Sampling Matrices from Harish-Chandra–Itzykson–Zuber Densities with Applications to Quantum Inference and Differential Privacy

Jonathan Leake
TU Berlin
Germany

Colin McSwiggen
University of Tokyo
Japan

Nisheeth K. Vishnoi
Yale University
USA

ABSTRACT

Given two Hermitian matrices Y and Λ, the Harish-Chandra–Itzykson–Zuber (HCIZ) distribution is given by the density $e^{\mathrm{Tr}(U\Lambda U^*Y)}$ with respect to the Haar measure on the unitary group. Random unitary matrices distributed according to the HCIZ distribution are important in various settings in physics and random matrix theory, but the problem of sampling efficiently from this distribution has remained open. We present two algorithms to sample matrices from distributions that are close to the HCIZ distribution. The first produces samples that are ξ-close in total variation distance, and the number of arithmetic operations required depends on $\mathrm{poly}(\log 1/\xi)$. The second produces samples that are ξ-close in infinity divergence, but with a $\mathrm{poly}(1/\xi)$ dependence. Our results have the following applications: 1) an efficient algorithm to sample from complex versions of matrix Langevin distributions studied in statistics, 2) an efficient algorithm to sample from continuous maximum entropy distributions over unitary orbits, which in turn implies an efficient algorithm to sample a pure quantum state from the entropy-maximizing ensemble representing a given density matrix, and 3) an efficient algorithm for differentially private rank-k approximation that comes with improved utility bounds for $k > 1$.

CCS CONCEPTS

• **Theory of computation** → **Design and analysis of algorithms**; *Quantum information theory*; • **Mathematics of computing** → *Probability and statistics*; • **Security and privacy**;

KEYWORDS

sampling from matrix manifolds, unitary orbits, exponential density, density matrices, differential privacy, low-rank approximation

1 INTRODUCTION

Let U(n) denote the group of $n \times n$ unitary matrices and let μ denote the Haar probability measure on U(n). Given $n \times n$ Hermitian matrices Y and Λ, consider the following measure on U(n):

$$e^{\mathrm{Tr}(U\Lambda U^*Y)}d\mu(U). \tag{1}$$

The corresponding density is referred to as the Harish-Chandra–Itzykson–Zuber (HCIZ) density and has been extensively studied, implicitly and explicitly, in physics, random matrix theory, statistics, and theoretical computer science. A major result about the HCIZ density is that its integral over U(n) admits an exact expression as a determinant.

THEOREM 1.1 (HCIZ INTEGRAL FORMULA). *For $n \times n$ Hermitian matrices Y and Λ with distinct eigenvalues $y_1 > \cdots > y_n$ and $\lambda_1 > \cdots > \lambda_n$ respectively, we have the following:*[1]

$$\int_{\mathrm{U}(n)} e^{\mathrm{Tr}(U\Lambda U^*Y)}d\mu(U) = \left(\prod_{p=1}^{n-1} p!\right) \frac{\det([e^{y_i\lambda_j}]_{1\le i,j\le n})}{\prod_{i<j}(y_i-y_j)(\lambda_i-\lambda_j)}. \tag{2}$$

Theorem 1.1 was proved by Harish-Chandra [13] and by Itzykson and Zuber [14]. See the post by Terry Tao [35] and the notes of the second author [26] for more on the HCIZ integral.

Physics and Random Matrix Theory. Matrices distributed according to the HCIZ density are important in various settings in physics and random matrix theory. For instance, they appear in multi-matrix models in quantum field theory and string theory [9, 14], and they are also related to models of coupled Gaussian matrices [14] that have been used to solve the Ising model on a planar random lattice [4, 19]. In particular, the moments of HCIZ distributed unitary matrices play a role in computing correlation functions for matrix models of gauge theories and have been studied extensively since the 1990's [10, 11, 27, 32, 33].

The HCIZ integral also arises in many other places in random matrix theory. Notably, it occurs in expressions for the joint spectral densities of a number of matrix ensembles, such as Wishart matrices and off-center Wigner matrices [12]. However, the basic question about sampling from the HCIZ distribution has remained open.

The problem of sampling from the HCIZ distribution can be equivalently cast as the problem of sampling according to an exponential density specified by Y on the U(n)-orbit of Λ. Let

$$O_\Lambda := \{U\Lambda U^* \mid U \in \mathrm{U}(n)\}$$

---

[1] Although (2) assumes that all y_i and λ_i are distinct, when this is not the case one can still use Theorem 1.1 to obtain an exact determinantal formula for the HCIZ integral, simply by applying L'Hôpital's rule to the right-hand side of (2).

denote the orbit of Λ under the conjugation action of $\mathrm{U}(n)$. Every such orbit contains a diagonal element $\mathrm{diag}(\lambda)$ where λ is the sequence of eigenvalues of Λ listed in non-increasing order, and thus we always assume $\Lambda = \mathrm{diag}(\lambda)$. Further, one can write any $X \in O_\Lambda$ as $X = U\Lambda U^*$ for some $U \in \mathrm{U}(n)$, so that the density of the measure (1) can be rewritten as

$$e^{\langle Y, X \rangle} = e^{\mathrm{Tr}(U\Lambda U^* Y)},$$

where $\langle Y, X \rangle := \mathrm{Tr}(Y^*X)$. Thus, we arrive at the following problem.

PROBLEM 1.2 (SAMPLING FROM UNITARY ORBITS). *Given two $n \times n$ Hermitian matrices Y and $\Lambda = \mathrm{diag}(\lambda)$, sample an $X \in O_\Lambda$ from the probability distribution*

$$d\nu(X) \propto e^{\langle Y, X \rangle} d\mu_\Lambda(X), \tag{3}$$

where μ_Λ is the $\mathrm{U}(n)$-invariant probability measure on O_Λ.

Statistics. Distributions of the kind mentioned in (3) have also been studied under the name *matrix Langevin* or *matrix Bingham* in statistics [7, 8]. The difference is that these distributions are supported on orbits of the orthogonal group rather than the unitary group. Obtaining efficient algorithms to sample from such distributions was left as an open problem; see Section 2.5.2 of [8].

Continuous Maximum Entropy Distributions over Matrix Manifolds. In recent works [21, 22], distributions as in (3) arose as solutions to maximum entropy problems over manifolds, with applications to computing the entropy-maximizing representation of a quantum density matrix as an ensemble of "pure states". Concretely, the authors study the following problem: given a matrix A in the convex hull of O_Λ, compute the probability density supported on O_Λ whose expected value is A and that minimizes the Kullback–Leibler divergence to μ_Λ. As an example, if we let Λ be the diagonal matrix with exactly one 1 and rest 0s, the convex hull of O_Λ is exactly the set of PSD matrices with trace one – density matrices. Thus, in this case, the solution to the above entropy problem gives a way to "infer" an ensemble of pure states corresponding to a given density matrix A, following the principle of maximum entropy [1, 15, 16, 34]. The authors show that the solution to the above optimization problem gives rise to the distribution of the form $e^{\langle Y^\star, X \rangle} d\mu_\Lambda(X)$ for some $Y^\star$. Their main result is a polynomial-time algorithm to find this optimal solution that runs in time, roughly, the number of bits needed to represent A and the distance of A to the "boundary" of the convex hull of O_Λ. While [21, 22] gave polynomial-time algorithms to compute the optimal value $Y^\star$, designing an algorithm to sample from the corresponding distribution was left as an open problem.

Differentially Private Algorithms for Low-Rank Approximation. An important technique to obtain differentially private algorithms is the exponential mechanism due to [25]. In the context of rank-k approximation of a given matrix, it amounts to sampling from an exponential density of the type (3) on the orbit corresponding to rank-k projections [6, 18]. To the best of our knowledge, the only result on this problem is an approximate algorithm for the rank-1 case given by [18]. They left it as an open problem to simplify their rank-1 algorithm and also to come up with an algorithm to sample from the corresponding exponential mechanism for the rank-k case when $k > 1$.

Our Contributions. We present two efficient algorithms to approximately sample from HCIZ distributions or, equivalently, from exponential densities on unitary orbits. The two algorithms differ in how they approximate the target HCIZ distribution. Our first algorithm approximates in the total variation distance and is sufficient for many applications.

THEOREM 1.3 (MAIN RESULT – TOTAL VARIATION DISTANCE). *There is an algorithm that, given a $\xi > 0$ and $n \times n$ Hermitian matrices $\Lambda = \mathrm{diag}(\lambda)$ and $Y = \mathrm{diag}(y)$, outputs a matrix X that is distributed according to a distribution that is ξ-close in total variation distance to $d\nu(X) \propto e^{\langle Y, X \rangle} d\mu_\Lambda(X)$. The number of arithmetic operations required to run the algorithm is polynomial in $\log \frac{1}{\xi}$ and the number bits required to represent y and λ.*

Our second algorithm approximates the target distribution in the (stronger) infinity divergence (Definition 2.6) and is used for the differential privacy application.

THEOREM 1.4 (MAIN RESULT – INFINITY DIVERGENCE). *There is an algorithm that, given a $\xi > 0$ and $n \times n$ Hermitian matrices $\Lambda = \mathrm{diag}(\lambda)$ and $Y = \mathrm{diag}(y)$, outputs a matrix X that is distributed according to a distribution that is ξ-close in infinity divergence distance to $d\nu(X) \propto e^{\langle Y, X \rangle} d\mu_\Lambda(X)$. The number of arithmetic operations required to run the algorithm is polynomial in $\lambda_{\max} - \lambda_{\min}$, $y_{\max} - y_{\min}$, $\frac{1}{\xi}$, and the number bits required to represent y and λ.*

Note that while the approximation guarantee of the second algorithm (Theorem 1.4) is better, the number of arithmetic operations it performs depends polynomially on $\frac{1}{\xi}$ as opposed to polylogarithmically in $\frac{1}{\xi}$ as in the first algorithm (Theorem 1.3). We leave it as an open problem to give an algorithm which samples from a distribution that is ξ-close in infinity divergence, but with arithmetic operations depending polynomially on $\log \frac{1}{\xi}$.

Theorem 1.3 enables efficient numerical simulation of models in physics and random matrix theory where HCIZ densities arise. Our algorithms also make progress on the open problems mentioned earlier. In particular, Theorem 1.3 immediately gives efficient algorithms to 1) sample from complex matrix Langevin distributions [7, 8] and 2) sample from continuous maximum entropy distributions on unitary orbits studied by [21, 22], implying an efficient algorithm to sample a pure quantum state from the entropy-maximizing ensemble corresponding to a given density matrix. Moreover, Theorem 1.4 implies an efficient algorithm for the exponential mechanism for differentially private rank-k approximation [6, 18, 25]. As a consequence, we show that Theorem 1.4 allows us to obtain an efficient differentially private rank-k approximation with improved utility.

THEOREM 1.5 (DIFFERENTIALLY PRIVATE LOW-RANK APPROXIMATION). *There is a randomized algorithm that, given a $d \times d$ positive definite matrix A and its eigenvalues $\gamma_1 \geq \cdots \geq \gamma_d > 0$, an integer $1 \leq k \leq d$, an $\varepsilon > 0$, and a small $\delta > 0$, outputs a rank-k $d \times d$ Hermitian projection P such that $\mathbb{E}_P [\langle A, P \rangle] \geq (1 - \delta) \sum_{i=1}^{k} \gamma_i$ as long as $\sum_{i=1}^{k} \gamma_i \geq C \cdot \frac{dk}{\varepsilon \delta} \cdot \log \frac{1}{\delta}$, where $C > 0$ is a universal constant. This algorithm is ε-differentially private and requires a number of arithmetic operations which is polynomial in $\gamma_1 - \gamma_d$, $\frac{1}{\varepsilon}$, and the number of bits required to represent γ.*

Note that the utility bound promised by Theorem 1.5 is about $\frac{dk}{\varepsilon}$, compared to the utility bound of roughly $\frac{dk^3}{\varepsilon}$ due to [18] for $k > 1$.

The proofs of Theorems 1.3 and 1.4 are identical except for an intermediate step that we mention below. One of the key difficulties in sampling from an HCIZ distribution is that its domain, a unitary orbit, is a non-convex algebraic manifold. The individual entries of the desired sample matrix are highly correlated due to the algebraic constraints that define the orbit, which makes it difficult to break the problem into lower-dimensional subproblems. Our main technical contribution is to reduce the problem of sampling from an exponential density on a unitary orbit to sampling from an exponential density on a bounded convex polytope. In particular, we use an alternative parameterization of unitary orbits based on the *Rayleigh map*, which sends a Hermitian matrix X to a natural organization of the eigenvalues of all leading principal submatrices of X. The image of each U(n)-orbit under the Rayleigh map is a convex polytope called a *Gelfand–Tsetlin (GT) polytope*, which is cut out by linear inequalities given by the interlacing properties of the eigenvalues. This mapping reveals a recursive structure intrinsic to U(n)-orbits, which is hard to see directly in the ambient space of matrices. The Rayleigh map from a given U(n)-orbit to the corresponding GT polytope is not injective. However, one can show that 1) the HCIZ density on the orbit pushes forward to an exponential density on the polytope, and 2) the HCIZ density is constant on the fibers of the Rayleigh map. Therefore, to solve the sampling problem on the orbit, it suffices to sample a point from an exponential density on the GT polytope, and then sample a Hermitian matrix uniformly at random from the fiber of the Rayleigh map over that point. To sample from the GT polytope, we use results of [23] for Theorem 1.3 and results of [3] for Theorem 1.4.

We give a detailed technical overview of the algorithms and the proofs of Theorems 1.3 and 1.4 in Section 3. The formal algorithms and proofs appear in Sections 4, 5, and 6. Some proofs and details have been omitted from this version of the paper due to space constraints (see [20] for the full version).

2 RAYLEIGH TRIANGLES AND GELFAND–TSETLIN POLYTOPES

In this section we introduce some definitions and facts that we will need in what follows. In particular, we discuss two types of combinatorial objects that are fundamental to the geometry of Hermitian matrices: Rayleigh triangles and Gelfand–Tsetlin polytopes.

DEFINITION 2.1 (RAYLEIGH TRIANGLE). *For an integer $n \geq 1$, a Rayleigh triangle is a triangular array of real numbers $R = (R_{i,j})_{1 \leq i \leq j \leq n}$ satisfying the interlacing relations*

$$R_{i,j} \geq R_{i,j-1} \geq R_{i+1,j} \qquad \text{for all } 1 \leq i < j \leq n. \tag{4}$$

The vector $R_{\bullet,j} = (R_{1,j}, \ldots, R_{j,j}) \in \mathbb{R}^j$ is called the jth row, and $R_{\bullet,n} \in \mathbb{R}^n$ is called the **top** *row. If we fix $R_{\bullet,n}$, we can regard the numbers $R_{i,j}, j \leq n-1$ as coordinates of a point in $\mathbb{R}^{n(n-1)/2}$.*

Note that the indexing for the Rayleigh triangle is different from that of matrix notation: the top row is indexed by n.

DEFINITION 2.2 (GELFAND–TSETLIN POLYTOPE). *Given a vector $\lambda \in \mathbb{R}^n$ with $\lambda_1 \geq \cdots \geq \lambda_n$, the Gelfand–Tsetlin polytope $GT(\lambda)$ is* the convex polytope in $\mathbb{R}^{n(n-1)/2}$ consisting of all Rayleigh triangles with top row equal to λ.

Thus, $GT(\lambda)$ is the polytope cut out by the interlacing inequalities (6), with $R_{\bullet,n} = \lambda$ fixed. In other words, the following system of n equalities and $n \times (n-1)$ inequalities determines $GT(\lambda)$:

$$R_{i,n} = \lambda_i \text{ for all } 1 \leq i \leq n, \tag{5}$$

$$R_{i,j} - R_{i,j-1} \geq 0 \text{ and } R_{i,j-1} - R_{i+1,j} \geq 0 \text{ for all } 1 \leq i < j \leq n. \tag{6}$$

Note that if all of the entries of λ are distinct, then $GT(\lambda)$ is full-dimensional in $\mathbb{R}^{n(n-1)/2}$ (with coordinates $R_{i,j}$ for $1 \leq i \leq j \leq n-1$), and every inequality given by (6) above is essential. However if entries of λ coincide, then some of the inequalities of (6) become equalities, and $GT(\lambda)$ lies in some affine subspace of $\mathbb{R}^{n(n-1)/2}$. In particular, if $\lambda_p = \lambda_{p+1} = \cdots = \lambda_q$ then for any $R \in GT(\lambda)$ we have that $R_{i,j} = \lambda_p$ for all i,j such that $p \leq i \leq q + j - n$. On the other hand, every inequality of (6) not associated to such a fixed entry $R_{i,j}$ is essential. Using this observation, for any fixed λ it is straightforward to determine the affine subspace in which $GT(\lambda)$ has non-empty interior.

The following is then a corollary of a classical result of linear algebra known as the Cauchy–Rayleigh interlacing theorem.

PROPOSITION 2.1 (INTERLACING AND RAYLEIGH TRIANGLES). *Given an $n \times n$ Hermitian matrix X, denote by $X[k]$ its kth leading principal submatrix (that is, the $k \times k$ submatrix in the upper left corner of X). Let $\lambda_{1,k} \geq \cdots \geq \lambda_{k,k}$ be the eigenvalues (which are real) of $X[k]$. Then the eigenvalues $(\lambda_{j,k})_{1 \leq j \leq k \leq n}$ of the leading submatrices of X form a Rayleigh triangle, which we write as $\mathcal{R}(X)$.*

We now need a few definitions.

DEFINITION 2.3 (TYPE VECTOR). *The type vector of R is defined by*

$$\text{type}(R) = \left(R_{1,1}, R_{1,2} + R_{2,2} - R_{1,1}, \ldots, \sum_{i=1}^{n} R_{i,n} - \sum_{j=1}^{n-1} R_{j,n-1} \right).$$

If $R = \mathcal{R}(X)$ for some Hermitian X, then $\text{type}(R) = (X_{11}, \ldots, X_{nn})$ is the diagonal of X.

DEFINITION 2.4 (ORBITS OF U(n)). *Given a vector $\lambda \in \mathbb{R}^n$ as above, write $\Lambda = \text{diag}(\lambda)$ and let $O_\Lambda = \{U \Lambda U^* \mid U \in U(n)\}$ be the unitary conjugation orbit of Λ. Let μ_Λ be the uniform probability measure on O_Λ, i.e., the unique probability measure on O_Λ that is invariant under the conjugation action of U(n).*

It can be shown that the image $\mathcal{R}(O_\Lambda)$ is $GT(\lambda)$. In fact, the following stronger result is true: the uniform measure on O_Λ maps to the uniform measure on $GT(\lambda)$; see e.g. [2, 29, 30, 38].

PROPOSITION 2.2 (PUSHFORWARD OF HAAR MEASURE). *The pushforward of the Haar measure μ_Λ on O_Λ by the map $\mathcal{R}$ is the uniform probability measure on $GT(\lambda)$.*

Note that in the above result, the pushforward distribution is a restriction of the Lebesgue measure on the *affine span* of $GT(\lambda)$, which is the minimal affine subspace of $\mathbb{R}^{n(n-1)/2}$ containing $GT(\lambda)$. This distinction only matters when not all λ_i are distinct, since in this case $GT(\lambda)$ has dimension less than $n(n-1)/2$ so that its volume in the ambient space $\mathbb{R}^{n(n-1)/2}$ is zero.

Proposition 2.2 allows us to prove the following crucial fact that the image of an exponential density on a unitary orbit is an exponential density on the GT polytope.

THEOREM 2.3 (PUSHFORWARD OF THE HCIZ DENSITY). *If* $Y = \text{diag}(y)$ *for some real vector* y, *then the pushforward of the measure* $e^{\langle Y, X \rangle} d\mu_\Lambda(X)$ *through the map* $\mathcal{R}$ *is*

$$\mathcal{R}_* \left[e^{\langle Y, X \rangle} d\mu_\Lambda(X) \right] = \text{Vol}(GT(\lambda))^{-1} e^{\langle y, \text{type}(P) \rangle} dP,$$

where dP *denotes the Lebesgue measure on the affine span of* $GT(\lambda)$.

PROOF. First note that $(X_{11}, \ldots, X_{nn}) = \text{type}(\mathcal{R}(X))$, as mentioned in Definition 2.3. Thus, we have

$$\langle Y, X \rangle = \sum_{i=1}^{n} y_i X_{ii} = \langle y, \text{type}(\mathcal{R}(X)) \rangle.$$

The result then follows from the fact that the pushforward of μ_Λ through $\mathcal{R}$ is uniform on $GT(\lambda)$ by Proposition 2.2. ∎

Finally, to prove the correctness of our sampling algorithm, we will need to describe the set of Hermitian matrices that map to a given Rayleigh triangle under $\mathcal{R}$.

DEFINITION 2.5 (FIBER OVER A RAYLEIGH TRIANGLE). *Given* $R \in GT(\lambda)$, *the fiber of the map* $\mathcal{R}$ *over* R *is the set* $\mathcal{R}^{-1}(R) = \{X \in O_\Lambda \mid \mathcal{R}(X) = R\}$.

The fiber $\mathcal{R}^{-1}(R)$ is a compact subset of O_Λ. The *uniform probability measure* on $\mathcal{R}^{-1}(R)$ is characterized by the property that if X is uniformly distributed in $\mathcal{R}^{-1}(R)$, then for $1 < k \leq n$, $X[k]$ is uniformly distributed on the compact manifold $\mathcal{H}(X[k-1]; R_{\bullet, k})$ of $k \times k$ Hermitian matrices with eigenvalues $R_{\bullet, k}$ and leading $(k-1) \times (k-1)$ submatrix equal to $X[k-1]$. We will show below in Lemma 4.10 that $\mathcal{H}(X[k-1]; R_{\bullet, k})$ is a product of spheres.

Before moving on, we formally define the notions of distance between distributions which are relevant to our main results.

DEFINITION 2.6 (NOTIONS OF DISTANCE BETWEEN DISTRIBUTIONS). *Given two distributions (i.e., Borel probability measures)* μ, ν *on* $S \subseteq \mathbb{R}^n$, *we define the* total variation distance *between* μ *and* ν *as* $\|\mu - \nu\|_{\text{TV}} := \sup_{S \subset \mathcal{S}} |\mu(S) - \nu(S)|$. *When* μ, ν *have continuous density functions* f, g *respectively (with respect to the same base measure) on* $\mathcal{S}$, *we further define the* infinity divergence *between* μ *and* ν *as* $D_\infty(\mu \| \nu) := \log \sup_{x \in \mathcal{S}} \frac{f(x)}{g(x)}$.

3 TECHNICAL OVERVIEW

In this section, we give an overview of the algorithms and the proof of our main results (Theorems 1.3 and 1.4), leaving the full details to Sections 4 and 5. Here we will emphasize the important ideas and and concepts from the various parts of the proof without going into too much detail. For the interested reader, we will provide links to other relevant sections of the full proof throughout this overview.

Throughout, Λ and Y will always be $n \times n$ real diagonal matrices. The unitary orbit O_Λ is defined as the set of all matrices $U\Lambda U^*$ obtained by conjugating Λ by any unitary matrix. The measure $d\mu_\Lambda(X)$ is the unitarily-invariant probability measure on O_Λ, which means that

$$\int_{O_\Lambda} f(X) \, d\mu_\Lambda(X) = \int_{O_\Lambda} f(UXU^*) \, d\mu_\Lambda(X)$$

for all integrable functions f and all unitary matrices $U \in U(n)$. The goal of our algorithms is to return a sample X from the unitary orbit O_Λ, such that the distribution of X is proportional to $e^{\langle Y, X \rangle} d\mu_\Lambda(X)$.

3.1 The Uniform Case

Let us first consider a simple case, when $Y = 0$. In this case the distribution we want to sample from is precisely the unitarily invariant (uniform) distribution on O_Λ. Unitary invariance implies that sampling X from this distribution on O_Λ is equivalent to sampling U from the Haar probability measure on $U(n)$ and taking $U\Lambda U^*$ as our sample in O_Λ. Sampling U from the Haar probability measure on $U(n)$ then has a classical solution: Inductively sample orthogonal unit vectors $u_1, u_2, \ldots, u_n$ from $\mathbb{C}^n$ by projecting and normalizing random Gaussian vectors. We can then construct a random matrix $U \in U(n)$ by setting $u_1, u_2, \ldots, u_n$ as the columns of U. This shows that sampling from O_Λ in the case of $Y = 0$ has a simple, intuitive solution.

Difficulty in Extending the Algorithm for the Uniform Case. For general Y however, the situation quickly becomes more complicated. The first observation is that unitary invariance is immediately lost, since generically we have

$$e^{\langle Y, X \rangle} \neq e^{\langle Y, UXU^* \rangle}.$$

This means the method used for $Y = 0$ breaks down, as there is no clear way to generalize the above simple algorithm to exponential weightings of the Haar measure. This is even true in the most basic case when O_Λ is the set of rank-one projections (when $\Lambda = \text{diag}(1, 0, \ldots, 0)$), and the difficulty in this case was already realized in several previous works [8, 18, 22].

Even though the density is not unitarily invariant, there is still significant symmetry coming from the structure of the orbit O_Λ. This symmetry leads to the HCIZ integral formula (Theorem 1.1), which gives an efficiently computable formula for the partition function of the HCIZ density. (It should be noted that the proof of this formula is highly non-trivial: Harish-Chandra's original proof from 1957 can be viewed as a starting point for much of the modern theory of quantum integrable systems [13, 26].) Typically, such an explicit formula for the partition function can be translated into an algorithm for sampling, but it is not clear how to do this for the unitary orbit O_Λ.

3.2 Searching for Self-Reducibility

In the world of discrete distributions, the seminal work of [17] gives a general way to sample from a distribution using an oracle for the associated partition function. The key property needed to utilize their results is that the distribution needs to be *self-reducible*. A problem is said to be self-reducible if, roughly speaking, a problem instance with input size n can be efficiently reduced to another instance of the same problem with input size $n - 1$.

As an example of where the ability to compute the partition function can lead to an efficient sampling algorithm, consider the case of sampling matchings from a graph. To uniformly sample a matching, one can first choose an edge e in the graph and then compute the number k_e of matchings that contain e, and the number l_e of matchings that do not contain e. The edge e is then included in the output matching with probability k_e/l_e, and the original problem can be reduced to finding a perfect matching in the smaller graph obtained by removing the vertices joined by e. To sample non-uniform matchings, the values of k_e and l_e are replaced by evaluations of the partition function.

In our world of continuous distributions on unitary orbits, it is not obvious how to perform a self-reduction similar to that of the discrete world, even though we have a formula for the partition function. The obstacle is that self-reducibility depends on preserving the original problem structure: we must reduce to an instance of the same problem, but with smaller input.

One approach towards this is to iteratively sample the individual entries or columns of the matrix, and then to interpret the remaining entries of the matrix as a smaller instance of the original problem conditioned on the previously selected entries. The issue with this approach is that the entries of a matrix X in the unitary orbit O_Λ are highly correlated due to the algebraic constraint that $X = U\Lambda U^*$ for some $UU^* = I$. This means that the problem of sampling from a given distribution on the orbit conditional on one or more matrix entries is a priori very different from the original problem, and much more complicated.

There is an alternative way to view a matrix $X \in O_\Lambda$: in terms of its eigenvalues. The eigenvalues and eigenvectors together determine the matrix X completely. Further, one can (almost) recover the eigenvectors of X using the eigenvalues of the principal submatrices of X (see [36]). When eigenvalues are distinct, one finds

$$|v_{i,j}|^2 = \frac{\prod_{k=1}^{n-1}(\lambda_i(X) - \lambda_k(X_j))}{\prod_{k \neq i}(\lambda_i(X) - \lambda_k(X))},$$

where $\lambda_i(X)$ is the ith largest eigenvalue of X, X_k is the principal submatrix of X with the kth row and column removed, and v_i is the eigenvector corresponding to $\lambda_i(X)$. That is, with the extra information of the eigenvalues of the principal submatrices of X, one can determine the eigenvectors of X up to the (complex) sign of the entries.

This is a good sign for us, as it hints at some inductive structure in the eigenvalues of X. Can we now understand this relationship between the eigenvectors and eigenvalues of principal submatrices in some recursive manner? As a matter of fact, by considering the matrix X in terms of all of its *leading* principal submatrices, we are able to prove a similar result (see Section 4.4). And not only that, but it turns out that this eigenvector information is sufficient for our purposes.

Self-Reducibility in the Space of Eigenvalues. This suggests a natural self-reducible structure for the unitary orbit O_Λ via the principal submatrices. The *Rayleigh map* $\mathcal{R}$ (Definition 2.1) maps a matrix $X \in O_\Lambda$ to the length-$\binom{n+1}{2}$ vector of the eigenvalues of all the leading principal minors of X. These eigenvalues are organized in the form of a triangle called the *Rayleigh triangle*, denoted $\mathcal{R}(X) = (R_{i,j})_{1 \leq i \leq j \leq n}$ where $R_{i,j}$ is the ith largest eigenvalue of the top left $j \times j$ principal submatrix. (Note that $R_{\bullet,n}$ are the eigenvalues of Λ, and counter to matrix indexing intuition, we refer to $R_{\bullet,n}$ is the *top row* of the Rayleigh triangle.) Organizing eigenvalues into a triangle like this then makes the self-reducible structure clear: fixing the top $n - k + 1$ rows of the triangle, $R_{\bullet,n}, \ldots, R_{\bullet,k}$, and leaving the bottom k rows free gives a lower-dimensional space of Rayleigh triangles that corresponds precisely to the $U(k)$-orbit of the $k \times k$ matrix $\text{diag}(R_{\bullet,k})$.

We now have a self-reducible way to view the elements $X \in O_\Lambda$ in terms of their eigenvalues, but how does this help us to sample from O_Λ? By the Cauchy–Rayleigh interlacing theorem, for all

$X \in O_\Lambda$ the Rayleigh triangle $\mathcal{R}(X)$ is a element of a polytope in $\mathbb{R}^{\binom{n+1}{2}}$ cut out by the inequalities

$$R_{i,j} \geq R_{i,j-1} \geq R_{i+1,j} \quad \text{for all valid } i, j.$$

In fact the converse is also true: the image $\mathcal{R}(O_\Lambda)$ is the whole polytope cut out by these inequalities, called the *Gelfand–Tsetlin (GT) polytope* and denoted $GT(\lambda)$ where λ is the vector of eigenvalues of Λ. What is special about the Rayleigh map $\mathcal{R}$ is then that it projects the uniform measure $d\mu_\Lambda$ on O_Λ to the Lebesgue (uniform) measure on the GT polytope. This leaves a few questions.

(1) How does the Rayleigh map $\mathcal{R}$ project the HCIZ density from the unitary orbit to $GT(\lambda)$?
(2) How do we sample from the corresponding distribution on the GT polytope?
(3) How do we transfer that sample back to the unitary orbit?

The answers here are reasonable: for (1) the Rayleigh map projects the exponential HCIZ density to an exponential density on the GT polytope (see Theorem 2.3), for (2) we can use powerful tools ([23] and [3]) to sample from this exponential density on a polytope cut out by polynomially many inequalities, and for (3) we have algorithms that utilize the symmetry of the orbit O_Λ and the HCIZ density (see Section 4.1). For the understanding of the reader, we first demonstrate this explicitly in the case of rank-one projections.

3.3 The Case of Rank-One Projections

Let us now look at the simplest choice of Λ: the case where Λ is the diagonal matrix with entries $1, 0, 0, \ldots, 0$. This means that O_Λ is the set of Hermitian positive semidefinite (PSD) rank-one projections. In this case, each leading principal submatrix of a given $X \in O_\Lambda$ has at most one non-zero eigenvalue. Thus, the entries of the Rayleigh triangle of X are all zero except for $R_{1,j}$, for which we have

$$1 = R_{1,n} \geq R_{1,n-1} \geq \cdots \geq R_{1,1} \geq 0.$$

This means that the GT polytope in this case is isomorphic to a simplex by considering the values of $R_{1,n} - R_{1,n-1}, \ldots, R_{1,2} - R_{1,1}$, and $R_{1,1}$, which sum to 1. Since the principal submatrices are all rank at most 1, these differences are the differences of the traces of the submatrices of X, which are precisely equal to the diagonal entries of X. Hence the map $\text{diag}(X)$, which picks out the diagonal entries of X, is equivalent to the Rayleigh map $\mathcal{R}$ in this case, and the image $\text{diag}(O_\Lambda)$ is the standard simplex Δ_n. Therefore $\text{diag}(X)$ maps the uniform measure $\mu_\Lambda(X)$ to the Lebesgue (uniform) measure on Δ_n, and we can determine the measure on the GT polytope corresponding to the HCIZ density via

$$e^{\langle Y, X \rangle} d\mu_\Lambda(X) = e^{\langle y, \text{diag}(X) \rangle} d\mu_\Lambda(X) \xrightarrow{\text{diag}} e^{\langle y, x \rangle} dx,$$

where y is the vector of diagonal entries of Y (which is itself a diagonal matrix). This suggests a method for sampling from our distribution $e^{\langle Y, X \rangle} d\mu_\Lambda(X)$ on O_Λ when Y is diagonal:

(1) Sample x from Δ_n according to the distribution $e^{\langle y, x \rangle} dx$ where $y = \text{diag}(Y)$.
(2) Convert x into a rank-one PSD projection $X \in O_\Lambda$.

This sampling problem is a special case of sampling from a log-concave density on a convex polytope, and there is a significant body of work which is geared towards coming up with algorithms for this general problem. In particular, to obtain a algorithm which

gives the TV distance bound promised by Theorem 1.3, we can appeal to Corollary 1.2 of [23]. To obtain an algorithm which gives the infinity divergence bound promised by Theorem 1.4, we appeal to Lemma 6.5 of the arXiv version of [3]. Since our function is log-linear, to use these results we just need to establish bounds on the outer and inner radius of the polytope (the simplex in this case) and the Lipschitz constant of the exponent. This is trivial in the case of the simplex, but it is also straightforward for the case of a general GT polytope; see Section 4.3.

However, we still need to convert x into a rank-one PSD projection $X \in O_\Lambda$ in a way which is compatible with our exponentially weighted distribution on O_Λ. Towards this, we first observe that the set of all X for which $\operatorname{diag}(X) = x$ is given by

$$\operatorname{diag}^{-1}(x) = \{X \in O_\Lambda : X = vv^*, \ v^* = (e^{-i\theta_1}\sqrt{x_1}, \ldots, e^{-i\theta_n}\sqrt{x_n})\}.$$

Second, for diagonal Y the density $e^{\langle Y, X \rangle}$ does not depend on the choice of $\theta_1, \ldots, \theta_n$. Therefore if we restrict our distribution on O_Λ to the subset $\operatorname{diag}^{-1}(x)$, we obtain the uniform distribution. This means we can convert x into a rank-one PSD projection by uniformly randomly sampling $e^{i\theta_1}, \ldots, e^{i\theta_n}$ independently from the unit circle and setting $X := vv^*$ where

$$v^* = (e^{-i\theta_1}\sqrt{x_1}, \ldots, e^{-i\theta_n}\sqrt{x_n}).$$

Combining these two steps—sampling from the simplex and then transferring that sample to O_Λ—gives us an algorithm for sampling from O_Λ according to our exponentially weighted density.

What remains is then to show that our bounds between the sampled and target distributions on the simplex (either TV distance or infinity divergence) transfer back to the respective distributions on O_Λ. In this case, the algorithm to transfer a sample from the simplex to a sample from O_Λ is very simple and explicit, as demonstrated above. In particular, given a point x of the simplex, we can sample *exactly* from the target (uniform) distribution on the associated fiber of x in O_Λ. This means that transferring from the simplex to O_Λ accumulates no extra error between the sampled and target distributions. Therefore the TV distance and infinity divergence bounds between the sampled and target distributions on O_Λ are *exactly* equal to the bounds achieved on the simplex Δ_n (see Appendix A of the arXiv version [20] for more details). Thus we have achieved our desired error bounds for the sampled distribution on O_Λ, completing the proof of the main results in the rank-one case.

Obstacles to Extending to General Λ. Unfortunately, extending this algorithm beyond the rank-one case immediately runs into issues. First, we need to know how the HCIZ density on a general orbit O_Λ transfers to the GT polytope through the Rayleigh map. In the case of rank-one projections, we obtained a log-linear density which was crucial to our sampling error bounds, and we need to be able to emulate this in the general case.

Beyond this, converting a sample from the GT polytope back to the unitary orbit is now more complicated. In the case of rank-one projections, determining the fiber $\operatorname{diag}^{-1}(x) \cong \mathcal{R}^{-1}(x)$ was straightforward and led to a simple method for sampling an element of the orbit O_Λ. For general Λ, the fiber $\mathcal{R}^{-1}(x)$ does not have such a clear description. We will need to study further the relationship between the unitary orbit O_Λ and the corresponding GT polytope

to understand how to generalize the sampling technique used for rank-one projections.

3.4 Moving to the GT Polytope in the Case of General Λ

In the case of general Λ, we can utilize the same overarching algorithm that was used in the rank-one case: sample from the GT polytope, and then transfer back to the unitary orbit. The first question we need to answer is what the distribution on the GT polytope should look like. We know that the Rayleigh map transfers the uniform distribution $d\mu_\Lambda(X)$ on the unitary orbit to the uniform distribution on the polytope, but what about the distribution $e^{\langle Y, X \rangle} d\mu_\Lambda(X)$?

Since Y is diagonal we can write

$$e^{\langle Y, X \rangle} d\mu_\Lambda(X) = e^{\langle y, \operatorname{diag}(X) \rangle} d\mu_\Lambda(X),$$

where $y = \operatorname{diag}(Y)$. In the rank-one case, the Rayleigh map $\mathcal{R}$ was equivalent to the $\operatorname{diag}(X)$ map, and this meant that the projected measure was given by $e^{\langle y, x \rangle} dx$. To handle the general case, we need a map which takes a Rayleigh triangle $R \in GT(\lambda)$ to the diagonal vector of the corresponding $X \in O_\Lambda$. This is precisely the $\operatorname{type}(R)$ map (Definition 2.3), which computes differences of the traces of the successive principal submatrices:

$$\operatorname{type}(R) = \left(R_{1,1}, \ R_{1,2} + R_{2,2} - R_{1,1}, \ \ldots, \ \sum_{i=1}^{n} R_{i,n} - \sum_{j=1}^{n-1} R_{j,n-1}\right).$$

This definition implies $\operatorname{type}(R) = \operatorname{diag}(X)$ whenever $R = \mathcal{R}(X)$. With this, we can more precisely state our sampling algorithm at a high level.

(1) Sample a Rayleigh triangle $R = (R_{i,j})$ from the associated GT polytope according to the distribution $e^{\langle y, \operatorname{type}(R) \rangle} dR$, where $y = \operatorname{diag}(Y)$.

(2) Convert R into an element $X \in O_\Lambda$ of the unitary orbit.

As in the simplex case, we can use the powerful tools of [23] and [3] to sample from the log-linear density on the GT polytope (see Section 4.3 for more details). However, we still must convert this into a sample from the HCIZ density on the orbit O_Λ.

3.5 From the GT Polytope back to the Unitary Orbit

Supposing we have a sample R distributed according to the log-linear density on the GT polytope, the final step is to convert R into a sample from the unitary orbit O_Λ. In the case of the simplex, this was easy because it is easy to describe the fiber $\operatorname{diag}^{-1}(x)$ as well as the uniform distribution on this fiber. In the case of the GT polytope and the Rayleigh map however, determining $\mathcal{R}^{-1}(R)$ and the associated distribution is more complicated.

Fortunately though, we can break the problem down into more manageable pieces corresponding to the row-by-row self-reducible structure discussed above. Observe that for any k we can identify $U(k-1)$ with the subgroup of $U(k)$ consisting of unitary matrices that have a 1 in the bottom right corner and zeros in all other positions of the last row and column. Inducting on this observation, we obtain inclusions

$$U(1) \hookrightarrow U(2) \hookrightarrow \cdots \hookrightarrow U(n-1) \hookrightarrow U(n),$$

and these inclusions correspond precisely to sub-triangles of our sampled Rayleigh triangle R (the bottom $1, 2, \ldots, n-1, n$ rows of the triangle respectively).

This allows us to induct on the rank of the unitary group. Assuming that we have an $(n-1) \times (n-1)$ matrix sample X_0 from the $U(n-1)$ orbit associated to the bottom $n-1$ rows of R, we just need to sample an $X \in O_\Lambda$ which has X_0 as its top-left principal submatrix. That is, given such an X_0, we need to sample some

$$X = \begin{bmatrix} X_0 & v \\ v^* & c \end{bmatrix} \in O_\Lambda.$$

To sample such an X, first note that the top row of R (that is, the eigenvalues of Λ) determines the trace of X, which specifies deterministically the value of c. This leads to a crucial observation: all possible values of the matrix X have the same diagonal entries, and hence our density function $e^{\langle Y, X \rangle}$ is constant since Y is a diagonal matrix. This means that we may sample X *uniformly* from the set of all X with the above block form.

Having made this observation, we now describe in detail how to sample such a matrix X (see Section 4.4).

The Case of Distinct Eigenvalues. We first demonstrate how to do this in a simplified case: when X_0 is a diagonal matrix with distinct eigenvalues. In this case, we make the following easy observation for $U \in U(n)$ where v, w are any vectors:

$$U \begin{bmatrix} X_0 & v \\ v^* & c \end{bmatrix} U^* = \begin{bmatrix} X_0 & w \\ w^* & c \end{bmatrix} \iff U = \mathrm{diag}(e^{i\theta_1}, \ldots, e^{i\theta_n}).$$

This immediately gives rise to an algorithm for sampling X of the above block form:

(1) Construct *any* $X \in O_\Lambda$.
(2) Sample $e^{i\theta_1}, \ldots, e^{i\theta_n}$ uniformly and independently from the unit circle.
(3) Defining $U := \mathrm{diag}(e^{i\theta_1}, \ldots, e^{i\theta_n})$, our sample is UXU^*.

What remains to be done then is to construct some $X \in O_\Lambda$, which is equivalent to constructing a valid value of v. To this end, we use the special form of X to write down its characteristic polynomial. Letting $X_0 = \mathrm{diag}(R_{1,n-1}, \ldots, R_{n-1,n-1})$, we want to choose v such that

$$\prod_{i=1}^{n}(t - \lambda_i) = \det\left(tI - \begin{bmatrix} X_0 & v \\ v^* & c \end{bmatrix}\right)$$

$$= (t - c)\prod_{i=1}^{n-1}(t - R_{i,n-1}) + \sum_{i=1}^{n-1} |v_i|^2 \prod_{j \neq i}(t - R_{j,n-1}).$$

Since the values of $R_{\bullet, n-1}$ are distinct, we obtain $n-1$ equations by plugging in $t = R_{k,n-1}$ for each value of $k \in \{1, \ldots, n-1\}$:

$$\prod_{i=1}^{n}(R_{k,n-1} - \lambda_i) = |v_k|^2 \prod_{i \neq k}(R_{k,n-1} - R_{i,n-1}).$$

This then implies $|v_k|^2 = \frac{\prod_{i=1}^{n}(R_{k,n-1} - \lambda_i)}{\prod_{i \neq k}(R_{k,n-1} - R_{i,n-1})}$, which gives us a formula for a choice of v_k so long as this ratio of products is non-negative. In fact, it is always non-negative because the values of $\lambda_\bullet$ and $R_{\bullet, n-1}$ are interlacing by the Cauchy–Rayleigh theorem. By choosing $v_k \geq 0$ which satisfy the above equalities, we have constructed a valid X from our orbit, and applying the above algorithm gives the desired sample of O_Λ.

The General Case. Handling the cases of non-distinct eigenvalues and non-diagonal X_0 is then straightforward. First, if X_0 is diagonal with non-distinct ordered eigenvalues, the set of unitary matrices which preserves the block form of X becomes larger:

distinct: $UXU^* \in O_\Lambda \iff U \in U(1) \times U(1) \times \cdots \times U(1)$

non-distinct: $UXU^* \in O_\Lambda \iff U \in U(m_1) \times \cdots \times U(m_p) \times U(1),$

where $m_1, m_2, \ldots, m_p$ are the multiplicities of the eigenvalues of X_0. That is, we need to replace step 2 of the above algorithm with

(2) Sample unitary matrices $U_1, \ldots, U_p, U_{p+1}$ uniformly from $U(m_1), \ldots, U(m_p), U(1)$ respectively.

Algorithms to sample uniformly from unitary groups are well-known and were discussed above.

Finally, handling the non-diagonal case is even easier. Letting $U_0 \in U(n-1)$ be such that $U_0 X_0 U_0^* = D_0$ is diagonal, we reduce to the previous cases by considering

$$\begin{bmatrix} U_0 & 0 \\ 0 & 1 \end{bmatrix} X \begin{bmatrix} U_0^* & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} U_0 X_0 U_0^* & U_0 v \\ (U_0 v)^* & c \end{bmatrix} = \begin{bmatrix} D_0 & U_0 v \\ (U_0 v)^* & c \end{bmatrix}.$$

We then first sample a matrix X' by applying the above algorithm to the right-hand side matrix above, since D_0 is diagonal. We then obtain our desired sample via the inverse conjugation by U_0:

$$X = \begin{bmatrix} U_0^* & 0 \\ 0 & 1 \end{bmatrix} X' \begin{bmatrix} U_0 & 0 \\ 0 & 1 \end{bmatrix}.$$

Combining all of this then yields an algorithm which constructs a matrix X in the unitary orbit O_Λ from the given Rayleigh triangle R in the GT polytope.

Sampling Error Bounds. The last thing we must do is show that our bounds between the sampled and target distributions on the GT polytope (either TV distance or infinity divergence) transfer back to the respective distributions on O_Λ. In the case of the simplex, the exact bounds transferred from the simplex to the orbit because we were able to exactly sample from the fibers $\mathcal{R}^{-1}(x)$ for any x in the simplex. Specifically, sampling from the fiber boiled down to sampling uniformly from a torus.

In the general case, we saw above that this torus sampling in the case of the simplex is replaced by an inductive sampling of unitary matrices from the uniform (Haar) distribution. There is a simple exact algorithm for sampling Haar-distributed unitary matrices, as discussed in Section 3.1. Therefore the same argument applies to the general case as applied to the case of rank-one projections and the simplex. Specifically, the bounds we achieve between the sampled and target distributions on the GT polytope transfer back exactly to the respective distributions on the orbit O_Λ.

This concludes the overview of the proofs of Theorems 1.3 and 1.4. To summarize, we first considered the pushforward measure of the HCIZ distribution on a unitary orbit O_Λ through the Rayleigh map $\mathcal{R}$ onto the GT polytope $GT(\lambda)$. This gave a density function on the GT polytope that was a log-linear function of the type vector. We then sampled a Rayleigh triangle $R \in GT(\lambda)$ according to this density using very general techniques for sampling from log-linear distributions on convex polytopes. Finally, we converted this sample $R \in GT(\lambda)$ into a sample from the orbit O_Λ by inductively sampling $k \times k$ matrices according to the bottom k rows of R. We refer the reader to Sections 4 and 5 for the remaining details of the proof.

4 THE SAMPLING ALGORITHMS OF THEOREMS 1.3 AND 1.4

In this section, we describe the main steps of the algorithms claimed in Theorems 1.3 and 1.4. We then prove that the steps produce the correct output and determine the number of arithmetic operations they require. We will mostly treat the two algorithms together, since they differ only in the procedure used to sample from the Gelfand–Tsetlin polytope.

REMARK 4.1. *Throughout we assume that we can exactly unitarily diagonalize Hermitian matrices for convenience. That said, the algorithms given in [31] approximate the eigenvalues and eigenvectors of a Hermitian $n \times n$ matrix H within relative error 2^{-L_H} in a number of arithmetic operations which is polynomial in n and $\log L_H$, where L_H is the number of bits required to represent H. To use this result, there is some extra error accounting which is required. In our case this can be handled, and we omit the details.*

Further, under this assumption we may also assume that the matrix Y which appears in the exponent of our density function $e^{\langle Y, X \rangle}$ is not diagonal but just Hermitian with eigenvalues $y_1, \ldots, y_n$. Indeed if $Y = U \cdot \mathrm{diag}(y) \cdot U^*$, then the fact that $\mu_\Lambda(X)$ is unitarily invariant implies $e^{\langle Y, X \rangle} d\mu_\Lambda(X) = e^{\langle \mathrm{diag}(y), U^* X U \rangle} d\mu_\Lambda(X) = e^{\langle \mathrm{diag}(y), X \rangle} d\mu_\Lambda(X)$. After sampling X according to this distribution on O_Λ, we then simply conjugate X by U to obtain a sample from the original target distribution on O_Λ.

4.1 Description of the Algorithms

Formally, the input and output of the algorithms are as follows.

- **Input:**
  (1) A vector $\lambda = (\lambda_1, \ldots, \lambda_n) \in \mathbb{R}^n$, with $\lambda_1 \geq \cdots \geq \lambda_n$.
  (2) A vector $y = (y_1, \ldots, y_n) \in \mathbb{R}^n$, with $y_1 \geq \cdots \geq y_n$.
  Write $\Lambda = \mathrm{diag}(\lambda)$, $Y = \mathrm{diag}(y)$.
- **Output:** An $n \times n$ Hermitian matrix with eigenvalues λ, distributed according to $d\nu(X) \propto e^{\langle Y, X \rangle} d\mu_\Lambda(X)$ on O_Λ.

At a high level, both algorithms then consist of the following steps.

(1) **Reduce sampling from O_Λ to sampling from $GT(\lambda)$.** Construct a membership oracle for $GT(\lambda)$ and an evaluation oracle for the correct exponential density on $GT(\lambda)$.

(2) **Sample a Rayleigh triangle from $GT(\lambda)$.** Sample a Rayleigh triangle $P = (P_{j,k})_{1 \leq k \leq j \leq n}$ from the density proportional to $e^{\langle y, \mathrm{type}(P) \rangle}$ on the polytope $GT(\lambda)$.

(3) **Sample from the fiber over P.** Sample a uniformly random matrix S from the fiber $\mathcal{R}^{-1}(P) = \{X \in O_\Lambda \mid \mathcal{R}(X) = P\}$.

(4) **Output S.**

We now describe Steps 1 and 3 in detail, and we also discuss the algorithms we cite and invoke for Step 2. In Section 5, we then complete the proofs of Theorems 1.3 and 1.4. Before describing the steps, we give one result which demonstrates that the steps of the above algorithm sample correctly from the orbit under the assumption that, in Step 2, we are able to sample exactly from the desired distribution with no error. We refer the reader to Appendix A of the arXiv version [20] for the straightforward details on handling the case where the distribution on $GT(\lambda)$ only approximates the target distribution.

PROPOSITION 4.2 (CORRECTNESS OF THE IDEAL ALGORITHM). *Let P be a random Rayleigh triangle distributed according to the distribution given by the density proportional to $e^{\langle y, \mathrm{type}(P) \rangle}$ on $GT(\lambda)$, and let S be a uniform random element of $\mathcal{R}^{-1}(P)$. Then S is distributed according to the measure $\nu(X) \propto e^{\langle Y, X \rangle} d\mu_\Lambda(X)$ on O_Λ.*

PROOF. By Theorem 2.3, the density function $e^{\langle Y, X \rangle}$ is constant on the fibers of $\mathcal{R}$, since $e^{\langle Y, X \rangle} = e^{\langle y, \mathrm{type}(\mathcal{R}(X)) \rangle}$. Using Theorem 2.3 again, the statement then follows immediately from the disintegration theorem for probability measures; see [5]. ∎

4.2 Step 1: Reduce Sampling from the Orbit to Sampling from the GT Polytope

In this section we describe the algorithm for constructing membership and evaluation oracles for the (unnormalized) exponential density function on the polytope $GT(\lambda)$.

Recall the following system of n equalities and $n \times (n-1)$ inequalities which determine if a Rayleigh triangle P is an element of $GT(\lambda)$ (see Equations 5 and 6):

$$P_{i,n} = \lambda_i \text{ for all } 1 \leq i \leq n,$$

$$P_{i,j} - P_{i,j-1} \geq 0 \text{ and } P_{i,j-1} - P_{i+1,j} \geq 0 \text{ for all } 1 \leq i < j \leq n.$$

Note that whenever some of the values of λ_i are actually equal, some of the inequalities will become equalities. In particular, if $\lambda_p = \lambda_{p+1} = \cdots = \lambda_q$ then for any $P \in GT(\lambda)$ we have that $P_{i,j} = \lambda_p$ for all i, j such that $p \leq i \leq q+j-n$. Using this observation, for any fixed λ it is straightforward to determine the ambient affine space in which $GT(\lambda)$ has non-empty interior.

The unnormalized density function on the polytope is then also easily described. Given a real vector y, Theorem 2.3 implies that the density function on $GT(\lambda)$ that we want to sample from is proportional to

$$f_0(P) = e^{\langle y, \mathrm{type}(P) \rangle}.$$

Recall from Definition 2.3 that $\mathrm{type}(P) \in \mathbb{R}^n$ is defined by

$$\mathrm{type}(P)_k := \sum_{i=1}^{k} P_{i,k} - \sum_{j=1}^{k-1} P_{j,k-1}.$$

Using this definition, we can write down the exponent of f_0 as a linear functional on P. We first have

$$\langle y, \mathrm{type}(P) \rangle = y_n \cdot \left[\sum_{i=1}^{n} \lambda_i \right] + \sum_{k=1}^{n-1} (y_k - y_{k+1}) \cdot \left[\sum_{i=1}^{k} P_{i,k} \right].$$

Notice that for fixed y and λ, we have that $y_n \cdot \left[\sum_{i=1}^{n} \lambda_i \right]$ is a constant in P. Therefore we can push this part of the exponent into the normalization factor. (For other entries of $P_{i,j}$ that are fixed by equalities in the λ vector, this can also be done.) We now define a triangle of values via $y_{i,j}^\Delta := y_j - y_{j+1}$ for $1 \leq i \leq j \leq n-1$. With this, we want to sample from a density function on $GT(\lambda)$ proportional to

$$f(P) = e^{\langle y^\Delta, P \rangle} \qquad \text{with} \qquad y_{i,j}^\Delta := y_j - y_{j+1}, \qquad (7)$$

where the top row of P is ignored. In particular, this means that the density function we want to sample from on $GT(\lambda)$ is in fact log-linear. Note further that shifting y by a multiple of the all-ones vector does not change the value of y^Δ. Therefore we may assume that $y_1 \geq 0 \geq y_n$ if desired.

The above discussion then implies the following algorithmic guarantees for construction and running of the oracles.

LEMMA 4.3 (MEMBERSHIP AND EXACT EVALUATION ORACLES). *There exists an algorithm such that, given $n \in \mathbb{N}$, $\lambda \in \mathbb{R}^n$, and $y \in \mathbb{R}^n$, outputs a membership oracle for $GT(\lambda)$ and an evaluation oracle for $f(P) = e^{\langle y, \text{type}(P) \rangle}$. The number of arithmetic operations required to run this algorithm is polynomial in n and the number of bits required to represent y and λ. Further, the number of arithmetic operations required to then run these oracles with input P is polynomial in the number of bits needed to represent λ, y, and P.*

4.3 Step 2: Sample a Rayleigh Triangle from the GT Polytope

To sample a Rayleigh triangle $P \in GT(\lambda)$ according to the log-linear density discussed above, we appeal to powerful tools for sampling from log-concave and log-Lipschitz densities on convex polytopes. Here we discuss two particular ways to do this, in terms of TV distance error and in terms of infinity divergence error.

Sampling from $GT(\lambda)$ with TV Distance Error. To sample from a distribution within TV distance ξ from the target exponential density on $GT(\lambda)$, we appeal to a result of Lovász and Vempala.[2]

THEOREM 4.4 (FOLLOWS FROM COROLLARY 1.2 OF [23]; SEE ALSO SECTION 2.1 OF [24]). *Let $K \subset \mathbb{R}^d$ be a convex polytope, and for $\ell \in \mathbb{R}^d$ let μ_ℓ denote the distribution on K defined by the density function proportional to $f_\ell(x) := e^{\langle \ell, x \rangle}$. There is an algorithm that, given a membership oracle for K, a vector $\ell \in \mathbb{R}^d$, a point $x_0 \in K$, an outer radius R of K, an inner radius r of K, and a $\xi > 0$, samples from a distribution $\tilde{\mu}_\ell$ on K with the property that*

$$\|\tilde{\mu}_\ell - \mu_\ell\|_{\text{TV}} < \xi.$$

The algorithm makes $\text{poly}(d, \log \|\ell\|, \log \frac{R}{r}, \log \frac{1}{\xi})$ calls to the membership and evaluation oracles.

Given the membership and evaluation oracles from Step 1 of the algorithm above, we can apply this result to sample from $GT(\lambda)$. Beyond the oracles, we also need the starting point x_0 and outer and inner balls for $GT(\lambda)$, which we discuss below.

Sampling from $GT(\lambda)$ with Infinity Divergence Error. To achieve the infinity divergence bound claimed in Theorem 1.4, we use a result of Bassily, Smith, and Thakurta. We state a simplified version of this result here for the convenience of the reader. Note that the dependence on $\log \frac{1}{r}$ appears because of the need to first put $GT(\lambda)$ in isotropic position; see Section 3.2 of the arXiv version of [3].

THEOREM 4.5 ([3], SEE LEMMA 6.5). *Let $K \subset \mathbb{R}^d$ be a convex polytope, and for $\ell \in \mathbb{R}^d$ let μ_ℓ denote the distribution on K defined by the density function proportional to $f_\ell(x) := e^{\langle \ell, x \rangle}$. There is an algorithm that, given a membership oracle for K, a vector $\ell \in \mathbb{R}^d$, an outer radius R of K, an inner radius r of K, and a $\xi > 0$, samples from a distribution $\tilde{\mu}_\ell$ on K with the property that $D_\infty(\tilde{\mu}_\ell \| \mu_\ell) < \xi$. The algorithm makes $\text{poly}(d, \|\ell\|, R, \log \frac{1}{r}, \frac{1}{\xi})$ calls to the membership and evaluation oracles.*

Given the membership and evaluation oracles from Step 1 of the algorithm above, we can apply this result to sample from $GT(\lambda)$. Beyond the oracles, we also need the outer and inner balls for $GT(\lambda)$, which we discuss below.

Before moving on, we note the main distinction between Theorems 4.5 and 4.4 above. That is, Theorem 4.5 achieves a stronger notion of approximation of the target exponential density at the cost of a larger number of oracle calls. Specifically, the number of oracle calls in Theorem 4.5 depends polynomially on $\|\ell\|$, R, and $\frac{1}{\xi}$, whereas in Theorem 4.4 these dependencies are polylogarithmic. We leave it as an open problem whether or not one can achieve infinity divergence error of ξ in $\text{poly}(d, \log \|\ell\|, \log \frac{R}{r}, \log \frac{1}{\xi})$ oracle calls.

Extra Inputs Required for the Polytope Sampling Algorithms. As discussed above, we also need to be able to compute some extra data to apply the above polytope sampling algorithms to the target exponential density function on $GT(\lambda)$. Specifically, we need a starting point P_0 for the algorithm, an outer radius R, and an inner radius r. We give this data in the following three lemmas. Note that by using a simpler argument than that of Lemma 4.8, one can achieve a worse bound on r which is good enough for our purposes; see Remark 4.9 below. For the proofs of these lemmas, see the arXiv version [20].

LEMMA 4.6 (STARTING POINT FOR SAMPLING FROM $GT(\lambda)$). *There is an algorithm that, given $\lambda \in \mathbb{R}^n$, samples uniformly from the polytope $GT(\lambda)$. The number of arithmetic operations required to run this algorithm is polynomial in the number of bits required to represent λ.*

LEMMA 4.7 (OUTER BALL FOR $GT(\lambda)$). *The polytope $GT(\lambda)$ is contained by a ball of radius $R = \sqrt{n} \cdot (\lambda_1 - \lambda_n)$.*

LEMMA 4.8 (INNER BALL FOR $GT(\lambda)$). *Let $q > 0$ be the minimal integer such that $\lambda_i = \frac{p_i}{q}$ for some integers $p_1, \ldots, p_n$. The polytope $GT(\lambda)$, considered as a subset of its affine span, contains a ball of radius $r = \frac{1}{8n^2 q}$.*

REMARK 4.9. *One can obtain a cheaper bound on the radius r of a small ball contained in $GT(\lambda)$, by defining $P \in GT(\lambda)$ inductively by simply choosing $P_{i,j}$ to be the midpoint between $P_{i,j+1}$ and $P_{i+1,j+1}$ for all valid i, j. Using this as the center of a small ball, one obtains a bound of $r \geq \frac{1}{2^{O(n)}q}$. Since the number of arithmetic operations required by our algorithms depends polylogarithmically on $\frac{1}{r}$, this bound would be enough for our purposes.*

4.4 Step 3: Sample a Uniformly Random Matrix from the Fiber

Once we have sampled P, it remains to sample a matrix S uniformly at random from the fiber $\mathcal{R}^{-1}(P)$. The uniform distribution on the fiber is defined by the property that if X is uniformly distributed in $\mathcal{R}^{-1}(P)$, then for $1 < k \leq n$, $X[k]$ is uniformly distributed on the compact manifold $\mathcal{H}(X[k-1]; P_{\bullet,k})$ of $k \times k$ Hermitian matrices with eigenvalues $P_{\bullet,k}$ and leading $(k-1) \times (k-1)$ submatrix equal to $X[k-1]$. Equivalently, the uniform measure on $\mathcal{R}^{-1}(P)$ is the disintegration (via the Rayleigh map) of the uniform measure on O_Λ, in the sense of [5]. We construct a uniform sample $S \in \mathcal{R}^{-1}(P)$

---

[2]The precise statement that we invoke here is not explicitly stated in their papers, but follows readily from the cited results and has been confirmed to us in personal correspondence [37].

using an inductive procedure, successively sampling the last row and column of each leading submatrix $S[k]$. We first define $S[1]$ to be the 1×1 matrix $[P_{1,1}]$, and we then sample each submatrix $S[k]$, for $1 < k \leq n$, such that $S[k]$ is uniformly distributed on $\mathcal{H}(S[k-1]; P_{\bullet,k})$. Explicitly, we sample $S[k]$ given $S[k-1]$ as follows.

Sampling Procedure for $S[k]$ given $S[k-1]$ and $P_{\bullet,k}$:

(1) **Diagonalize $S[k-1]$:** Compute a unitary matrix $U \in \mathrm{U}(k-1)$ such that $U^* \cdot S[k-1] \cdot U$ is diagonal.

(2) **Compute the new diagonal entry of $S[k]$:** Write

$$S[k] = \begin{bmatrix} S[k-1] & Uv \\ (Uv)^* & c \end{bmatrix}, \qquad (8)$$

where $v \in \mathbb{C}^{k-1}$ and $c \in \mathbb{R}$ are to be determined. Since the diagonal entries of $S[k]$ are just the type vector of $\mathcal{R}(S[k])$, we can compute

$$c = \sum_{i=1}^{k} P_{i,k} - \sum_{j=1}^{k-1} P_{j,k-1}.$$

(3) **Compute the magnitudes of the new off-diagonal entries of $S[k]$:** It remains to sample v uniformly at random from the set of vectors in $\mathbb{C}^{k-1}$ such that the matrix $S[k]$ in (8) has spectrum $P_{\bullet,k}$. We prove below in Lemma 4.10 that this can be done using the following procedure. Let $\delta_1 > \cdots > \delta_m$ be the distinct entries of $P_{\bullet,k-1}$, where δ_i has multiplicity n_i, so that $n_1 + \cdots + n_m = k - 1$. The interlacing relations (6) imply that each value δ_i occurs in $P_{\bullet,k}$ with multiplicity at least $n_i - 1$. Let $(\mu_1, \ldots, \mu_{m+1})$ be the vector obtained by removing $n_i - 1$ entries equal to δ_i from $P_{\bullet,k}$, for each i. Then define

$$r_i = \sqrt{-\frac{\prod_{j=1}^{m+1}(\delta_i - \mu_j)}{\prod_{j \neq i}(\delta_i - \delta_j)}}.$$

The interlacing relations guarantee that the quantity under the square root above is non-negative, so that r_i is well defined. As shown below, the vector v is distributed uniformly on a product of complex spheres of radii $r_1, \ldots, r_m$.

(4) **Sample the phases of the new off-diagonal entries of $S[k]$:** For each $i = 1, \ldots, m$, we then sample the n_i coordinates

$$\left(v_{1+\sum_{j=1}^{i-1} n_j}, \ \ldots, \ v_{n_i + \sum_{j=1}^{i-1} n_j} \right) \in \mathbb{C}^{n_i}$$

uniformly at random from the sphere of radius r_i in $\mathbb{C}^{n_i}$. This last step can be accomplished by well-known methods; see e.g. [28].

Output: Finally, after iteratively sampling all of the leading submatrices, we output $S = S[n]$.

Correctness and Number of Operations of the Iterative Algorithm. We now prove that the above algorithm samples from the correct distribution on the fiber $\mathcal{R}^{-1}(P)$ of P, and then we bound the number of operations the algorithm requires.

LEMMA 4.10 (SAMPLING FROM THE FIBER OVER P: CORRECTNESS). *The above algorithm, given a Rayleigh triangle $P \in GT(\lambda)$, returns a uniform random element of the fiber $\mathcal{R}^{-1}(P) = \{S \in O_\Lambda : \mathcal{R}(S) = P\}$.*

PROOF. Again we write $\mathcal{H}(S[k-1]; P_{\bullet,k})$ for the set of $k \times k$ Hermitian matrices with eigenvalues $P_{\bullet,k}$ and $(k-1)$th leading submatrix equal to $S[k-1]$. It only remains to show that $\mathcal{H}(S[k-1]; P_{\bullet,k})$ is a product of spheres as described above. Specifically, let $U \in \mathrm{U}(k-1)$ be a unitary matrix diagonalizing $S[k-1]$, so that $U^* \cdot S[k-1] \cdot U = \mathrm{diag}(P_{\bullet,k-1})$. We will show

$$\mathcal{H}(S[k-1]; P_{\bullet,k}) = \left\{ \begin{bmatrix} S[k-1] & Uv \\ (Uv)^* & c \end{bmatrix} \ \middle| \ \sum_{\ell=1}^{n_i} |v_{\ell + \sum_{j=1}^{i-1} n_j}|^2 = r_i^2, \forall i \right\}, \qquad (9)$$

where we necessarily have

$$c = \sum_{i=1}^{k} P_{i,k} - \sum_{j=1}^{k-1} P_{j,k-1}$$

due the the fact that the diagonal of any Hermitian matrix X is equal to $\mathrm{type}(\mathcal{R}(X))$. Write $D = \mathrm{diag}(P_{\bullet,k-1})$. To establish (9), we must show that a matrix of the form

$$S = \begin{bmatrix} U & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} D & v \\ v^* & c \end{bmatrix} \begin{bmatrix} U^* & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} S[k-1] & Uv \\ (Uv)^* & c \end{bmatrix}$$

has eigenvalues $P_{\bullet,k}$ if and only if $\sum_{\ell=1}^{n_i} |v_{l + \sum_{j=1}^{i-1} n_j}|^2 = r_i^2$ for $i = 1, \ldots, m$. We prove this by writing the characteristic polynomial of S in two different ways. First, if S has eigenvalues $P_{\bullet,k}$ then

$$\det(tI - S) = \prod_{i=1}^{k}(t - P_{i,k}). \qquad (10)$$

On the other hand, we must have

$$\det(tI - S) = \det\left(tI - \begin{bmatrix} D & v \\ v^* & c \end{bmatrix}\right) = \det\begin{bmatrix} tI - D & v \\ v^* & t - c \end{bmatrix},$$

and expanding along the last row and column we find that this equals

$$(t - c)\prod_{j=1}^{m}(t - \delta_j)^{n_j}$$
$$- \sum_{i=1}^{m}\left(\sum_{\ell=1}^{n_i} |v_{n_1 + \cdots + n_{i-1} + \ell}|^2\right)(t - \delta_i)^{n_i - 1}\prod_{j \neq i}(t - \delta_j)^{n_j}. \qquad (11)$$

We have $S \in \mathcal{H}(S[k-1]; P_{\bullet,k})$ exactly when (10) equals (11). Equating these two expressions for the characteristic polynomial and recalling that interlacing of $P_{\bullet,k}$ and $P_{\bullet,k-1}$ implies that $P_{\bullet,k}$ contains the value δ_i with multiplicity at least $n_i - 1$ for all i, we can divide through both sides by $(t - \delta_i)^{n_i - 1}$ for all i to obtain

$$\prod_{i=1}^{m+1}(t - \mu_i) = (t - c)\prod_{j=1}^{m}(t - \delta_j)$$
$$- \sum_{i=1}^{m}\left(\sum_{\ell=1}^{n_i} |v_{n_1 + \cdots + n_{i-1} + \ell}|^2\right)\prod_{j \neq i}(t - \delta_j). \qquad (12)$$

Note that both sides of (12) are monic polynomials of degree $m + 1$, and

$$c = \sum_{i=1}^{k} P_{i,k} - \sum_{j=1}^{k-1} P_{j,k-1} = \sum_{i=1}^{m+1}\mu_i - \sum_{j=1}^{m}\delta_j$$

implies the coefficients of t^{m+1} and t^m on both sides are equal. Therefore the polynomials in (12) are equal if and only if they

are equal at m distinct points. Evaluating both sides at $t = \delta_i$ for $i = 1, \ldots, m$, we find that they are equal exactly when

$$\sum_{l=1}^{n_i} |v_{l+\sum_{j=1}^{i-1} n_j}|^2 = r_i^2$$

for $i = 1, \ldots, m$, which is the desired result. ∎

Lemma 4.11 (Sampling from the fiber over P: Number of operations). *The number of arithmetic operations the above algorithm requires to sample uniformly from the fiber $\mathcal{R}^{-1}(P)$ is polynomial in the number of bits required to represent the entries of P.*

Proof. To determine the number of arithmetic operations required by this part of the algorithm, we first determine the number of operations for each of the steps described above. Step 1 amounts to unitarily diagonalizing a Hermitian matrix $H = U^* D U$, and this can be done in a number of operations which is polynomial in the size of the matrix k and the bit complexity of the entries of the matrix H. (We will discuss below the bit complexity of H.) Step 2 involves basic matrix operations with U and elements of the Rayleigh triangle P, which again depends polynomially on k and the bit complexity of H and P. Step 3 does basic arithmetic on the entries of P, requiring a number of operations which is polynomial in k and the bit complexity of the entries of the Rayleigh triangle P. Step 4 requires sampling of elements of the unit sphere and multiplying those samples by the magnitudes computed in step 3, and this also can be done in a number of operations which is polynomial in k and the bit complexity of the entries of P.

The whole iterative process to construct $S = S[n]$ then requires n iterations of the above 4 steps, where the output to each iteration is $S[k]$ and the input to each iteration is the Rayleigh triangle P along with the output $S[k-1]$ of the previous iteration. Note that steps 2, 3, and 4 only refer to the entries of P and not to the output of the previous iteration. The new entries of $S[k]$ constructed from steps 2, 3, and 4 then require $\mathrm{poly}(n, L_P)$ bits to represent, where L_P is the number of bits needed to represent the entries of P. Thus in each iteration we add new entries, with bit complexity $\mathrm{poly}(n, L_P)$, to $S[k-1]$ to construct the output $S[k]$. The unitary diagonalization of $S[k-1]$ in step 1 then requires a number of operations which is polynomial in the number of bits needed to represent the entries of $S[k-1]$. And since we are only adding new entries to $S[k-1]$ to construct $S[k]$ (not changing previously constructed entries), after each iteration the entries of $S[k]$ require $\mathrm{poly}(n, L_P)$ bits to represent. After all n iterations, the algorithm has sampled $S = S[n]$ in a number of arithmetic operations that is polynomial in n and in the number of bits required to represent the entries of P. ∎

5 PROOFS OF THEOREMS 1.3 AND 1.4

In this section, we complete the proofs of Theorems 1.3 and 1.4 using the results of the previous section. We first prove correctness of the algorithms, and then we prove bounds on the required number of arithmetic operations.

5.1 Correctness

For Theorem 1.3, we want to show that the algorithm from the previous section samples from O_Λ according to the exponential density proportional to $e^{\langle Y, X \rangle} d\mu_\Lambda(X)$. Recall that the algorithm consists of two main steps: sampling P from $GT(\lambda)$ (called Step 2 above), and then sampling from the fiber $\mathcal{R}^{-1}(P)$ (called Step 3 above). To sample from $GT(\lambda)$, we use one of two algorithms: the algorithm given by Theorem 4.4 for TV distance error claimed in Theorem 1.3, or the algorithm given by Theorem 4.5 for infinity divergence error claimed in Theorem 1.4. These algorithms require a membership oracle for $GT(\lambda)$ (given by Lemma 4.3), a vector ℓ and an evaluation oracle for the target density $g(P) \propto e^{\langle \ell, P \rangle}$ on $GT(\lambda)$ (also given by Lemma 4.3), a starting point for the algorithm (given by Lemma 4.6), and outer and inner balls for $GT(\lambda)$ (given by Lemmas 4.7 and 4.8 respectively).

Once we have our sample P from $GT(\lambda)$, we use it to sample uniformly from the fiber $\mathcal{R}^{-1}(P)$ via Lemma 4.10. The last thing we need to prove then is that, by sampling from $GT(\lambda)$ and then from the corresponding fiber, we are in fact sampling from O_Λ according to the exponential density proportional to $e^{\langle Y, X \rangle} d\mu_\Lambda(X)$ as claimed. For this, we handle the cases of Theorems 1.3 and 1.4 separately.

Correctness for Theorem 1.3. Let ν be the target distribution on $GT(\lambda)$ associated to the unnormalized density function $f(P) = e^{\langle y, \mathrm{type}(P) \rangle}$. Equation 7 shows that $f(P) = e^{\langle y^\Delta, P \rangle}$, and thus we can apply Theorem 4.4 to f to sample P from $GT(\lambda)$ according to a distribution $\hat{\nu}$ for which $\|\hat{\nu} - \nu\|_{\mathrm{TV}} < \xi$.

Now given P in $GT(\lambda)$, Lemma 4.10 then says that the algorithm of Section 4.4 samples uniformly from the fiber of P. By standard arguments, the uniform distribution on the fiber $\mathcal{R}^{-1}(P)$ is the disintegrated measure of the target distribution on O_Λ, and the overall algorithm samples from O_Λ according to a distribution which is within TV distance error ξ of the target. (See Appendix A in the arXiv version [20] for more discussion of these standard arguments, and see also Proposition 4.2 for a similar result in the ideal case.) This completes the proof of correctness of the algorithm of Theorem 1.3.

Correctness for Theorem 1.4. Let ν be the target distribution on $GT(\lambda)$ associated to the unnormalized density function $f(P) = e^{\langle y, \mathrm{type}(P) \rangle}$. Equation 7 shows that $f(P) = e^{\langle y^\Delta, P \rangle}$, and thus we can apply Theorem 4.5 to f to sample P from $GT(\lambda)$ according to a distribution $\hat{\nu}$ for which $D_\infty(\hat{\nu} \| \nu) < \xi$.

Given P in $GT(\lambda)$, Lemma 4.10 then says that the algorithm of Section 4.4 samples uniformly from the fiber of P. As above, the uniform distribution on the fiber $\mathcal{R}^{-1}(P)$ is the disintegrated measure of the target distribution on O_Λ, and the overall algorithm samples from O_Λ according to a distribution which is within infinity divergence error ξ of the target. (See Appendix A in the arXiv version [20] for more discussion, and see also Proposition 4.2 for a similar result in the ideal case.) This completes the proof of correctness of the algorithm of Theorem 1.4.

5.2 Number of Operations

We now determine the number of arithmetic operations required of the algorithms of Theorems 1.3 and 1.4. For both algorithms we are given $n \in \mathbb{N}$, $\lambda \in \mathbb{R}^n$, $y \in \mathbb{R}^n$, and a desired error bound $\xi > 0$. As described in Section 4.1, we need to (1) construct the membership oracles, (2) use them to sample P from the polytope $GT(\lambda)$, and (3) then sample uniformly from the fiber $\mathcal{R}^{-1}(P)$ over P. Steps 1 and 3

are exactly the same for both algorithms. Lemma 4.3 implies the necessary oracles can be constructed using a number of operations which is polynomial in n and in the number of bits required to represent y and λ. Lemma 4.11 implies we can sample from the fiber over P in a number of operations which is polynomial in n and in the number of bits required to represent P. We now discuss the number of arithmetic operations required of the algorithms used to sample P from $GT(\lambda)$.

Recall from Section 4.2 that the target distribution on $GT(\lambda)$ is given by a density proportional to

$$f(P) = e^{\langle y^{\Delta}, P \rangle} \quad \text{with} \quad y^{\Delta}_{i,j} := y_j - y_{j+1}.$$

From this we achieve the bound

$$\|y^{\Delta}\| \leq n^2 (y_1 - y_n).$$

Further, we also have the outer and inner balls for $GT(\lambda)$ via Lemmas 4.7 and 4.8, given as

$$R = \sqrt{n} \cdot (\lambda_1 - \lambda_n) \quad \text{and} \quad \frac{1}{r} = 8n^2 q,$$

where $q > 0$ is an integer such that $\lambda_i = \frac{p_i}{q}$ for some integers $p_1, \ldots, p_n$. We now use these bounds in order to finish the analysis of the algorithms.

Number of Operations for Theorem 1.3. For Theorem 1.3, we apply Theorem 4.4 as described above which implies we can sample from a distribution on $GT(\lambda)$ within TV distance error ξ of the target distribution in $\mathrm{poly}(n, \log \frac{1}{\xi}, \log(y_1 - y_n), \log(\lambda_1 - \lambda_n), L_y, L_\lambda)$ calls to the membership and evaluation oracles, where L_y and L_λ are the number of bits required to represent y and λ respectively. Since $\log(y_1 - y_n)$ and $\log(\lambda_1 - \lambda_n)$ are bounded above by L_y and L_λ respectively, we have that the above sampling can be done in $\mathrm{poly}(n, \log \frac{1}{\xi}, L_y, L_\lambda)$ calls to the membership and evaluation oracles. The bits then required to represent the sample P from $GT(\lambda)$ can then be no larger than the number of oracle calls. Combining this with Lemma 4.3 and the above discussion implies the number of arithmetic operations required to run the algorithm claimed by Theorem 1.3 is polynomial in n, $\log \frac{1}{\xi}$, and the number of bits needed to represent y and λ. Since the number of bits needed to represent y (or λ) is at least n, we can drop the explicit dependence on n.

Number of Operations for Theorem 1.4. For Theorem 1.4, we apply Theorem 4.5 as described above which implies we can sample from a distribution on $GT(\lambda)$ within infinity divergence error ξ of the target distribution in $\mathrm{poly}(y_1 - y_n, \lambda_1 - \lambda_n, \frac{1}{\xi}, L_y, L_\lambda)$ calls to the membership and evaluation oracles, where L_y and L_λ are the number of bits required to represent y and λ respectively. As above, the bits then required to represent the sample P from $GT(\lambda)$ can then be no larger than the number of oracle calls. Combining this with Lemma 4.3 and the above discussion implies the number of arithmetic operations required to run the algorithm claimed by Theorem 1.4 is polynomial in $y_1 - y_n$, $\lambda_1 - \lambda_n$, $\frac{1}{\xi}$, and the number of bits needed to represent y and λ.

6 DIFFERENTIALLY PRIVATE RANK-k APPROXIMATION

We consider the problem of differentially private low-rank approximation. In the low-rank approximation problem, we are given a $d \times d$ real positive semidefinite (PSD) matrix A and $1 \leq k \leq n$, and the goal is to output the space spanned by the top k eigenvectors of A. Let $\mathcal{P}_k$ denote the set of $d \times d$ rank-k Hermitian PSD projection matrices, considered as a subset of the space of complex Hermitian matrices. It is easy to see that

$$\max_{P \in \mathcal{P}_k} \langle P, A \rangle = \sum_{i=1}^{k} \gamma_i,$$

where $\gamma_1 \geq \cdots \geq \gamma_d \geq 0$ are the eigenvalues of A.

Differential Privacy. Let $\mathcal{U}$ be the universe of users. For each $u \in \mathcal{U}$, we have a vector $v_u \in \mathbb{R}^d$ such that $\|v_u\|_2 \leq 1$. Given a dataset $D \subseteq \mathcal{U}$, define $A := \sum_{u \in D} v_u v_u^*$.

DEFINITION 6.1. *Given an $\varepsilon > 0$ and a set R, a randomized mechanism $\mathcal{M} : \mathbb{R}^{d \times d} \to R$ is said to be $(\varepsilon, 0)$-differentially private if for all $S \subseteq R$ and for all $D, D' \subseteq \mathcal{U}$ such that the symmetric set difference $D \Delta D'$ has cardinality 2, one has*

$$\Pr[\mathcal{M}(A) \in S] \leq e^{\varepsilon} \Pr[\mathcal{M}(A') \in S].$$

Here $A := \sum_{u \in D} v_u v_u^*$ and $A' := \sum_{u \in D'} v_u v_u^*$.

In our setting, R is the space of $d \times d$ and rank-k Hermitian matrices. We now copy Theorem 1.5 from the introduction, which we prove in this section.

THEOREM 6.1 (DIFFERENTIALLY PRIVATE LOW-RANK APPROXIMATION). *There is a randomized algorithm that, given a positive semidefinite $d \times d$ matrix A and its eigenvalues $\gamma_1 \geq \cdots \geq \gamma_d$, an integer $1 \leq k \leq d$, and an $\varepsilon > 0$, outputs a rank-k $d \times d$ Hermitian projection P that is $(\varepsilon, 0)$-differentially private. Moreover, there is a universal constant $C > 0$ such that, if there is a $\delta > 0$ satisfying $\sum_{i=1}^{k} \gamma_i \geq C \cdot \frac{dk}{\varepsilon \delta} \cdot \log \frac{1}{\delta}$, then we have:*

$$\mathbb{E}_P \left[\langle A, P \rangle \right] \geq (1 - \delta) \sum_{i=1}^{k} \gamma_i.$$

The number of arithmetic operations required by this algorithm is polynomial in $\frac{1}{\varepsilon}$, $\gamma_1 - \gamma_d$, and the number of bits needed to represent γ.

This result generalizes a Hermitian version of Theorem 1.1 of [18], where the above result is given in the case of $k = 1$ for real symmetric rank-one matrices. Specifically their Theorem 1.1 gives an algorithm which outputs an $(\varepsilon, 0)$-differentially private real unit vector v for which the expected value of $v^{\top} A v = \langle A, v v^{\top} \rangle$ is bounded below by $(1 - \delta)\gamma_1$ whenever $\gamma_1 \geq \Omega(\frac{d}{\varepsilon \delta} \cdot \log \frac{1}{\delta})$. They then use the rank-one case to prove a somewhat similar result in the general rank-k case, which we state now. The main difference here is that their Theorem 1.2 stated below outputs a real symmetric positive semidefinite matrix which approximates A, while our Theorem 6.1 above outputs a Hermitian projection P which projects onto a k-dimensional subspace for which $\langle A, P \rangle$ approximates the sum of the top k eigenvalues of A.

Theorem 6.2 (Theorem 1.2 of [18]). *Let A be a $d \times d$ real symmetric positive semidefinite matrix with eigenvalues $\gamma_1 \geq \cdots \geq \gamma_d$. There exists an $(\varepsilon, 0)$-differentially private polynomial-time algorithm for computing a matrix A_k of rank at most k so that $\|A - A_k\|_2 \leq \gamma_{k+1} + \delta\gamma_1$ as long as $\gamma_1 \geq \Omega(\frac{dk^3}{\varepsilon\delta^6})$.*

We now compare the respective utility bounds for the two differentially private rank-k mechanisms. For their mechanism the "utility" can be described by the error term $\delta\gamma_1$, which is bounded below by

$$\delta\gamma_1 \geq \Omega\left(\frac{dk^3}{\varepsilon\delta^5}\right).$$

For our mechanism the "utility" can be described by the error term $\delta\sum_{i=1}^{k} \gamma_i$, which is bounded below by

$$\delta\sum_{i=1}^{k} \gamma_i \geq \Omega\left(\frac{dk}{\varepsilon} \cdot \log\frac{1}{\delta}\right).$$

Since δ is assumed to be small, our rank-k mechanism improves upon the utility (error bound) of the rank-k mechanism from [18].

The Proof of Theorem 6.1. We now prove Theorem 6.1 by combining the exponential mechanism framework due to [25] with Theorem 1.4. Given a $\sigma > 0$, we define $\mathcal{M}' = \mathcal{M}'(A)$ to be the mechanism which is given by the sampling algorithm of Theorem 1.4 with λ being the vector that has k ones and $d - k$ zeros, y being the eigenvalues of A multiplied by $\frac{\varepsilon}{4\sigma}$, and $\xi = \frac{\varepsilon}{2}$. Therefore $\mathcal{M}' = \mathcal{M}'(A)$ outputs a sample from a distribution $\tilde{v}'_A$ on the set of $n \times n$ rank-k PSD projections which is within infinity divergence error $\frac{\varepsilon}{2}$ of the distribution v'_A given by the density $e^{\frac{\varepsilon}{4\sigma} \langle \text{diag}(\gamma), P \rangle}$.

Next we diagonalize A to determine the unitary matrix U for which $A = U \cdot \text{diag}(\gamma) \cdot U^*$. With this, we define $\mathcal{M} = \mathcal{M}(A)$ to be the mechanism which is given by sampling P from $\mathcal{M}'(A)$ and then outputting UPU^*. Since $\langle A, UPU^* \rangle = \langle U \cdot \text{diag}(\gamma) \cdot U^*, UPU^* \rangle = \langle \text{diag}(\gamma), P \rangle$ and $\mathcal{P}_k$ is unitarily invariant, we have that $\mathcal{M}$ outputs a sample from a distribution $\tilde{v}_A$ on the set of $n \times n$ rank-k PSD projections which is within infinity divergence error $\frac{\varepsilon}{2}$ of the target distribution v_A given by the density $e^{\frac{\varepsilon}{4\sigma} \langle A, P \rangle}$.

Now suppose σ is an upper bound on the following "sensitivity" of the function $\langle A, P \rangle$:

$$\sup_{A, A'} \sup_{P} |\langle A, P \rangle - \langle A', P \rangle|,$$

where A, A' are such that

$$A' = A - v_1 v_1^* + v_2 v_2^*$$

for some $\|v_1\|_2, \|v_2\|_2 \leq 1$ and P is a rank-k PSD projection matrix. Then Lemma 6.3 says that we can choose $\sigma = 1$, and with this Lemma 6.4 implies that $\mathcal{M}$ is $(\varepsilon, 0)$-differentially private.

The number of arithmetic operations required for this algorithm then can be bounded by applying Theorem 1.4 directly with our specified inputs. The number of operation required in Theorem 1.4 is polynomial in d, $\lambda_1 - \lambda_d$, $y_1 - y_d$, $\frac{1}{\varepsilon}$, and the number of bits required to represent λ and y. In our case, λ is a vector of 0's and 1's, and $y = \gamma$ is the sequence of eigenvalues of A. Therefore the number of arithmetic operations required to run the algorithm is polynomial in $\frac{1}{\varepsilon}$, $\gamma_1 - \gamma_d$, and the number of bits needed to represent γ (which is at least d).

6.1 Correctness: Privacy Guarantee

The privacy guarantee given below in Lemma 6.4 requires the following lemma on the sensitivity of the function $\langle A, P \rangle$. For the proofs of these lemmas, see the arXiv version [20].

Lemma 6.3 (Sensitivity). *For all A, A' PSD with $A' = A - v_1 v_1^* + v_2 v_2^*$ for some v_1, v_2 such that $\|v_1\|_2, \|v_2\|_2 \leq 1$ and for all rank-k PSD projection matrices P, we have that $|\langle A, P \rangle - \langle A', P \rangle| \leq 1 =: \sigma$.*

Lemma 6.4 (Privacy via the exponential mechanism). *For $\sigma = 1$, the mechanism $\mathcal{M}$ is $(\varepsilon, 0)$-differentially private.*

6.2 The Utility Bound

The utility bound given below in Lemma 6.6 requires the following lemma on the covering number for the orbit $\mathcal{P}_k$. See the arXiv version [20] for the proof of Lemma 6.5.

Lemma 6.5 (Covering number for $\mathcal{P}_k$). *Let $\mathcal{P}_k$ denote the set of $d \times d$ rank-k Hermitian PSD projection matrices, considered as a subset of the space of Hermitian matrices equipped with the ℓ^2 operator norm. For any $\zeta > 0$, the number of balls centered in $\mathcal{P}_k$ of radius ζ required to cover the set $\mathcal{P}_k$ is at most $\left(1 + \frac{8}{\zeta}\right)^{2dk}$.*

Lemma 6.6 (Utility bound). *The rank-k exponential mechanism, given a $d \times d$ Hermitian positive definite matrix A with eigenvalues $\gamma_1 \geq \gamma_2 \geq \cdots \geq \gamma_d$, outputs a $d \times d$ rank-k Hermitian PSD projection P such that*

$$\mathbb{E}_P\left[\langle A, P \rangle\right] \geq (1 - \delta) \sum_{i=1}^{k} \gamma_i$$

as long as $\sum_{i=1}^{k} \gamma_i \geq C \cdot \frac{dk}{\varepsilon\delta} \cdot \log\frac{1}{\delta}$ for small $\delta > 0$ and an absolute constant $C > 0$.

Proof. We first define "good" and "bad" sets via

$$G := \left\{ P \in \mathcal{P}_k : \langle A, P \rangle \geq \left(1 - \frac{\delta}{2}\right) \sum_{i=1}^{k} \gamma_i \right\},$$

$$B := \left\{ P \in \mathcal{P}_k : \langle A, P \rangle \leq (1 - \delta) \sum_{i=1}^{k} \gamma_i \right\}.$$

Let P_0 be the projection associated to the top k eigenvectors of A, and define $A_0 := AP_0$ so that the top k eigenpairs of A_0 agree with that of A and the rest of the eigenvalues are 0. Now fix any $P \in \mathcal{P}_k$ such that $\|P - P_0\|_2 < \frac{\delta}{2}$, where $\|\cdot\|_2$ denotes the ℓ^2 operator norm. Since the ℓ^2 operator norm is the ∞-norm on the singular values, we can apply Hölder's inequality to get

$$\langle A, P \rangle \geq \langle A_0, P \rangle = \langle A_0, P_0 \rangle - \langle A_0, P_0 - P \rangle$$

$$\geq \sum_{i=1}^{k} \gamma_i - \|P_0 - P\|_2 \sum_{i=1}^{k} \gamma_i > \left(1 - \frac{\delta}{2}\right) \sum_{i=1}^{k} \gamma_i.$$

That is, every $P \in \mathcal{P}_k$ contained in the ball of radius $\frac{\delta}{2}$ about P_0 is also contained in G.

Letting μ_k be the unitarily invariant probability measure on $\mathcal{P}_k$, the covering number lemma (Lemma 6.5) implies there is some ball $B_{\delta/2}(P')$ centered at $P' \in \mathcal{P}_k$ of radius $\frac{\delta}{2}$ is such that

$\mu_k(B_{\delta/2}(P')) \geq (1 + \frac{16}{\delta})^{-2dk}$. By unitary invariance of μ_k, we have

$$\mu_k(G) \geq \mu_k(B_{\delta/2}(P_0)) = \mu_k(B_{\delta/2}(P'))$$
$$\geq e^{-2dk \log(1 + \frac{16}{\delta})} \geq e^{-C' \cdot dk \log \frac{1}{\delta}}$$

for some absolute $C' \geq 2$ whenever δ is small. Now let

$$f_A(P) := e^{\frac{\varepsilon}{2} \langle A, P \rangle}$$

denote the unnormalized probability density function of the exponential mechanism, and let Z be the normalization constant. Then whenever $\sum_{i=1}^{k} \gamma_i \geq 8C' \cdot \frac{dk}{\varepsilon \delta} \cdot \log \frac{1}{\delta}$, we have

$$\frac{\mathbb{P}[P \in B]}{\mathbb{P}[P \in G]} \leq \frac{\mu_k(B) \cdot \max_{P \in B} \frac{f_A(P)}{Z}}{\mu_k(G) \cdot \min_{P \in G} \frac{f_A(P)}{Z}}$$

$$\leq \frac{1 \cdot e^{\frac{\varepsilon}{2}(1-\delta) \sum_{i=1}^{k} \gamma_i}}{e^{-C' \cdot dk \log \frac{1}{\delta}} \cdot e^{\frac{\varepsilon}{2}(1-\frac{\delta}{2}) \sum_{i=1}^{k} \gamma_i}} = \frac{e^{\frac{\varepsilon}{2}(-\frac{\delta}{2}) \sum_{i=1}^{k} \gamma_i}}{e^{-C' \cdot dk \log \frac{1}{\delta}}}$$

$$\leq \frac{e^{-2C' \cdot dk \log \frac{1}{\delta}}}{e^{-C' \cdot dk \log \frac{1}{\delta}}} = e^{-C' \cdot dk \log \frac{1}{\delta}}.$$

For $C' \geq 2$ and $\delta > 0$ small, a basic computation then yields

$$\mathbb{E}_P[\langle A, P \rangle] \geq (1 - \delta) \sum_{i=1}^{k} \gamma_i.$$

(See the arXiv version [20] for the full computation.) ∎

ACKNOWLEDGEMENTS

REFERENCES

[1] William Band and James L. Park. 1976. New information-theoretic foundations for quantum statistics. *Foundations of Physics* 6, 3 (01 June 1976), 249–262. https://doi.org/10.1007/BF00708800

[2] Yu. Baryshnikov. 2001. GUEs and queues. *Probab. Theory Relat. Fields* 119 (2001), 256–274.

[3] Raef Bassily, Adam Smith, and Abhradeep Thakurta. 2014. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*. IEEE, 464–473.

[4] D. V. Boulatov and V. A. Kazakov. 1987. The Ising model on a random planar lattice: The structure of the phase transition and the exact critical exponents. *Physics Letters B* 186 (1987), 379–384.

[5] Joseph T. Chang and David Pollard. 1997. Conditioning as disintegration. *Statistica Neerlandica* 51, 3 (1997), 287–317.

[6] Kamalika Chaudhuri, Anand Sarwate, and Kaushik Sinha. 2012. Near-optimal Differentially Private Principal Components. In *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger (Eds.). Curran Associates, Inc., 989–997. http://papers.nips.cc/paper/4565-near-optimal-differentially-private-principal-components.pdf

[7] Yasuko Chikuse. 2003. Concentrated matrix Langevin distributions. *Journal of Multivariate Analysis* 85, 2 (2003), 375 – 394. https://doi.org/10.1016/S0047-259X(02)00065-9

[8] Y. Chikuse. 2012. *Statistics on Special Manifolds*. Springer New York. https://books.google.com/books?id=7lX1BwAAQBAJ

[9] P. Di Francesco, P. Ginsparg, and J. Zinn-Justin. 1995. 2D gravity and random matrices. *Physics Reports* 254 (1995), 1–133. http://arxiv.org/abs/hep-th/9306153.

[10] B. Eynard. 2004. A short note about Morozov's formula. Service de Physique Théorique de Saclay, report no. SPHT-T04-077. https://arxiv.org/abs/math-ph/0406063.

[11] B. Eynard and A. Prats Ferrer. 2006. 2-matrix versus complex matrix model, integrals over the unitary group as triangular integrals. *Commun. Math. Phys.* 264 (2006), 115–144. https://arxiv.org/abs/hep-th/0502041.

[12] A. Guionnet. 2004. Large deviations and stochastic calculus for large random matrices. *Probability Surveys* 1 (2004), 72–172. https://arxiv.org/abs/math/0409277.

[13] Harish-Chandra. 1957. Differential Operators on a Semisimple Lie Algebra. *American Journal of Mathematics* 79, 1 (1957), 87–120. http://www.jstor.org/stable/2372387

[14] C. Itzykson and J.-B. Zuber. 1980. The planar approximation. II. *J. Math. Phys.* 21 (1980), 411–421.

[15] Edwin T. Jaynes. 1957. Information theory and statistical mechanics. *Physical Review* 106 (May 1957), 620–630. https://doi.org/10.1103/PhysRev.106.620

[16] Edwin T. Jaynes. 1957. Information theory and statistical mechanics. II. *Physical Review* 108 (Oct. 1957), 171–190. https://doi.org/10.1103/PhysRev.108.171

[17] Mark Jerrum, Leslie G. Valiant, and Vijay V. Vazirani. 1986. Random Generation of Combinatorial Structures from a Uniform Distribution. *Theor. Comput. Sci.* 43 (1986), 169–188. https://doi.org/10.1016/0304-3975(86)90174-X

[18] Michael Kapralov and Kunal Talwar. 2013. On Differentially Private Low Rank Approximation. In *Proceedings of the Twenty-Fourth Annual ACM-SIAM Symposium on Discrete Algorithms* (New Orleans, Louisiana) *(SODA '13)*. Society for Industrial and Applied Mathematics, USA, 1395âĂŞ1414.

[19] V. A. Kazakov. 1986. Ising model on a dynamical planar random lattice: Exact solution. *Physics Letters A* 119 (1986), 140–144.

[20] Jonathan Leake, Colin McSwiggen, and Nisheeth K. Vishnoi. 2020. Sampling Matrices from Harish-Chandra–Itzykson–Zuber Densities with Applications to Quantum Inference and Differential Privacy. https://arxiv.org/abs/2011.05417.

[21] Jonathan Leake and Nisheeth K. Vishnoi. 2020. On the Computability of Continuous Maximum Entropy Distributions: Adjoint Orbits of Lie Groups. In *arXiv 2011.01851*.

[22] Jonathan Leake and Nisheeth K. Vishnoi. 2020. On the Computability of Continuous Maximum Entropy Distributions with Applications. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing* (Chicago, IL, USA) *(STOC 2020)*. Association for Computing Machinery, New York, NY, USA, 930âĂŞ943. https://doi.org/10.1145/3357713.3384302

[23] László Lovász and Santosh Vempala. 2006. Fast algorithms for logconcave functions: Sampling, rounding, integration and optimization. In *2006 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06)*. IEEE, 57–68.

[24] László Lovász and Santosh Vempala. 2006. Simulated annealing in convex bodies and an $O^*(n^4)$ volume algorithm. *J. Comput. System Sci.* 72, 2 (2006), 392–417.

[25] F. McSherry and K. Talwar. 2007. Mechanism Design via Differential Privacy. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07)*. 94–103. https://doi.org/10.1109/FOCS.2007.66

[26] C. McSwiggen. 2018. The Harish-Chandra integral: An introduction with examples. https://arxiv.org/abs/1806.11155.

[27] A. Morozov. 1992. Pair correlator in the Itzykson–Zuber integral. *Modern Physics Letters A* 7 (1992), 3503–3507. https://arxiv.org/abs/hep-th/9209074.

[28] M. E. Muller. 1959. A Note on a Method for Generating Points Uniformly on N-Dimensional Spheres. *Comm. Assoc. Comput. Mach.* 2 (1959), 19–20.

[29] Yu. A. Neretin. 2003. Rayleigh triangles and non-matrix interpolation of matrix beta integrals. *Sbornik: Mathematics* 194, 4 (April 2003), 515–540. https://doi.org/10.1070/sm2003v194n04abeh000727

[30] N. O'Connell. 2014. Whittaker functions and related stochastic processes. *MSRI Publications: Random Matrix Theory, Interacting Particle Systems and Integrable Systems* 65 (2014), 385–409.

[31] Victor Y. Pan and Zhao Q. Chen. 1999. The complexity of the matrix eigenproblem. In *Proceedings of the Thirty-First Annual ACM Symposium on Theory of Computing*. 507–516.

[32] A. Prats Ferrer, B. Eynard, P. Di Francesco, and J.-B. Zuber. 2007. Correlation functions of Harish-Chandra integrals over the orthogonal and the symplectic groups. *Journal of Statistical Physics* 129 (2007), 885–935. https://arxiv.org/abs/math-ph/0610049.

[33] S. L. Shatashvili. 1993. Correlation functions in the Itzykson–Zuber model. *Communications in Mathematical Physics* 154 (1993), 421–432. https://arxiv.org/abs/hep-th/9209083.

[34] Paul B. Slater. 1991. Relations between the barycentric and von Neumann entropies of a density matrix. *Physics Letters A* 159, 8 (1991), 411 – 414. https://doi.org/10.1016/0375-9601(91)90371-E

[35] Terrence Tao. 2013. *The Harish-Chandra-Itzykson-Zuber integral formula*. https://terrytao.wordpress.com/2013/02/08/the-harish-chandra-itzykson-zuber-integral-formula/.

[36] Terrence Tao. 2019. *Eigenvectors from eigenvalues*. https://terrytao.wordpress.com/2019/08/13/eigenvectors-from-eigenvalues/.

[37] Santosh Vempala. [n.d.]. Personal communication.

[38] J.-B. Zuber. 2020. On the minor problem and branching coefficients. https://arxiv.org/abs/2006.03006.