

Classical and Quantum Algorithms for Tensor Principal Component Analysis

Matthew B. Hastings^{1,2}

¹Station Q, Microsoft Research, Santa Barbara, CA 93106-6105, USA

²Microsoft Quantum and Microsoft Research, Redmond, WA 98052, USA

We present classical and quantum algorithms based on spectral methods for a problem in tensor principal component analysis. The quantum algorithm achieves a *quartic* speedup while using exponentially smaller space than the fastest classical spectral algorithm, and a super-polynomial speedup over classical algorithms that use only polynomial space. The classical algorithms that we present are related to, but slightly different from those presented recently in Ref. 1. In particular, we have an improved threshold for recovery and the algorithms we present work for both even and odd order tensors. These results suggest that large-scale inference problems are a promising future application for quantum computers.

Principal component analysis is a fundamental technique that finds applications in reducing the dimensionality of data and denoising. While an optimal choice of principal components for a matrix can be computed efficiently using linear algebra, the corresponding problem for tensors is much less well-understood. Ref. 2 introduced a simple statistical model for tensor principal component analysis, termed the “spiked tensor” problem, and this paper has lead to a large amount of follow-up research. The model consists of (see below for more precise definitions) randomly choosing some unknown “signal vector” $v_{\text{sig}} \in \mathbb{R}^N$; then, the p -th order tensor

$$T_0 = \lambda v_{\text{sig}}^{\otimes p} + G \quad (1)$$

is formed, where $G \in (\mathbb{R}^N)^{\otimes p}$ is noise chosen from some random distribution and where λ is some scalar representing a signal-to-noise ratio. One task called recovery is to infer v_{sig} (to some accuracy) given T_0 . A simpler task called detection is to distinguish the case $\lambda = 0$ from $\lambda = \bar{\lambda}$ for some $\bar{\lambda} > 0$, again just given T_0 .

Ref. 2 presented a variety of algorithms for this problem. Following the normalization of Ref. 1, the entries of G are chosen independently from a Gaussian distribution of zero mean and unit variance, with $|v_{\text{sig}}| = \sqrt{N}$. Then, information theoretically, it is possible to recover for λ much larger than $N^{1-p/2}$ [2, 3]. However, no polynomial time algorithm is known that achieves this performance. Rather, the two best known algorithms are spectral and sum-of-squares. Spectral algorithms were first suggested in Ref. 2. There, a matrix is formed from T_0 (if p is even, the matrix is $N^{p/2}$ -by- $N^{p/2}$, with its entries given by entries of T_0) and the leading eigenvector of the matrix is used to determine v_{sig} . For even p , this method works for λ much larger than $N^{-p/4}$, and a variant of it is conjectured to perform similarly for odd p . Methods based on the sum-of-squares also perform similarly to the spectral method. The sum-of-squares method[4] for this problem gives rise to a sequence of algorithms[5, 6], in which one can recover at λ smaller than $N^{-p/4}$ at the cost of runtime and space increasing exponentially in $\text{polylog}(N)N^{-p/4}/\lambda$. In Ref. 1, a sequence of spectral algorithms with similar performance was shown.

In this paper, we present another spectral algorithm for the problem. Our spectral algorithm for even p is closely related to that of Ref. 1 which we became aware of while this paper was in preparation, and we use the normalization in that paper. However, we make several changes. Our main technical results are the following. First, we prove an improved threshold for recovery for even p using our algorithm; this relies on a randomized method of recovering. Second, we provide a different algorithm for odd p with provable guarantees on while no guarantees were given in Ref. 1 for odd p . For both even and odd p , we have provable bounds on recovery for λ of order $N^{-p/4}$ (without any polylogarithmic factors) and we have a sequence of algorithms similar to that above for λ small compared to $N^{-p/4}$. Third, we give a quantum algorithm for our spectral method, achieving a quartic speedup and exponential reduction in space. This quantum algorithm involves two main ideas. The first uses phase estimation and amplitude amplification to obtain a quadratic speedup in computing the largest eigenvector. The second idea uses a chosen input state to obtain a further quadratic speedup, giving the overall quartic speedup.

We emphasize that the quantum speedup is quartic compared to classical spectral algorithms presented here and in previous work. We are not able to make an accurate comparison of the runtime to sum-of-squares methods. In part, given that the runtime of all of these algorithms increases exponentially in λ^{-1} , a change in prefactors in some estimates for threshold can give rise to polynomial changes in runtime. We expect that many of these estimates of thresholds are not tight (indeed, we expect that they are off by a polylogarithmic factor), and so either improved analytic methods or numerical simulations are needed to give an accurate comparison.

At a more heuristic level, we present a rather different motivation for our spectral algorithm compared to Ref. 1. Rather than being motivated by the so-called Kikuchi free energy, we instead are motivated by mean-field approximations to quantum many-body systems. We consider a system of a some number n_{bos} of qudits, each of dimension

N , and use the tensor T_0 to construct a quantum Hamiltonian on these qudits. Increasing n_{bos} gives rise to a similar sequence of algorithms as above, with increased runtime but improved performance: the required n_{bos} increases polynomially in λ^{-1} as $\text{polylog}(N)(N^{-p/4}/\lambda)^{4/(p-2)}$, but the runtime increases exponentially.

Restricting to the symmetric subspace, these n_{bos} qudits can be thought of as a system of bosons. In the case $p = 4$, for example, our Hamiltonian has pairwise interaction terms for all pairs of qudits. It is natural from the viewpoint of mean-field theory in physics then to expect that the leading eigenvector of the problem, for large n_{bos} , can be approximated by a product state. While the bounds for arbitrary pairwise Hamiltonians would require rather large n_{bos} for given N in order for such a mean-field approximation to be accurate[7–9], we will be able to prove that for the statistical model above the mean-field approximation becomes accurate with high probability at much smaller n_{bos} , depending upon the value of λ . In this mean-field regime, the product state is an n_{bos} -fold tensor product of a single particle state, and this single particle state is given by v_{sig} in an obvious way, regarding the vector v_{sig} as a vector in the single-particle Hilbert space. While we will not prove that this state is a good approximation to the leading eigenvector, it will be a good approximation to some state in an eigenspace with large eigenvalue. Then, the single particle density matrix allows one to infer v_{sig} (a similar matrix was used in Ref. 1 where it was termed a voting matrix).

Classically, implementing this spectral algorithm requires high-dimensional linear algebra, in particular finding the leading eigenvector of a matrix of dimension $\approx N^{n_{bos}}$. This makes it a natural candidate for a quantum algorithm. Since the Hamiltonian here is fairly simple, it can be simulated efficiently using standard techniques in the literature reviewed later. This allows us to give a simple algorithm based on preparing a random initial state and then phase estimating in an attempt to project onto the leading eigenvector. The probability of success in this projection is inverse in the dimension of the matrix, so this simple algorithm leads to no speedup over classical. However, we show that it is possible to apply amplitude amplification to give a quadratic speedup over classical. More surprisingly, we show that one can use the tensor T_0 to prepare an input state to the algorithm with improved overlap with the leading eigenvector, giving the quantum algorithm a quartic speedup over classical. Here, when comparing to classical we are considering classical algorithms based on the power method or similar algorithms such as Lanczos; these algorithms require exponential space while the quantum algorithm uses only polynomial space. We also consider classical algorithms based on ideas in Ref. 10 which use polynomial space but the quantum algorithm is super-polynomially faster than these algorithms. We also present some minor improvements to the quantum algorithm which may be useful in practice.

A. Definitions, Random Ensembles, and Notation

Let us make some formal definitions. A tensor T of order p and dimension N is a multi-dimensional array. The entries of the tensor are written $T_{\mu_1, \mu_2, \dots, \mu_p}$ where $p \geq 1$ is an integer and each μ_a ranges from $1, \dots, N$. Generalizing previous work on this problem, we consider two possible cases, one in which entries of a tensor are chosen to be real numbers, and one in which they may be complex numbers, so that either $T \in (\mathbb{R}^N)^{\otimes p}$ or $T \in (\mathbb{C}^N)^{\otimes p}$; we explain later the reason for this generalization; a tensor with all entries real will be called a real tensor. A symmetric tensor is one that is invariant under permutation of its indices. The symmetrization of a tensor is equal to $1/p!$ times the sum of tensors given by permuting indices.

The spiked tensor model for given N, p is defined as follows. Let v_{sig} be a vector in \mathbb{R}^N , normalized by $|v_{\text{sig}}| = \sqrt{N}$, chosen from some probability distribution; this is the “signal vector”. Let G be a real tensor of order p with entries chosen from a Gaussian distribution with vanishing mean. We let $T_0 = \lambda v_{\text{sig}}^{\otimes p} + G$ as above, where $v_{\text{sig}}^{\otimes p}$ is defined to be the tensor with entries

$$(v_{\text{sig}}^{\otimes p})_{\mu_1, \dots, \mu_p} = \prod_{a=1}^p (v_{\text{sig}})_{\mu_a}.$$

Here we use the notation that a subscript on a vector denotes an entry of that vector; we use a similar notation for matrices later.

Remark: some of the best sum-of-squares results are for a different distribution in which the entries of T_0 are chosen from a biased distribution on $\{-1, +1\}$, rather than for the Gaussian distribution. We expect that using that distribution would not affect the results here too much, but we avoid treating that case also for simplicity.

Since the tensor $v_{\text{sig}}^{\otimes p}$ is symmetric, of course it is natural to replace T_0 by its symmetrization. Indeed, no information can be lost by this replacement since given a tensor T_0 one can symmetrize the tensor, and then add back in Gaussian noise chosen to vanish under symmetrization to obtain a tensor drawn from the same distribution as T_0 was. That is, the cases in which G is symmetrized or not can be reduced to each other.

A generalization of this problem is the case in which G is chosen to have complex entries, with each entry having real and imaginary parts chosen from a Gaussian distribution with vanishing mean and variance $1/2$. We refer to

this as the complex ensemble, while we refer to the case where G has real entries as the real ensemble; the choice of reducing the variance to $1/2$ is a convenient normalization for later. It is clear that since v_{sig} is real, the case of complex G can be reduced to the real case (up to an overall rescaling for the different variance) simply by taking the real part of T_0 , and similarly the real case can be reduced to the complex case (again up to an overall rescaling) by adding Gaussian distributed imaginary terms to the entries of T_0 . We will see that for odd p , at least for reasons of analyzing the algorithms, it is convenient not to symmetrize T_0 and to take complex G , while for even p this is not necessary. It may be possible to avoid doing this for odd p (which may improve the detection and recovery threshold of the algorithm by constant factors) and we comment on this later.

We treat p as fixed in the asymptotic notation, but consider the dependence on N, n_{bos} . So, throughout this paper, when we refer to a polynomial in N , we mean a polynomial independent of the parameter n_{bos} . The polynomial may, however, depend on p , such as N^p .

We will assume that $n_{\text{bos}} = O(N^\theta)$ for some p -dependent constant $\theta > 0$ chosen sufficiently small; this is useful to simplify some of the statements of the results to avoid having to specify the allowed range of n_{bos} in each case. For example, we will say that a quantity such as n_{bos}^p/N is $o(1)$, meaning that we must take $\theta < 1/p$. We do not specify the exact value of θ but it can be deduced from the proofs if desired.

Similarly, we will also assume that λ is $\Omega(N^{-\theta'})$ for some p -dependent constant $\theta' > p/4$; again this helps simplify some of the statements of the results. Since we have assumed that $n_{\text{bos}} = O(N^\theta)$ and we will see that the required n_{bos} increases polynomially with λ^{-1} , this assumed lower bound on λ is not a further restriction on our results.

In big-O notation, we also always assume that $\lambda = O(N^{-p/4})$; of course, there is no reason to consider λ larger than this since simple spectral methods succeed if λ is $\omega(N^{-p/4})$.

We write $\mathbb{E}[\dots]$ to denote expectation values and $\Pr[\dots]$ to denote a probability. Usually these are expectation values or probabilities over choices of G , though in some cases we consider expectation values over other random variables. We use $\|\dots\|$ to denote the operator norm of an operator, i.e., the absolute value of the largest singular value. We use $|\dots|$ to denote either the ℓ_2 norm of a tensor or the ℓ_2 norm of a quantum state. All logarithms are natural logarithms.

We use $\langle \dots | \dots \rangle$ to denote inner products and use bra-ket notation both for vectors and for quantum mechanical states.

We will need to compute expectation values over random choices of G and also compute expectation values of certain operators in quantum states, such as $\langle \psi | O | \psi \rangle$ for some state ψ and operator O . We refer to the latter as a *quantum expectation value* of O in state ψ to distinguish it from an expectation value over random variables.

B. Outline

In section I, we review some results on recovery and boosting from Ref. 1 and present a randomized recovery procedure that will help in improving the recovery threshold. In section II, we give give spectral algorithms for the spiked tensor problem for the case of both even and odd p . In that section, we present algorithms in terms of eigenvalues and eigenvectors of a matrix (more precisely, vectors in some eigenspace and quantum expectation values of operators in those vectors) that we call a *Hamiltonian*. We leave the method to compute these eigenvalues and expectation values for later in section IV, where we give classical and quantum algorithms for this and give time bounds for those algorithms. In section III, we give some results on the spectra of random tensors needed for the analysis of these algorithms. A key idea here is reducing the case of a p -th order tensor for odd p to the case of a q -th order tensor for even $q = 2(p-1)$. One interesting corollary of this technique, see corollary 2, is that for odd p and for the minimal value of n_{bos} we are able remove a logarithmic factor in some of the bounds. An appendix A gives an introduction to some techniques used to evaluate expectation values of tensors networks whose tensors are chosen from a Gaussian distribution; these techniques are used earlier in the paper. In section V, we further discuss tensor networks and use this to consider limitations of certain algorithms. In section VI, we discuss some extensions of the results.

The proof of detection is in theorem 2 for the even p case and 4 for the odd p case. The proof of recovery is in theorem 3 for the even p case and theorem 5 for the odd p case. The runtime bound for the fastest quantum algorithm is in theorem 6. This theorem gives a quartic improvement in the runtime compared to the fastest classical spectral algorithm; more precisely the log of the runtime with the quantum algorithm divided by the log of the runtime of the classical algorithm approaches $1/4$ as $N \rightarrow \infty$ at fixed $N^{-p/4}/\lambda$.

I. RECOVERY

In this section we discuss recovery and define randomized procedures for recovery that will be useful in boosting the threshold. Following the notation of Ref. 1, define the *correlation* between two vectors by a normalized overlap

$$\text{corr}(x, y) = \frac{|\langle x | y \rangle|}{|x| \cdot |y|}. \quad (2)$$

The goal of an algorithm is to produce a vector x with large $\text{corr}(x, v_{\text{sig}})$. Note that we take the absolute value in the correlation, ignoring the sign. For even p , the sign of v_{sig} is irrelevant, while for odd p it is easy, given a guess of v_{sig} up to sign, to try both choices of sign and see which is most likely.

Strong recovery means that $\text{corr}(x, v_{\text{sig}}) = 1 - o(1)$. Proposition 2.6 of Ref. 1, which is noted in that reference as being implicit in Ref. 2, shows how to “boost” a weaker correlation to strong recovery. It is shown that given a vector u one can apply a single iteration of the tensor power algorithm to obtain a new vector x such that, with high probability,

$$\text{corr}(x, v_{\text{sig}}) \geq 1 - c\lambda^{-1}\text{corr}(u, v_{\text{sig}})^{1-p}N^{(1-p)/2}, \quad (3)$$

where c is a constant depending on p . So, for any $\lambda\omega(N^{(1-p)/2})$, if $\text{corr}(u, v_{\text{sig}}) = \Omega(1)$, we have $\text{corr}(x, v_{\text{sig}}) = 1 - o(1)$.

Thus, it suffices to construct which outputs some vector u which has, with high probability, $\text{corr}(u, v_{\text{sig}}) = \Omega(1)$. This is termed *weak recovery*; indeed, one can be satisfied with even weaker assumptions depending on the value of λ ; for λ close to $N^{-p/4}$, one may have polynomially small $\text{corr}(u, v_{\text{sig}})$.

The spectral algorithms that we construct later will output a matrix that we will write ρ_1 . This matrix will be a positive semi-definite Hermitian matrix with trace 1. In physics terms, such a matrix is termed a *density matrix*. For sufficiently large λ , the leading eigenvector of the matrix will have a large overlap with v_{sig} . However, for smaller λ , we will still have some lower bound that, with high probability, $\langle v_{\text{sig}} | \rho_1 | v_{\text{sig}} \rangle = \Omega(1)|v_{\text{sig}}|^2$. The randomized algorithm 1 below shows how to use this to obtain weak recovery. This randomized algorithm allows us to improve the threshold over the recovery method[1] of simply using the leading eigenvector ρ_1 since it works even in some cases where the leading eigenvector has small correlation with v_{sig} .

Putting these results together we find that

Corollary 1. *Given an algorithm that, with high probability, outputs a density matrix ρ with*

$$\frac{\langle v_{\text{sig}} | \rho_1 | v_{\text{sig}} \rangle}{N} = \Omega(1),$$

then in polynomial time, with high probability, strong recovery is possible.

We will simply say that such an algorithm achieves recovery.

We present the algorithm for the case that the matrix ρ_1 may be complex; if the matrix is real, one can instead sample from a real Gaussian distribution and the proof of lemma 1 goes through will slightly different constants.

Algorithm 1 Input: density matrix ρ . Output, some vector w obeying bounds described above

1. Randomly sample a vector u with entries chosen from a correlated complex Gaussian distribution with zero mean and with covariance $\mathbb{E}[\bar{u}_i u_j] = (\rho)_{ij}$, with $\mathbb{E}[u_i u_j] = \mathbb{E}[\bar{u}_i \bar{u}_j] = 0$.
 2. Let $w = u/|u|$.
-

We have

Lemma 1. *For algorithm 1, with probability at least $1/2$,*

$$|\langle w | v_{\text{sig}} \rangle| \geq c' \sqrt{\langle v_{\text{sig}} | \rho | v_{\text{sig}} \rangle}, \quad (4)$$

for some scalar $c' > 0$.

Proof. We have $\mathbb{E}[|u|^2] = \text{tr}(\rho) = 1$. Hence, with probability at least $3/4$ we have that $|u|^2 \leq 4/3$. We have $\mathbb{E}[|\langle u | v_{\text{sig}} \rangle|^2] = \langle v_{\text{sig}} | \rho | v_{\text{sig}} \rangle$ and since $\langle u | v_{\text{sig}} \rangle$ is a Gaussian random variable with mean 0, with probability at least $3/4$ its absolute value is at least some positive constant c'' (the exact constant can be deduced from the error function) times its standard deviation. Hence, the lemma follows for $c' = (3/4)c''$. \square

II. SPECTRAL ALGORITHM

We now give the spectral algorithms for the spiked tensor problem for the case of both even and odd p . In subsection II A, we define a Hamiltonian $H(T)$ given an arbitrary tensor T . Then, in subsection II B, we present the spectral algorithm in terms of $H(T_0)$.

For even p , the Hamiltonian that we present is very similar to the matrix Y given in Ref. 1 but it has some minor differences. In our language (explained below), this matrix Y is obtained by projecting our Hamiltonian of Eq. (5) into the subspace of the “symmetric subspace” spanned by $|\mu_1\rangle \times |\mu_2\rangle \otimes \dots \otimes |\mu_{n_{bos}}\rangle$ with $\mu_1, \dots, \mu_{n_{bos}}$ all distinct. The relative reduction in the size of the matrix is only $O(1/N)$ in the limit of large N .

Also, in our method, we have an $O(N)$ rotational symmetry of the basis which is very useful in analysis, for example showing that the eigenvalues of $H(\lambda v_{\text{sig}}^{\otimes p})$ are independent of choice of v_{sig} . For the matrix Y of 1, this is not obvious to us and we do not fully understand the claimed behavior of the largest eigenvalue in that case. We will use a different notation, using creation and annihilation operators, which will help make this rotational symmetry more explicit.

For odd p , the Hamiltonian that we use is unrelated to that of Ref. 1.

A. Hamiltonian Definition

For even p , given a tensor T we define an linear operator $H(T)$ that we call a *Hamiltonian* as follows. This Hamiltonian is a linear operator on a vector space $(\mathbb{R}^N)^{\otimes n_{bos}}$ or $(\mathbb{C}^N)^{\otimes n_{bos}}$, for integer $n_{bos} \geq 1$ chosen below. We write basis elements of this space as $|\mu_1\rangle \times |\mu_2\rangle \otimes \dots \otimes |\mu_{n_{bos}}\rangle$, and we call this space the *full Hilbert space*. We define

$$H(T) = \frac{1}{2} \sum_{i_1, \dots, i_{p/2}} \left(\sum_{\mu_1, \dots, \mu_p} T_{\mu_1, \mu_2, \dots, \mu_p} |\mu_1\rangle_{i_1} \langle \mu_1 + p/2| \otimes |\mu_2\rangle_{i_2} \langle \mu_2 + p/2| \otimes \dots \otimes |\mu_{p/2}\rangle_{i_{p/2}} \langle \mu_p| + \text{h.c.} \right), \quad (5)$$

where the sum is over distinct $i_1, i_2, \dots, i_{p/2}$ so that there are $(p/2!)(\binom{n_{bos}}{p/2})$ terms in the sum and where h.c. means adding the Hermitian conjugate of the given terms, so that $H(T)$ is Hermitian and where $|\mu\rangle_i \langle \nu|$ denotes the operator $|\mu\rangle \langle \nu|$ on qudit i . We require that $n_{bos} \geq p/2$ or else $H(T)$ is trivially zero.

Note of course that if T is real and symmetric, then the term $\sum_{\mu_1, \dots, \mu_p} T_{\mu_1, \mu_2, \dots, \mu_p} |\mu_1\rangle_{i_1} \langle \mu_1 + p/2| \otimes |\mu_2\rangle_{i_2} \langle \mu_2 + p/2| \otimes \dots \otimes |\mu_{p/2}\rangle_{i_{p/2}} \langle \mu_p|$ is already Hermitian. $H(T)$ can be regarded as a Hamiltonian acting on a space of n_{bos} qudits, each of dimension N , and with interaction between sets of $p/2$ particles at a time.

Even if T is not symmetrized, $H(T)$ is unchanged if one applies an arbitrary permutation to the first $p/2$ indices of T and applies the same permutation to the last $p/2$ indices of T .

We may restrict to the symmetric subspace of this Hilbert space. We write $D(N, n_{bos})$ to indicate the dimension of this subspace. For $N \gg n_{bos}$, we can approximate $D(N, n_{bos}) \approx N^{n_{bos}}/n_{bos}!$.

Within the symmetric subspace, we can write this Hamiltonian in a so-called “second-quantized” form:

$$H(T) = \frac{1}{2} \left(\sum_{\mu_1, \dots, \mu_p} T_{\mu_1, \mu_2, \dots, \mu_p} \left(\prod_{i=1}^{p/2} a_{\mu_i}^\dagger \right) \left(\prod_{i=p/2+1}^p a_{\mu_i} \right) + \text{h.c.} \right). \quad (6)$$

This replacement by a second-quantized Hamiltonian is simply a convenient notation. The operators a_μ^\dagger, a_μ are bosonic creation and annihilation operators, obeying canonical commutation relations $[a_\mu, a_\nu^\dagger] = \delta_{\mu, \nu}$. We restrict to the subspace with a total of n_{bos} bosons, i.e., we define the number operator n by

$$n \equiv \sum_{\mu} a_\mu^\dagger a_\mu, \quad (7)$$

and restrict to $n = n_{bos}$. An orthonormal basis of states for this symmetric subspace is given by all states equal to some normalization constant multiplying $a_{\mu_1}^\dagger a_{\mu_2}^\dagger \dots a_{\mu_{n_{bos}}}^\dagger |0\rangle$, where $|0\rangle$ is the *vacuum* state (i.e., the state annihilated by a_μ for all μ), and where $\mu_1 \leq \mu_2 \leq \dots \leq \mu_{n_{bos}}$ is some sequence.

The second quantized Hamiltonian for the symmetric subspace is unchanged under arbitrary permutation of the first $p/2$ indices of T and arbitrary (not necessarily the same) permutation of the last $p/2$ indices of T .

For odd p , we define the Hamiltonian $H(T)$ as follows. Given a tensor T of odd order p , define a new tensor \tilde{T} of even order $q = 2(p-1)$ with components

$$\tilde{T}_{\mu_1, \dots, \mu_{(p-1)/2}, \nu_1, \dots, \nu_{(p-1)/2}, \mu_{(p-1)/2+1}, \dots, \mu_{p-1}, \nu_{(p-1)/2}, \dots, \nu_{p-1}} = \sum_{\sigma} T_{\mu_1, \dots, \mu_{p-1}, \sigma} T_{\nu_1, \dots, \nu_{p-1}, \sigma}. \quad (8)$$

Then define $H(T) = H(\tilde{T})$, using the definition (5) for $H(\tilde{T})$. Note the order of indices on the left-hand side of Eq. (8). Using the second-quantized notation, this gives for odd p :

$$H(T) = \frac{1}{2} \left(\sum_{\mu_1, \dots, \mu_{p-1}} \sum_{\nu_1, \dots, \nu_{p-1}} \sum_{\sigma} T_{\mu_1, \mu_2, \dots, \mu_{p-1}, \sigma} T_{\nu_1, \nu_2, \dots, \nu_{p-1}, \sigma} \left(\prod_{i=1}^{(p-1)/2} a_{\mu_i}^\dagger a_{\nu_i}^\dagger \right) \left(\prod_{i=(p-1)/2+1}^{p-1} a_{\mu_i} a_{\nu_i} \right) + \text{h.c.} \right), \quad (9)$$

Now we require that $n_{bos} \geq p-1$ as otherwise $H(T)$ is trivially zero. For this Hamiltonian, it is convenient to take G from the complex ensemble because, as we explain more below, it makes $\mathbb{E}[H(G)]$ equal to zero, as well as canceling out certain terms in higher order moments, making the proof of the spectral properties of $H(G)$ simpler. We discuss later to what extent we can avoid using the complex ensemble.

B. Spectral Algorithms

The spectral algorithm for detection and recovery is algorithm 2. In this subsection we prove correctness of this algorithm, using statistical properties of $H(G)$ proven later.

This algorithm uses quantities E_0 and E_{max} defined later; roughly E_{max} is an upper bound on the eigenvalues of $H(G)$ and E_0 is the largest eigenvalue of $H(\lambda v^{\otimes p})$. The algorithm can then achieve detection by verifying that the largest eigenvalue is significantly larger than E_{max} , which occurs when E_0 is large enough. Indeed, we will see that it suffices to have $E_0 = (1+c)E_{max}$ for any *fixed* $c > 0$ (some results can be extended to the case that c decays slowly with N but we omit this for brevity). This fixes the scaling of n_{bos} with λ so that we need (up to polylogarithmic factors) $n_{bos} \gtrsim (N^{-p/4}/\lambda)^{4/(p-2)}$.

One interesting feature of the algorithm is that in step 3, we compute the density matrix of the leading eigenvector or of any vector in the eigenspace of eigenvalue $\geq E_{cut}$, for $E_{cut} = (E_0 + E_{max})/2$ defined in the algorithm. This might seem surprising, given that the leading eigenvector is computed in step 1; one might wonder why some other vector should be taken. We describe the algorithm in this way since, in later classical and quantum algorithms that we give to compute the spectral properties of the matrix, we might not extract the leading eigenvector but instead extract only some vector in this eigenspace due to use of the power method in a classical algorithm or due to approximations in phase estimation in a quantum algorithm. Thus, much of our analysis is given to showing that *some* eigenvalue larger than E_{cut} exists by lower bounding the leading eigenvalue λ_1 , but given that some such eigenvalue exists, we do not worry too much about exactly what mixture of eigenvectors in the given eigenspace we compute.

Algorithm 2 Spectral algorithm. This algorithm takes a tensor T_0 as input and also a scalar $\bar{\lambda}$ and an integer n_{bos} . The output is a decision about whether $\lambda = \bar{\lambda}$ or $\lambda = 0$, and, if the algorithm reports that $\lambda = \bar{\lambda}$, it also returns an approximation of v_{sig} (up to an overall sign). The quantity n_{bos} is chosen depending upon the value of $\bar{\lambda}$; smaller values of λ require larger values of n_{bos} in order for E_0 to be sufficiently larger than E_{max} for the algorithm to be accurate. See theorems 2,4,3,5. For $E_0 \geq (1+c)E_{max}$ for any $c > 0$, the algorithm achieves recovery.

1. Compute the eigenvector of $H(T_0)$ and the leading eigenvalue, denoted λ_1 .
2. (Detection) If

$$\lambda_1 > E_{cut} \equiv (E_0 + E_{max})/2,$$

where $E_0 = \bar{\lambda}(p/2)! \binom{n_{bos}}{p/2} N^{p/2}$ for even p , and $E_0 = \bar{\lambda}^2(p-1)! \binom{n_{bos}}{p-1} N^p$ for odd p , and where E_{max} is defined in theorem 1, then report that $\lambda = \bar{\lambda}$. Otherwise report $\lambda = 0$.

3. (Recovery) Compute the single particle density matrix (defined below) of the leading eigenvector or of any vector in the eigenspace of eigenvalue $\geq E_{cut}$. Apply algorithm 1 to recover an approximation to v_{sig} .
-

In section III, we consider the largest eigenvalue λ_1 of $H(G)$ and show the following theorem which summarizes the results of lemmas 5,7. Roughly, up to prefactors, the result for odd p is given by considering the result for a q -th order tensor for even $q = 2(p-1)$ and then multiplying the largest eigenvalue by a factor of \sqrt{N} .

Theorem 1. *Let λ_1 be the largest eigenvalue of G . For even p let*

$$E_{max} = \sqrt{2J \log(N)} n_{bos}^{p/4+1/2} N^{p/4},$$

and for odd p let

$$E_{max} = 2\sqrt{J \log(N)} n_{bos}^{p/2} N^{p/2},$$

where J is a scalar depends that implicitly on p, n_{bos}, N and tends to some function depending only on p for large n_{bos}, N . More precisely, for even p , J is equal to $(p/2)! \binom{n_{bos}}{p/2} / n_{bos}^{p/2} + o(1)$ for the real ensemble and is twice that for the complex ensemble, and for odd p , J is equal to that for the even case for $2(p-1)$.

Then, for any x ,

$$\Pr[\lambda_1 \geq x] \leq \exp\left(-\frac{x - E_{max}}{\xi}\right), \quad (10)$$

with for even p

$$\xi = \frac{\sqrt{J} n_{bos}^{p/4-1/2} N^{p/4}}{\sqrt{2 \log(N)}} \quad (11)$$

and for odd p

$$\xi = \frac{\sqrt{J} n_{bos}^{p/2-1} N^{p/2}}{\sqrt{\log(N)}}. \quad (12)$$

So, for any E' which is $\omega(\xi)$, with high probability $\lambda_1 \leq E_{max} + E'$.

Now consider the eigenvectors and eigenvalues of $H(T_0)$. For any symmetric tensor T of order n_{bos} , let $|T\rangle$ be the vector on n_{bos} qudits (each of dimension N) with amplitudes given by the entries of the tensor in the obvious way:

$$|T\rangle = \sum_{\mu_1, \dots, \mu_{n_{bos}}} T_{\mu_1, \dots, \mu_{n_{bos}}} |\mu_1\rangle \otimes \dots \otimes |\mu_{n_{bos}}\rangle.$$

This vector is only normalized if $|T| = 1$. So, $\Psi_{\text{sig}} \equiv N^{-n_{bos}/2} |v_{\text{sig}}^{\otimes n_{bos}}\rangle$ is a normalized vector.

We have the following simple property:

Lemma 2. *Let λ_1 be the largest eigenvalues of $H(T_0)$. Then, $\lambda_1 \geq E \equiv \langle \Psi_{\text{sig}} | H(T) | \Psi_{\text{sig}} \rangle$.*

Proof. Immediate from the variational principle. □

1. Even p Case

We now show correctness of the algorithm. All results in this subsection refer to even p even if we do not state it explicitly. First we estimate E of lemma 2 to show detection. Then, we show recovery.

We have

$$E = \langle \Psi_{\text{sig}} | H(\lambda v_{\text{sig}}^{\otimes p}) | \Psi_{\text{sig}} \rangle + \langle \Psi_{\text{sig}} | H(G) | \Psi_{\text{sig}} \rangle \quad (13)$$

$$= E_0 + \langle \Psi_{\text{sig}} | H(G) | \Psi_{\text{sig}} \rangle, \quad (14)$$

where

$$E_0 = \lambda (p/2)! \binom{n_{bos}}{p/2} N^{p/2}. \quad (15)$$

To evaluate $\langle \Psi_{\text{sig}} | H(G) | \Psi_{\text{sig}} \rangle$ it is convenient to exploit a rotational invariance of this problem. We can apply a rotation using any matrix O in the orthogonal group $O(N)$, rotating the creation and annihilation operators by making the replacement

$$a_\mu^\dagger \rightarrow \sum_\nu O_{\mu\nu} a_\nu^\dagger$$

and

$$a_\mu \rightarrow \sum_\nu O_{\mu\nu} a_\nu.$$

This rotation preserves the canonical commutation relations and is equivalent to rotating the basis states on each qudit by O . To preserve the Hamiltonian $H(T_0)$, we rotate each leg of the tensor T_0 :

$$(T_0)_{\mu_1, \dots, \mu_p} \rightarrow \sum_{\nu_1, \dots, \nu_p} (T_0)_{\nu_1, \dots, \nu_p} \prod_i O_{\mu_i, \nu_i}.$$

This rotation preserves the Gaussian measure on G but changes v_{sig} . So, we can rotate so that v_{sig} is some fixed vector, say $(\sqrt{N}, 0, 0, \dots)$ so $(v_{\text{sig}})_1 = \sqrt{N}$. Then $\langle \Psi_{\text{sig}} | H(G) | \Psi_{\text{sig}} \rangle$ is equal to $(p/2)! \binom{n_{\text{bos}}}{p/2}$ multiplied by a single entry of G , i.e., the entries with all indices equal to 1, which is some quantity chosen from a Gaussian distribution with zero mean and unit variance. So, with high probability, $E = \lambda(p/2)! \binom{n_{\text{bos}}}{p/2} N^{p/2} + n_{\text{bos}}^{p/2} O(N^\eta)$ for any $\eta > 0$.

Hence,

Theorem 2. *If $\lambda(p/2)! \binom{n_{\text{bos}}}{p/2} N^{p/2} - E_{\text{max}} = \omega\left(\frac{\sqrt{J} n_{\text{bos}}^{p/4 - 1/2} N^{p/4}}{\sqrt{2 \log(N)}}\right)$, then with high probability algorithm 2 correctly determines whether $\lambda = 0$ or $\lambda = \bar{\lambda}$.*

Proof. This follows from the estimate of E which lowers bounds the largest eigenvalue when $\lambda = \bar{\lambda}$ and from theorem 1 which upper bounds $\|H(G)\|$. \square

Given any state, define the *single particle density matrix* ρ_1 in the basis for the full Hilbert space used in Eq. (5) to be the reduced density matrix on any of the qudits. Equivalently, this single particle density matrix can be expressed in terms of creation and annihilation operators by

$$(\rho_1)_{\mu\nu} = \frac{1}{n_{\text{bos}}} a_\mu^\dagger a_\nu. \quad (16)$$

Note that ρ_1 is positive semi-definite, Hermitian, and trace 1.

We have

Theorem 3. *Given any vector Ψ such that $\langle \Psi | H(T_0) | \Psi \rangle \geq (1 + c') E_{\text{max}}$ for any scalar $c' > 0$, then with high probability the corresponding single particle density matrix ρ_1 obeys*

$$\frac{\langle v_{\text{sig}} | \rho_1 | v_{\text{sig}} \rangle}{N} \geq (c' - o(1)) \frac{E_{\text{max}}}{E_0}. \quad (17)$$

In particular, for $\langle \Psi | H(T_0) | \Psi \rangle \geq E_{\text{cut}}$ with $E_0 \geq (1 + c) E_{\text{max}}$ we have $c' \geq c/2$ so with high probability

$$\frac{\langle v_{\text{sig}} | \rho_1 | v_{\text{sig}} \rangle}{N} \geq \frac{1}{2} \frac{c}{1 + c}. \quad (18)$$

Hence, algorithm 2 achieves recovery.

Proof. We have $\langle \Psi | H(\lambda v_{\text{sig}}^{\otimes p}) | \Psi \rangle + \langle \Psi | H(G) | \Psi \rangle \geq (1 + c) E_{\text{max}}$. By theorem 1, with high probability $\langle \Psi | H(G) | \Psi \rangle \leq (1 + o(1)) E_{\text{max}}$. Hence, with high probability $\langle \Psi | H(\lambda v_{\text{sig}}^{\otimes p}) | \Psi \rangle \geq (c - o(1)) E_{\text{max}}$.

Rotate so that $v_{\text{sig}} = (\sqrt{N}, 0, 0, \dots)$. Then, the Hamiltonian $H(\lambda v_{\text{sig}}^{\otimes p})$ is diagonal in the computational basis for the full Hilbert space. Let $P(m)$ be the probability, for state Ψ , that exactly m out of n_{bos} qudits are in state $|1\rangle$. Then, $\langle \Psi | H(\lambda v_{\text{sig}}^{\otimes p}) | \Psi \rangle = \sum_m \lambda(p/2)! \binom{m}{p/2} N^{p/2} P(m) \leq \sum_m m (E_0/n_{\text{bos}}) P(m) = E_0 \langle v_{\text{sig}} | \rho_1 | v_{\text{sig}} \rangle / N$. \square

Remark: more generally, the same result holds for any mixed state: given any mixed state σ such that $\text{tr}(\sigma H(T_0)) \geq (1 + c) E_{\text{max}}$ for any scalar $c > 0$, then with high probability the corresponding single particle density matrix ρ_1 obeys Eq. (17).

2. Odd p Case

We now show correctness of the algorithm for odd p . All results in this subsection refer to odd p even if we do not state it explicitly.

Let us first estimate E to show detection. We have $E = \langle \Psi_{\text{sig}} | H(T_0) | \Psi_{\text{sig}} \rangle$, but now $H(T_0)$ is a quadratic function of T_0 so there are some cross-terms. We have

$$\langle \Psi_{\text{sig}} | H(v_{\text{sig}}^{\otimes p}) | \Psi_{\text{sig}} \rangle = \lambda^2 (p-1)! \binom{n_{\text{bos}}}{p-1} N^p \equiv E_0. \quad (19)$$

The cross-term is

$$\lambda \langle \Psi_{\text{sig}} | \sum_{\mu_1, \dots, \mu_{p-1}} \sum_{\nu_1, \dots, \nu_{p-1}} \sum_{\sigma} (v_{\text{sig}}^{\otimes p})_{\mu_1, \mu_2, \dots, \mu_{p-1}, \sigma} G_{\nu_1, \nu_2, \dots, \nu_{p-1}, \sigma} \left(\prod_{i=1}^{(p-1)/2} a_{\mu_i}^{\dagger} a_{\nu_i}^{\dagger} \right) \left(\prod_{i=(p-1)/2+1}^{p-1} a_{\mu_i} a_{\nu_i} \right) + \text{h.c.} | \Psi_{\text{sig}} \rangle.$$

Exploiting the same rotational invariance as in the even p case, this is equal to $2\lambda(p-1)!(\binom{n_{\text{bos}}}{p-1})N^{p/2}$ multiplied by the real part of a single entry of G , i.e., the entry with all indices equal to 1. So, with high probability, this cross-term is bounded by $\lambda n_{\text{bos}}^{p-1}O(N^{p/2+\eta})$ for any $\eta > 0$.

The term quadratic in G is

$$\frac{1}{2} \langle \Psi_{\text{sig}} | \sum_{\mu_1, \dots, \mu_{p-1}} \sum_{\nu_1, \dots, \nu_{p-1}} \sum_{\sigma} G_{\mu_1, \mu_2, \dots, \mu_{p-1}, \sigma} G_{\nu_1, \nu_2, \dots, \nu_{p-1}, \sigma} \left(\prod_{i=1}^{(p-1)/2} a_{\mu_i}^{\dagger} a_{\nu_i}^{\dagger} \right) \left(\prod_{i=(p-1)/2+1}^{p-1} a_{\mu_i} a_{\nu_i} \right) + \text{h.c.} | \Psi_{\text{sig}} \rangle.$$

Exploiting the same rotational invariance as before, fixing $v_{\text{sig}} = (\sqrt{N}, 0, 0, \dots)$, we must have all $\mu_i, \nu_i = 1$. So, this is a sum of squares of N different entries of G , corresponding to N possible choices of σ ; since we use the complex ensemble, this has vanishing mean. So, with high probability this term is $n_{\text{bos}}^{p-1}O(N^{1/2+\eta})$ for any $\eta > 0$.

So, with high probability,

$$E \geq E_0 - \lambda n_{\text{bos}}^{p-1}O(N^{p/2+\eta}) - n_{\text{bos}}^{p-1}O(N^{1/2+\eta}) \quad (20)$$

for any $\eta > 0$.

Hence,

Theorem 4. *If $\lambda^2(p-1)!(\binom{n_{\text{bos}}}{p-1})N^p - E_{\max} = \omega\left(\frac{\sqrt{J}n_{\text{bos}}^{p/2-1}N^{p/2}}{\sqrt{\log(N)}}\right)$, then with high probability algorithm 2 correctly determines whether $\lambda = 0$ or $\lambda = \bar{\lambda}$.*

Proof. This follows from the estimate of E in Eq. (20) which lowers bounds the largest eigenvalue when $\lambda = \bar{\lambda}$ and from theorem 1 which upper bounds $\|H(G)\|$. Note that for $\lambda = O(N^{-p/4})$, the terms $-\lambda n_{\text{bos}}^{p-1}O(N^{p/2+\eta}) - n_{\text{bos}}^{p-1}O(N^{1/2+\eta})$ on the right-hand side of this equation are negligible as they are $o\left(\frac{\sqrt{J}n_{\text{bos}}^{p/2-1}N^{p/2}}{\sqrt{\log(N)}}\right)$. \square

We now consider recovery. We will first give a more general bound on cross-terms that will be useful. We have

$$H(T_0) = H(\lambda v_{\text{sig}}^{\otimes p}) + H(G) + \text{cross terms.}$$

Let us first bound the cross-terms.

Lemma 3. *With high probability, the operator norm of the cross-terms is bounded by $\lambda N^{p/2}O(N^{(p-1)/4})n_{\text{bos}}^{p-1}$.*

Proof. The cross-terms are equal to $2H(T_{\text{cross}})$ where T_{cross} is a tensor for even $q = 2(p-1)$ which has components

$$(T_{\text{cross}})_{\mu_1, \dots, \mu_{(p-1)/2}, \nu_1, \dots, \nu_{(p-1)/2}, \mu_{(p-1)/2+1}, \dots, \mu_p, \nu_{(p-1)/2+1}, \dots, \nu_p} = \lambda \sum_{\sigma} T_{\mu_1, \dots, \mu_{p-1}, \sigma} \cdot \left(\prod_{i=1}^{p-1} (v_{\text{sig}})_{\nu_i} \right) \cdot (v_{\text{sig}})_{\sigma}. \quad (21)$$

Rotating so that $v_{\text{sig}} = (\sqrt{N}, 0, 0, \dots)$, we have

$$(T_{\text{cross}})_{\mu_1, \dots, \mu_{(p-1)/2}, \nu_1, \dots, \nu_{(p-1)/2}, \mu_{(p-1)/2+1}, \dots, \mu_p, \nu_{(p-1)/2+1}, \dots, \nu_p} = \lambda N^{p/2} \sum_{\sigma} T_{\mu_1, \dots, \mu_{p-1}, 1} \prod_{i=1}^{p-1} \delta_{\nu_i, 1}. \quad (22)$$

Clearly, $\|H(T_{\text{cross}})\| \leq n_{\text{bos}}^{p-1}\|M_{\text{cross}}\|$, where M_{cross} is an N^{p-1} -by- N^{p-1} matrix, whose entries are determined by the entries of T_{cross} in an obvious way so that the first $p-1$ indices of T_{cross} determine a column index in M and the last $p-1$ indices of T_{cross} determine a row index of M . Regard $T_{\mu_1, \dots, \mu_{p-1}, 1}$ as being the entries of some tensor of order $p-1$ and let M' be the $N^{(p-1)/2}$ -by- $N^{(p-1)/2}$ matrix whose entries are determined by the entries of this tensor, again in the obvious way so that the first $(p-1)/2$ indices of the tensor determine a column index and the last $(p-1)/2$ indices determine a row index.

Then,

$$\|M_{\text{cross}}\| = \lambda N^{p/2}\|M'\|. \quad (23)$$

However, since the entries of M' are independent Gaussian random variables, it follows from standard random matrix theory results that with high probability $\|M'\| = O(N^{(p-1)/4})$. Remark: the dependence on n_{bos} is not tight here. \square

So as in the even case, we have

Theorem 5. *Given any vector Ψ such that $\langle \Psi | H(T_0) | \Psi \rangle \geq (1 + c')E_{max}$ for any scalar $c' > 0$, then with high probability the corresponding single particle density matrix ρ_1 obeys*

$$\frac{\langle v_{sig} | \rho_1 | v_{sig} \rangle}{N} \geq (c' - o(1)) \frac{E_{max}}{E_0}. \quad (24)$$

In particular, for $\langle \Psi | H(T_0) | \Psi \rangle \geq E_{cut}$ with $E_0 \geq (1 + c)E_{max}$ we have $c' \geq c/2$ so with high probability

$$\frac{\langle v_{sig} | \rho_1 | v_{sig} \rangle}{N} \geq \frac{1}{2} \frac{c}{1 + c}. \quad (25)$$

Hence, algorithm 2 achieves recovery.

Proof. We use lemma 3 to bound $\|H(T_{cross})\|$. This bound is asymptotically negligible compared to E_{max} . So, we have $\langle \Psi | H(\lambda v_{sig}^{\otimes p} | \Psi) + \langle \Psi | H(G) | \Psi \rangle \geq (1 + c - o(1))E_{max}$ where the $o(1)$ denotes the $o(1)$ contribution from the cross-terms.

Then, the rest of the proof is the same as in the even case, except for a replacement of $p/2$ by $p - 1$. In detail: by theorem 1, with high probability $\langle \Psi | H(G) | \Psi \rangle \leq (1 + o(1))E_{max}$. Hence, $\langle \Psi | H(\lambda v_{sig}^{\otimes p} | \Psi) \geq (c - o(1))E_{max}$.

Rotate so that $v_{sig} = (\sqrt{N}, 0, 0, \dots)$. Then, the Hamiltonian $H(\lambda v_{sig}^{\otimes p})$ is diagonal in the computational basis for the full Hilbert space. Let $P(m)$ be the probability, for state Ψ , that exactly m out of n_{bos} qudits are in state $|1\rangle$. Then, $\langle \Psi | H(\lambda v_{sig}^{\otimes p}) | \Psi \rangle = \sum_m \lambda^2 (p-1)! \binom{m}{p-1} N^p P(m) \leq \sum_m m (E_0/n_{bos}) P(m) = E_0 \langle v_{sig} | \rho_1 | v_{sig} \rangle / N$. \square

III. SPECTRUM OF RANDOM HAMILTONIAN

In this section, we will estimate the eigenvalues of $H(G)$. We consider first the case of even p . Here our proof is very similar to the of Ref. 1, though the method here also suggests some heuristics that may lead to a tighter bound in the future. Then, we consider the case of odd p by reducing it to the case of even p .

A. Even p

We first consider the case of even p . Let $Z(\tau, p, N, n_{bos})$ denote $\mathbb{E}[\text{tr}(\exp\{\tau H(G)\})]$ for a tensor G of rank p , with entries of G chosen from the Gaussian distribution, with given N, n_{bos} , for some real scalar τ . In this subsection, G may be symmetrized or not, and may be chosen from either the real or complex ensemble.

The main result in this section is the following lemma:

Lemma 4. *For each ensemble, real or complex, symmetrized or not, we have*

$$Z(\tau, p, N, n_{bos}) \leq D(N, n_{bos}) \exp(\tau^2 (J/2) n_{bos}^{p/2} N^{p/2}), \quad (26)$$

where J is a scalar depends that implicitly on p, n_{bos}, N and tends to some function depending only on p for large n_{bos}, N . More precisely, J is equal to $(p/2)! \binom{n_{bos}}{p/2} / n_{bos}^{p/2} + o(1)$ for the real ensemble and is twice that for the complex ensemble.

Proof. We first give a brief derivation of Eq. (28) below which is a standard result using quantum field theory techniques. Note that $\tau H(G) = H(\tau G)$ and τG is a tensor with entries chosen from a Gaussian distribution with zero mean and variance τ^2 . Hence for any $\tau' > \tau$ we have

$$Z(\tau', p, N, n_{bos}) = \mathbb{E}_{G, \eta}[\text{tr}(\exp\{H(\tau G + \eta)\})], \quad (27)$$

where $\mathbb{E}_{G, \eta}[\dots]$ denotes the expectation value over G and η , with the tensor η having Gaussian entries with zero mean and variance $(\tau')^2 - \tau^2$. Taking the expectation value over η , for $\tau'^2 - \tau^2$ small, we need to keep only the zeroth and second order terms on the right-hand side. So, we find that

$$\partial_{\tau'}^2 Z(\tau, p, N, n_{bos}) = \int_0^1 ds_2 \int_0^{s_2} ds_1 \mathbb{E}_{G, \eta}[\text{tr}(\exp\{(1 - s_2)\tau H(G)\} H(\eta) \exp\{(s_2 - s_1)\tau H(G)\} H(\eta) \exp\{s_1 \tau H(G)\})]. \quad (28)$$

Using cyclic properties of the trace, this can be simplified to

$$\partial_\tau^2 Z(\tau, p, N, n_{bos}) = \frac{1}{2} \int_0^1 ds_1 \mathbb{E}_{G, \eta} [\text{tr}(\exp\{(1-s_1)\tau H(G)\} H(\eta) \exp\{s_1 \tau H(G)\} H(\eta))]. \quad (29)$$

We now use a general result. Consider any Hermitian H (we will use $H = \tau H(G)$) and any operator O (we will use $O = H(\eta) = H(\eta)^\dagger$) and any $s_1 \in [0, 1]$. We claim that

$$\text{tr}(\exp(H) O^\dagger O) + O \leftrightarrow O^\dagger \geq \text{tr}(\exp((1-s_1)H) O^\dagger \exp(s_1 H) O) + O \leftrightarrow O^\dagger,$$

where $O \leftrightarrow O^\dagger$ indicates the previous term with O, O^\dagger interchanged. Proof of claim: work in an eigenbasis of H . It suffices to consider the case that $O = |b\rangle\langle a|$ where $|a\rangle, |b\rangle$ are eigenvectors of H with eigenvalues E_a, E_b . Then the right-hand side is equal to $\exp(s_1 E_a + (1-s_1)E_b) + \exp(s_1 E_b + (1-s_1)E_a) = 2 \exp((E_a + E_b)/2) \cosh((s_1 - 1/2)(E_b - E_a))$. The cosh function is maximized on the interval $[0, 1]$ at $s_1 = 0, 1$ when the right-hand side becomes equal to the left-hand side. So,

$$\partial_\tau^2 Z(\tau, p, N, n_{bos}) \leq \frac{1}{2} \mathbb{E}_{G, \eta} [\text{tr}(\exp(\tau H(G)) H(\eta)^2)]. \quad (30)$$

For the real ensemble without symmetrization, we have

$$\mathbb{E}[H(\eta)^2] = \sum_{\mu_1, \dots, \mu_{p/2}} \sum_{\nu_1, \dots, \nu_{p/2}} \prod_{i=1}^{p/2} (a_{\mu_i}^\dagger a_{\nu_i}) \prod_{i=1}^{p/2} (a_{\nu_i}^\dagger a_{\mu_i})$$

To leading order in N , we may approximate $\sum_\nu a_\nu a_\nu^\dagger = N$ so on the given Hilbert space, $\mathbb{E}[H(\eta)^2]$ is a scalar equal to $(p/2)! \binom{n_{bos}}{p/2} N^{p/2} + O(N^{p/2-1})$. In general, for the complex or real ensemble, symmetrized or not, we find that $\mathbb{E}[H(\eta)^2]$ is a scalar equal to $J(p/2)! \binom{n_{bos}}{p/2} N^{p/2}$, where J obeys the claims of the lemma. To verify that $\mathbb{E}[H(\eta)^2]$ is a scalar and to compute the scalar to all orders in N , commute the annihilation operators a_ν to the right. The result is some linear combination of operators with all annihilation operators to the right of creation operators which can be written in the form $\sum_{\mu_1, \dots, \mu_k} (\prod_{i=1}^k a_{\mu_i}^\dagger) (\prod_{i=1}^k a_{\mu_i})$, and each such operator is equal to $k! \binom{n_{bos}}{k}$.

Hence, from Eq. (30), $\partial_\tau^2 \log(Z(\tau, p, N, n_{bos})) \leq (J/2) n_{bos}^{p/2} N^{p/2}$. \square

Remark: this result is clearly not tight in the regime where random matrix theory is accurate ($n_{bos} = p/2$). It is interesting to see what happens there. The correlation function $\mathbb{E}_{G, \eta} [\text{tr}(\exp\{(1-s_1)\tau H(G)\} H(\eta) \exp\{s_1 \tau H(G)\} H(\eta))]$ is not independent of s_1 , but rather decays as a function of s_1 for $s_1 \leq 1/2$ (of course, it increases again as s_1 becomes larger than $1/2$). Considering the regime $s_1 \ll 1/2$, using the square-root singularity at the edge of the Wigner semi-circle we can estimate that it decays as $s_1^{-3/2}$. This means that the integral of this correlation function over s_1 is dominated by its value for small s_1 of order $1/\tau$ so that for τ large compared to the inverse width of the semi-circle (though of course τ not too large) the integral becomes of order $1/\tau$. This is stronger than the upper bounds here where we have bounded by the integral by something independent of τ . We may guess that a tighter analysis will show that a similar effect will happen in the case of $n_{bos} \gg p/2$; however, an important difference occurs. If we take an eigenstate of $H(G)$ with some eigenvalue λ_0 , and apply $H(\eta)$ for random η , this only changes $p/2$ out of the n_{bos} qudits in the state. So, one might guess that the resulting state will have expectation value of $H(G)$ that is $\lambda_0(1 - p/(2n_{bos}))$ rather than an (as in the random matrix case) expectation value of $H(G)$ which is zero. So, we may guess that the correlation function will be non-negligible for $s_1 \lesssim (n_{bos}/p)\tau^{-1}$. A heuristic estimate in this fashion suggests that the lemma below for the eigenvalue is tight up to logarithmic factors.

From lemma 4, the following lemma is an immediate corollary:

Lemma 5. *Let λ_1 be the largest eigenvalue of G . Let*

$$E_{max} = \sqrt{2J \log(N) n_{bos}^{p/4+1/2} N^{p/4}}.$$

Then, for any x ,

$$\Pr[\lambda_1 \geq x] \leq \exp\left(-\frac{x - E_{max}}{\xi}\right), \quad (31)$$

with

$$\xi = \frac{\sqrt{J} n_{bos}^{p/4-1/2} N^{p/4}}{\sqrt{2 \log(N)}} \quad (32)$$

So, for any E' which is $\omega(\xi)$, with high probability $\lambda_1 \leq E_{max} + E'$.

Proof. We have $\text{tr}(\exp\{\tau H(G)\}) \geq \exp(\tau \lambda_1)$. Hence, for any x , $\Pr[\lambda_1 \geq x] \leq Z(\tau, p, N, n_{bos}) / \exp(\tau x)$. Since $D(N, n_{bos}) \leq N^{n_{bos}}$, minimizing over τ , we find that

$$\Pr[\lambda_1 \geq x] \leq \exp\left(n_{bos} \log(N) - \frac{x^2}{2J n_{bos}^{p/2} N^{p/2}}\right).$$

For $x = E_{max}$, the right-hand is equal to 1 and for $x > E_{max}$ the right-hand side decays exponentially. \square

B. Odd p

We now consider the case of odd p . Let $Z(\tau, p, N, n_{bos})$ denote $\mathbb{E}[\text{tr}(\exp(\tau H(G)))]$ for a tensor G of rank p , with entries of G chosen from the Gaussian distribution, with given N, n_{bos} . In this subsection, G is complex and not symmetrized. The Hamiltonian $H(G)$ is given by Eq. (6) for even p and by Eq. (9) for odd p .

We will reduce the calculation for odd p to the case for even p , up to a bounded error, showing the following

Lemma 6. For odd p , for $n_{bos}^{p-1} \tau N^{1/3} = o(1)$,

$$Z(\sqrt{2N}\tau, 2(p-1), N, n_{bos}) \leq Z(\tau, p, N, n_{bos}) \leq Z(\sqrt{2N}\tau, 2(p-1), N, n_{bos}) \exp(o(N^{(p-1)/2})). \quad (33)$$

All occurrences of $Z(\cdot)$ in the above equation refer to the complex ensemble without symmetrizing.

Remark: the assumptions require that τ is $o(1)$; however, that is still large enough τ to be useful later in bounding the spectrum since the largest eigenvalues of $H(G)$ are typically large compared to 1.

Proof. For arbitrary $H(G)$, the exponential $\exp\{\tau H(G)\}$ can be expanded as a series

$$1 + \tau H(G) + \frac{\tau^2}{2} H(G)^2 + \dots$$

We bring the expectation value (over G) inside the trace, and compute the expectation value of this series. This expectation value can be computed using Wick's theorem to compute the expectation value of a moment of a Gaussian distribution.

For odd p , each term in H is a sum of two terms, one depending on the product of two tensors G and one depending on the product of two tensors \overline{G} , where the overline denotes complex conjugation. Hence, the m -th order term $\frac{\tau^m}{m!} H(G)^m$ is a sum of 2^m terms, corresponding to m different choices of G or \overline{G} in each $H(G)$; we call each such choice a *history*. Each term is a product of creation and annihilation operators depending on a product of $2m$ tensors, some of which are G and some of which are \overline{G} . This expectation value of such a term is non-vanishing only if there are m tensors G and m tensors \overline{G} . In that case, the expectation value is given by summing over ways of pairing each tensor G with a distinct tensor \overline{G} . There are $m!$ such pairings. Then, for each such pairing, one computes an operator by taking the operator $\frac{\tau^m}{m!} H(G)^m$ and replacing, for every pair, the two tensors $G_{\mu_1, \dots, \mu_p} \overline{G}_{\nu_1, \dots, \nu_p}$ in that pair with a product of δ -functions

$$\prod_{a=1}^p \delta_{\mu_a, \nu_a} = \mathbb{E}[G_{\mu_1, \dots, \mu_p} \overline{G}_{\nu_1, \dots, \nu_p}].$$

Summing this operator over pairings gives the expectation value.

Note also that here we have not symmetrized G . If we were to symmetrize G , then $\mathbb{E}[G_{\mu_1, \dots, \mu_p} \overline{G}_{\nu_1, \dots, \nu_p}]$ is given by $1/p!$ times a sum of $p!$ different products of δ -functions, i.e. $\sum_{\pi} \delta_{\mu_a, \nu_{\pi(a)}}$, where π is a permutation. This would lead to additional terms that we need to compute and would make the analysis more difficult (though in practice may lead to better performance).

For given m , given history and given pairing, let us define a *cluster* as follows: define a graph with $2m$ vertices, each vertex corresponding to a single tensor, either G or \overline{G} . Let there be an edge between any two tensors which both appear in the same term in the Hamiltonian. Let there also be an edge between any two tensors which are in a pair.

Hence, this is a graph of degree 2 (we allow the possibility of multiple edges connecting two vertices if we pair two tensor which both appear in the same term in the Hamiltonian). A *cluster* is a connected component of this graph.

We refer to a cluster containing four vertices as a *minimal* cluster. A cluster with six or more vertices is called a *non-minimal* cluster. Note that there are no clusters containing only two terms because each term in $H(G)$ depends on a product of two tensors G or two tensors \bar{G} ; if we had not taken the complex ensemble and instead taken G to be real, we would instead have these clusters with two terms. We discuss the case of the real ensemble further after the proof of the lemma.

The minimal clusters will turn out to give the dominant order contribution in an expansion in N . The non-minimal clusters will be subleading order.

We have expressed $\frac{\tau^m}{m!} H(G)^m$ as a sum over histories and pairings and so $\mathbb{E}[\exp(\tau H(G))]$ is a sum over m , histories, and pairings. Each term in the sum over m , histories, and pairings for $\mathbb{E}[\exp(\tau H(G))]$ is an operator. Hence, $Z(\tau, p, N, n_{bos})$ is also given by a sum over m , histories, and pairings as one may take the trace of each term in $\mathbb{E}[\exp(\tau H(G))]$. Note that each m , history, and pairing gives a non-negative contribution to $Z(\tau, p, N, n_{bos})$, i.e., it has a non-negative trace.

Lower Bound— Now we prove the first inequality in Eq. (33), namely that $Z(\sqrt{2N}\tau, 2(p-1), N, n_{bos}) \leq Z(\tau, p, N, n_{bos})$. To do this, we first define a similar summation over m , histories, and pairings to compute $Z(\tau, q, N, n_{bos})$ for even $q = 2(p-1)$ and then we compare the two summations.

For even q , each term in H is a sum of two terms, one depending linearly on a tensor G and one depending on tensor \bar{G} . As before, the m -th order term $\frac{\tau^m}{m!} H(G)^m$ is a sum of 2^m *histories*, with each history corresponding to m different choices of G or \bar{G} in each $H(G)$. Each term is a product of creation and annihilation operators depending on a product of m tensors, some of which are G and some of which are \bar{G} ; note the difference here from the odd case as now there are only m tensors, rather than $2m$. This expectation value of such a term is non-vanishing only if there are $m/2$ tensors G and $m/2$ tensors \bar{G} . In that case, the expectation value is given as before by summing over pairings and replacing the two tensors in the pair with δ -functions.

We claim that if we consider $Z(\tau, p, N, n_{bos})$ for odd p and consider only the sum of pairings for which all clusters are minimal, this gives precisely the sum of terms for $Z(\sqrt{2N}\tau, 2(p-1), N, n_{bos})$. Since all terms contribute non-negatively to the trace, this will prove the first inequality.

To show this claim, let $L_{2(p-1)}$ label a given choice of m , history, and pairing for $Z(\sqrt{2N}\tau, 2(p-1), N, n_{bos})$ and let L_p label a given choice of m , history, and clusters for $Z(\tau, p, N, n_{bos})$ for which all clusters are minimal. There are $2^{m/2}$ different pairings for the given choice of clusters labelled by L_p .

We will construct a one-to-one correspondence between L_p and $L_{2(p-1)}$ and show that the sum of the terms labelled by L_p is equal to the term labelled by $L_{2(p-1)}$, i.e., that they give the same operator, and hence have the same trace. Consider given L_p . Then, $L_{2(p-1)}$ is as follows. The value of m is the same. The history is also the same: for a given sequence of choices of G or \bar{G} for L_p , we use the same sequence for $L_{2(p-1)}$. Note that in L_p , each of the m choices of G or \bar{G} denotes a choice that both tensors in $H(G)$ are equal to G or both are equal to \bar{G} , while in $L_{2(p-1)}$ each of the m choices is only a choice about a single tensor in $H(G)$.

The pairing is as follows. We introduce notation to label the tensors in $H(G)^m$. For even p , we label the tensors by an integer $i \in \{1, 2, \dots, m\}$ depending which of the m factors of $H(G)$ it is in. Here, we mean that the tensors appear in the product $H(G)^m = H(G) \cdot H(G) \cdot \dots \cdot H(G)$; we label each of the different factors in the product $1, 2, \dots, m$ in sequence. For odd p , we label the tensors by a pair (i, w) for $i \in \{1, 2, \dots, m\}$ and $w \in \{1, 2\}$. The integer i labels which of the m factors of $H(G)$ it is in and w labels whether it is the first or second of the two tensors in $H(G)$. For a pairing for $Z(\tau, p, N, n_{bos})$ with all clusters minimal, then each cluster is of the form that for some i, j that we pair $(i, 1)$ with $(j, 1)$ and $(i, 2)$ with $(j, 2)$ or of the form that we pair $(i, 1)$ with $(j, 2)$ and $(i, 2)$ with $(j, 1)$. For a given choice of clusters, the corresponding pairing for $L_{2(p-1)}$ pairs i with j .

We sketch the claim about the terms. For a term in L_p , for each cluster we have two tensors T and two tensors \bar{T} . We replace these tensors with the expectation value

$$\mathbb{E}\left[\sum_{\mu_1, \dots, \mu_{p-1}} \sum_{\nu_1, \dots, \nu_{p-1}} \sum_{\sigma} T_{\mu_1, \mu_2, \dots, \mu_{p-1}, \sigma} T_{\nu_1, \nu_2, \dots, \nu_{p-1}, \sigma} \sum_{\alpha_1, \dots, \alpha_{p-1}} \sum_{\beta_1, \dots, \beta_{p-1}} \sum_{\bar{\sigma}} \bar{T}_{\alpha_1, \alpha_2, \dots, \alpha_{p-1}, \bar{\sigma}} \bar{T}_{\beta_1, \beta_2, \dots, \beta_{p-1}, \bar{\sigma}}\right],$$

which is equal to some product of δ -functions. This expectation value then multiplies the operators $\left(\prod_{i=1}^{(p-1)/2} a_{\mu_i}^\dagger a_{\nu_i}^\dagger\right) \left(\prod_{i=(p-1)/2+1}^{p-1} a_{\mu_i} a_{\nu_i}\right)$ and $\left(\prod_{i=1}^{(p-1)/2} a_{\alpha_i}^\dagger a_{\beta_i}^\dagger\right) \left(\prod_{i=(p-1)/2+1}^{p-1} a_{\alpha_i} a_{\beta_i}\right)$ inserted at the appropriate places into the product $H(G)^m$. The δ -functions constrain $\sigma = \bar{\sigma}$; summing over this gives a factor of N for each cluster, while there are two pairings for each cluster giving another factor of 2 for each cluster. The number of clusters is equal to $m/2$, giving an overall factor $(2N)^{m/2}$.

This proves the first inequality.

Upper bound— Now we prove the second inequality in Eq. (33). To do this, we define the following quantity for q which is a multiple of 4 (note that $q = 2(p-1)$ is a multiple of 4 if p is odd):

$$Z'(\tau, \tau', q, N, n_{bos}) \equiv \mathbb{E}[\text{tr}(\exp\{\tau H(G) + \tau' H(G')^2\})], \quad (34)$$

where the expectation value is over tensors G of order q chosen from the complex ensemble and tensors G' of order $q/2$ chosen also from the complex ensemble (note that $q/2$ is even). Note that we square $H(G')$ in the exponent.

We will prove that

$$Z(\tau, p, N, n_{bos}) \leq Z'(\sqrt{2N}\tau, \tau', 2(p-1), N, n_{bos}) \quad (35)$$

for $\tau' = N^{1/3}\tau$. From this, the second inequality in Eq. (33) follows. To see this, we have, for any G' , $\|H(G')^2\| \leq O(n_{bos}^{p-1})\|G'\|^2$ where the operator norm $\|H(G')^2\|$ denotes the largest eigenvalue in absolute value and $\|G'\|$ denotes the largest singular value of G' , regarding G' as a matrix of size $N^{q/4}$ -by- $N^{q/4}$. So, by the Golden-Thompson inequality,

$$\begin{aligned} \mathbb{E}_G[\text{tr}(\exp\{\tau H(G) + \tau' H(G')^2\})] &\leq \mathbb{E}_G[\text{tr}(\exp\{\tau H(G)\})] \mathbb{E}_{G'}[\exp(O(n_{bos}^{p-1})\tau'\|G'\|^2)] \\ &= Z(\sqrt{2N}\tau, 2(p-1), N, n_{bos}) \mathbb{E}_{G'}[\exp(O(n_{bos}^{p-1})\tau'\|G'\|^2)]. \end{aligned} \quad (36)$$

where the subscript G or G' denotes the expectation over G or G' . The matrix G' has complex entries; we can bound its operator norm by the sum of the operator norms of its Hermitian and anti-Hermitian parts. For $n_{bos}^{p-1}\tau' = o(1)$, we have $\mathbb{E}_{G'}[\exp(O(n_{bos}^{p-1})\tau'\|G'\|^2)] = \exp[O(n_{bos}^{p-1})\tau'O(N^{q/4})]$ as can be shown using Hermite polynomials[11] to compute the probability distribution of the eigenvalues of a random matrix and using the decay of a Hermite polynomial multiplying a Gaussian (the study of the largest eigenvalue of a random matrix is a rich field if we consider the probability distribution near the edge of the Wigner semi-circle but here we are satisfied to consider the probability distribution for eigenvalues which are some constant factor > 1 multiplying the edge of the semi-circle).

To show Eq. (35), let L_p label the set of terms for $Z(\tau, p, N, n_{bos})$ with a given choice of m , history, clusters (the clusters need not be minimal), and choice of pairing for all tensors in non-minimal clusters (we do not specify the pairing of the tensors in the minimal clusters; if there are n_{min} minimal clusters then there are $2^{n_{min}}$ terms in the set). Let $L'_{2(p-1)}$ label a term for $Z'(\sqrt{2N}\tau, \tau', 2(p-1), N, n_{bos})$ with given m , history, clusters, and pairing. Here, for Z' , a *history* corresponds to m choices of either $H(G)$ or $H'(G)^2$ and also to choices of G or \bar{G} , or G' or \bar{G}' , for each $H(G)$ or $H(G')$. We define a map from choice of L_p to choice of $L'_{2(p-1)}$ so that the sum of terms labelled by L_p is equal to the term labelled by the corresponding $L'_{2(p-1)}$. This map will be one-to-one but it will not be onto. However, since all terms are non-negative, this will establish the needed inequality.

The map is as follows. The m will be the same. Using the notation above, if tensor $(i, 1)$ is in a minimal cluster (and hence, so is $(i, 2)$) in the set labelled by L_p , then in the history for $L'_{2(p-1)}$ for the i -th term we choose $\tau H(G)$, while if $(i, 1)$ is not in a minimal cluster, then we choose $\tau' H(G')^2$. Given a history labelled by L_p , corresponding to a choice of tensor or its complex conjugate (i.e., either G or \bar{G}) in each of the m different $H(G)$ in a term in the series for $Z(\tau, p, N, n_{bos})$, then we make the same choice of tensor or its complex conjugate in each of the m different $H(G)$ or $H(G')^2$ in the history labelled by $L'_{2(p-1)}$. That is, if we choose G in the i -th $H(G)$ in some terms in the series for $Z(\tau, p, N, n_{bos})$, then we choose G in $H(G)$ or G' in *both* terms in $H(G')^2$ and similarly if we choose \bar{G} we choose \bar{G} in $H(G)$ and \bar{G}' in both terms in $H(G')^2$.

We label the tensors in the expansion for $Z'(\sqrt{2N}\tau, \tau', 2(p-1), N, n_{bos})$ either by an integer i , if it appears in $H(G)$, or by a pair (i, w) , if it appears in $H(G')^2$, in which case the index $w \in \{1, 2\}$ labels which of the two $H(G')$ it is in. Finally, we define the pairing labelled by $L'_{2(p-1)}$. For a minimal cluster labelled by L_p , pairing $(j, 1)$ and $(i, 2)$ with $(j, 2)$ or $(i, 1)$ with $(j, 2)$ and $(i, 2)$ with $(j, 1)$, then in the pairing labelled by $L'_{2(p-1)}$ we pair i with j .

If a cluster is non-minimal, then we simply use the same pairing for the corresponding tensors in $L'_{2(p-1)}$. That is, suppose a cluster pairs (i_1, w_1) with (i_2, w'_2) , and pairs (i_2, w_2) with (i_3, w'_3) , and so on, where $w'_a = 1$ if $w_a = 2$ and $w'_a = 2$ if $w_a = 1$. Then, we also pair (i_1, w_1) with (i_2, w'_2) , and pairs (i_2, w_2) with (i_3, w'_3) , and so on.

The smallest non-minimal cluster has six vertices. In every cluster, minimal or not, there is a sum over some index (for example, the sum over the index $\sigma = \sigma'$ in the lower bound calculation above) which gives a factor N . Thus, taking $\tau' = \tau N^{1/3}$ accounts for this factor. No factor of 2 occurs for the non-minimal clusters. \square

Remark: since we have chosen the entries of G from the complex ensemble, we have that $\mathbb{E}[H(G)] = 0$ for p odd. If instead we had chosen the entries of G from the real ensemble we would have (considering the specific case $p = 3$ for simplicity and not symmetrizing G , again for simplicity) a non-vanishing expectation value since

$$\mathbb{E}[T_{\mu_1, \mu_2, \sigma} T_{\nu_1, \nu_2, \sigma}] = N \delta_{\mu_1, \nu_1} \delta_{\mu_2, \nu_2}, \quad (37)$$

so that $\mathbb{E}[H(G)] \propto N \sum_{\mu_1, \mu_2} a_{\mu_2}^\dagger a_{\mu_2}^\dagger a_{\mu_1} a_{\mu_1}$. Such a term is sometimes called a pairing term or a “cooperon” in the study of disordered system in physics. In the case $n_{bos} = 2$ (the smallest possible for $p = 3$), this term has operator norm N^2 . This is much larger than $N^{1/2}$ times the expected operator norm of $H(G)$ for $p = 2(p-1) = 4$, i.e., that expected operator norm is proportional to N by random matrix theory, and $N^2 \gg N^{3/2}$.

There may be other ways to deal with this non-vanishing expectation value other than using the complex ensemble. One way is to use the real ensemble, but to consider the Hamiltonian $H(T_0) - M$, where we define $M = N \sum_{\mu_1, \mu_2} a_{\mu_2}^\dagger a_{\mu_2}^\dagger a_{\mu_1} a_{\mu_1}$ for $p = 3$. In this case, the added term $-M$ cancels the expectation value of $H(G)$ term-by-term in the perturbation expansion. However, if we do this we still have some additional terms when we consider clusters of four or more vertices. We expect that the clusters of six or more vertices are still negligible, but the structure of the clusters of four vertices becomes more complicated. We leave the analysis of this case for the future, but we expect that it would work and may be practically useful.

From lemma 6, the following lemma is an immediate corollary:

Lemma 7. *Let λ_1 be the largest eigenvalue of G . Let*

$$E_{max} = 2\sqrt{J \log(N)} n_{bos}^{p/2} N^{p/2},$$

where J is the J of lemma 4 for $Z(\tau, 2(p-1), N, n_{bos})$. Then, for any x ,

$$\Pr[\lambda_1 \geq x] \leq \exp\left(-\frac{x - E_{max}}{\xi}\right), \quad (38)$$

with

$$\xi = \frac{\sqrt{J} n_{bos}^{p/2-1} N^{p/2}}{\sqrt{\log(N)}}. \quad (39)$$

So, for any E' which is $\omega(\xi)$, with high probability $\lambda_1 \leq E_{max} + E'$.

Proof. From lemmas 4,6, for $n_{bos}^{p-1} \tau N^{1/3} = o(1)$, we have $Z(\tau, p, N, n_{bos}) \leq \exp(\tau^2 J n_{bos}^{p-1} N^p) \exp(o(N^{(p-1)/2}))$.

Let $\tau = n_{bos}^{1-p/2} N^{-p/2} \sqrt{\log(N)} / \sqrt{J}$. For $n_{bos} = o(N^{(p/2-1/3)/(1+p/2)} \log(N)^{1/(2+p)})$, the condition $n_{bos}^{p-1} \tau N^{1/3} = o(1)$ holds. So, after some algebra, for any x , the result follows. \square

It is worth noting that lemma 6 has the following corollary:

Corollary 2. *For odd p , for $n_{bos} = p-1$, with high probability $\lambda_1 = O(N^{p/2})$.*

Proof. To prove this we use the existence of tighter bounds for even $q = 2(p-1)$ when $n_{bos} = q/2$. By Eq. (33), for $n_{bos}^p \tau N^{1/3} = o(1)$, we have $Z(\tau, p, N, n_{bos}) \leq Z(\sqrt{2N}\tau, 2(p-1), N, n_{bos}) \exp(o(N^{(p-1)/2}))$. Since we are considering fixed n_{bos} , this holds for $\tau N^{1/3} = o(1)$. We have that $Z(\sqrt{2N}\tau, 2(p-1), N, p-1) = \mathbb{E}_G[\exp(\sqrt{2N}\tau H(G))]$, but the Hamiltonian $H(G)$ is a random matrix of size N^{p-1} -by- N^{p-1} chosen from the so-called Gaussian unitary ensemble[11]. With high probability, the largest eigenvalue of this matrix is $\Theta(N^{(p-1)/2})$. Further, we can choose $\tau = \omega(\log(N) N^{-p/2})$ so that $Z(\sqrt{2N}\tau, 2(p-1), N, p-1) = N \exp(O(\tau N^{p/2}))$; for example, this can be shown by using orthogonal polynomials to bound the probability distribution of the largest eigenvalue as discussed previously. We have $\Pr[\lambda_1 > x] \leq Z(\tau, p, N, n_{bos}) \exp(-\tau x)$. For x sufficiently large compared to $N^{p/2}$, for $\tau = \omega(\log(N) N^{-p/2})$, the right-hand side is $o(1)$. \square

IV. QUANTUM AND CLASSICAL ALGORITHMS

We now discuss the complexity of classical and quantum algorithms to implement the needed linear algebra. In particular, we need to determine if that largest eigenvalue is larger than E_{cut} and we need to find some vector in the eigenspace of eigenvalue $\geq E_{cut}$. We emphasize that it is not necessary to find the leading eigenvector itself.

We will use ψ_{target} to denote this leading eigenvector. Note that if we lower bound the squared overlap of some vector with ψ_{target} , this will lower bound the probability of success of phase estimation.

We have defined algorithm 2 so that in the detection step it uses a hard cutoff on eigenvalue: if the leading eigenvalue is $\geq E_{cut}$ it reports that $\lambda = \bar{\lambda}$ while otherwise it reports that $\lambda = 0$. However, no algorithm, classical or quantum, will be able to compute the leading eigenvalue exactly; there will always be some limit on the precision. Fortunately, the proofs of theorems 2,4 show that if $E_0 \geq (1+c)E_{max}$ for any $c > 0$, then if $\lambda = \bar{\lambda}$ then with high probability $\lambda_1 \geq (1-\eta)E_0 + \eta E_{max}$ for any $0 < \eta < 1$ while if $\lambda = 0$ then with high probability $\lambda_1 \leq (1-\eta')E_0 + \eta' E_{max}$ for

any $0 < \eta' < 1$. For example, we might take $\eta = 1/8$ and $\eta' = 1/2$. So, it suffices instead to implement some “soft” estimate of the leading eigenvalue which will be very likely to give one result (i.e., reporting that $\lambda = \bar{\lambda}$) if the leading eigenvalue is larger than $(7/8)E_0 + (1/8)E_{max}$ but very unlikely to give that result if the leading eigenvalue is $\leq E_{cut}$.

In the quantum algorithms, to obtain some vector in the eigenspace of eigenvalue $\geq E_{cut}$ and to do this soft estimate, we will implement an approximate projector by phase estimation in the quantum algorithms onto the eigenspace of eigenvalue $\geq (E_0 + E_{cut})/2 = (3/4)E_0 + (1/4)E_{cut}$. By doing this, the phase estimation error will become negligible when considering the projection of the resulting vector onto the eigenspace with eigenvalue $< E_{cut}$. Similarly, in the classical power method we will take the number of iterations sufficiently large that the vector has negligible projection on the eigenspace with eigenvalue $< E_{cut}$ and further so that it has expectation value for $H(T_0)$ greater than E_{cut} .

We begin with a description of some classical algorithms. The time and space requirements for the classical algorithms are of course not intended to represent a lower bound; rather, they represent times that can be achieved using standard algorithms in the literature. We then give quantum algorithms. Finally, we give a further improvement to the quantum algorithm that may be useful in practice.

When we refer to “space” in a classical algorithm, if we store a $D(N, n_{bos})$ -dimensional vector, the space requirement is equal to $D(N, n_{bos})$ multiplied by the number of bits to store a single entry of the vector. In the classical algorithms, we will not discuss issues with finite precision arithmetic in detail. Since we will be applying operators of the form $H(T_0)^m$ to vectors in the “path integral” methods, we might need to approximate each entry of the vector to accuracy $D(N, n_{bos})^{-1} \|H(T_0)\|^{-m} = O(\text{poly}(N^{n_{bos}m}))^{-1}$. However, the required number of bits is then only $O(mn_{bos} \log(N))$ and m will be logarithmic in N so the required number of bits will be only polylogarithmic in N .

A. Classical Algorithms

Classically, the most obvious algorithm is to perform an eigendecomposition on $H(T_0)$. This requires storing matrices of size $D(N, n_{bos})$ -by- $D(N, n_{bos})$ so that the space required is $\tilde{O}(D(N, n_{bos})^2)$ and the time is $D(N, n_{bos})^\omega$ where ω is the matrix multiplication exponent[12], though of course in practice the time is closer to $D(N, n_{bos})^3$.

However, there is no need to perform a full eigendecomposition. One can instead initialize a random vector and then apply the power method to extract some eigenvector of $H(T_0)$ in the eigenspace with eigenvalue $\geq E_{cut}$. The space required is then only $\tilde{O}(D(N, n_{bos}))$. The time required for a single iteration of the power method is $\tilde{O}(D(N, n_{bos}))$. If $\lambda = \bar{\lambda}$, then in $O(\log(D(N, n_{bos}))/\log(E_0/E_{max}))$ iterations, the resulting vector will have an $1 - o(1)$ projection onto the eigenspace with eigenvalue $\geq (E_0 + E_{cut})/2$ and a negligible projection onto the eigenspace with eigenvalue $< E_{cut}$. So, after this many iterations, one can compute the expectation value of $H(T_0)$ on that vector to perform detection, i.e., if $\lambda = \bar{\lambda}$ the expectation will be larger than E_{cut} but if $\lambda = 0$ the expectation will be close to E_{max} , and one can compute the single particle density matrix of the vector after those iterations. So, the time is $\tilde{O}(D(N, n_{bos}))O(1/\log(E_0/E_{max}))$.

This power method still requires exponential (in n_{bos}) space. In fact, one can use only polynomial space. One obvious choice is to perform a “path integral”. That is, given a random initial state Ψ_{random} , the power method computes the state $H(T_0)^m |\Psi_{\text{random}}\rangle$, for some exponent m which gives the number of iterations in the power method. So, we wish to compute

$$\frac{\langle \Psi_{\text{random}} | H(T_0)^m a_\mu^\dagger a_\nu H(T_0)^m | \Psi_{\text{random}} \rangle}{\langle \Psi_{\text{random}} | H(T_0)^{2m} | \Psi_{\text{random}} \rangle},$$

where the denominator is to correctly normalize the state. Let us choose the initial state to be a (normalized) random state from the basis for the symmetric basis given before. We make this choice to make the “path integral” simpler and since this is a complete orthonormal basis of states, a random state from this basis has expected overlap with the largest eigenvector of $H(T_0)$ equal to $1/N^{n_{bos}}$. Then, both the numerator and denominator above can be expressed by a summation over intermediate states from this basis, requiring only space $\tilde{O}(\log(D(N, n_{bos}))m)$; this summation is a “path integral”. The time required however is now $\tilde{O}(D(N, n_{bos})^m)$ and so becomes significantly worse if the number of iterations is much more than 1.

An improvement to this time can be obtained by using the algorithm of theorem 4.1 of Ref. 10. This algorithm is expressed in terms of qubits, but we can translate the algorithm to get an algorithm for a single qudit of dimension $D(N, n_{bos})$, or, using the tradeoffs discussed there, n_{bos} qudits each of dimension N .

Let us explain this in detail, following the discussion there. We wish to compute $\langle y | C | x \rangle$, for some fixed basis states x, y from the basis above (in our application $x = y$, where C is a “circuit” of some depth d (for example $C = 2m + 1$ for the numerator above and $C = m$ for the denominator above). While in Ref. 10, a circuit was assumed to be unitary, there in fact is no need to assume that; for non-unitary circuits, the issues with finite precision arithmetic do become more important but as remarked above this can be handled with only polylogarithmic overhead. For us,

a circuit is not necessarily unitary; rather it is built out of a sequence of operations, each of which is either $H(T_0)$ or $a_\mu^\dagger a_\nu$; more generally a circuit might include any operations built out of some tensor of order $O(1)$ multiplying some number of creation and annihilation operators.

For $d = 1$, $\langle y|C|x \rangle$ can be computed in time $\text{poly}(n_{bos})$. For $d > 1$, we have

$$\langle y|C|x \rangle = \sum_z \langle y|C_{[d \leftarrow d/2+1]}|z \rangle \cdot \langle z|C_{[d/2 \leftarrow 1]}|x \rangle, \quad (40)$$

where the summation is over states z in the basis and where $C_{[d \leftarrow d/2+1]}$ and $C_{[d/2 \leftarrow 1]}$ are subcircuits from the second and first half of C . If $F(d)$ is the runtime at depth d , we have $F(d) \leq 2 \cdot D(N, n_{bos}) \cdot F(\lceil d/2 \rceil)$. So, $F(d) \leq \text{poly}(n_{bos})(2D(N, n_{bos}))^{\lceil \log(d) \rceil}$. So, the runtime is $\tilde{O}((2D(N, n_{bos}))^{\lceil \log(2m+1) \rceil})$. This is much faster than the path integral method but potentially much slower than the power method, depending on the required m , i.e., for $E_0/E_{max} = \Theta(1)$, we need m proportional to $\log(D(N, n_{bos}))$ and so the time required is superpolynomially worse than the time required if one stores the full vector.

B. Quantum Algorithms

We now discuss quantum algorithms for the same problem. In contrast to the classical algorithms above, all these algorithms take only polynomial space. First let us describe a simple algorithm, given as algorithm 3. We then describe a sequence of improvements.

For algorithm 3 and all subsequent algorithms, we analyze assuming that $\lambda = \bar{\lambda}$, i.e., for purposes of analysis we consider the problem of recovery rather than detection. All these algorithms report success or failure and, if they fail, they are rerun until they succeed. We give bounds on the expected runtime under the assumption that $\lambda = \bar{\lambda}$. If we consider the problem of detection, and if $\lambda = 0$, then the algorithm will not report success in the given runtime (since it will be unable to succeed in a certain phase estimation procedure) and so all these algorithms can also be used for detection by running them for some multiple of the given runtime and reporting that $\lambda = 0$ if success is not reported in that time.

1. Maximally Entangled (or Maximally Mixed) Input State

Algorithms 3,4 in this subsection work on a Hilbert space which is the tensor product of two Hilbert spaces, each of which have dimension $D(N, n_{bos})$. (In some versions of the Hamiltonian simulation used in the algorithms, it is convenient to embed the symmetric subspace of dimension $D(N, n_{bos})$ within the full Hilbert space.)

We use a tensor product notation $A \otimes B$ to denote an operator that is a tensor product of two operators A, B on the two different tensor factors.

Algorithm 3 Quantum Algorithm (simplest, unamplified version). This and all other quantum algorithms have the same inputs, outputs, and parameter choices as algorithm 2.

1. Prepare a maximally entangled state between the two qudits.
 2. Apply phase estimation using $H(T_0) \otimes I$. Let ψ be the resulting state. If the resulting eigenvalue is larger than $(E_0 + E_{cut})/2$, report “success”. Otherwise, report “failure”.
 3. If success is reported, measure and return $\langle \psi | a_\mu^\dagger a_\nu \otimes I | \psi \rangle$.
-

Steps 1 – 2 are designed to prepare a state whose density matrix on the first qudit has large projection onto the eigenspace of eigenvalue $\geq E_{cut}$. For purposes of analysis, we trace out the second qudit, so that the input state on the first qudit is a maximally mixed state. If success is reported then (ignoring phase estimation error) we have indeed projected onto this eigenspace. Ignoring phase estimation error, the probability of success is $\geq 1/D(N, n_{bos})$.

Remark: we prepare a maximally entangled state between the two qudits so that the density matrix on the first state is maximally mixed; we could equally well modify the algorithm to use only a single qudit (reducing the space by a factor of 2) and prepare a random state on the first qudit. This modification however requires some care when we discuss a version of the algorithm that uses amplitude amplification below. In the unamplified version, we can choose a new random state (for example, choosing it uniformly from any orthogonal basis) each time we perform phase estimation; however, in the amplified version, we must choose a fixed random initial state on the first qudit and amplify the algorithm with that choice of initial state. We claim (we omit the proof) that if one picks a tensor product of random single qudit states $|v_1 \otimes v_2 \otimes \dots \otimes v_{n_{bos}} \rangle$ where v_a are independently chosen from a Haar uniform

distribution on the sphere $|v_a| = 1$, with high probability the state has squared overlap with the leading eigenvector close to $N^{-n_{bos}}$; more precisely, “close” means that with high probability the logarithm of the squared overlap is at least $-n_{bos} \log(N) \cdot (1 + o(1))$. Note that we might not have this property of the overlap if we had chosen the initial state from the computational basis, for example. Note also that we can efficiently prepare states from this distribution. Sketch of proof of claim: consider the logarithm of the probability that the sequence of measurements $|v_i\rangle_i \langle v_i|$ succeeds for $i = 1, \dots, n_{bos}$ in sequence. The probability that the i -th measurement succeeds, conditioned on the previous measurements succeeding, can be computed from the trace of $|v_i\rangle \langle v_i|$ with the reduced density matrix on the i -th qudit of some state (i.e., the leading eigenvector projected by the previous measurements), and for any such reduced density matrix with high probability the logarithm of the trace is at least $-\log(N) \cdot (1 + o(1))$.

Step 3 of the algorithm measures some property of a state in the eigenspace with eigenvalue $\geq E_{cut}$. It is possible that each time the algorithm is run, one obtains a different energy measurement and hence a different state, so that measuring this property of the state gives some expectation value of $a_\mu^\dagger a_\nu$ in a mixed state. This does not matter since theorems 3 or 5 also hold for mixed states.

We explain the measurement in step 3 in more detail below. The simplest possibility is to simply measure one matrix element of $a_\mu^\dagger a_\nu$. Since there are N^2 matrix elements, we then need to repeat the algorithm $\text{poly}(N, \log(\epsilon))$ times to measure each matrix element to accuracy ϵ . We explain improvements to this in subsection IV C.

We explain the phase estimation in more detail below. First, let us analyze the algorithm in a rough outline. Let the phase estimation be carried out to a precision sufficiently smaller than $E_0 - E_{max}$. To define this, we work in an eigenbasis of $H(T_0)$. Let $\tilde{\epsilon}$ be a bound on the error probability of the phase estimation. More precisely, we will say that phase estimation implements an operator E_{PE} which is diagonal in the eigenbasis such that on a normalized eigenvector v_i with eigenvalue λ_i we have

$$\begin{aligned} \lambda_i < E_{cut} &\rightarrow \langle v_i | E_{PE} | v_i \rangle \leq \tilde{\epsilon}, \\ \lambda_i > (7/8)E_0 + (1/8)E_{max} &\rightarrow \langle v_i | E_{PE} | v_i \rangle \geq 1 - \tilde{\epsilon}. \end{aligned} \quad (41)$$

The reader should appreciate that there are a few energy scales here, chosen rather arbitrarily. The exact value of the energies are not too important. We have picked E_{cut} to be partway between E_0 and E_{max} so that it is very unlikely that the largest eigenvalue of $H(G)$ is above E_{cut} and also very unlikely that the $\lambda_1 < E_{cut}$. We have picked the energy cutoff in step 2 to be $(E_0 + E_{cut})/2$ simply to pick some energy partway between E_0 and E_{cut} so that it is very unlikely that phase estimation reports success on an eigenstate with energy $< E_{cut}$; see first line of Eq. (41). In the last line of Eq. (41) we wrote $(7/8)E_0 + (1/8)E_{max}$ simply to pick some energy scale slightly below E_0 above which it is very likely that phase estimation reports success; for us later, it would suffice to have any instead a bound for $\lambda_i \geq (1 - \eta)E_0 + \eta E_{max}$ for any $\eta > 0$. Of course, the two lines in Eq. (41) are not completely symmetric about $(E_0 + E_{cut})/2$.

Then, choose $\tilde{\epsilon} = \epsilon/D(N, n_{bos})$, so that the algorithm reports success with probability at least $(1 - \tilde{\epsilon})/D(N, n_{bos})$ and, given that the algorithm reports success, the resulting state has projection onto the eigenspace with eigenvalue $\geq E_{cut}$ which is greater than or equal to

$$\frac{(1 - \tilde{\epsilon})}{(1 - \tilde{\epsilon}) + (D(N, n_{bos}) - 1)\tilde{\epsilon}} = 1 - O(\epsilon),$$

where the first term in the denominator is the probability that it reports success on ψ_{target} as input and the second term is the probability of reporting success on a state in the eigenspace with eigenvalue $< E_{cut}$, multiplied by $D(N, n_{bos}) - 1$, i.e., multiplied by an upper bound on the dimensionality of that eigenspace. Taking $\epsilon \ll 1$, we can obtain a large projection onto the eigenspace with eigenvalue $\geq E_{cut}$, so that the cost of phase estimation increases logarithmically with $D(N, n_{bos})/\epsilon$.

The success probability for $\epsilon \ll 1$ is greater than or equal to $(1/D(N, n_{bos}))(1 - \epsilon/D(N, n_{bos}))$, so for small ϵ it is very close to $1/D(N, n_{bos})$. Hence, repeating the algorithm until it succeeds, the expected runtime to obtain a single measurement of one matrix element of $a_\mu^\dagger a_\nu$ is bounded the time for phase estimation multiplied by $O(1/D(N, n_{bos}))$.

To perform phase estimation, we use controlled simulation of the Hamiltonian $H(T_0)$. There are a large number of quantum simulation algorithms which would work here, such as Refs. 13–17 to name just a few. There are two broad possibilities. The first possibility is to work in the symmetric subspace of dimension $D(N, n_{bos})$. In this case, $H(T_0)$ is a sparse Hamiltonian, and sparse simulation algorithms apply. The second possibility is to use the Hamiltonian of Eq. (5) and embed the symmetric subspace into the full Hilbert space; in this case, $H(T_0)$ is a local Hamiltonian, in that each terms acts on a small number of qudits, each of dimension N , and local simulation algorithms apply. The cost for these algorithms to simulate for a time t to error $\tilde{\epsilon}$ is $\text{poly}(t\|H(T_0)\|, n_{bos}, N, \log(\tilde{\epsilon}))$.

Using the simplest phase estimation algorithm of Ref. 18, the number of bits that we need to phase estimate is $s = O(\log(\|H(T_0)\|/(E_0 - E_{max})))$. The most expensive bit to obtain is the least significant bit, since obtaining the j -th least significant bit requires simulating for a time proportional to $2^{s-j}(E_0 - E_{max})^{-1}$. So, we can obtain the

least significant bit to error $\tilde{\epsilon}/2$, then obtain the next least significant bit to error $\tilde{\epsilon}/4$, and so on, making the total error $\tilde{\epsilon}$. Of course, a large number of variations of the Kitaev phase estimation algorithm exist in the literature, and any could be used here.

With high probability, $\|H(T_0)\|$ is $\text{poly}(N)$. Thus, with high probability the time for phase estimation is $\text{poly}(N, n_{bos}, 1/(E_0 - E_{max}), \log(D(N, n_{bos})/\epsilon))$, giving an algorithm runtime $D(N, n_{bos})\text{poly}(N, n_{bos}, 1/(E_0 - E_{max}), \log(D(N, n_{bos})/\epsilon))$.

We can speed this algorithm up quadratically by applying amplitude amplification[19]. Modify the phase estimation step 2 of algorithm 3 so that the algorithm phase estimates the eigenvalue, determines if the eigenvalue is larger than E_{cut} , then uncomputes the eigenvalue, returning just a single bit of success or failure. See algorithm 4. Then, applying amplitude amplification, with high probability the algorithm succeeds in expected time $D(N, n_{bos})^{1/2}\text{poly}(N, n_{bos}, 1/(E_0 - E_{max}), \log(D(N, n_{bos})/\epsilon))$. Multiplying by $\text{poly}(N, 1/\epsilon)$ to measure $a_\mu^\dagger a_\nu$ to accuracy ϵ , the the expected time is still

$$D(N, n_{bos})^{1/2}\text{poly}(N, n_{bos}, 1/(E_0 - E_{max}), \log(D(N, n_{bos})/\epsilon)),$$

giving a quadratic time improvement, up to $\text{poly}(N)$ factors, and an exponential space improvement, over the fastest classical algorithm described above.

Algorithm 4 Quantum Algorithm (amplified version)

1. Apply amplitude amplification to steps 1 – 2 of algorithm 3, modifying step 2 to uncompute the eigenvalue and return only success or failure.
 2. If success is reported, measure and return $\langle \psi | a_\mu^\dagger a_\nu \otimes I | \psi \rangle$.
-

2. Chosen Input State: Simple Version

We can obtain a further quadratic speedup by modifying the initial state that we phase estimate. In this subsubsection, let us first explain an algorithm that gives the basic idea of the initial state preparation; we will only be able to prove some slightly weaker results for this algorithm (in particular we will prove a lower bound on the average inverse runtime, rather than an upper bound on the average runtime). We expect that this is primarily a technical issue and that some concentration of measure argument should allow us to prove an upper bound on the average runtime. We will then describe in the next subsubsection a modification to the algorithm which avoids this technical difficulty and for which we can prove a quadratic improvement without further assumption.

Instead of a maximally entangled state, we can use the tensor T_0 to prepare a state with a larger projection onto ψ_{target} . In this subsubsection, we will work in the $D^{n_{bos}}$ -dimensional Hilbert space of Eq. (5), so we will use n_{bos} qudits each of dimension N . The unamplified version is algorithm 5 and a version with amplitude amplification is algorithm 6.

The most important new step that must be explained is the initial state preparation (we discuss some other details at the end of this subsubsection). We use the fact that, given a classical list of amplitudes for some M -dimensional vector, with the vector having unit norm, we can prepare a quantum state on an M -dimensional qudit with the given amplitudes (up to an ill-defined overall phase, of course) using a quantum circuit of depth $O(M)$ and using $O(M)$ classical computation. For example, labelling the basis states $|0\rangle, |1\rangle, \dots, |M-1\rangle$, one can start with initial state $|0\rangle$ and apply a sequence of $M-1$ rotations in the two dimensional subspaces spanned by $|i\rangle, |i+1\rangle$ for $i = 0, \dots, M-2$.

Algorithm 5 Quantum Algorithm (improved input state, unamplified version)

1. Use T_0 to prepare the initial state Ψ_{input} of Eq. (42).
 2. If the initial state is not in the symmetric subspace, report “failure”. If the state is in the symmetric subspace, apply phase estimation using $H(T_0)$. Let ψ be the resulting state. If the resulting eigenvalue is larger than $(E_0 + E_{cut})/2$, report “success”. Otherwise, report “failure”.
 3. If success is reported, measure and return $\langle \psi | a_\mu^\dagger a_\nu | \psi \rangle$.
-

In this subsubsection, for simplicity in analysis, we assume that the error ϵ in phase estimation is small enough to be negligible.

We use the same method to produce the input state for both even and odd p . First consider the case that n_{bos} is an integer multiple of p . For any tensor T of order p , let $|T\rangle$ denote the vector on p qudits (each of dimension N)

Algorithm 6 Quantum Algorithm (amplified version)

1. Apply amplitude amplification to steps 1–2 of algorithm 5, modifying step 2 to uncompute the eigenvalue and uncompute the determination of whether the state is in the symmetric subspace and to return only success or failure.
 2. If success is reported, measure and return $\langle \psi | a_\mu^\dagger a_\nu | \psi \rangle$.
-

with amplitudes given by the entries of the tensor. This vector is correctly normalized if $|T| = 1$. We prepare the input state

$$\Psi_{\text{input}} = \frac{1}{|T_0|^{n_{\text{bos}}/p}} |T_0\rangle^{\otimes n_{\text{bos}}/p}. \quad (42)$$

Preparing this state takes circuit depth $O(N^p)$ since we can prepare n_{bos}/p copies of the state $\frac{1}{|T_0|} |T_0\rangle$ in parallel.

We want to know the expectation value $\langle \Psi_{\text{input}} | E_{PE} | \Psi_{\text{input}} \rangle$, but to get oriented, let us estimate the overlap $\langle \Psi_{\text{input}} | \Psi_{\text{sig}} \rangle$.

We have

$$\langle v_{\text{sig}}^{\otimes p} | T_0 \rangle = \lambda N^p + \langle v_{\text{sig}}^{\otimes p} | G \rangle. \quad (43)$$

The probability distribution of $\langle v_{\text{sig}}^{\otimes p} | G \rangle$ is a Gaussian with zero mean and unit variance, so with high probability, $\langle v_{\text{sig}}^{\otimes p} | G \rangle$ is $o(\lambda N^p)$; indeed, for any increasing function of N which diverges as $N \rightarrow \infty$, with high probability it is bounded by that function.

Hence, with high probability,

$$\langle v_{\text{sig}}^{\otimes n_{\text{bos}}} | T_0^{\otimes n_{\text{bos}}/p} \rangle = (1 - o(1)) \cdot \lambda^{n_{\text{bos}}/p} N^{n_{\text{bos}}}. \quad (44)$$

At the same time,

$$\langle T_0^{\otimes n_{\text{bos}}/p} | T_0^{\otimes n_{\text{bos}}/p} \rangle = |T_0|^{2n_{\text{bos}}/p}. \quad (45)$$

We have $\mathbb{E}[|G|^2] = O(N^p)$, where the precise constant in the big-O notation depends on whether we symmetrize G or not and whether we use complex entries or not. Further, $|G|^2$ is a sum of squares of independent random variables (the entries of G). So, by central limit, with high probability $|G|^2$ is bounded by $O(N^p)$. So, with high probability, $|T_0|^{2n_{\text{bos}}/p} = O(N^{n_{\text{bos}}})$.

So, with high probability,

$$\frac{\left| \langle v_{\text{sig}}^{\otimes n_{\text{bos}}} | T_0^{\otimes n_{\text{bos}}/p} \rangle \right|^2}{\langle v_{\text{sig}}^{\otimes n_{\text{bos}}} | v_{\text{sig}}^{\otimes n_{\text{bos}}} \rangle \cdot \langle T_0^{\otimes n_{\text{bos}}/p} | T_0^{\otimes n_{\text{bos}}/p} \rangle} \geq (1 - o(1)) \cdot \lambda^{2n_{\text{bos}}/p}. \quad (46)$$

For $\lambda = CN^{-p/4}$, this is $(1 - o(1))C^{2n_{\text{bos}}/p}N^{-n_{\text{bos}}/2}$.

If ψ_{target} were equal to $\Psi_{\text{sig}} = N^{-n_{\text{bos}}/2} |v_{\text{sig}}^{\otimes n_{\text{bos}}}\rangle$, then for fixed $N^{-p/4}/\lambda$, Eq. (46) would give a lower bound to the squared overlap of the initial state with ψ_{target} which would be quadratically better (in terms of its scaling with N) than the squared overlap for the maximally entangled input state. So, after applying amplitude amplification, this would give a quartic improvement over the fastest classical algorithm.

However, ψ_{target} is not equal to $\Psi_{\text{sig}} = N^{-n_{\text{bos}}/2} |v_{\text{sig}}^{\otimes n_{\text{bos}}}\rangle$ and so this does not give a lower bound on $\langle \psi_{\text{target}} | \Psi_{\text{input}} \rangle$.

However, we have:

Lemma 8. *Suppose that $E_0 \geq (1 + c)E_{\text{max}}$. Then, the projection of Ψ_{sig} onto the eigenspace with eigenvalue $\geq (7/8)E_0 + (1/8)E_{\text{max}}$ is greater than or equal to $\Omega(1) \cdot c/(1 + c)$.*

Proof. The expectation value $\langle \Psi_{\text{sig}} | H(T_0) | \Psi_{\text{sig}} \rangle$ was estimated in subsections IIB 1, IIB 2. We have that with high probability, this expectation value is $\geq (1 - o(1))E_0$. With high probability, the largest eigenvalue of $H(T_0)$ in absolute value is bounded by $(1 + o(1))(E_0 + E_{\text{max}})$; for the even case this is just the triangle inequality, while for the odd case this uses lemma 3. Hence, by Markov's inequality applied to $\lambda_1 - H(T_0)$, the projection of Ψ_{sig} onto the eigenspace with eigenvalue $\geq (7/8)E_0 + (1/8)E_{\text{max}}$ is greater than or equal to $(1 - o(1))E_0 - ((7/8)E_0 + (1/8)E_{\text{max}})/((1 + o(1))(E_0 + E_{\text{max}}))$ which is $\geq \Omega(1) \cdot c/(1 + c)$. \square

So, Ψ_{sig} has some non-negligible projection onto the desired eigenspace. This does not however yet give us a lower bound on $\langle \Psi_{\text{input}} | E_{PE} | \Psi_{\text{input}} \rangle$: we can expand Ψ_{input} as a linear combination of Ψ_{sig} and some orthogonal state but we have not bounded the cross-terms in the expectation value.

However, we now give heuristic evidence (not a proof) for a lower bound on $\mathbb{E}[\langle \Psi_{\text{input}} | E_{PE} | \Psi_{\text{input}} \rangle]$. The main assumption we will make is that $\lambda_1 = E_0 \cdot (1 + o(1/\log(N)))$; we expect that that assumption can be proven to hold with high probability. We consider just the case of even p (we expect that odd p can be handled similarly).

The main reason that we do not give a full proof is that a lower bound on $\mathbb{E}[\langle \Psi_{\text{input}} | E_{PE} | \Psi_{\text{input}} \rangle]$ will only imply a lower bound on the expected inverse squared runtime, rather than an upper bound on the expected runtime, which is what we really want. Instead in the next subsection we give a modified algorithm with an upper bound on the expected runtime. Let us note that we conjecture that algorithm 6 does have a quartic speedup for the expected runtime with high probability. One might guess that this could be proven using the bound on expectation value of the inverse squared runtime and some concentration of measure argument. However, we have not been able to make this precise.

Let us clarify some terminology to distinguish two meanings of the word “expected”, corresponding to averages over G or averages over outcomes of a quantum algorithm, i.e., to the “expected runtime”. From here on, when we refer to the “runtime” of a phase estimation algorithm, this is a short way of saying the expected runtime for a given choice of G . When we refer to the “expectation value of the runtime”, we mean the expectation value over G of this expected runtime. Applying amplitude amplification, the runtime is bounded by the time for the state preparation and phase estimation multiplied by the inverse square-root of $\langle \Psi_{\text{input}} | E_{PE} | \Psi_{\text{input}} \rangle$. So, lower bounding $\mathbb{E}_G[\langle \Psi_{\text{input}} | E_{PE} | \Psi_{\text{input}} \rangle]$ will upper bound the expectation value over G of the inverse squared runtime and we will find a bound on the expectation value of the inverse squared runtime by

$$\left(N^{n_{\text{bos}}/4} \text{poly}(N, n_{\text{bos}}, 1/(E_0 - E_{\text{max}}), \log(D(N, n_{\text{bos}})/\epsilon)) \right)^{-2} \left(N^{-p/4}/\lambda \right)^{-2n_{\text{bos}}/p}$$

in the case that $E_0 \geq E_{\text{max}} \cdot (1 + c)$ for any $c > 0$. For fixed $N^{-p/4}/\lambda$, this gives a further quadratic improvement, in terms of the scaling of the runtime with N , over algorithm 4.

Given the existence of that modified algorithm of subsection IV B 3, we will just sketch an outline of a possible proof of the lower bound on $\mathbb{E}[\langle \Psi_{\text{input}} | E_{PE} | \Psi_{\text{input}} \rangle]$, leaving several details out. The basic idea is to lower bound this expectation value by a tensor network, working in some approximation which amounts to ignoring fluctuations in λ_1 , then average the tensor network by a sum of pairings, and use this sum to lower bound the expectation value. Roughly the physical idea is that the terms in Ψ_{input} proportional to G will tend on average to *increase* the overlap $\langle \Psi_{\text{input}} | \psi_{\text{target}} \rangle$, rather than decrease it.

Consider the operator $\lambda_1^{-m} H(T_0)^m$ for large m . If we take m sufficiently large compared to $n_{\text{bos}} \log(N)/\log(E_0/E_{\text{max}})$, this will operator will lower bound E_{PE} up to some negligible error. That is, we take m large enough that $\lambda_1^{-m} H(T_0)^m$ is negligibly small acting on any eigenvector v_i with eigenvalue $\lambda_i \leq (7/8)E_0 + (1/8)E_{\text{max}}$, and for $\lambda_i \geq (7/8)E_0 + (1/8)E_{\text{max}}$, Eq. (41) gives a lower bound on E_{PE} that is equal to 1 up to some negligible phase estimation error $\tilde{\epsilon}$ while clearly $\lambda_1^{-m} H(T_0)^m \leq 1$. For fixed E_0/E_{max} , it suffices to take $m = O(n_{\text{bos}} \log(N))$.

For the range of m that we consider here, by assumption we can ignore the fluctuations in λ_1 , i.e., we approximate

$$|\langle \Psi_{\text{input}} | E_{PE} | \Psi_{\text{input}} \rangle|^2 \gtrsim (\mathbb{E}[\lambda_1])^{-m} \mathbb{E}[\langle \Psi_{\text{input}} | H(T_0)^m | \Psi_{\text{input}} \rangle] \quad (47)$$

$$\approx (\mathbb{E}[\lambda_1])^{-m} \frac{1}{\mathbb{E}[\langle T_0^{\otimes n_{\text{bos}}/p} | T_0^{\otimes n_{\text{bos}}/p} \rangle]} [\mathbb{E}[\langle T_0^{\otimes n_{\text{bos}}/p} | H(T_0)^m | T_0^{\otimes n_{\text{bos}}/p} \rangle]], \quad (48)$$

where in the second line we further approximate that we can ignore fluctuations in the norm $\langle T_0^{\otimes n_{\text{bos}}/p} | T_0^{\otimes n_{\text{bos}}/p} \rangle$ and treat it as a constant (the proof is a standard large deviation argument on the norm of the tensor).

The quantity $\langle T_0^{\otimes n_{\text{bos}}/p} | H(T_0)^m | T_0^{\otimes n_{\text{bos}}/p} \rangle$ can be evaluated by a sum of tensor networks, using the full Hilbert space, i.e., for each of m choices of $i_1, \dots, i_{p/2}$ in each of the m factors of $H(T_0)$ we have a tensor network). We can then write each tensor network as a sum of tensor networks, inserting either $\lambda v_{\text{sig}}^{\otimes n}$ or G for each tensor, and we can average these tensor networks over G using the methods of appendix A by summing over pairings. Since every term in this sum over networks and pairings is positive, if we restrict to some subset of terms, we get a lower bound. Let us restrict to the terms in which for Ψ_{input} , we choose $\lambda v_{\text{sig}}^{\otimes p}$ for every tensor. For this set of terms, the tensor network computes precisely $\mathbb{E}[\langle \lambda^{n_{\text{bos}}/p} v_{\text{sig}}^{\otimes n_{\text{bos}}} | H(T_0)^m | \lambda^{n_{\text{bos}}/p} v_{\text{sig}}^{\otimes n_{\text{bos}}} \rangle]$. This in turn is $\geq (E_0 \cdot (1 - o(1/m)))^m |\lambda^{n_{\text{bos}}/p} v_{\text{sig}}^{\otimes n_{\text{bos}}}|^2$, since $\langle \Psi_{\text{sig}} | H(T_0) | \Psi_{\text{sig}} \rangle \geq E_0 \cdot (1 - o(1/m))$ for $m = O(n_{\text{bos}} \log(N))$. So the tensor network is lower bounded by $E_0^m |\lambda^{n_{\text{bos}}/p} v_{\text{sig}}^{\otimes n_{\text{bos}}}|^2$.

So, with these approximations we have lower bounded $\mathbb{E}[\langle \Psi_{\text{input}} | E_{PE} | \Psi_{\text{input}} \rangle]$.

We make some implementation remarks on the algorithm. The algorithm as described requires measuring whether we are in the symmetric subspace. Note that the input state Ψ_{input} need not be in the symmetric subspace. Such a projection can be done for example by phase estimating a Hamiltonian which is a sum of permutation operators. One can also omit this projection onto the symmetric subspace since our upper bounds on $H(G)$ holds both in the full Hilbert space and in the symmetric subspace.

We have considered the case that n_{bos} is an integer multiple of p . If $n_{\text{bos}} = kp + l$ for some integers k, l with $0 < l < p$, then one can use l ancilla qudits, and prepare an input state which is equal to

$$\frac{1}{|T_0|^k} |T_0\rangle^{\otimes k},$$

on kp qudits, tensored with a maximally entangled state between the remaining l qudits and the remaining l ancillas. The idea is that we get the additional quadratic improvement in overlap on kp of the qudits, and the remaining l ancilla only cost $1/\text{poly}(N)$ overlap since $l = O(1)$.

3. Chosen Input State: Modified Version

We now modify the algorithm 5 (and its amplified version) to obtain an algorithm for which we can prove the quadratic improvement over algorithm 4 without any assumption. Consider given T_0 . Let Δ be a p -th order tensor, chosen from the same distribution as G . Consider the tensor

$$\begin{aligned} T'_0 &= T_0 + x\Delta \\ &\equiv \lambda v_{\text{sig}}^{\otimes p} + G' \end{aligned} \tag{49}$$

for some real scalar x , where the tensor $G' \equiv G + x\Delta$ has Gaussian entries with variance of the entries equal to $1 + x^2$. We will assume $x = O(1)$; indeed later we will choose $x = o(1)$. Let us write $\Psi_{\text{input}}(T) \equiv |T|^{-n_{\text{bos}}/p} |T^{\otimes n_{\text{bos}}/p}\rangle$ and $E_{PE}(T)$ to denote the phase estimation operator E_{PE} for Hamiltonian $H(T)$.

We have

$$\begin{aligned} G &= \frac{1}{1+x^2}(G+x\Delta) + \frac{x}{1+x^2}(xG-\Delta) \\ &= \frac{1}{1+x^2}G' + \frac{x}{\sqrt{1+x^2}}\sqrt{1+x^2}(xG-\Delta) \\ &= \frac{1}{1+x^2}G' + \frac{x}{\sqrt{1+x^2}}\delta, \end{aligned} \tag{50}$$

where $\delta = \sqrt{1+x^2}(xG-\Delta)$. The two random variables G' and δ have vanishing covariance, so Eq. (50) expresses G as a scalar multiple of G' plus an additional Gaussian random variable δ which is independent of G' .

Given G and G' , let

$$G(y) = yG' + (1-y)G = (y + \frac{1-y}{1+x^2})G' + \frac{x(1-y)}{\sqrt{1+x^2}}\delta. \tag{51}$$

so that $G(y)$ linearly interpolates between G and G' .

The idea behind the algorithm is to take a given G as input, randomly perturb to produce G' , and then consider several input states $\Psi_{\text{input}}(\lambda v_{\text{sig}}^{\otimes p} + G(y))$ with different choices of $y \in [0, 1]$.

Let us first recall a property of normalization. We have $\Psi_{\text{input}} = \frac{1}{|T_0|^{n_{\text{bos}}/p}} |T_0\rangle^{\otimes n_{\text{bos}}/p}$. Let us write

$$Z = |T_0|^{2n_{\text{bos}}/p},$$

so $\Psi_{\text{input}} = Z^{-1/2} |T_0\rangle^{\otimes n_{\text{bos}}/p}$. As shown before, with high probability $Z^{1/2} = |T_0|^{n_{\text{bos}}/p} = O(N^{n_{\text{bos}}/2})$. Further, with high probability the fluctuations of $Z^{1/2}$ are $o(1)$ compared to its expectation value. So, from here on we will treat this normalization factor Z as a constant, i.e., of course the normalization depends on N, n_{bos} but we will ignore its dependence on G . We emphasize that we are not making any additional assumption here as with high probability the fluctuations are asymptotically negligible; we are simply choosing not to write the normalization explicitly. (Remark: indeed, all we really need is an upper bound on $|T_0|^{n_{\text{bos}}/p}$ that holds with high probability, since the normalization constant $|T_0|^{n_{\text{bos}}/p}$ always appears in the denominator.) Further, we will, without remarking on it further, treat

other normalization factors such as $|\Psi_{\text{input}}(\lambda v_{\text{sig}}^{\otimes p} + G(y))|$ as constants, and we will introduce other notation for those constants. Indeed, because we treat the normalization factors such as Z as constants, we will mostly work with un-normalized states which simplifies some of the calculations.

In an abuse of notation, let us define $\Psi_{\text{input}}(y) = \Psi_{\text{input}}(\lambda v_{\text{sig}}^{\otimes p} + G(y))$, and write $Z(y) = |\lambda v_{\text{sig}}^{\otimes p} + G(y)|^{2n_{\text{bos}}/p}$. Let us write $T_0(y) = \lambda v_{\text{sig}}^{\otimes p} + G(y)$.

Let $\Psi(y)$ denote the un-normalized state $|T_0(y)\rangle^{\otimes n_{\text{bos}}/p}$ so that $\Psi_{\text{input}}(y) = Z(y)^{-1/2}\Psi(y)$. Let us expand $\Psi(y)$ as a series in δ and define $\Psi^0(y)$ to denote the zeroth order term in δ .

As a warmup, let us consider $\langle \Psi^0(y) | E_{PE}(T'_0) | \Psi^0(y) \rangle$. We consider the higher order terms in δ later.

We expand the state $\Psi^0(y)$ as a series in $(y + \frac{1-y}{1+x^2})$. Doing this means that we express $\langle \Psi^0(y) | E_{PE}(T'_0) | \Psi^0(y) \rangle$ as a polynomial of degree $2n_{\text{bos}}$ which we write as $\sum_{i \geq 0} a_i (y + \frac{1-y}{1+x^2})^i$. The zero-th order term a_0 is simply equal to $\langle (\lambda v_{\text{sig}}^{\otimes p})^{\otimes n_{\text{bos}}/p} | E_{PE}(T'_0) | (\lambda v_{\text{sig}}^{\otimes p})^{\otimes n_{\text{bos}}/p} \rangle$, which is lower bounded in lemma 8. Hence,

$$a_0 \geq \lambda^{2n_{\text{bos}}/p} N^{n_{\text{bos}}} \Omega(1) \cdot c / (1 + c) \quad (52)$$

for $E_0 \geq (1 + c)E_{\text{max}}$.

Now we use a result about polynomials:

Lemma 9. *Let $p(z)$ be a polynomial of degree $2n_{\text{bos}}$. Let $[a, b]$ be an interval with $0 \leq a < b$. Then,*

$$\max_{z \in [a, b]} |p(z)| \geq \left| \frac{a+b}{b-a} \right|^{-2n_{\text{bos}}} \exp(-O(n_{\text{bos}})) |p(0)|. \quad (53)$$

Proof. We minimize $\max_{z \in [a, b]} |p(z)|$ over polynomials of degree $2n_{\text{bos}}$ with given value $p(0)$. Applying an affine transformation $z \rightarrow (2/(b-a))(z - (a+b)/2)$, this is equivalent to minimizing $\max_{z \in [-1, 1]} |p(z)|$ over polynomials of degree 2 with given value $p(z_0)$ for $z_0 = (a+b)/(a-b)$. We claim that this is minimized by

$$\frac{p(z_0)}{T_{2n_{\text{bos}}}(z_0)} T_{2n_{\text{bos}}}(z),$$

where $T_{2n_{\text{bos}}}$ is a Chebyshev polynomial. Proof of claim: suppose some other polynomial $q(z)$ has a smaller maximum absolute value on $[-1, 1]$ with $q(z_0) = p(z_0)$. Then the polynomial $p(z) - q(z)$ has a zero at z_0 but also has at least $2n_{\text{bos}}$ zeros on the interval $[-1, 1]$; this follows from the intermediate value theorem because $T_{2n_{\text{bos}}}$ has $2n_{\text{bos}} + 1$ extreme points on the interval which alternate signs. This gives a contradiction since $p(z) - q(z)$ is degree at most $2n_{\text{bos}}$.

So, $\max_{z \in [a, b]} |p(z)| \geq |p(z_0)| / |T_{2n_{\text{bos}}}(z_0)|$. If $0 \notin [a, b]$ then $|z_0| > 1$. We can bound $T_{2n_{\text{bos}}}(z_0)$ for $|z_0| > 1$ by $|z_0|^{2n_{\text{bos}}}$ times the sum of absolute values of coefficients of $T_{2n_{\text{bos}}}$. This sum of coefficients is bounded by $\exp(O(n_{\text{bos}}))$. \square

We apply this lemma to lower bound $\max_{y \in [0, 1]} |\sum_{i \geq 0} a_i (y + \frac{1-y}{1+x^2})^i|$. For $z = y + \frac{1-y}{1+x^2}$, with $b = 1$ and $a = (1+x^2)^{-1}$, this is $\max_{z \in [a, b]} |p(z)|$ with $p(z) = \sum_i a_i z^i$. So, using that $(a+b)/(b-a) \leq (2/x^2)$, we have

$$\max_{y \in [0, 1]} |\sum_{i \geq 0} a_i (y + \frac{1-y}{1+x^2})^i| \geq x^{4n_{\text{bos}}} \exp(-O(n_{\text{bos}})) |a_0|. \quad (54)$$

However, for purposes of an algorithm, we need to consider not just the maximum of this polynomial $p(z)$ over an interval, but also consider whether we can come close to this maximum by sampling it at some small number of discrete points. Fortunately we have the following lemma:

Lemma 10. *Let $p(z)$ be a polynomial of degree $2n_{\text{bos}}$. Let $[a, b]$ be an interval which does not contain 0. Then, with probability at least $1/2$, if one selects a random point z in the interval $[a, b]$ from the uniform distribution, we have*

$$|p(z)| \geq \left| \frac{a+b}{b-a} \right|^{-2n_{\text{bos}}} \exp(-O(n_{\text{bos}})) |p(0)|.$$

Proof. Apply an affine transformation $z \rightarrow (2/(b-a))(z - (a+b)/2)$, which maps a to -1 , b to $+1$ and 0 to $(a+b)/(a-b)$. Write $p(z) = \prod_i (z - z_i)$ where z_i are zeros of the polynomial; we multiply by $p(z)$ by a normalization constant so that the highest order coefficient is equal to 1. Then $\log(|p(z)|) = \sum_i \text{Re}(\log(z - z_i))$. Let A denote the average of this logarithm over the interval $[-1, +1]$; by calculus this is

$$A = \frac{1}{2} \sum_i \text{Re} \left((z - z_i) \log(z - z_i) - (z - z_i) \right) \Big|_{z=-1}^{z=+1}.$$

We claim that

$$\log(p((a+b)/(a-b))) - A \leq 2n_{bos} \cdot \left(\log((a+b)/(a-b)) + O(1) \right). \quad (55)$$

To show this let T_i denote a given term in the sum for A , i.e. $T_i = (1/2)\text{Re}((z - z_i) \log(z - z_i) - (z - z_i)) \Big|_{z=-1}^{z=+1}$. Let $D_i \equiv \text{Re}(\log((a+b)/(a-b) - z_i)) - T_i$. By considering various cases, we will show that $D_i \leq \log((a+b)/(a-b)) + O(1)$ which implies Eq. (55). Proof of claim: for $|z_i|$ less than or equal to some fixed constant (for example, $|z_i| \leq 10$), T_i is lower bounded by some absolute constant, so $D_i \leq \log((a+b)/(a-b)) + O(1)$. For $|z_i|$ larger than this fixed constant (for example, $|z_i| > 10$), T_i is at least $\log(|z_i|)$ minus some other absolute constant, so $D_i \leq \text{Re}(\log((a+b)/(a-b) - z_i) - \log(z_i)) + O(1) = \text{Re}(\log((a+b)/(a-b))) + O(1)$.

Further, for each i , we see that $\max_{z \in [-1, +1]} \text{Re}(\log(z - z_i))$ is upper bounded by $T_i + O(1)$. Hence, $\max_{z \in [-1, +1]} \log(|p(z)|)$ is upper bounded by $A + O(n_{bos})$. Hence, with probability at least $1/2$, for z chosen uniformly in $[-1, +1]$, we have that $\log(|p(z)|) \geq A - O(n_{bos})$.

Hence, with probability at least $1/2$, we have that $\log(|p(z)|) \geq A - 2n_{bos} \cdot (\log((a+b)/(a-b)) - O(n_{bos}))$. \square

So, noting that for $z = y + \frac{1-y}{1+x^2}$, a uniform choice of y on $[0, 1]$ is the same as a uniform choice of z on $[1/(1+x^2), 1]$, we have

Lemma 11. *For y chosen randomly from the uniform distribution on $[0, 1]$, the quantity $\langle \Psi^0(y) | E_{PE}(T'_0) | \Psi^0(y) \rangle$ is greater than or equal to $\langle (\lambda v_{\text{sig}}^{\otimes p})^{\otimes n_{bos}/p} | E_{PE}(T'_0) | (\lambda v_{\text{sig}}^{\otimes p})^{\otimes n_{bos}/p} \rangle \cdot \exp(-O(n_{bos})) x^{4n_{bos}}$.*

Using Eq. (52),

$$\langle \Psi^0(y) | E_{PE}(T'_0) | \Psi^0(y) \rangle \geq \lambda^{2n_{bos}/p} N^{n_{bos}} \exp(-O(n_{bos})) x^{4n_{bos}} \cdot c/(1+c). \quad (56)$$

We later will worry about optimizing over x . Increasing x will increase the right-hand side of Eq. (56), but it will also change the normalization constant $Z(y)$ that we must multiply by to correctly normalize the input state and also will worsen the threshold for recovery (since E_{max} will increase). For now, let us note for orientation that if we choose, for example, $x = 0.01$, then the tensor T'_0 is chosen from the same distribution as T_0 up to a multiplication by a factor of $(1 + 0.01^2)^{1/2}$ which is very close to 1. This leads to only a very small reduction in the threshold λ at which recovery is possible. At the same time, the change in normalization $(1 + 0.01^2)^{n_{bos}}$ is asymptotically negligible (for fixed n_{bos}) compared to the improvement which is polynomial in $N^{n_{bos}}$.

We now consider the additional terms depending on δ , i.e., due to the difference $\Psi(y) - \Psi^0(y)$. At first glance, we have not gained much by the above trick introducing T' since we still have this Gaussian random variable δ to consider. However, we have the advantage now that in $\langle \Psi(y) | E_{PE}(T'_0) | \Psi(y) \rangle$, the operator $E_{PE}(T'_0)$ does not depend on δ and the states in the bra and ket depend on δ only polynomially (treating the overall normalization as a constant).

Let us outline our approach. The main worry that we have is that this quantity $\langle \Psi(y) | E_{PE}(T'_0) | \Psi(y) \rangle$ will have a probability distribution that is peaked near some “small” value, i.e., much less than the value of the overlap at $\delta = 0$, rather than some “large” value, i.e., roughly comparable to the value of the overlap at $\delta = 0$. To deal with this, we will use a trick of adding additional noise to make the expectation value of the overlap large, i.e., roughly comparable to the value of the overlap at $\delta = 0$. Now we still need to worry about whether the probability function might have some large probability of having a small value. So, we will then appeal to a theorem of Carbery-Wright on “anti-concentration” of polynomials of Gaussian random variables, bounding the probability that the probability lies in some small interval. This theorem gives us a useful bound unless the variance of the polynomial is small; however, in that case we can show that the polynomial is likely to be close to its expectation value.

The trick of adding additional noise is as follows. We perturb the input state by adding additional Gaussian random noise. Let $\Psi(y, x')$ denote the unnormalized state

$$|\lambda v_{\text{sig}}^{\otimes p} + (y + \frac{1-y}{1+x^2})G' + \frac{x(1-y)}{\sqrt{1+x^2}}(\delta + \delta')\rangle^{\otimes n_{bos}/p},$$

where δ' is an additional tensor chosen from a Gaussian distribution with some standard deviation x' . So, the tensor $\delta + \delta'$ is sampled from some distribution with variance $1 + x'^2$. Let $Z(y, x') = |\Psi(y, x')|^2$. Let $\Psi_{\text{input}}(y, x') = Z(y, x')^{-1/2} \Psi(y, x')$ be the “perturbed input state”.

Then, we consider the expectation value over δ' of $\langle \Psi(y, x') | E_{PE}(T'_0) | \Psi(y, x') \rangle$; we consider this expectation value as a series in the variable $1 + x'^2$. Using the same treatment as above of this expectation value as a polynomial, we find that if x'^2 is chosen uniformly from the interval $[0, x^2]$ then, with probability at least $1/2$, the quantum expectation value $\langle \Psi(y, x') | E_{PE}(T'_0) | \Psi(y, x') \rangle$ is at least equal to the quantum expectation value with $\delta' = 0$ multiplied by $\exp(-O(n_{bos})) x^{8n_{bos}}$.

Hence,

Lemma 12. *With probability at least $1/4$, for uniform random choices of y and x'^2 on $[0, 1]$ and $[0, x^2]$, the expectation value over δ, δ' of $\langle \Psi(y, x') | E_{PE}(T'_0) | \Psi(y, x') \rangle$ is at least $\langle (\lambda v_{\text{sig}}^{\otimes p})^{\otimes n_{\text{bos}}/p} | E_{PE}(T'_0) | (\lambda v_{\text{sig}}^{\otimes p})^{\otimes n_{\text{bos}}/p} \rangle \cdot \exp(-O(n_{\text{bos}})) x^{8n_{\text{bos}}}$.*

We emphasize that throughout we are treating G' as fixed and considering δ, δ' as random. Now we use anti-concentration in the Gaussian random variables δ, δ' . The following lemma is a consequence of the Carbery-Wright theorem[20]

Lemma 13. *Let $p(z_1, \dots, z_n)$ be a polynomial of degree at most k in n independent normally distributed random variables. Then, for any $t \in \mathbb{R}$ and any $\delta > 0$ with $|t - \mathbb{E}[p]| > \delta$,*

$$\begin{aligned} \Pr[|p(z_1, \dots, z_n) - t| \leq \delta] &\leq \left(\frac{\delta}{|t - \mathbb{E}[p]| - \delta} \right)^{2/(2k+1)} O(k)^{2k/(2k+1)} \\ &\leq \left(\frac{\delta}{|t - \mathbb{E}[p]| - \delta} \right)^{2/(2k+1)} O(k), \end{aligned} \quad (57)$$

where the big- O notation here hides a universal constant.

As a corollary, choosing $t = 0$,

$$\Pr[|p| \leq \delta |\mathbb{E}[p]|] \leq \left(\frac{\delta}{1 - \delta} \right)^{2/(2k+1)} O(k) = O(\delta^{2/(2k+1)}) O(k). \quad (58)$$

Proof. Let $\text{Var}(p)$ denote the variance of $p(\cdot)$. As a trivial bound,

$$\Pr[|p(z_1, \dots, z_n) - t| \leq \delta] \leq \text{Var}(p) / \left(|t - \mathbb{E}[p]| - \delta \right)^2. \quad (59)$$

By Carbery-Wright,

$$\Pr[|p(z_1, \dots, z_n) - t| \leq \delta] \leq O(k) \cdot (\delta / \sqrt{\text{Var}(p)})^{1/k}. \quad (60)$$

Maximizing the bound from Eqs. (59,60) over $\text{Var}(p)$, Eq. (57) follows. \square

Hence, from Eq. (58),

Lemma 14. *Consider a choice of y, x' such that the expectation value over δ, δ' of the quantum expectation value $\langle \Psi(y, x') | E_{PE}(T'_0) | \Psi(y, x') \rangle$ is at least $\langle (\lambda v_{\text{sig}}^{\otimes p})^{\otimes n_{\text{bos}}/p} | E_{PE}(T'_0) | (\lambda v_{\text{sig}}^{\otimes p})^{\otimes n_{\text{bos}}/p} \rangle \cdot \exp(-O(n_{\text{bos}})) x^{8n_{\text{bos}}}$. Then, for random choice of δ, δ' , the quantum expectation value $\langle \Psi(y, x') | E_{PE}(T'_0) | \Psi(y, x') \rangle$ is at least*

$$\langle (\lambda v_{\text{sig}}^{\otimes p})^{\otimes n_{\text{bos}}/p} | E_{PE}(T'_0) | (\lambda v_{\text{sig}}^{\otimes p})^{\otimes n_{\text{bos}}/p} \rangle \cdot \exp(-O(n_{\text{bos}})) x^{8n_{\text{bos}}} \delta,$$

with probability at least $1 - O(n_{\text{bos}}) O(\delta)^{2/(2n_{\text{bos}}+1)}$.

Lemma 14 considers unnormalized input states. Dividing by normalization constant $Z(y, x')$ we can get a lower bound on the expectation value for the normalized input state $\Psi_{\text{input}}(y, x')$ by (for $E_0 \geq (1+c)E_{\text{max}}$):

$$\begin{aligned} &\frac{1}{Z(y, x')} \langle (\lambda v_{\text{sig}}^{\otimes p})^{\otimes n_{\text{bos}}/p} | E_{PE}(T'_0) | (\lambda v_{\text{sig}}^{\otimes p})^{\otimes n_{\text{bos}}/p} \rangle \cdot \exp(-O(n_{\text{bos}})) x^{8n_{\text{bos}}} \delta \\ &= \lambda^{2n_{\text{bos}}/p} \exp(-O(n_{\text{bos}})) x^{8n_{\text{bos}}} \delta \cdot \Omega(1) \\ &= \left(\frac{\lambda}{N^{-p/4}} \right)^{2n_{\text{bos}}/p} N^{-n_{\text{bos}}/2} x^{8n_{\text{bos}}} \delta \cdot \Omega(1) \\ &= \left(\frac{\lambda}{N^{-p/4}} \right)^{2n_{\text{bos}}/p} D(N, n_{\text{bos}})^{-1/2} x^{8n_{\text{bos}}} \delta \cdot \Omega(1). \end{aligned} \quad (61)$$

Finally, let us pick $x = 1/\log(N)$. Then, we consider the following algorithm 7.

We apply amplitude amplification to algorithm 7. We apply amplitude amplification under the assumption that indeed $\langle \Psi(y, x') | E_{PE}(T'_0) | \Psi(y, x') \rangle$ is at least $\langle (\lambda v_{\text{sig}}^{\otimes p})^{\otimes n_{\text{bos}}/p} | E_{PE}(T'_0) | (\lambda v_{\text{sig}}^{\otimes p})^{\otimes n_{\text{bos}}/p} \rangle \cdot \exp(-O(n_{\text{bos}})) x^{4n_{\text{bos}}} \delta$. From the theorems above, with high probability in δ' , this happens at least $1/4$ of the time. If the assumption does not hold, we re-sample y, x' and try again. Then, we find that for $E_0 \geq E_{\text{max}} \cdot (1 + O(1/\log(N)))$, with probability at least $1 - O(n_{\text{bos}}) \delta^{2/(2n_{\text{bos}}+1)}$, the expected runtime of the algorithm is at most $\text{poly}(N, n_{\text{bos}}, 1/(E_0 - E_{\text{max}}), \log(D(N, n_{\text{bos}})/\epsilon)) \exp(O(n_{\text{bos}})) \log(N)^{4n_{\text{bos}}} \left(\frac{N^{-p/4}}{\lambda} \right)^{n_{\text{bos}}/p} D(N, n_{\text{bos}})^{1/4} \delta^{-1/2}$.

Picking δ to be a slowly decaying function which is $o(1)$, such as $\delta = 1/\log(N)$, we find that

Algorithm 7 Quantum Algorithm (modified improved input state, unamplified version)

1. Choose random y, x' . Sample T'_0, δ' randomly. Prepare input state $\Psi_{\text{input}}(y, x')$.
 2. If the initial state is not in the symmetric subspace, report “failure”. If the state is in the symmetric subspace, apply phase estimation using $H(T_0)$. Let ψ be the resulting state. If the resulting eigenvalue is larger than $(E_0 + E_{\text{cut}})/2$, report “success”. Otherwise, report “failure”.
 3. If success is reported, measure and return $\langle \psi | a_\mu^\dagger a_\nu | \psi \rangle$.
-

Theorem 6. For $E_0 \geq E_{\text{max}} \cdot (1 + c)$, for any $c > 0$, with high probability, the expected runtime of the algorithm is at most

$$\text{poly}(N, n_{\text{bos}}, 1/(E_0 - E_{\text{max}}, \log(D(N, n_{\text{bos}})/\epsilon)) \exp(O(n_{\text{bos}})) \log(N)^{4n_{\text{bos}}} \left(\frac{N^{-p/4}}{\lambda}\right)^{n_{\text{bos}}/p} D(N, n_{\text{bos}})^{1/4}.$$

Remark: the reader may wonder why we picked $x = 1/\log(N)$ rather than a more slowly decaying function. Indeed, any choice of x which is $o(1)$ would suffice, and this would replace the factor $\log(N)^{4n_{\text{bos}}}$ with some other slowly decaying function.

C. Further Improvements

In this section, we sketch one improvement that might be practically useful even if it is not theoretically interesting. For that reason, we do not give any formal proofs in this section. As we have explained the algorithm above, once the algorithm succeeds in projecting onto the leading eigenvector, it measures some element of the single particle density matrix. It is necessary to repeat this operation, including both preparing the leading eigenvector and measuring, a total of $\text{poly}(N, 1/\log(\epsilon))$ times to measure the single particle density matrix to error ϵ , and, as explained, the algorithm needs to try again to prepare the leading eigenvector each time. The multiplicative overhead $\text{poly}(N, 1/\log(\epsilon))$ can be turned into an additive overhead using the non-destructive measurement technique introduced in Ref. 21. To implement this, we work in the full Hilbert space (the leading eigenvector preparation can be implemented in the symmetric subspace, but after preparing the leading eigenvector we work in the full Hilbert space). Then, we implement a two-outcome projective measurement on a single qudit (this measurement takes us outside the symmetric subspace though it does leave us in a subspace which is symmetric under permutation of a subset of $n_{\text{bos}} - 1$ of the qudits), such as $|\mu\rangle_1 \langle \mu|$ for some μ . Using the non-destructive measurement techniques of Ref. 21, we can alternate this two-outcome projective measurement with a projector onto the leading eigenvector (implemented approximately using phase estimation as above) to recover the leading eigenvector. See Ref. 21 for more details.

V. NETWORKS

In this paper, we have used the language of tensor networks to perform some of the estimates. In this section, we explore the relation to tensor networks further. Let us consider the simplest task of detection so that the spectral algorithms need only to estimate the largest eigenvalue of $H(T_0)$. In this case, it suffices to evaluate $\text{tr}(H(T_0)^m)$ to sufficiently large power m . This trace can be written as a sum of tensor networks: the sum is present because $H(T_0)$ is a sum of terms. If we restrict to the minimum n_{bos} (i.e. $n_{\text{bos}} = p/2$ for even p and $n_{\text{bos}} = p - 1$ for odd p) then no such sum is present.

Consider a given tensor network $N(T)$ in the sum with N_T tensors in the network. We write the network $N(T)$ to indicate that the graph and ordering of vertices is given, but that the tensor corresponding to each vertex will be determined by some tensor T (for some vertices it will be T and for some it will be \bar{T}). We have $N_T = m$ for even p or $2m$ for odd p . Thus, the network has $N_E = pN_T/2$ edges. The tensor network evaluates to some scalar that we write $\text{Val}(N(T))$. To get oriented, suppose that we insert $\lambda v_{\text{sig}}^{\otimes p}$ for every tensor in the network. In this case, the scalar $\text{Val}(N(\lambda v_{\text{sig}}^{\otimes p}))$ is equal to $\lambda^{N_T} N^{N_E}$. Suppose instead that we insert G or \bar{G} for every tensor in the network. It is possible that $\mathbb{E}[\text{Val}(N)]$ is zero and it is possible that it is nonzero. However, let us compute $\mathbb{E}[|\text{Val}(N(G))|^2]$. This expectation value can be evaluated using the method of appendix A. Note that $\text{Val}(N(G))^2$ is equal to the evaluation of some tensor network with $2N_T$ tensors; this network is given by two copies of N , with all the tensors complex conjugated in one of the two copies. We refer to the two copies as the first and second copy. We evaluate this network with $2N_T$ tensors by summing over pairings. Consider just the pairing in which we pair each tensor in the first copy with the corresponding tensor in the second copy. It is easy to see that this pairing contributes

N^{N_E} to the expectation value. Hence, since all pairings contribute positively to the expectation value, we have that $\mathbb{E}[|\text{Val}(N)|^2]^{1/2} \geq N^{N_E/2}$.

So, we see that $\mathbb{E}[|\text{Val}(N(G))|^2]^{1/2} \gg \text{Val}(N(\lambda v_{\text{sig}}^{\otimes p}))$ for $N^{N_E/2} \gg \lambda^{N_T} N^{N_E}$, i.e., for $\lambda \ll N^{-p/4}$. However, this does not yet imply that $\mathbb{E}[|\text{Val}(N(G))|^2]^{1/2} \gg \mathbb{E}[\text{Val}(N(T_0))]$, because, in particular, $\mathbb{E}[\text{Val}(N(T_0))]$ may be larger than $\text{Val}(N(\lambda v_{\text{sig}}^{\otimes p}))$. However, for any network N suppose we insert $\lambda v_{\text{sig}}^{\otimes p}$ on some set of tensors S in the network and insert G for the remaining tensors in the network. In this case, the expectation value is some number given by summing over pairings of the $N_T - q$ insertions of G . For a given pairing P , consider the following pairing Q on the tensor network which computes $\mathbb{E}[|\text{Val}(N(G))|^2]$. For each pair of tensors T_1, T_2 in P , we pair the first copy of T_1 with the first copy of T_2 and we pair the second copy of T_1 with the second copy of T_2 . On the other hand, if a tensor T is not paired in P (so T is in S), we pair the first copy of T with the second copy of T . Using this pairing, we again find, for $\lambda \ll N^{-p/4}$, for the given choices of inserting $\lambda v_{\text{sig}}^{\otimes p}$ and G into tensors in the network for $\mathbb{E}[\text{Val}(N(T_0))]$, that the contribution of pairing Q to $\mathbb{E}[|\text{Val}(N(G))|^2]$ is much larger than the square of the contribution of pairing P to $\mathbb{E}[\text{Val}(N(T_0))]$.

So, in this regime, the rms fluctuations in the expectation value are large compared to the expectation value itself, i.e., it seems like detection is not possible. So, to what extent does this result on tensor networks imply that spectral algorithms cannot perform inference for $\lambda < N^{-p/4}$? Of course, such inference is possible by taking n_{bos} large so certainly no general impossibility statement can be made. One way in which one can evade this result is that the result is for a specific tensor network, while the algorithms compute a sum over tensor networks. Hence, it is possible that by taking a large number of networks in the sum, the rms fluctuations in the sum over networks are small compared to the expectation value of the sum of tensor networks. This seems to be what happens for the specific algorithms here by taking n_{bos} large giving a large number of networks in the sum. This connection to tensor networks may be worth exploring further.

VI. DISCUSSION

The main new results of this paper are a sequence of spectral algorithms for odd p , and a quantum algorithm to compute the single particle density matrix of some vector in an eigenspace with large eigenvalue. The quantum algorithm could be applied to other matrices other than $H(T_0)$ considered here such as Y from Ref. 1; it is not certain though if the speedup using an input state chosen based on T_0 would apply to such other matrices.

In this paper, we have related the spectrum for odd p to that for even $q = 2(p-1)$, up to an overall scale factor \sqrt{N} , at least in the case of Gaussian distributed entries of G . This enables us to prove correctness of the algorithm for odd p . Also, for $n_{\text{bos}} = p-1$, the spectrum for $q = 2(p-1)$ is given by random matrix theory, removing a logarithmic factor present for odd p in some previous work. See corollary 2. We have not considered the case of entries of T_0 being chosen from a discrete distribution, however, we believe that a similar reduction from odd p to even $q = 2(p-1)$ could be proven in that case.

We have used entries of G chosen from a complex Gaussian distribution in the case of odd p to simplify the proof of the bounds on the spectrum for odd p . We conjecture however that a modification discussed above to cancel certain “pairing terms” would work for real entries of G .

-
- [1] A. S. Wein, A. E. Alaoui, and C. Moore, arXiv preprint arXiv:1904.03858 (2019).
 - [2] A. Montanari and E. Richard, in *Advances in Neural Information Processing Systems* (2014), pp. 2897–2905.
 - [3] T. Lesieur, L. Miolane, M. Lelarge, F. Krzakala, and L. Zdeborova, in *2017 IEEE International Symposium on Information Theory (ISIT)* (IEEE, 2017), URL <https://doi.org/10.1109%2Fisit.2017.8006580>.
 - [4] S. B. Hopkins, T. Schramm, J. Shi, and D. Steurer, in *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing - STOC 2016* (ACM Press, 2016), URL <https://doi.org/10.1145%2F2897518.2897529>.
 - [5] V. V. S. P. Bhattiprolu, M. Ghosh, E. Lee, V. Guruswami, and M. Tulsiani, CoRR **abs/1611.05998** (2016), 1611.05998, URL <http://arxiv.org/abs/1611.05998>.
 - [6] V. Bhattiprolu, M. Ghosh, V. Guruswami, E. Lee, and M. Tulsiani, in *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)* (IEEE, 2017), URL <https://doi.org/10.1109%2Ffocs.2017.97>.
 - [7] R. Werner, in *Quantum Probability and Related Topics* (WORLD SCIENTIFIC, 1992), pp. 349–381, URL https://doi.org/10.1142%2F9789814354783_0024.
 - [8] C. V. Kraus, M. Lewenstein, and J. I. Cirac, *Physical Review A* **88** (2013), URL <https://doi.org/10.1103%2Fphysreva.88.022335>.
 - [9] F. G. S. L. Brandão and A. W. Harrow, *Communications in Mathematical Physics* **342**, 47 (2016), URL <https://doi.org/10.1007%2Fs00220-016-2575-1>.

- [10] S. Aaronson and L. Chen, arXiv preprint arXiv:1612.05903 (2016).
- [11] M. L. Mehta, *Random matrices*, vol. 142 (Elsevier, 2004).
- [12] J. Demmel, I. Dumitriu, and O. Holtz, *Numerische Mathematik* **108**, 59 (2007), URL <https://doi.org/10.1007/2Fs00211-007-0114-x>.
- [13] G. H. Low and I. L. Chuang, *Phys. Rev. Lett.* **118**, 010501 (2017), 1606.02685v2.
- [14] G. H. Low and I. L. Chuang (2016), 1610.06546.
- [15] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma (2014), pp. 283–292, 1312.1414.
- [16] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, *Phys. Rev. Lett.* **114**, 090502 (2015), 1412.4687.
- [17] D. W. Berry, A. M. Childs, and R. Kothari, in *2015 IEEE 56th Annual Symposium on Foundations of Computer Science* (2015), pp. 792–809, ISSN 0272-5428, 1501.01715.
- [18] A. Y. Kitaev, A. Shen, and M. N. Vyalii, *Classical and quantum computation*, vol. 47 (American Mathematical Society Providence, 2002).
- [19] G. Brassard, P. Hoyer, M. Mosca, and A. Tapp, *Contemporary Mathematics* **305**, 53 (2002).
- [20] A. Carbery and J. Wright, *Mathematical Research Letters* **8**, 233 (2001), URL <https://doi.org/10.4310/2Fmrl.2001.v8.n3.a1>.
- [21] D. Wecker, M. B. Hastings, N. Wiebe, B. K. Clark, C. Nayak, and M. Troyer, *Physical Review A* **92** (2015), URL <https://doi.org/10.1103/2Fphysreva.92.062318>.
- [22] M. B. Hastings, *Communications in Mathematical Physics* **351**, 387 (2016), URL <https://doi.org/10.1007/2Fs00220-016-2791-8>.

Appendix A: Contracting Tensor Networks

In this appendix, we briefly explain how to estimate the average of a tensor network over tensors with entries chosen from a Gaussian distribution, either real or complex, in both cases with zero mean and unit variance. This is a fairly standard technique, and Ref. 22 is an example that uses this technique heavily and can be used as a reference for some of the ideas.

A tensor network is defined by some graph, with each vertex being labelled by some tensor. The degree of each vertex is equal to the order of the corresponding tensor. If the tensors are not symmetric, then for each vertex we also specify some ordering on the edges attached to that vertex.

A tensor network L with m tensors evaluates to a scalar $\text{Val}(L)$ that we will call the value of the tensor network. The value is computed as follows: for each way of assigning an index to each edge (there are $N^{pm/2}$ such choices if there are m tensors in the network each of order p), for every vertex v we compute some scalar; this scalar is equal to a particular entry of the tensor T^v corresponding to that vertex v . The entry that is chosen is given by the indices on the edges attached to that vertex (if the tensor is not symmetric, one uses the ordering on edges of that vertex to determine the order of indices in the tensor). Then, the product of this scalar over all vertices v gives some scalar for each way of assigning an index to each edge. Finally, the sum of this scalar over all possible assignments to edges is $\text{Val}(L)$, so that

$$\text{Val}(L) = \sum_{\text{assignments}} \prod_{\text{vertex } v} (T^v)_{\text{edges}}, \quad (\text{A1})$$

where the sum is over choices of indices, the product is over vertices v , and $(T^v)_{\text{edges}}$ denotes the entry of a tensor corresponding to the given vertex v determined by the indices on edges attached to that vertex.

Thus given m tensors of order p , the value is an m -th order polynomial in the entries. Equivalently, it is a sum over mp different indices (one for each leg attached to each tensor), with $mp/2$ δ -functions (one for each edge) constraining any pair of legs on two different tensors which are connected by an edge to have the same index.

Consider a tensor network in which vertices are labelled by G or \bar{G} in the complex ensemble or by G in the real ensemble. We wish to average the value of this network over Gaussian choices of G . Let us first consider the case in which the tensors are *not* symmetrized. By Wick's theorem, this is computed by a sum over pairings. Each pairing is either (if all tensors are chosen from a real Gaussian distribution) a way of partitioning the set of tensors into distinct pairs or (if tensors are chosen from the complex distribution) a way of pairing the set into distinct pairs such that we only pair G with \bar{G} . If there are m tensors in the network, then in the real case there are $m!!$ pairings and in the complex case there are $(m/2)!$ pairings assuming that there are $m/2$ tensors G and $m/2$ tensors \bar{G} (if not, the expectation value vanishes).

For each pairing, we compute a scalar as follows. For every pair T and T' , replace the two tensors T_{μ_1, \dots, μ_p} and T'_{ν_1, \dots, ν_p} in the polynomial with a product of δ -functions $\prod_{a=1}^p \delta_{\mu_a, \nu_a}$. After making this replacement, sum over all indices. The result will be equal to $N^{n_{loop}}$ where n_{loop} is some positive integer. Summing $N^{n_{loop}}$ over all pairings computes the expectation value of the value of the network.

This can conveniently be represented graphically as follows: given a tensor network, and given any tensors that are paired, “cut” those two tensors out of the network, leaving $2p$ open edges in the network (the edges that were attached to those tensors). Then, attach those edges to each other, attaching each of the p edges that were attached to the first tensor to the corresponding one that was attached to the second tensor. After doing this for every pair, what is left is some collection of n_{loop} closed loops.

If the tensors are symmetrized, for each pair we must sum over $p!$ different ways of attaching the edges that were attached to the first tensor to those that were attached to the second tensor.

One can easily also consider the case in which we insert tensors $v_{\text{sig}}^{\otimes p}$ into the network. This can be done by applying the rotation to make $v_{\text{sig}} = (\sqrt{N}, 0, 0, \dots)$. Then, replace each tensor $v_{\text{sig}}^{\otimes p}$ with \sqrt{N} multiplying a product of δ -functions on its edge, $\prod_{a=1}^p \delta_{\mu_a, 1}$. We can represent this graphically simply by writing those δ -functions as open edges; again one sums $N^{n_{loop}}$ over pairings to compute the expectation value of the value of the network, but any open edge with open ends does not count as a closed loop.