2018 IEEE 59th Annual Symposium on Foundations of Computer Science

Approximating the Permanent of a Random Matrix with Vanishing Mean

Lior Eldar
Email: eldar.lior@gmail.com

Saeed Mehraban
EECS MIT
Cambridge, MA, USA
Email: mehraban@mit.edu

***Abstract*—The permanent is $\#P$-hard to compute exactly on average for natural random matrices including matrices over finite fields or Gaussian ensembles. Should we expect that it remains $\#P$-hard to compute on average if we only care about approximation instead of exact computation?**

In this work we take a first step towards resolving this question: We present a quasi-polynomial time deterministic algorithm for approximating the permanent of a typical $n\times n$ random matrix with unit variance and vanishing mean $\mu = O(\ln\ln n)^{-1/8}$ to within inverse polynomial multiplicative error. (alternatively, one can achieve permanent approximation for matrices with mean $\mu = 1/\mathrm{polylog}(n)$ in time $2^{n^{\varepsilon}}$, for arbitrarily small $\varepsilon > 0$).

The proposed algorithm significantly extends the regime of matrices for which efficient approximation of the permanent is known. This is because unlike previous algorithms which require a stringent correlation between the signs of the entries of the matrix [1], [2] it can tolerate random ensembles in which this correlation is negligible (albeit non-zero). Among important special cases we note:

1) **Biased Gaussian: each entry is a complex Gaussian with unit variance 1 and mean μ.**
2) **Biased Bernoulli: each entry is $-1+\mu$ with probability $1/2$, and 1 with probability $1/2$.**

These results counter the common intuition that the difficulty of computing the permanent, even approximately, stems merely from our inability to treat matrices with many opposing signs. The Gaussian ensemble approaches the threshold of a conjectured hardness [3] of computing the permanent of a zero mean Gaussian matrix. This conjecture is one of the baseline assumptions of the BosonSampling paradigm that has received vast attention in recent years in the context of quantum supremacy experiments.

We furthermore show that the permanent of the biased Gaussian ensemble is $\#P$-hard to compute exactly on average. To our knowledge, this is the first natural example of a counting problem that becomes easy only when average case analysis and approximation are combined.

On a technical level, our approach stems from a recent approach taken by Barvinok [1], [4], [5], [6] who used Taylor series approximation of the logarithm of a certain univariate polynomial related to the permanent. Our main contribution is to introduce an average-case analysis of such related polynomials. We complement our approach with a new technique for iteratively computing a Taylor series approximation of a function that is analytical in the vicinity of a curve in the complex plane. This method can be viewed as a computational version of analytic continuation in complex analysis.

***Keywords*-permanent; approximation algorithm; average-case complexity; Boson Sampling; anti-concentration; complex analysis;**

I. Introduction

A. Complexity of computing the permanent

The permanent of an $n \times n$ matrix A is the following degree n polynomial in the entries of A:

$$\mathrm{Per(A)} := \sum_{\sigma\in \mathrm{S_n}} \prod_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{A_{i,\sigma(i)}}, \tag{1}$$

where S_n is the symmetric group over n elements. Its computation has been the subject of intense research [2], [3], [5], [7], [8], [9] and has been connected to subjects ranging from random sampling of bi-partite matchings [2] to establishing a so-called "quantum supremacy" using linear optical experiments [3].

Much of this interest is centered around the computational complexity of computing the permanent. The permanent is known to be $\#P$-hard to compute exactly [8], [10] and we only know exponential time algorithms for computing the permanent of a general matrix A, the fastest of which is the Ryser formula.

Because of this hardness in the worst case, research has also focused on computing an approximation for the value of permanent: For multiplicative approximation, approximation schemes are known for several special cases of matrices.

2575-8454/18/$31.00 ©2018 IEEE
DOI 10.1109/FOCS.2018.00012

Perhaps most prominently is the work of Jerrum, Sinclair and Vigoda [2] who showed a randomized polynomial time algorithm to compute a $1 + 1/\mathrm{poly}(n)$ multiplicative approximation of the permanent for matrices with non-negative entries. More recently [11] have shown how to approximate the permanent of a PSD matrix to simply exponential factor in polynomial time.

Still, if one allows the matrix to have an arbitrary number of negative values it is even $\#P$-hard to compute the sign of the permanent [10], which rules out a multiplicative approximation. This suggests that part of the computational hardness of computing the permanent comes from alternating signs in a matrix. Hence, for general matrices, it seems that efficient approximation of the permanent remains well out of reach and progress is limited by a an "interference barrier" by which the positive and negative entries of a matrix may generate an intricate interference pattern that is hard to approximate.

Given the apparent difficulty in computing the permanent exactly and approximately one may ask a different question: Is the computation of permanent still hard on a faction of inputs? It turns out that it is still difficult to compute the permanent even on a small fraction of inputs. For example, it has been shown [12] that the permanent of a matrix over a finite field is $\#P$-hard to compute exactly even on a $1/\mathrm{poly}(n)$ fraction of such matrices. Specifically for the case of complex Gaussian matrices a somewhat weaker statement is known: It is $\#P$-hard to compute $\mathrm{Per(A)}$ exactly for a Gaussian A with probability greater than $3/4 + 1/\mathrm{poly}(n)$ [3].

Faced with the apparent robustness of the complexity of computing the permanent against both error on a fraction of inputs and against approximation, Aaronson and Arkhipov [3] designed a beautiful paradigm called $\mathrm{BosonSampling}$ for demonstrating a so-called quantum supremacy over classical computers. It hinges upon permanent computation remaining $\#P$-hard even when we simultaneously allow error on some inputs (see table I), and allow an approximate value for the inputs for which we do handle. The intuitive difficulty of approximating the permanent on such a distribution stems from the same "interference barrier" described above. Namely, that quantum computers, using complex amplitudes encoded in quantum states can "bypass" this barrier naturally, whereas classical computers fall short.

	worst case	average case
exact	$\#P$-hard	$\#P$-hard
approximate	$\#P$-hard	?

Table I: The computational complexity of computing the permanent of a zero mean complex Gaussian matrix. We know that: Permanent is $\#P$-hard to compute exactly in the worst case or the average case. We also know that permanent is $\#P$-hard to compute exactly on average. However, nothing is known about the approximation of permanent in the average case.

The range of parameters in our result approaches the threshold of this conjectured hardness. Hence, it raises the intriguing question of whether there exists a yet undiscovered phenomenon, that occurs only for zero mean, which prohibits efficient approximation, or whether that conjecture is false. We discuss this further in Section I-F1.

B. A complex function perspective

Recent works of Barvinok [1], [4], [5], [6], [13] have outlined a new approach to computing the permanent, and in fact a large class of high-degree polynomials in matrices such as the partition function [14], [15]. His approach, stated here for the permanent of an $n \times n$ matrix A, is quite intuitive: instead of trying to compute $\mathrm{Per(A)}$ directly, one computes an additive approximation of $\ln(\mathrm{Per(A)})$ and then exponentiates the result. Let J be the all ones $n \times n$ matrix. The additive approximation of $\ln(\mathrm{Per(A)})$ is then computed as a Taylor series approximation of the complex log of the univariate polynomial $g_A(z) = \mathrm{Per}(\mathrm{J} \cdot (1 - \mathrm{z}) + \mathrm{z} \cdot \mathrm{A})$ around the point $z = 0$, and evaluated at point $z = 1$. The crucial point is that one can compute the lowest m derivatives of f at point $z = 0$ relatively efficiently, i.e., in time $n^{O(m)}$ since they correspond essentially to a linear combination of the permanents of submatrices of A of size at most $m \times m$. The additive approximation error of the Taylor series expansion decays exponentially fast in m. So choosing $m = O(\ln(n))$ implies an algorithm for a multiplicative error of $1 + 1/\mathrm{poly}(n)$ that runs in time at most $\binom{n}{m} = 2^{O(\ln^2(n))}$.

In order to apply this Taylor series technique there is a significant limitation that must be carefully addressed: The Taylor approximation of $\ln(g(z))$ about a point $z = x_0$ is valid only in a disk around x_0 that contains no poles of $f(z)$ (the roots of $g(z)$). Therefore, this approach inherently requires knowledge about the location of the roots

of the univariate polynomial used for interpolating the value of the permanent from an easy-to-compute matrix to the target matrix A.

Using combinatorial arguments Barvinok characterized the location of the roots of $g_A(z)$ for certain complex matrices: For example those that that satisfy $\max_{i,j}|A_{i,j}-1| \leq 0.5$ [1] and diagonally dominant matrices [13]. This then implied quasi-polynomial time algorithms for these classes of matrices.

Hence, his approach is the first to break the "sign barrier" for approximating the permanent - i.e. the ability to approximate the permanent for matrices that have many entries with opposing signs, thus extending our ability to compute the permanent beyond matrices with non-negative entries for which the algorithm by Jerrum, Sinclair and Vigoda [2] is famously known for. Still, this divergence from non-negative entry matrices was only quite mild: The entries of the matrix types that the algorithm handles are still highly correlated in a specific direction and so have a very negligible interference pattern.

That said, this approach opens up a wide range of possibilities for computing these otherwise intractable polynomials: Instead of thinking about them as combinatorial objects, one can ask a completely different set of questions: What can be said about the location of the roots of $g_A(z)$? Can one detour around these roots in order to reach "interesting" points z where the value of $g_A(z)$ is non-trivial? Yet another set of questions can then be phrased for a "random ensemble" of matrices A: What is the behavior of the roots of $g_A(z)$ for typical A and can they be detoured "on average", in an efficient way? Answering such questions analytically, and subsequently efficiently as an algorithm is the focus of our work.

C. Main results

Consider a random matrix where each entry is sampled independently from a distribution of complex valued random variables with mean 0 and variance 1. We refer to such matrix a random matrix with mean 0 and variance 1. An example of such a matrix is a Bernoulli or Gaussian matrix.

Theorem (Informal statement of Theorem 17)**.** *Let n be some sufficiently large integer and $\mu = 1/\mathrm{polyloglog}n$. Let A be an $n \times n$ random matrix with mean μ and variance 1. There is a deterministic quasi-polynomial time algorithm that for $1-o(1)$ fraction of random matrices A outputs a number that is within inverse polynomial relative error of the permanent* $\mathrm{Per(A)}$.

	worst case	average case
exact	$\#P$-hard	$\#P$-hard
approximate	$\#P$-hard	Efficient

Table II: The computational complexity of computing the permanent of a complex Gaussian matrix with nonzero but vanishing mean.

We note that one can also achieve a mean value parameter of $\mu = 1/\mathrm{polylog}n$ with a run time of $2^{n^{\varepsilon}}$ for some small $\varepsilon > 0$. The details of the proof can be found in [16].

One can ask whether perhaps allowing a non-vanishing mean devoids the permanent of its average-case hardness. Thus we show a complementary claim whose proof appears in [16]

Theorem (Average-case hardness for the permanent of a nonzero mean Gaussian)**.** *Let $\mu > 0$ and let $\mathcal{O}$ be the oracle that for any $\mu' \geq \mu$ computes the permanent of $7/8+1/\mathrm{poly}(n)$ fraction of matrices from the ensemble $\mathcal{N}^{n\times n}(\mu', 1, \mathbb{C})$ exactly. Then $P^{\mathcal{O}} = P^{\#P}$.*

Hence our results establish a natural ensemble of matrices for which permanent approximation is efficient on average even though the exact computation is not (see Table II). This should be contrasted with the central BosonSampling assumption which is that permanent approximation remains hard for zero mean.

D. Roots of random interpolating polynomials

In this work we consider the Taylor series technique of Barvinok (see Section I-B) in a random setting: We ask given an ensemble of random matrices A what can be said about the typical location of roots of some interpolating polynomial related to the permanent of A, say $g_A(z) = \mathrm{Per}((1-\mathrm{z})\mathrm{J}+\mathrm{zA})$?

Notably, this question completely changes the flavor of the analysis: Instead of considering families of matrices with some fixed property (say diagonal dominance) and analyzing them combinatorially, we consider random matrices, and then treat $g_A(z)$ as a random polynomial which allows us to bring to bear techniques from analysis of random polynomials.

First, to simplify matters we consider instead the polynomial

$$g_A(z) = \mathrm{Per}(\mathrm{J} + \mathrm{zA}), \tag{2}$$

and observe that for any non-zero z if A is a random matrix with mean 0, then $g_A(z)$ is (up to normalization by z^n) the permanent of a biased version of A with mean $1/z$. Given this polynomial we ask: What is the distribution of roots of $g_A(z)$? Our goal is to show that we can find a sequence of overlapping disks, whose radii is not too small, that allows us to apply *analytic continuation* (see e.g. [17]) from the value of the function $g_A(z)$ at $z = 0$ to $g_A(z)$ for some $|z| \gg 1$.

A useful technique in the analysis of random polynomials is Jensen's formula which states that for a function $g(z)$ with zeros $z_1, z_2, \ldots$ that is analytic in the complex disk of radius r centered at the origin and $g(0) \neq 0$ we have

$$\int_0^{2\pi} \ln(|g_A(re^{i\theta})|)\frac{d\theta}{2\pi} - \ln(|g(0)|) = \sum_{|z_j|\leq r} \ln\frac{r}{|z_j|}. \quad (3)$$

In order to apply Jensen's formula we observe that the left hand side of the formula, for zero mean random matrices A, and $g_A(z) = \mathrm{Per}(\mathrm{J} + \mathrm{zA})$ is essentially bounded from above by the the logarithm of the second moment $g_A(r)$. We prove in Lemma 6 (which is based on Proposition 1 of Rempała and Wesołowski [18]) that this second moment is upper-bounded by a term that scales at most exponentially with the square of r:

$$\mathbf{E}_A[|g_A(r)|^2] \leq (n!)^2 \cdot e^{r^2}. \quad (4)$$

Together with Jensen's formula this bound implies two interesting facts about the roots of $g_A(z) = \mathrm{Per}(\mathrm{J} + \mathrm{zA})$ summarized in Proposition 7: That typical matrices A are such that $g_A(z)$ has no roots inside the disk of radius $|z|$ for $|z| \ll 1$, and very few (say $O(\ln(n))$) roots in the disk of radius $|z| = \sqrt{\ln(n)}$.

These two facts imply together the existence of analytic curves arching from $z = 0$ to some value z, $|z| \gg 1$ with the following property: For a typical A from the distribution, these curves are at distance at least some small but large enough $\varepsilon > 0$ from any root of $g_A(z)$. These curves are depicted in Figure 1. The specific choice of curves that we use in this work is according to Figure 2. Proposition 7 implies that most curves of this form avoid all roots of $g_A(z)$ with margin at least ε for most A's, so our algorithm samples such a curve at random and use it to interpolate the value of $g(z)$ from $z = 0$ to a large value of $|z|$.

E. Turning into an algorithm

To recap the previous section, our overall strategy is to compute a Taylor series expansion of the logarithm of a polynomial $g_A(z)$ related to the permanent of a random matrix A. In order to do that, we characterize the location of the roots of $g_A(z)$ for a typical matrix A which allows us to find simple curves in the complex plane which are at distance at least some small but large enough $\varepsilon > 0$ from any root of $g_A(z)$ for most A's. This would imply that for such matrices A the function $f_A(z) = \ln(g_A(z))$ is analytic on any point along these curves, up to radius ε.

However, it is not immediately clear that this analytic continuation strategy can be turned into an algorithm. Suppose $g_A(z)$ is root-free within ε-neighborhood of each point of the segment $[0, 1]$. In his work [5] Barvinok suggested composing $g_A(z)$ with an auxiliary polynomial ϕ related to $e^{O(1/\varepsilon)}$ terms in the Taylor expansion of $\varepsilon \ln \frac{1}{1-z}$ around $z = 0$. He showed that $g_A \circ \phi$ has indeed no roots inside a disk of radius $1 - e^{-O(1/\varepsilon)}$ around $z = 0$. See Lemma 8.1 of [5] and Section 2.2 of [19] for more details.

It is somewhat less clear however, whether one can use the auxiliary map strategy for tubes around more elaborate curves like a piecewise linear curve (which we consider in this work), and whether such maps would result in good error parameters. In order to extend Barvinok's approach to such curves one needs to compose ϕ with another auxiliary low-degree polynomial map which maps the ε neighborhood of $[0, 1]$ to an $O(\varepsilon)$ neighborhood of these curves. We however use a different method which allows us to interpolate along an arbitrary (even non-differentiable) curve. We name this complementary algorithmic technique "computational analytic continuation".

1) Computational analytic continuation (CAC): CAC is an algorithm that for a degree n polynomial $g(z)$ computes the value of $f(z) = \ln(g(z))$ at some value z given an oracle access to derivatives of $f(z)$ at $z = 0$, and a curve γ, $\gamma(0) = 0, \gamma(1) = z$ that is at distance at least some $\varepsilon > 0$ from all roots of $g(z)$.

Let $g(z)$ be a univariate polynomial $g : \mathbb{C} \to \mathbb{C}$ and assume we have an oracle which computes the derivatives of this function at $z = 0$. Let γ be some curve in the complex plane that is at distance at least R from any root of $g(z)$. We would like to approximate $f(\sigma) = \ln(g(\sigma))$ for

some complex number $\sigma = \gamma(1) \neq 0$, using a Taylor series expansion of order m, that is small as possible. For simplicity, we assume that γ is piece-wise linear and divide each segment of γ into small intervals. We denote the entire sequence of intervals as $\Delta_1, \ldots, \Delta_t$. For each i the length of Δ_i is at most R/β for all i, for some $\beta > 1$.

We then use a sequence of iterations to compute many derivatives at each step: At the first step $z = 0$ we compute some m derivatives, where m is suitably chosen for small approximation error. Then at the next step, we compute $O(m/\ln(m))$ derivatives at point Δ_1. The update rule of k-th derivative at step i, denoted by $\hat{f}_i^{(k)}$ is merely the Taylor series approximation of the k-th derivative $f^{(k)}$ as an analytical function to order m:

$$\hat{f}_i^{(k)} \leftarrow \sum_{j=0}^{m} \frac{f_{i-1}^{(k+j)} \Delta_1^k}{k!}. \tag{5}$$

In general at each step i we compute s_i derivatives where the number of derivatives compute at each step is reduced sharply with i:

$$s_i \approx O(s_{i-1}/\ln(s_{i-1})). \tag{6}$$

We choose m sufficiently large so that s_t is sufficiently large to imply a $1/\mathrm{poly}(n)$ additive error in the final Taylor series approximation of $f_t^{(0)}$. Intuitively, if t is the number of overlapping disks, then we need to fix $m \approx \ln n \times (\ln \ln n)^t$.

Since γ is at distance at least R from any root of $g(z)$, and the step sizes Δ_i are chosen to be sufficiently small compared to the convergence radius R around the previous point $\sum_{j=0}^{i-1} \Delta_j$ it follows that the Taylor series converges quickly to the true function value at each step. We prove a quantitative estimate:

Lemma (Sketch)**.** *Let $\Delta_{min} = \min_i |\Delta_i|$ and suppose that $\Delta_{min} \leq R/\beta$ where R is the convergence radius of $\ln(g(z))$ around point $\sum_{j=0}^{i-1} \Delta_j$, minimized over all $1 \leq i < t$. Consider the update rule in Equation 5 that computes s_i derivatives $f_i^{(k)}$ using s_{i-1} previously computed derivatives $f_{i-1}^{(k)}$. Suppose that $s_0 = \ln \frac{n\sigma}{\delta\Delta} O(\ln 1/\Delta_{min})^t$ for some error parameter $\delta = 1/\mathrm{poly}(n)$. Then*

$$\left| \hat{f}_t^{(0)} - f(\sigma) \right| \leq \delta. \tag{7}$$

We then show that for a specific choice of parameters $\sigma = O(\ln n)$, $\Delta_{min} > 1/\mathrm{polylog}(n)$, and $t = \frac{\ln \ln n}{\ln \ln \ln n}$ we get an inverse polynomial error

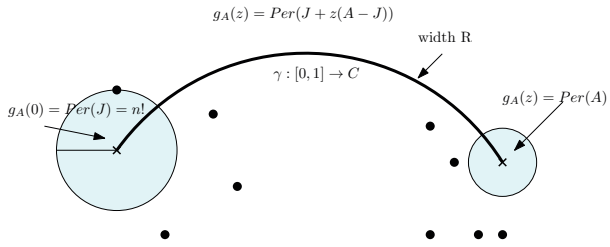

Figure 1: The curve γ connects $z = 0$ and some value $z \neq 0$ and is at distance at least R from any root of $g_A(z)$. Note that interpolation along the curve in small segments can reach a lot further than taking a single interpolation step which is bounded by blue-shaded region - dictated by the location of the nearest root to the point $z = 0$.

by using a poly-logarithmic number of derivatives. Since $e^{f^{(t)}} = g(\sigma) = \mathrm{Per}(\mathrm{J} + \sigma\mathrm{A})$ for zero mean random matrix A then $\sigma^{-n} e^{\hat{f}_t^{(0)}}$ is a $1 + 1/\mathrm{poly}(n)$ multiplicative approximation of the same random matrix but with vanishing mean, i.e. $\mu = 1/\sigma$.

F. Discussion and future work

Our study extends the line of work pioneered by Barvinok that views the computation of the permanent of a matrix A as a question about a related complex-valued polynomial $g_A(z)$. In this work we allow A to be a random matrix and hence recast the question about the permanent of a random matrix A as a question about the location of roots of a random polynomial related to A. We characterize the behavior of these polynomials for some random matrices A, and then provide an algorithmic technique which allows us to turn this knowledge into a quasi-polynomial algorithm for approximating the permanent of these matrices.

For a while now, it has been a folklore notion that the permanent is difficult to compute for general matrices mainly because of the "sign problem" - namely the fact that entries with opposing signs generate intricate interference patterns. Such matrices avoid by definition the regime of matrices for which efficient algorithms are known, most prominent of which is the algorithm of Jerrum, Sinclair and Vigoda [2] for non-negative entry matrices. Our work, we believe, places this notion in serious doubt - we show natural random ensembles with very little (i.e. vanishing) correlation between any pair of entries - and yet, we are able to approximate the permanent for such matrices quite efficiently. Hence, it seems that if in fact approximation of the permanent on average is a difficult task, it must be due to another, yet uncharacterized phenomenon. Furthermore, our study makes the

hardness of approximating the permanent on average to an even more intriguing problem: It seems that several natural ensembles do in fact admit an efficient approximation - but is it the case for other ensembles? Most notably, one would like to consider the case of zero mean complex Gaussian matrices, the presumed hardness of which is the basis for the BosonSampling paradigm, discussed in the following section.

1) Implications to BosonSampling*:* In [3] the authors consider the following computational problem:

Definition 1 ($\mathrm{GPE}_{\times}^{\mu}$)**.** *Given $A \sim \mathcal{N}^{n\times n}(\mu, 1, \mathbb{C})$, ε, δ, output a number Q such that with probability at least $1-\delta$, $\left|Q - \mathrm{Per(A)}\right| \leq \varepsilon|\mathrm{Per(A)}|$ in $\mathrm{poly}(n, 1/\varepsilon, 1/\delta)$. We use the notation $\mathrm{GPE}_{\times}^{\mu}(\varepsilon, \delta)$ for the specific choice of parameters.*

They conjecture that

Conjecture 2. $\mathrm{GPE}_{\times}^{0}$ *is $\#P$-hard to compute.*

Together with another conjecture on the anti-concentration of the permanent of complex Gaussian matrices this conjecture implies that BPP simulation of the linear-optical experiment called BosonSampling to within vanishing total variation distance implies collapse of the polynomial hierarchy to the third level, thereby establishing a so-called "quantum supremacy" of the outcomes of these physical experiments.

Using the same anti-concentration assumption on the permanent of zero mean Gaussian matrices one can show that in fact the above conjecture is true also for complex Gaussian matrices with mean $\mu = 1/\mathrm{poly}(n)$. More concretely

Theorem 3. *Assume that the following conjecture on anti-concentration of the permanent holds for the standard Gaussian matrix $A \sim \mathcal{N}^{n\times n}(0, 1, \mathbb{C})$:*

$$\Pr_{A}\left(|\mathrm{Per(A)}| > \frac{\sqrt{\mathrm{n!}}}{\mathrm{n^c}}\right) \geq 1 - \delta. \tag{8}$$

Then for any $K > 0$ and $\mu \leq \frac{1}{\sqrt{n-1}}$:

$$\mathrm{GPE}_{\times}^{0}\left((\varepsilon + \mu \cdot nK)n^c, \quad \delta + \delta' + \tfrac{1}{K^2}\right)$$

$$\preceq$$

$$\mathrm{GPE}_{\times}^{\mu}(\varepsilon, \delta), \tag{9}$$

where $\mathrm{GPE}_{\times}^{0}(\varepsilon, \delta)$ is defined in definition 1.

The proof of this theorem can be found in [16].

On the other hand, our main theorem implies that

$$\mathrm{GPE}_{\times}^{1/\mathrm{polyloglog}(n)} \in \mathrm{DTIME}\left(2^{\mathrm{polylog}(n)}\right), \tag{10}$$

and hence $\mathrm{GPE}_{\times}^{1/\mathrm{polyloglog}(n)}$ is very unlikely to be $\#P$-hard.

This raises the following intriguing question: It seems that the hardness of the permanent of complex Gaussian matrices (or general random matrices for that matter) is not due to the common intuition that the different signs of the entries prohibit combinatorial treatment of the matrix, as a graph, in the spirit of [2]. Hence, if indeed $\mathrm{GPE}_{\times}^{0}$ is hard there must exist another phenomenon governing the behavior of the permanent of complex Gaussian matrices with mean values between $\mu = 1/\mathrm{poly}(n)$ and $\mu = 1/\mathrm{polyloglog}(n)$ which makes computation intractable.

2) Reducing the mean value: A natural next step for our approach is to attempt to further increase the value of z for which we evaluate $g(z)$. Approximating $g_A(z)$ for typical A at $|z| = 1/\mu$ implies an approximation of the permanent for a random matrix with mean μ and variance 1.

However, one can see from the upper-bound on the interpolation error above that in order to achieve sufficiently small error the number of derivatives we initially compute must scale doubly exponentially in the ratio σ/Δ, namely the ratio of the interpolation length σ, and the step size Δ. Since $\Delta \sim 1/N_\sigma$ where $N_\sigma = \Omega(\sigma^2)$ is the number of roots in the disk of radius σ, then $\Delta \sim 1/\sigma^2$ which implies that the number of required derivatives is exponential in $\mathrm{poly}(\sigma)$.

Thus to improve on our technique new ideas would be required which would make a more economic use of the derivatives computed, and not "discard" at each step i a fraction $1/\ln(s_i)$ of the s_i derivatives computed at that step. Another approach would be to tighten the bound on the number of roots inside the disk of a given radius for the polynomial $g_A(z)$ or some other related polynomial.

3) Anti-concentration of the permanent: We point out that our algorithm, and also that of [5] and [6] are not merely a sequence of computational steps, but actually show that the permanent of typical complex Gaussian (or more generally, random) matrices $A \sim \mathcal{N}^{n\times n}(\mu, 1, \mathbb{C})$ are well approximated by a low-degree polynomial in the entries of $A \sim \mathcal{N}^{n\times n}(0, 1, \mathbb{C})$ - in fact a polynomial

of degree $\mathrm{polylog}(n)$. Such a statement strongly indicates that for this range the permanent of complex Gaussian matrices is anti-concentrated, since it is well-known by a theorem of Carbery and Wright [20] that any polynomial of low-degree in standard i.i.d. complex Gaussian variables is anti-concentrated.

However, we have been unable to use this theorem directly to derive a formal anti-concentration statement. We note that the anti-concentration of the Gaussian permanent is also a necessary conjecture of the BosonSampling paradigm in [3] along with the conjectured $\#P$-hardness of this problem. Hence, intriguingly it seems that the conjectures on the computability, and statistics (i.e. anti-concentration) of the permanent of the complex Gaussian matrices are closely related: Our results suggest that for mean values $1/\mathrm{polylog}(n)$ the anti-concentration conjecture holds, but ironically this is because the second conjecture does not hold for this regime - i.e. the permanent is relatively easy to compute via a polynomial of low degree.

II. Preliminaries

A. Notation

$\mathbb{C}$ denotes the complex numbers, and $\mathbb{R}$ denotes the real numbers. $\mathcal{N}(\mu, \sigma, \mathbb{C})$ is the complex Gaussian distribution of mean μ and variance σ^2. $\mathrm{Bern}(\mu)$ is the biased-Bernoulli random variable - it is 1 with probability $1/2$ and $-1+\mu$ with probability $1/2$. $\mathrm{Per}(\mathrm{X})$ is the permanent of matrix X. $\ln(x)$ is the natural complex logarithm of x, defined up to an additive factor $2\pi i k$ for integer k. $\mathcal{B}_r(z)$ denotes the closed disk in the complex plane of radius r around point z. For computational problems A, B we denote $A \preceq B$ if there exists a poly-time reduction from A to B - i.e. A is no harder than B.

Definition 4 (Random Matrix)**.** *An $n \times n$ matrix A is called a random matrix and denoted by $A \sim \mathcal{D}^{n\times n}(\mu, \sigma^2)$, if each entry of A is independently drawn from some complex valued distribution with mean μ and variance σ^2.*

The entries of A in the above definition do not have to be identically distributed. We denote the distribution of complex Gaussian matrices with mean μ and variance σ^2 with $\mathcal{N}^{n\times n}(\mu, \sigma^2, \mathbb{C})$.

III. Permanent-interpolating-polynomial as a random polynomial

As described in Section I-B recent studies designed algorithms for evaluating multi-variate functions like the permanent or Ising model by considering a related univariate polynomial $g(z)$. These schemes used this polynomial to interpolate the value of $\ln(g(z))$ at some point of interest $z = z_0$ using knowledge of the derivatives at $z = 0$.

For example, in his work, Barvinok [5] used the polynomial $g(z) = \mathrm{Per}(\mathrm{J}\cdot(1-\mathrm{z}) + \mathrm{A}\cdot\mathrm{z})$. In a more recent work [15] the authors choose a different polynomial $g(z) = Z_\beta(z)$ - namely the Ising partition function. In both of these works the authors characterized the location of the roots of $g(z)$ in order to establish that $\ln(g(z))$ is analytical in the region of interest. Indeed, in his work, Barvinok showed that $g(z)$ has no roots in the unit disk for any matrix A that satisfies $\max_{ij}|A_{ij} - J| \leq 0.5$ Likewise, the polynomial in the work of [15] the polynomial $g(z)$ was shown to have no roots inside the unit disk using the Lee-Yang theorem.

In our work we consider the polynomial

$$g_A(z) := \mathrm{Per}(\mathrm{J} + \mathrm{z}\cdot\mathrm{A}), \tag{11}$$

and then analyze it as a random polynomial in order to gain insight into the distribution of its roots. The choice of this polynomial has a more natural interpretation in the context of random matrices: For a zero mean random matrix A, the value of $g_A(z)/z^n$ for nonzero z is the value of the permanent of a matrix drawn from the ensemble A when shifted with a mean of $1/z$ and variance 1. Another reason why we choose this polynomial is that it is easier to bound its roots compared to $\mathrm{Per}((1-\mathrm{z})\mathrm{J} + \mathrm{z}\cdot\mathrm{A})$.

We begin with the following definition

Definition 5 (Average sensitivity)**.** *Let $g_A(z)$ be a random polynomial where A is a random matrix. For any real $r > 0$ the stability of $g_A(z)$ at point r is defined as*

$$\kappa(r) \equiv \kappa_g(r) := \mathop{\mathbb{E}}_{\theta}\mathop{\mathbb{E}}_{A}\left[\frac{|g_A(re^{i\theta})|^2}{|g_A(0)|^2}\right], \tag{12}$$

where $\mathbb{E}_\theta[\cdot] = \int_{\theta=0}^{2\pi}[\cdot]\frac{d\theta}{2\pi}$ is the expectation over θ from a uniform distribution over $[0, 2\pi)$.

We begin with an upper-bound on the average sensitivity of the permanent of a random matrix that can also be derived from the work of Rempała and Wesołowski [18] (see Proposition 1).

Lemma 6. *Let $A \sim \mathcal{D}^{n\times n}(0,1)$. Then*

$$\kappa_g(r) \le e^{r^2}. \tag{13}$$

A somewhat simpler and more intuitive proof of this lemma is given in [16].

Our next result is to relate the number of roots of g_A inside a disk of certain radius around origin to its average sensitivity around the boundary of that disk. Jensen's formula provides this connection.

Proposition 7. *Let*

$$g_A(z) = \mathrm{Per}(\mathrm{J} + \mathrm{z}\cdot\mathrm{A}), \tag{14}$$

and $A \sim \mathcal{D}^{n\times n}(0,1)$. Then if N_r is the number of roots of g_A inside a disk of radius r, $\mathbb{E}_A[N_r] \le 4r^2$. In particular:

1) *Let $r > 1$. With probability at least $1 - \frac{1}{r}$ the polynomial $g_A(z)$ has at most $4r^3$ roots inside the disk with radius r around $z = 0$.*
2) *Let $r < 1/2$. With probability at least $1 - 4r^2$ there are no roots inside the disk with radius r around $z = 0$.*

The above claim immediately implies:

Corollary 8. *Let ε be real number $0 < \varepsilon < 0.5$. For at least $1 - 3\varepsilon$ fraction of matrices $A \sim \mathcal{D}^{n\times n}(0,1)$, g_A has no roots inside $\mathcal{B}_\varepsilon(0)$ and has at most $32/\varepsilon^3$ roots inside $\mathcal{B}_{2/\varepsilon}(0)$.*

Proof: The proof follows by a union bound on the two items in Proposition 7. In particular, the probability that $\mathcal{B}_\varepsilon(0)$ is not root free is at most $4\varepsilon^2$, and the probability that $\mathcal{B}_{2/\varepsilon}(0)$ has more than $32/\varepsilon^3$ roots is at most $\varepsilon/2$, so using union bound for $\varepsilon < 0.5$ the probability of error amounts to $\varepsilon/2 + 4\varepsilon^2 < 3\varepsilon$. ∎

Proof of Proposition 7: We use Jensen's formula. Let $g : \mathbb{C} \to \mathbb{C}$ be an arbitrary polynomial such that $g_A(0) \neq 0$. Let N_r be the number of zeros of g inside a disk of radius r centered around $z = 0$, and let $z_1, \ldots, z_{N_r} \in \mathbb{C}$ be these zeros. Then Jensen's formula is

$$\sum_{|z_j|\le r} \ln\frac{r}{|z_j|} + \ln|g_A(0)| = \mathop{\mathbb{E}}_{\theta} \ln|g_A(re^{i\theta})|, \tag{15}$$

where we have used the notation

$$\int_{\theta=0}^{2\pi}[\cdot]\frac{d\theta}{2\pi} =: \mathop{\mathbb{E}}_{\theta}[\cdot]. \tag{16}$$

Let $0 < \delta < 1$ be a real number. We first use the following bound

$$\begin{aligned}
\sum_{|z_j|\le r} \ln\frac{r}{|z_j|} &\ge \sum_{|z_j|\le r(1-\delta)} \ln\frac{r}{|z_j|}, && (17)\\
&\ge \sum_{|z_j|\le r(1-\delta)} \ln\frac{r}{r(1-\delta)}, && (18)\\
&\ge \delta\cdot N_{r(1-\delta)}. && (19)
\end{aligned}$$

We now pick $A \sim \mathcal{D}^{n\times n}(0,1,\mathbb{C})$ and view the variables in the Jensen's formula above as random variables depending on A. By Jensen's formula

$$\begin{aligned}
\delta\cdot N_{r(1-\delta)} &\le \mathop{\mathbb{E}}_{\theta}\left[\ln(|g_A(re^{i\theta})|)\right] - \ln(|g_A(0)|), && (20)\\
&= \mathop{\mathbb{E}}_{\theta}\left[\ln\left(\left|\frac{g_A(re^{i\theta})}{n!}\right|\right)\right], && (21)\\
&= \frac{1}{2}\mathop{\mathbb{E}}_{\theta}\left[\ln\left(\left|\frac{g_A(re^{i\theta})}{n!}\right|^2\right)\right], && (22)\\
&= \frac{1}{2}\cdot\mathop{\mathbb{E}}_{\theta}[\ln b(r,\theta)], && (23)
\end{aligned}$$

where $b(r,\theta) := \left|\frac{g_A(re^{i\theta})}{n!}\right|^2$. Then by Lemma 6

$$\forall\theta,\quad \mathop{\mathbb{E}}_{A}[b(r,\theta)] \le e^{r^2}, \tag{24}$$

and so in particular

$$\mathop{\mathbb{E}}_{\theta,A}[b(r,\theta)] \le e^{r^2}. \tag{25}$$

So by concavity of the logarithm function

$$\begin{aligned}
\delta\cdot\mathop{\mathbb{E}}_{A} N_{r(1-\delta)} &\le \frac{1}{2}\mathop{\mathbb{E}}_{\theta,A}\left[\ln\Big(b(r,\theta)\Big)\right], && (26)\\
&\le \frac{1}{2}\ln\Big(\mathop{\mathbb{E}}_{\theta,A}[b(r,\theta)]\Big), && (27)\\
&\le r^2/2. && (28)
\end{aligned}$$

As a result, choosing $\delta = 1/2$ and doing the change of variable $r \leftarrow r/2$,

$$\mathop{\mathbb{E}}_{A} N_r \le 4r^2. \tag{29}$$

Thus by Markov's inequality, when $r > 1$:

$$\Pr_A(N_r \ge 4r^3) \le \frac{1}{r}. \tag{30}$$

Next we consider the case when $r < 1$. In this case we can directly use the following Markov's inequality using equation 29:

$$\Pr(N_r \ge 1) \le 4r^2. \tag{31}$$

Note this bound is useful only when $r < 1/2$. ∎

A. Root-avoiding curves

We now use the above insight on the distribution of roots of the random polynomial $g_A(z)$ to edge closer to an algorithm. We define:

Definition 9 (Root-free area)**.** *A subset $S \subseteq \mathbb{C}$ is root-free w.r.t. polynomial $g(z)$ if it contains no roots of g.*

Definition 10 (Tube of width w around a curve)**.** *Let $w > 0$ be a real number, and let $\gamma : [0,1] \to \mathbb{C}$ denote some parameterized curve in the complex plane. The tube of width w around γ denoted by $\mathcal{T}(\gamma, w)$, is the set of points defined as $\cup_{x\in\gamma}\mathcal{B}_w(x)$, where $\mathcal{B}_w(x)$ is the closed w-ball centered around $x \in \mathbb{C}$. In other words, for each point on the curve we include the w-ball around it in the tube.*

We will denote by $L(a,b)$ the linear segment in $\mathbb{C}$ between $a, b \in \mathbb{C}$.

Lemma 11 (Finding a root avoiding tube)**.** *Let $0 < \varepsilon < 0.1$ and $A \sim \mathcal{D}^{n\times n}(0,1)$. Fix $w = \pi\varepsilon^6$. For numbers $a \in \mathbb{C}, b \in \mathbb{R}$ consider the following curve*

$$\gamma_{a,b}(t) = \begin{cases} at & t \in [0, \frac{1}{2}) \\ a(1-t) + 2b(t - \frac{1}{2}) & t \in [\frac{1}{2}, 1] \end{cases} \tag{32}$$

There exists $a \in \mathbb{C}$, $|a| \leq 2\varepsilon$ and $b \in \mathbb{R}$, $b \in [1/\varepsilon, 1/\varepsilon + 2\varepsilon]$ such that the tube $\mathcal{T}(\gamma_{a,b}, w)$ is root-free with respect to $g_A(z)$ for a fraction at least $1 - 4\varepsilon$ over choices of A. The total length of such a tube is at most $2/\varepsilon$.

Proof: Fix $M = 32/\varepsilon^5$. For each $j \in [M]$ define the piece-wise linear curve γ_j made of two segments

$$L(0, 2\varepsilon + i \cdot 2\varepsilon \tan(2\pi j/M)), \tag{33}$$

and,

$$L(2\varepsilon + i \cdot 2\varepsilon \tan(2\pi j/M), 1/\varepsilon \cdot \tan(2\pi j/M)). \tag{34}$$

See Figure 2. Note that specifically for a small subset of indices $j \in [M/8, \ldots, M/8 + \varepsilon M]$ the value of b is locked in a tight range:

$$|a| \leq 2\varepsilon, \quad b \in [1/\varepsilon, 1/\varepsilon + 2\varepsilon]. \tag{35}$$

Let $\mathcal{T}_j = \mathcal{T}(\gamma_j, w)$. Then by Equation 35 the union of these tubes is contained inside a ball that is not too large:

$$\bigcup_j \mathcal{T}_j \subset \mathcal{B}_{2/\varepsilon}(0). \tag{36}$$

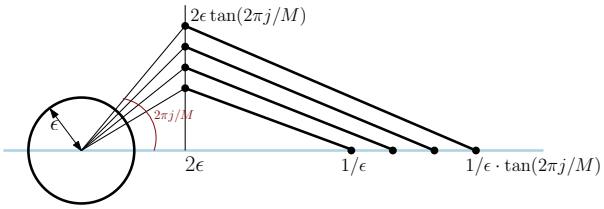

Figure 2: The family of 2-piecewise linear curves interpolating from $z = 0$ to some point z on the real line. The curves branch out from the unit circle at an angle between $\pi/4$ and $\pi/4 + \varepsilon$. Each curve starts out at an angle of $2\pi j/M$ for some integer j and hits the imaginary axis at $\Re(z) = 2\varepsilon$. Hence the imaginary magnitude of the end-point of the first segment in the j-th curve is $2\varepsilon\tan(2\pi j/M)$. After hitting the imaginary axis at $\Re(z) = 2\varepsilon$, they descend back to the real line in *parallel*, thus hitting the real line at different points. The bottom-most curve hits the real line at $z = 1/\varepsilon$. Note that by this definition tubes of weight w around each curve do not overlap outside the ball of radius ε and in particular when they hit the imaginary axis at $\Re(z) = 2\varepsilon$.

and that by our definition of w these tubes are disjoint outside a ball of radius ε:

$$\forall j \neq k, \quad \mathcal{T}_j \cap \mathcal{T}_k - \mathcal{B}_\varepsilon(0) = \emptyset. \tag{37}$$

Let E the event of no roots at distance at most ε to the origin and few roots inside the disk of radius $2/\varepsilon$:

$$E = \{A : N_\varepsilon = 0, \quad \wedge \quad N_{2/\varepsilon} \leq 32/\varepsilon^3\}. \tag{38}$$

By Corollary 8

$$\Pr(E) \geq 1 - 3\varepsilon. \tag{39}$$

Condition on E and for each $j \in [M/8, \ldots, M/8 + \varepsilon M]$ and matrix A let $m_{j,A}$ denote the number of roots of $g_A(z)$ inside $\mathcal{T}_j - \{0\}$. Then by Equation 37 and the definition of E we have

$$\forall A \in E, \quad \sum_{j=M/8}^{M/8+\varepsilon M} m_{j,A} \leq N_{2/\varepsilon}, \tag{40}$$

and so for uniform random j the average number of roots is small

$$\forall A \in E, \quad \mathbf{E}_j\left[m_{j,A}\right] \leq N_{2/\varepsilon}/(\varepsilon M) \leq \varepsilon, \tag{41}$$

and so in particular this holds for average matrix A (conditioned on E):

$$\mathbf{E}_{A|E}\mathbf{E}_j\left[m_{j,A}\right] \leq \varepsilon, \tag{42}$$

so by linearity of expectation

$$\mathbf{E}_j\mathbf{E}_{A|E}\left[m_{j,A}\right] \leq \varepsilon, \tag{43}$$

which implies that there exists $j_0 \in \{M/8, \ldots, M/8 + M\varepsilon\}$ (an index that minimizes $\mathbf{E}_{A|E}[m_{j,A}]$) such that

$$\mathbf{E}_{A|E}[m_{j_0,A}] \leq \varepsilon, \tag{44}$$

hence for j_0 we have

$$\Pr_{A|E}(m_{j_0,A} > 0) \leq \varepsilon, \tag{45}$$

and so by the union bound with the probability of E from Equation 39

$$\Pr_{A}(m_{j_0,A} > 0) \leq \varepsilon + 3\varepsilon = 4\varepsilon. \tag{46}$$

Note that the total length of this tube is at most $2/\varepsilon$.

■

IV. Computational analytic continuation

In this section we pose the question of analytic continuation as a computational task and devise a method to derive the value of the function $g(z)$ at some point $z = r$ using it's derivatives at $z = 0$, assuming that g has a root-free tube around a curve γ, $\gamma(0) = 0, \gamma(1) = z$. We require the following result of Barvinok:

Lemma 12 (Efficient computation of derivatives [5])**.**

1) *Let A be an $n \times n$ complex matrix and let $g_A(z) := \mathrm{Per}(\mathrm{J+zA})$, where J is all ones matrix. There exists a deterministic algorithm that runs in time $n^{O(l)}$ and computes the l-th derivative of $g_A(z)$ at point $z = 0$.*
2) *Let $g(z)$ be a polynomial of degree n. Given an algorithm that runs in time t and computes the first ℓ derivatives of $g(z)$ at point $z = 0$, one can compute in time $O(\ell^2 t)$ the first ℓ derivatives of $f(z) = \ln(g(z))$ at $z = 0$.*

We also need the following technical numerical result the proof of which is deferred to [16].

Lemma 13. *For all $m \geq 2l$ and $\beta \geq e$*

$$\sum_{k=m}^{\infty} \beta^{-k} \cdot k^l \leq 3 \cdot \beta^{-m} m^l. \tag{47}$$

We now present a deterministic algorithm for computing the analytic continuation (see [17]) of a degree n polynomial $g : \mathbb{C} \to \mathbb{C}$.

Algorithm 14 (Computational analytic continuation).

1) ***Input:*** *An oracle $\mathcal{O}_g$ that takes a number m as input and outputs the first m derivatives of g at $z = 0$. t complex numbers $\Delta_1, \ldots, \Delta_t$, a number $\beta > 1$, precision parameter $\delta > 0$, and integer m - the number of derivatives computed at the 0-th step.*
2) ***Fixed parameters:***
   a) $\Delta_{min} = \min_i |\Delta_i|$.
   b) $y_0 = 0$ *and* $y_i = \sum_{j=1}^{i-1} \Delta_j$ *for each* $1 \leq i \leq t-1$
3) ***Variables:***
   a) $\hat{f}_i^{(l)}$ *for* $1 \leq l \leq m$ *and* $0 \leq i \leq t-1$ % *the l-th derivative of f at y_i.*
   b) s_i *for* $0 \leq i \leq t-1$ % *the number of derivatives at each point y_i.*
   c) $s_0 \leftarrow m$.
4) ***Main:***
   a) *Query $\mathcal{O}_g(m)$ to obtain $g^{(0)}(0), \ldots, g^{(m)}(0)$*
   b) *Using Lemma 12 and derivatives from step I compute $\hat{f}_0^{(l)} \leftarrow f^{(l)}(y_0)$ for $1 \leq l \leq m$.*
   c) *For each $i = 0..t-1$:*
      - *Set:* $s_{i+1} \leftarrow \frac{\ln \beta}{2} \frac{s_i}{\ln(2s_i/\Delta_{min})}$.
      - *Compute:* $\forall k \in [s_{i+1}]$,

$$\hat{f}_{i+1}^{(k)} = \sum_{p=0}^{s_i - k} \frac{\hat{f}_i^{(p+k)}}{p!} \Delta_i^p. \tag{48}$$

5) ***Output:***
   Let $\hat{f} := \hat{f}_t^{(0)}$ and return $\mathcal{O} = e^{\hat{f}}$.

Prior to establishing the correctness of the algorithm, we define shifted versions of $g(z)$ as follows:

$$\forall i \in [t], \quad \tilde{g}_i(z) = g(z + y_i), \tag{49}$$

and

$$\tilde{f}_i(z) = \ln(\tilde{g}_i(z)), \tag{50}$$

and denote $f_i^{(l)} = \tilde{f}_i^{(l)}(0)$. Note y_i's are defined in algorithm 14. We need the following elementary fact which we leave without proof.

Lemma 15. *If the closest root of g to the point y_i in the complex plane is λ, then the closest root of $\tilde{g}_i$ to $z = 0$ is also λ.*

We now establish correctness:

Claim 16. *[Correctness of algorithm 14] Let $g(z)$ be a polynomial of degree at most n, and $f(z) = \ln(g(z))$. Suppose the inputs to algorithm 14 satisfy the following conditions:*

1) *Precision parameter: $\delta \geq n^{-c_1}$ for some constant $c_1 > 0$.*
2) *Interpolation length: $\sigma_t := \sum_i |\Delta_i| \leq c_2 \ln(n)$ for some constant $c_2 > 0$.*
3) *Minimal segment length: $\Delta_{min} = \min_i |\Delta_i| \geq \ln^{-c_3}(n)$ for some constant $c_3 > 0$.*
4) *Number of iterations: $t \leq c_4 \ln\ln(n) / \ln\ln\ln(n)$ for some constant $c_4 > 0$.*

5) *For each i the ratio of the distance of the closest root of $g(z)$ to y_i to step size $|\Delta_{i+1}|$ is at least $\beta \geq e$.*
6) *Number of derivatives at step 0: $m \leq \ln^{c_5}(n)$ for some constant $c_5 > 0$.*

Then the following holds: there exists a constant c_0 such that if the number of derivatives m that the algorithm queries from $\mathcal{O}_g$ at step 0 is at least

$$\ln(n) \cdot (c_0 \cdot \ln\ln(n))^t, \tag{51}$$

then output of the algorithm satisfies

$$\mathcal{O} = e^{\hat{f}} = g(y_t) \cdot (1 + \mathcal{E}), \quad |\mathcal{E}| = O(\delta). \tag{52}$$

where $\mathcal{E}$ is a complex number.

The proof of this claim is very technical and can be found in [16].

V. Approximation of permanents of random matrices of vanishing mean

A. Main theorem

Theorem 17. *Let $\delta = 1/\mathrm{poly}(n)$ and $\varepsilon \geq [\ln\ln\ln(n)/\ln\ln(n)]^{1/7}$. There exists $\mu \in [\varepsilon, \varepsilon + 2\varepsilon^2]$ such that for any distribution $\mathcal{D}^{n\times n}(\mu, 1)$ there exists an algorithm running in quasi-polynomial time that computes a number $\mathcal{O}$ that for $1 - o(1)$ fraction of matrices $A \sim \mathcal{D}^{n\times n}(\mu, 1)$ satisfies*

$$\mathcal{O} = \mathrm{Per(A)} \cdot (1 + \mathcal{E}), \quad \mathcal{E} \in \mathbb{C}, \quad |\mathcal{E}| = \mathrm{O}(\delta). \tag{53}$$

In particular the algorithm solves $\mathrm{GPE}^{\varepsilon}_{\times}$.

Proof: Let

$$A' \sim \mathcal{D}^{n\times n}(0, 1). \tag{54}$$

Set $w = \pi\varepsilon^6$. By Lemma 11 there exists a tube $\mathcal{T}(\gamma_{a,b}, w)$ for $|a| \leq 2\varepsilon, b \in [1/\varepsilon, 1/\varepsilon + 2\varepsilon]$ that is root-free with probability $g_{A'}(z)$ at least $1 - 4\varepsilon$ over choices of A'. We will now use the curve $\gamma_{a,b}$ to interpolate the value of the function $g(z) = \mathrm{Per(J + zA')}$ from $z = 0$ to $z = b \in \mathbb{R}$.

Suppose A' is such a matrix. Divide each of the two linear segments comprising $\gamma_{a,b}$ into small equal segments of size $\Delta_{min} = w/e = \pi\varepsilon^6/e$ each. Enumerate these segments together as $\Delta_1, \ldots, \Delta_t$ Then the number of segments t is at most the length of $\gamma_{a,b}$ divided by Δ_{min}. By Lemma 11 the length of $\gamma_{a,b}$ is at most $2/\varepsilon$ and so

$$t = |\gamma_{a,b}|/\Delta_{min} \leq 2e/(\varepsilon w) \leq \frac{2e}{\pi}\varepsilon^{-7} \leq e\varepsilon^{-7}. \tag{55}$$

We run algorithm 14 for the following parameters:
1) matrix A'
2) $\Delta_1, \ldots, \Delta_t$. Note that $y_t = \sum_{i=1}^{t} \Delta_i = b$ by definition of $\gamma_{a,b}$.
3) $\beta = e$.
4) precision parameter δ
5) $m = \ln(n) \cdot (c_0 \cdot \ln\ln(n))^t$, where c_0 is the constant implied by Claim 16

Recall the conditions of Claim 16:

$$\delta = n^{-\Omega(1)}, \sigma_t = O(\log(n)), \Delta_{min} = 1/\mathrm{polylog}(n), \tag{56}$$

$$t = O(\ln\ln(n)/\ln\ln\ln(n)), \beta = O(1), m = \mathrm{polylog}(n). \tag{57}$$

We now verify the conditions of Claim 16 in the order they appear:
1) $\delta = n^{-\Omega(1)}$ by assumption.
2) The total length of all segments is at most $2/\varepsilon$ by Lemma 11.
3) $\Delta_{min} = \frac{\pi}{e}\varepsilon^6 = \Theta(\frac{\ln\ln\ln n}{\ln\ln n})^{6/7} > 1/\mathrm{polylog}(n)$.
4) $t \leq e\varepsilon^{-7} = O(\frac{\ln\ln n}{\ln\ln\ln n})$.
5) For each i the ratio of the distance of the closest root of $g(z)$ to y_i to step size $|\Delta_{i+1}|$ is at least $\beta \geq e$: this follows by construction since $\mathcal{T}$ is root-free with parameter w, and $\Delta_{min} = w/e$.
6) The above value of t implies that

$$m = \ln(n) \cdot (c_0 \ln\ln(n))^t, \tag{58}$$
$$= \ln(n) \cdot \ln\ln(n)^{O(\ln\ln(n)/\ln\ln\ln(n))}, \tag{59}$$
$$= \mathrm{polylog}(n). \tag{60}$$

Hence we can invoke Claim 16. By this claim for $|\mathcal{E}| \leq \delta$ we have:

$$e^{\hat{f}} = \mathrm{Per(J + y_t \cdot A')} \cdot (1 + \mathcal{E}), \tag{61}$$
$$= \mathrm{Per(J + b \cdot A')} \cdot (1 + \mathcal{E}). \tag{62}$$

The matrix $J + b \cdot A'$ is distributed as $b \cdot A$ where $A \sim \mathcal{D}^{n\times n}(\mu, 1)$ for some $\mu \in [\varepsilon, \varepsilon + 2\varepsilon^2]$. Thus

$$b^{-n}e^{\hat{f}} = \mathrm{Per(A)} \cdot (1 + \mathcal{E}), \quad |\mathcal{E}| \leq \delta \tag{63}$$

where $A \sim \mathcal{D}^{n\times n}(\mu, 1)$.

Run-time:: Algorithm 14 requires an oracle $\mathcal{O}_g$ at step 4a for computing derivatives of g at $z = y_0 = 0$: Using item (2) of Lemma 12 we can implement $\mathcal{O}_g$ in $n^{O(m)}$ time. Next, to compute the m derivatives of $f(z) = \ln(g(z))$ at $z = y_0$ at step 4b using $\mathcal{O}_g$ we invoke item (1) of the lemma, and compute them in time at most $O(m^2 n^{O(m)}) = n^{O(m)}$. Finally, we update at most m derivatives along $t = O(\ln(n))$ steps, where each update requires at most m summands. This results in total complexity

$$O(t \cdot m^2) \cdot n^{O(m)} = n^{O(m)} = 2^{\mathrm{polylog}(n)}. \tag{64}$$

■

B. Natural biased distributions

The following corollaries immediately follow from this theorem by choosing $\varepsilon = 1/\mathrm{polyloglog}(n)$:

Corollary 18.

1) ***"Slightly-biased" Gaussian:*** *There exists* $\mu = 1/\mathrm{polyloglog}(n), \delta = 1/\mathrm{poly}(n)$ *and a deterministic algorithm that for* $A \sim \mathcal{N}^{n\times n}(\mu, 1, \mathbb{C})$ *computes a* $1+\delta$ *multiplicative approximation of* $\mathrm{Per(A)}$ *on a* $1 - o(1)$ *fraction of matrices* A.
2) ***"Slightly-biased" Bernoulli:*** *There exists* $\mu = 1/\mathrm{polyloglog}(n), \delta = 1/\mathrm{poly}(n)$ *and a deterministic algorithm that for* $A \sim \mathrm{Bern}(\mu)^{n\times n}$ *computes a* $1+\delta$ *multiplicative approximation of* $\mathrm{Per(A)}$ *on a* $1 - o(1)$ *fraction of matrices* A.

Acknowledgement

We thank Scott Aaronson, Alexander Barvinok, and Aram Harrow for helpful discussions. LE and SM were partially supported by NSF grant CCF-1629809.

References

[1] A. Barvinok. Computing the permanent of (some) complex matrices. *Foundations of Computational Mathematics*, 16(2):329–342, Apr 2016. arXiv: 1405.1303

[2] M. Jerrum, A. Sinclair, and E. Vigoda. A polynomial-time approximation algorithm for the permanent of a matrix with nonnegative entries. *J. ACM*, 51(4):671–697, July 2004.

[3] S. Aaronson and A. Arkhipov. The computational complexity of linear optics. *Theory of Computing (ToC)*, Volume 9 (2013) Article 4 pp. 143-252. arXiv: 1011.3245

[4] A. Barvinok. Computing the partition function for cliques in a graph. *Theory of Computing*, 11(13):339–355, 2015. arXiv: 1405.1974

[5] A. Barvinok. Approximating permanents and hafnians. *Journal of Discrete Analysis*. 2017, Paper No. 2, 34 pp arXiv:1601.07518

[6] A. Barvinok and P. Soberón. Computing the partition function for graph homomorphisms. *Combinatorica*, 37(4):633–650, Aug 2017. arXiv: 1410.1842

[7] L. Gurvits. On the complexity of mixed discriminants and related problems, pages 447–458. *Springer*, Berlin Heidelberg, Berlin, Heidelberg, 2005.

[8] L. Valiant. The complexity of computing the permanent. *Theoretical Computer Science*, 8(2):189 – 201, 1979.

[9] N. Linial, A. Samorodnitsky, and A. Wigderson. A deterministic strongly polynomial algorithm for matrix scaling and approximate permanents. *Combinatorica*, 20(4):545–568, Apr 2000.

[10] S. Aaronson. A linear-optical proof that the permanent is #P-hard. *Proc. R. Soc. A*, 467(2136):3393–3405, 2011. arXiv: 1109.1674

[11] N. Anari, L. Gurvits, S. Oveis Gharan, and A. Saberi. Simply exponential approximation of the permanent of positive semidefinite matrices. *58th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 04 2017. arXiv: 1704.03486

[12] J.-Y. Cai, A. Pavan, and D. Sivakumar. On the hardness of permanent. In C. Meinel and S. Tison, editors, *STACS 99*, pages 90–99, Berlin, Heidelberg, 1999. Springer Berlin Heidelberg.

[13] A. I. Barvinok. Computing permanents of complex diagonally dominant matrices and tensors. arXiv: 1801.04191

[14] V. Patel and G. Regts. Deterministic polynomial-time approximation algorithms for partition functions and graph polynomials. *Electronic Notes in Discrete Mathematics*, 61:971 – 977, 2017. The European Conference on Combinatorics, Graph Theory and Applications (EUROCOMB'17). arXiv: 1607.01167

[15] J. Liu, A. Sinclair, and P. Srivastava. The ising partition function: zeros and deterministic approximation. arXiv:1704.06493.

[16] L. Eldar and S. Mehraban. Approximating the permanent of a random matrix with vanishing mean. arXiv:1711.09457

[17] L. Ahlfors. Complex Analysis. *McGraw-Hill*, 3 edition, 1979.

[18] G. Rempała and J. Wesołowski. Limiting behavior of random permanents. *Statistics & probability letters*, 45 (2):149–158, 1999 (Elsevier).

[19] A. Barvinok. Combinatorics and complexity of partition functions. *Algorithms and Combinatorics.* 30. Springer, Cham, 2016. vi+303 pp.

[20] A. Carbery and J. Wright. Distributional and ℓ^q-norm inequalities for polynomials over convex bodies in $\mathbf{R}^n$. *Math. Res. Lett.*, 8(3):233–248, 2001.