|)^2} \leq \frac{1}{\mathrm{E}(|S_t|)} \leq \frac{1}{8}$.

Putting the above two points together, we see that with probability at least $\frac{3}{4}$ the estimator $\hat{N} = 2^{t^*}$ falls in the range $[2^{k-4}, 2^{k+3}]$, which is within a constant factor 16 of the true value N. Finally we can use the amplification technique of Section 4.1 to improve the constant factor approximation to $1+\varepsilon$ for any desired ε, since the generic problem of counting accepting computations of a TM can be amplified in that way [**Exercise:** Verify this!] □

References

[Bro41] R.L. Brooks. On colouring the nodes of a network. *Mathematical Proceedings of the Cambridge Philosophical Society*, 37:194–197, 1941.

[GŠV15] A. Galanis, D. Štefankovič, and E. Vigoda. Inapproximability for antiferromagnetic spin systems in the tree nonuniqueness region. *Journal of the ACM*, 62:1–60, 2015.

[JVV86] M.R. Jerrum, L.G. Valiant, and V.V. Vazirani. Random generation of combinatorial structures from a uniform distribution. *Theoretical Computer Science*, 43:169–188, 1986.

[Sip83] M. Sipser. A complexity-theoretic approach to randomness. *Proceedings of the 15th ACM Symposium on Theory of Computing (STOC)*, pages 330–335, 1983.

[SJ89] A. Sinclair and M. Jerrum. Approximate counting, uniform generation and rapidly mixing Markov chains. *Information & Computation*, 82:93–133, 1989.

[Sto83] L. Stockmeyer. The complexity of approximate counting. *Proceedings of the 15th ACM Symposium on Theory of Computing (STOC)*, pages 118–126, 1983.

[ŠVV09] D. Štefankovič, S. Vempala, and E. Vigoda. Adaptive simulated annealing: A near-optimal connection between sampling and counting. *Journal of the ACM*, 56:1–36, 2009.

[VV86] L.G. Valiant and V.V. Vazirani. NP is as easy as detecting unique solutions. *Theoretical Computer Science*, 47:85–93, 1986.