Estimating the mana of magical matrix product operators

Siddhartha Harmalkar

November 2020

Contents

1 Introduction

Quantum error correction codes based on stabilizer circuits provide a robust theory of fault-tolerant, albeit non-universal, quantum computation. To achieve universal quantum computation, an extra resource of non-stabilizer gates or states prepared using such gates - "magic" states - can be injected into the system. A resource theory of this model of universal quantum computation has been developed to quantify the amount of magical resource required to perform tasks such as preparing a magical state on a quantum computer [1].

Using this resource theory to quantify such tasks involves the computation of measures of magic - the *mana* and *relative entropy of magic*, for example [1] - which are exponentially hard to compute in the size of the system. Here we introduce a new measure which allows for a polynomial-time computation when applied to matrix product operators.

2 Approximating the mana $\mathcal{M}$ with a related quantity $\mathcal{N}$

Stabilizer states are the only pure states with non-negative Wigner function in Hilbert spaces of odd dimension [2]. All mixtures of stabilizer states therefore have non-negative Wigner function[1]. Moreover, the degree to which a state's wigner function is negative gives us information about how much magical resource is required to create it: The *mana* of a mixed state

$$\mathcal{M}(\rho) = \ln \sum_u |W_\rho(u)| \tag{1}$$

quantifies the amount of magical resource needed to prepare magical states using procedures such as magical state distillation [1], where $W_u(\rho)$ is the Wigner function of the state ρ - see Section A.3 for details. Computation of the mana of an arbitrary

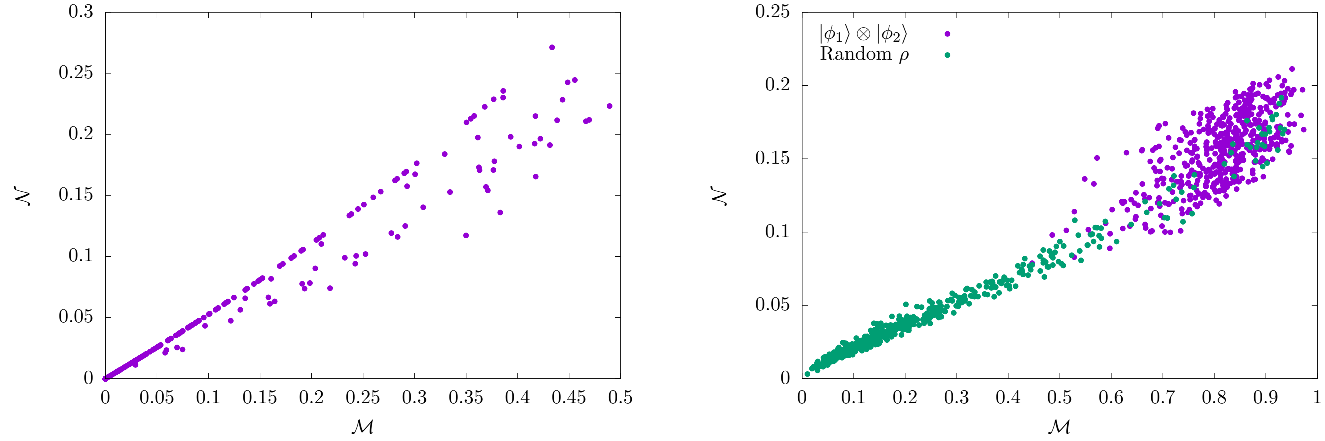
[1] However, there do exist mixed states with non-negative Wigner function which are not mixtures of stabilizer states - see section V of [2]

state is exponentially hard in the size of the system, as it requires us to know $W_u(\rho)$ for all d^{2n} possible strings $u \in \mathbb{Z}_d^{2n}$, an intractable computation for large system sizes.

Instead of calculating the mana, we can approximate the magic of a state with the quantity

$$\mathcal{N}(\rho) = \sqrt{\sum_{u \in N(\rho)} W_\rho(u)^2}, \tag{2}$$

where $N(\rho)$ is the set of strings satisfying $W_u(\rho) < 0$. While $\mathcal{N}$ does not satisfy the same properties as $\mathcal{M}$ which would allow us to use it to quantify the amount of resource needed for tasks such as magic state distillation, it is clearly larger for more states with larger mana, and is emperically highly correlated with $\mathcal{M}$ for a generic mixed state:



(a) $\mathcal{M}$ vs. $\mathcal{N}$ for mixed single qutrit states

(b) $\mathcal{M}$ vs. $\mathcal{N}$ for mixed two qutrit states of two types - product states and arbitrary mixed states

Figure 1: Comparison of $\mathcal{M}$ and $\mathcal{N}$ on a variety of mixed states. Each mixed state in the plots is randomly generated according to the algorithm described in Section B out of N randomly sampled pure states, where N was uniformly sampled between 1 and 20.

We find that $\mathcal{N}$ is much easier to work with than $\mathcal{M}$[2]. In particular, we can efficiently approximate $\mathcal{N}$ by noticing that it is the distance between the state ρ and the space of operators[3] with non-negative Wigner function:

$$\mathcal{N}(\rho) = \min_{\mathcal{M}(\sigma)=0} d(\rho, \sigma), \tag{3}$$

where

$$d(\rho, \sigma) \equiv \sqrt{\sum_{u \in \mathbb{Z}_d^{2n}} \left(W_\rho(u) - W_\sigma(u)\right)^2} \tag{4}$$

is the distance between two operators when viewed as vectors in the space of Wigner function coefficients. Therefore, we can approximate $M_2(\rho)$ by minimizing $d(\rho, \sigma)$ in the space of non-negative Wigner function operators σ.

3 Efficient approximation of $\mathcal{N}$ for a matrix product operator

Given a local operator basis A_i of the space dual to the Hilbert space $\mathcal{H}$, the tensor product $A_{i_1} \otimes A_{i_2} \otimes \cdots \otimes A_{i_n}$ forms a basis of the n-site dual space $(\mathcal{H}^*)^{\otimes n}$. A matrix product operator (MPO) is an operator $\mathcal{O} \in (\mathcal{H}^*)^{\otimes n}$ of the form

$$\mathcal{O} = \sum_u \left(\sum_\alpha M^{(1)u_1}_{\alpha_1 \alpha_2} M^{(2)u_2}_{\alpha_2 \alpha_3} \cdots M^{(n)u_n}_{\alpha_n \alpha_{n+1}} \right) A_{u_1} \otimes A_{u_2} \otimes \cdots \otimes A_{u_n}, \tag{5}$$

where $\{M^{(i)u}\}_{u=1}^{d}$ is a collection of d matrices "at site i", indexed by u, with dimension[4] $D \times D$. D is referred to as the *bond dimension* of the MPO.

[2] This is reminiscent of how the standard deviation of a distribution, $\mathrm{E}[(X - \mathrm{E}[X])^2]^{1/2}$, is an easier quantity to work with than the more direct measure of the degree to which samples deviate from the mean: $\mathrm{E}[|X - \mathrm{E}[X]|]$

[3] Note that not all operators satisfying $W_u(\rho) \geq 0 \; \forall u$ represent states - only those with $\operatorname{Tr} \rho = 1$. Hermiticity is gauranteed, though, due to the fact that $W_u(\rho) \in \mathbb{R}$.

[4] The first and last matrices have dimension $1 \times D$ and $D \times 1$, respectively - α_1 and α_{n+1} are dummy indices which only take on one value

The structure of matrix product operators can be exploited to reduce computational costs which are generically exponential in the d to ones which are linear in the d and polynomial in D. The procedure of estimating $\mathcal{N}(\rho)$ outlined above allows for such a reduction in computation time if ρ is an MPO and the variational state σ is also an MPO, as we can efficiently perform the contractions required in the computation of $d(\rho,\sigma)$ and its derivatives with respect to the degrees of freedom in σ at a single site - see Section C

The procedure is then to sweep through sites in a DMRG-like fashion, optimizing $d(\rho,\sigma)$ with respect to the degrees of freedom at each site in σ, under the constraint that σ remains in the space of states with non-negative Wigner function:

Algorithm for approximating $\mathcal{N}(\rho)$: Given a matrix product operator $\rho = \sum_\alpha \otimes_k M^{(k)u_k}_{\alpha_k\alpha_{k+1}} A_k$ with bond dimension D_ρ, where A_i are the phase space operators[5] (18):

1. Let $\sigma = \sum_{u,\alpha_i} \otimes_k \sigma^{(k)u_k}_{\alpha_k\alpha_{k+1}} A_{u_k}$ be a random MPO with bond dimension D_σ such that $\sigma^{(k)u}_{\alpha\beta}\ \forall k \in \mathbb{Z}_n, u \in \mathbb{Z}_d^{2n}, \alpha, \beta \in Z_{D_\sigma}$.
2. Set $i \leftarrow 1$. While $i \leq n$:
   (a) Precompute the contractions required in the computation of the derivatives $\frac{\partial d(\rho,\sigma)}{\sigma^{(i)a}_{\alpha\beta}}$ - see Section C.

   (b) Let $\mathcal{F}$ be the set of $d \times D_\sigma \times D_\sigma$ sized tensors[6] with non-negative elements: $x \in \mathcal{F} \Rightarrow x^u_{\alpha,\beta} \geq 0\ \forall u \in \mathbb{Z}_d, \alpha, \beta \in \mathbb{Z}_{D_\sigma}$, and $\hat{\sigma}(x)$ be the function which replaces the matrix in σ at site i with the matrix x:

   $$\hat{\sigma} : x \in \mathcal{F} \mapsto \sum_u \left(\sum_\alpha \sigma^{(1)u_1}_{\alpha_1\alpha_2} \cdots \sigma^{(i-1)u_{i-1}}_{\alpha_{i-1}\alpha_i} x^{u_i}_{\alpha_i\alpha_{i+1}} \sigma^{(i+1)u_{i+1}}_{\alpha_{i+1}\alpha_{i+2}} \cdots \sigma^{(n)u_n}_{\alpha_L\alpha_{L+1}} \right) A_{u_1} \otimes A_{u_2} \otimes \cdots \otimes A_{u_n} \tag{6}$$

   (c) Solve the constrained optimization problem $\sigma^{(i)*} = \underset{x\in\mathcal{F}}{\operatorname{argmin}}\ d(\rho, \hat{\sigma}(x))$ numerically, using a procedure such as [3].

   (d) $\sigma^{(i)} \leftarrow \sigma^{(i)*}$

   (e) $i \leftarrow i+1$
3. Repeat step 2 until $d(\rho,\sigma)$ asymptotes[7]
4. Increase the bond dimension D_σ. Repeat steps 2 and 3 until $d(\rho,\sigma)$ asymptotes with respect to bond dimension (or the computational resources required in step 2 become intractable).

4 Caveats

A few things should be noted about this optimization:

- Even if the optimization is done perfectly, the final distance $d(\rho,\sigma)$ will only be an upper bound for $\mathcal{N}(\rho)$. This is because the set of states we can reach in our optimization algorithm is only a subset of the set of operators with non-negative Wigner function, due to the fact that σ has a matrix product structure with fixed bond dimension and $\sigma^{(i)u_i}_{\alpha\beta} \geq 0\ \forall i, u_i, \alpha, \beta$, which is more restrictive than the requirement that σ have non-negative Wigner function representation, i.e $W_u(\sigma) = \sum_\alpha \prod_i \sigma^{(i)u_i}_{\alpha_i,\alpha_{i+1}} \geq 0\ \forall u \in \mathbb{Z}_d^{2n}$. In practice, we alleviate the first restriction slightly by using larger bond dimensions until we either see our estimate for $\mathcal{N}$ asymptote or we run out of resources.
- The space of MPOs with fixed bond dimension is not convex - adding two MPOs generally increases the bond dimension of the MPO. This means that our contrained optimization problem might settle on local minima.

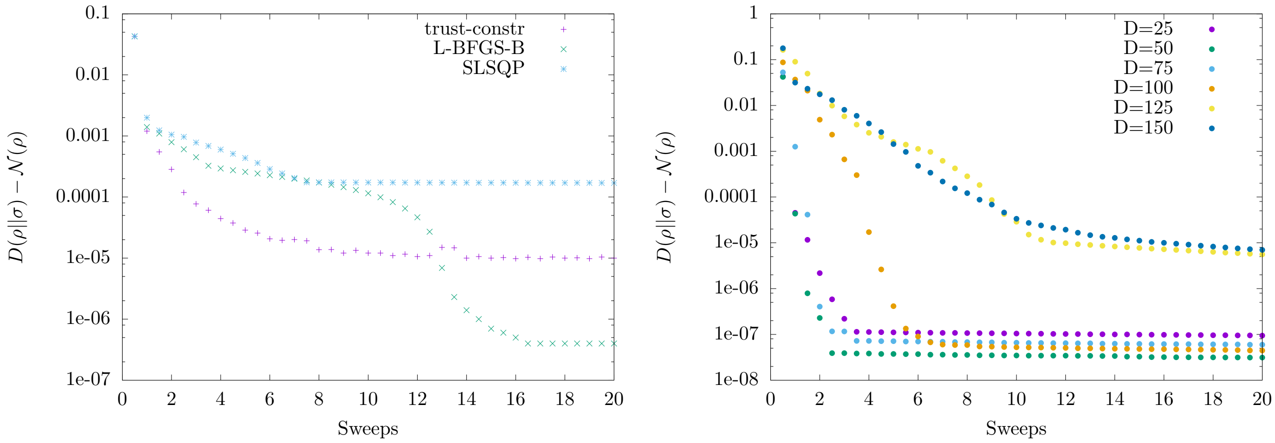
[5] If ρ is represented in a different basis, perform a local change of basis to find its Wigner representation first: Let $\rho = \sum_\alpha \otimes_k M^{(k)u_k}_{\alpha_k\alpha_{k+1}} B_k$, where B_k is some local operator basis. Then this consists of performing the change of basis $\rho^{(i)u}_{\alpha\beta} = \sum_v U_{uv} \rho^{(i)v}_{\alpha\beta}$, where $U_{uv} = \mathrm{Tr}[A_u B_v]$

[6] Since the first and last bond indices are dummy indices, the tensors have size $d \times 1 \times D_\sigma$ for $i = 1$, and $d \times D_\sigma \times 1$ for $i = n$

[7] Note that optimization algorithms might appear to asymptote but can later perform better if allowed to continue

5 Results

Running the above algorithm with the L-BFGS-B constrained optimization method (As seen below, the L-BFGS-B method [4, 3] for constrained optimization works well in practice, outperforming an interior point method solver ("trust-constr") and a sequential least squares quadratic programming method ("SLSQP")) on a random magical mixed state ρ generated using $N = 20$ randomly sampled pure states as described in Section B shows that the approximation for $\mathcal{N}(\rho)$ can get better with larger bond dimension, but starts to require more sweeps to asymptote after a certain point.



(a) Distance between a randomly generated 2-site MPO and the optimized MPO vs. sweeps for various optimizers.

(b) Runs with L-BFGS-B for varying bond dimensions

Figure 2: Results from running the algorithm for approximating $\mathcal{N}(\rho)$ of a random mixed state ρ. Each run consists of 20 sweeps, and generates 40 points due to two optimizations being done during each sweep - one for each site.

6 Next steps

We need to analyze optimization function on a variety of test cases to understand how it performs. for small n, we can exactly compute the true value of the optimal cost function and compare. Some ideas of families of states to look at:

- $S_\rho(\alpha) = \alpha\rho + \frac{(1-\alpha)}{d}I$, where ρ is a random pure state. As we vary α from 0 to 1, the state should start to gain more magic as soon as it clears the stabilizer hull. Sampling a set of random ρ uniformly and computing the average and standard deviation of the error in the result after optimization as a function of n will give a good estimate of how well the optimizer does as the system size increases.

- $S_{a,b}(\alpha) = aS_1 + bS_2 - \alpha I/d^n$: Here the idea is to mix two stabilizer states S_1 and S_2 and walk in the negative direction of identity. This is a good test because the distance $d(S_{a,b}(\alpha), I/d^n)$ should be linear in α even for large n.

- MPSs with truncated bond dimensions: Create a random state, turn it into an MPS with truncated bond dimension, or just do SVDs at each site and project onto the first few singular components. Truncate enough so that the mana is tractable even at large n, to compute the error in optimization exactly.

7 Future work

- Extend results to larger system sizes
- Estimate the magic of the ground state of a CFT, such as the spin-1 Potts model near criticality, for large system sizes
- Improve the algorithm for approximating $\mathcal{N}(\rho)$. Some ideas for doing so:
  - Understand why the constrained optimization procedure requires significantly more sweeps to converge when the bond dimension of σ is increased
  - Understand why the value which the procedure asymptotes to does not always decrease with increasing bond dimension (what's most likely happening is that the optimization is taking so long to asymptote that it seems to asymptote early at a non-optimal value)
  - Initialize the optimization with a clever initial condition, which might help alleviate the issues above

  - Relax the constraint that all matrix elements of $\sigma^{(i)u}$ be non-negative, while ensuring that $W_u(\sigma) > 0 \; \forall u \in \mathbb{Z}_d^{2n}$. This would allow for better approximation of $\mathcal{N}(\rho)$

- Attempt to optimize the *relative entropy* [1] $\rho \log \rho - \rho \log \sigma$ with the constraint that σ is a stabilizer state, which can more directly quantify the magic of ρ. This would require including a normalization constraint in the optimization. It's not clear how to ensure that σ is a stabilizer state during the optimization, however, due to the fact that some states with nonnegative Wigner representation are not stabilizers, so we would most likely have to just approximate the stabilizer hull with the space of nonnegative Wigner representation states as well.

A Stabilizer formalism

A.1 Heisenberg-Weyl operator basis

Mixed states of qubits can be nicely represented using the familiar pauli matrices σ_i of 2-component quantum mechanics, which can be used to construct a set of 4 operators which form a basis of hermitian 2×2 matrices: $\sigma_\alpha = \frac{1}{2}(1, \sigma_i)$. The factor of $\frac{1}{2}$ gives orthonormality with respect to the frobenius norm: $(\sigma^i)^2 = 1$ and $\operatorname{Tr} \sigma^i = 0$ imply that $\operatorname{Tr}[(\sigma_\alpha)^\dagger \sigma_\beta] = \delta_{\alpha\beta}$, allowing for any mixed state to be written as $\rho = \sum_\alpha \operatorname{Tr}[\sigma_\alpha \rho]\sigma_\alpha$. A basis for mixed states of n qubits is then given by "strings" of pauli matrices tensored together: the string $u = (u_1, u_2, \ldots, u_n) \in \mathbb{Z}_4^n$ indexes the basis element $\otimes_{i=1}^n \sigma_{u_i}$.

For a general qudit, a set of d^2 operators is needed to decompose qudit states. If $d > 2$ is prime, one choice of operators is the so-called *generalized pauli matrices* or *Heisenberg-Weyl operators*:

$$T_u = \omega^{-(d+1)ab/2} Z^a X^b = \omega^{-2^{-1}ab} Z^a X^b, \tag{7}$$

where $u = (a, b) \in \mathbb{Z}_d \times \mathbb{Z}_d$, $\omega = e^{2\pi i/d}$ is the d-th root of unity, $2^{-1} = \frac{d+1}{2}$ is the multiplicative inverse of 2 modulo d if d is odd (which it is for prime $d > 2$), and X and Z are the *boost* and *shift* operators, respectively, whose action on the computational basis is

$$X\left|k\right\rangle = \left|k \oplus 1\right\rangle \text{ and } Z\left|k\right\rangle = \omega^k \left|k\right\rangle, \tag{8}$$

where $\oplus$ denotes addition modulo d. Note that any operation on a and b can be thought of as being modulo d, because $\omega^d = X^d = Z^d = 1$:

$$X^d\left|k\right\rangle = \left|k \oplus d\right\rangle = \left|k\right\rangle, Z^d\left|k\right\rangle = \omega^d\left|k\right\rangle = \left|k\right\rangle \tag{9}$$

The boost and shift operators are unitary:

$$\left\langle i\right| X^\dagger X \left|j\right\rangle = \left\langle i \oplus 1 | j \oplus 1\right\rangle = \delta_{ij} \Rightarrow X^\dagger = X^{-1},\ \left\langle i\right| Z^\dagger Z \left|j\right\rangle = \left\langle i\right| \omega^{-i}\omega^j \left|j\right\rangle = \omega^{j-i}\delta_{ij} = \delta_{ij} \Rightarrow Z^\dagger = Z^{-1} \tag{10}$$

and have commutation structure $[Z, X] = (\omega - 1)XZ$:

$$XZ\left|k\right\rangle = \omega^k \left|k \oplus 1\right\rangle,\ ZX\left|k\right\rangle = \omega^{k+1}\left|k \oplus 1\right\rangle \Rightarrow ZX = \omega XZ \tag{11}$$

They are orthonormal up to a factor of d:

$$\operatorname{Tr}[X^a] = \sum_k \left\langle k | k \oplus a\right\rangle = d\delta_{a,0},\ \operatorname{Tr}[Z^b] = \sum_k (\omega^b)^k = d\delta_{b,0} + \frac{1 - \omega^{bd}}{1 - \omega^b}(1 - \delta_{b,0}) = d\delta_{b,0} \Rightarrow \operatorname{Tr}[Z^a X^b] = d\delta_{a,0}\delta_{b,0}, \tag{12}$$

which implies that the generalized pauli operators satisfy $\operatorname{Tr}[T_{(a,b)}] = d\delta_{a,0}\delta_{b,0}$ and

$$\operatorname{Tr}[T_{(a,b)}T_{(a',b')}] = \omega^{-2^{-1}ab}\omega^{-2^{-1}a'b'} \operatorname{Tr}[Z^a X^b Z^{a'} X^{b'}] = \omega^{-2^{-1}(ab+a'b')+a'b} \operatorname{Tr}[Z^{a+a'} X^{b+b'}] \tag{13}$$

$$= \omega^{-2^{-1}(b(a+a')+a'(b+b'))} d\delta_{a,-a'}\delta_{b,-b'} = d\delta_{a,-a'}\delta_{b,-b'} \tag{14}$$

For a system of n qudits, the generalized pauli matrices are built up from their single qudit analogs in the same way as the pauli matrices are, and can be indexed by a string $u = (a_1, b_1, a_2, b_2, \ldots, a_n, b_n) \in \mathbb{Z}_d^{2n}$:

$$T_u = \otimes_{i=1}^n T_{(a_i, b_i)}. \tag{15}$$

The generalized pauli matrices, unlike the pauli matrices, are unitary - they are composed of a product of unitary operators. They are not hermitian and therefore do not form an orthonormal basis of operators that can be used to represent density matrices. However, they fail to be hermitian in a very specific way: The hermitian conjugate of a generalized pauli matrix is another generalized pauli matrix:

$$T^\dagger_{(a,b)} = \left(\omega^{-2^{-1}ab} Z^a X^b\right)^\dagger = \left(\omega^{-2^{-1}ab+ab} X^b Z^a\right)^\dagger = \omega^{-2^{-1}ab} Z^{-a} X^{-b} = T_{(-a,-b)}, \tag{16}$$

where we've used the fact that X and Z are unitary and are related to each other by (11), which implies that $Z^a X^b = \omega^{ab} X^b Z^a$.

This means that the sum of all generalized pauli matrices is hermitian:

$$\left(\sum_{(a,b)} T_{(a,b)}\right)^\dagger = \sum_{(a,b)} T^\dagger_{(a,b)} = \sum_{(a,b)} T_{(-a,-b)} = \sum_{(a',b')} T_{(a',b')} \tag{17}$$

This fact will allow us to construct a basis of hermitian operators.

A.2 Phase space operator basis

The *phase space operators* for are

$$A_u = \frac{1}{d^n} T_u \left(\sum_{v \in \mathbb{Z}_d^{2n}} T_v \right) T_u^\dagger, \tag{18}$$

and satisfy the property

$$A_u = \otimes_{i=1}^n A_{(a_i,b_i)}. \tag{19}$$

They are hermitian, as they are formed by unitary conjugation of the hermitian operator $\sum_u T_u$. The single qudit operators have unit trace:

$$\operatorname{Tr}[A_u] = \frac{1}{d} \operatorname{Tr}[T_u^\dagger (\sum_v T_v) T_u] = \frac{1}{d} \operatorname{Tr}[\sum_v T_v] = \frac{1}{d} \operatorname{Tr}[1] = 1, \tag{20}$$

and are trace orthonormal up to a factor of d:

$$\operatorname{Tr}[A_{(a,b)} A_{(a',b')}] = \frac{1}{d^2} \sum_{i,j,i',j'} \operatorname{Tr}[T_{(a,b)} T_{(i,j)} T_{(a,b)}^\dagger T_{(a',b')} T_{(i',j')} T_{(a',b')}^\dagger] = \frac{1}{d^2} \sum_{i,j,i',j'} \omega^{ib-aj+i'b'-a'j'} \operatorname{Tr}[T_{(i,j)} T_{(i',j')}] \tag{21}$$

$$= \frac{1}{d} \sum_{i,j} \omega^{ib-aj-ib'+a'j} = \frac{1}{d} \sum_i \omega^{i(b-b')} \sum_j \omega^{j(a'-a)} = d\delta_{a,a'}\delta_{b,b'}, \tag{22}$$

where we've used the fact that

$$T_{(a,b)} T_{(a'b')} = \omega^{-2^{-1}(ab+a'b')} Z^a X^b Z^{a'} X^{b'} = \omega^{-2^{-1}(ab+a'b')} \omega^{a'b-ab'} Z^{a'} X^{b'} Z^a X^b = \omega^{a'b-ab'} T_{(a',b')} T_{(a,b)}. \tag{23}$$

A.3 Wigner function representation of a mixed state

The *wigner function* of a state ρ of n qudits is a function of a string $u \in \mathbb{Z}_d^{2n}$ representing the coefficient of ρ in the phase space basis element A_u:

$$W_u(\rho) = \frac{1}{d^n} \operatorname{Tr}[A_u \rho] \Rightarrow \rho = \sum_u W_u(\rho) A_u. \tag{24}$$

The coefficients sum to the trace of ρ due to (20):

$$\operatorname{Tr} \rho = \operatorname{Tr} \left[\sum_u W_u(\rho) A_u \right] = \sum_u W_u(\rho) \operatorname{Tr}[A_u] = \sum_u W_u(\rho), \tag{25}$$

so for a properly normalized density matrix we have

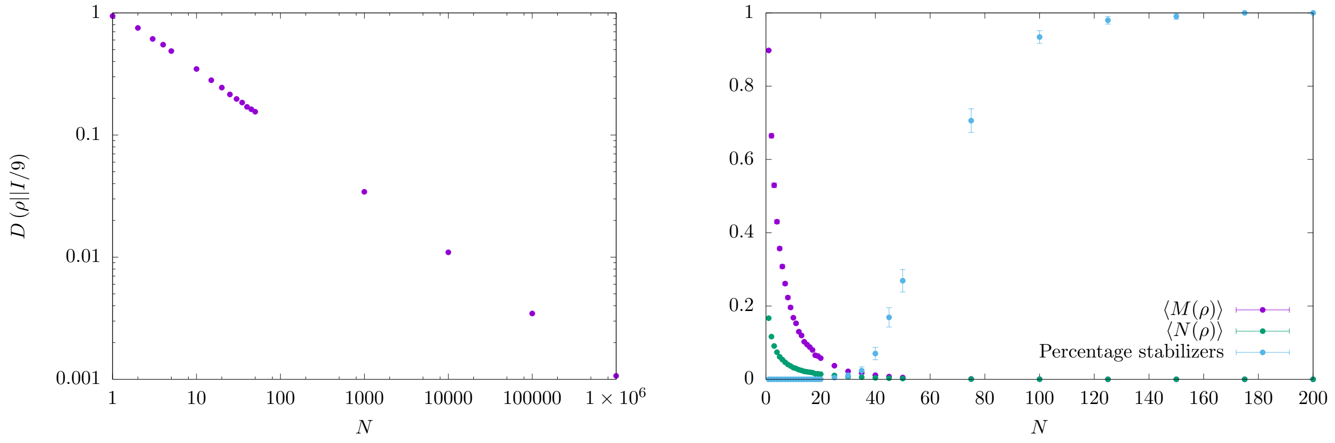
$$\sum_u W_u(\rho) = 1 \tag{26}$$

B Random state generation

To generate random mixed states, we first generate N random pure states $|\mathcal{S}_j\rangle = \frac{1}{\sqrt{\sum_k c_{jk}^2}} c_{jk} |\phi_k\rangle$, where $|\phi_j\rangle$ is the computational basis, by generating a sample of normally-distributed complex coefficients c_{jk}. After normalizing, generate N real numbers r_i uniformly on $[0, 1]$ and form the mixed state

$$\rho = \frac{1}{\sum_j r_j} \sum_{i=1}^N r_i |\mathcal{S}_i\rangle \langle \mathcal{S}_i| \tag{27}$$

When $N = 1$, the "mixed" state is really just a random pure state. As $N \to \infty$, it becomes maximally mixed. So fixing N to a finite value gives us a moderately mixed state. We can get a sense of what N to choose by looking at the average distance $d(\rho, \frac{1}{d}I) = \sqrt{\operatorname{Tr}[(\rho - \frac{1}{d}I)^2]}$ between the maximally mixed state and a randomly sampled state of N pure states, where d is the hilbert space dimension: For $N = 1$, all states will have a fixed distance $\sqrt{\frac{d-1}{d}}$ from the maximally mixed state. As $N \to \infty$, $D \to 0$.



(a) Average distance between a randomly generated mixed state of 2 qutrits and the maximally mixed state $\frac{I}{3^2}$

(b) Average values of both quantifiers of magic - $\mathcal{M}$ and $\mathcal{N}$, and the average percentage of states which lie in the stabilizer hull

Figure 3: Analysis of random state generation protocol as a function of the number N of pure qudit states sampled in each mixed state. 100 mixed states were used to generate each point in the first plot, and 200 in the second. Error bars on both plots are generated using 100 bootstrap resamples - they are too small to see in the first plot.

C Efficient contraction scheme to compute $d(\rho,\sigma)$ and its derivatives

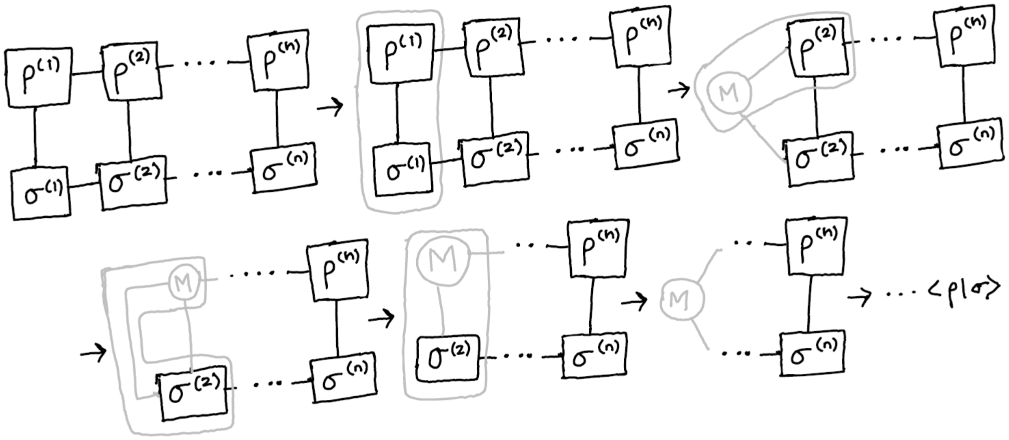
Let $\rho = \sum_{u,\alpha_i} \otimes_k \rho^{(k)u_k}_{\alpha_k\alpha_{k+1}} A_{u_k}$ and $\sigma = \sum_{u,\alpha_i} \otimes_k \sigma^{(k)u_k}_{\alpha_k\alpha_{k+1}} A_{u_k}$ be MPOs in the phase space basis A_u (18) with bond dimensions D_ρ and D_σ, respectively. In this basis, the Wigner coefficients are simply given by the matrix products $W_u(\rho) = \prod_i \rho^{(i)u_i}$ and $W_u(\sigma) = \prod_i \sigma^{(i)u_i}$ due to (24), and the distance between them when viewed as vectors in the Wigner function representation space is simply

$$d(\rho,\sigma) = \left(\sum_u \left(\prod_i \rho^{(i)u_i} - \prod_j \rho^{(j)u_j}\right)^2\right)^{1/2} = d^{-n/2}\sqrt{\langle\rho|\rho\rangle - 2\langle\rho|\sigma\rangle + \langle\sigma|\sigma\rangle} \qquad (28)$$

Where $\langle\rho|\sigma\rangle$ denotes the Frobenius overlap between the states ρ and σ[8]:

$$\langle\rho|\sigma\rangle = \mathrm{Tr}\left[\rho^\dagger\sigma\right] = d^{-n}\sum_u \left(\prod_i \rho^{(i)u_i}\right)\left(\prod_j \sigma^{(i)u_j}\right) \qquad (29)$$

This overlap can be calculated using a contraction order which allows for only a linear number of computations in d and n and cubic in D_σ and D_ρ:

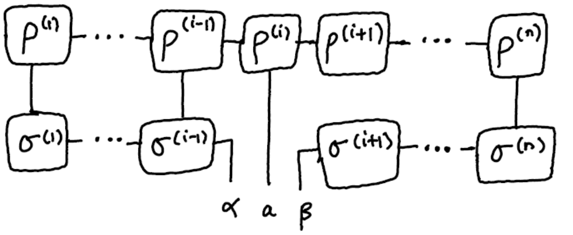


In this diagram, vertical and horizontal lines indicate contraction over “physical” indices u_i with dimension d and “bond” indices with dimension D_ρ (or D_σ), respectively. The tensor M represents the tensor in memory during the contraction,

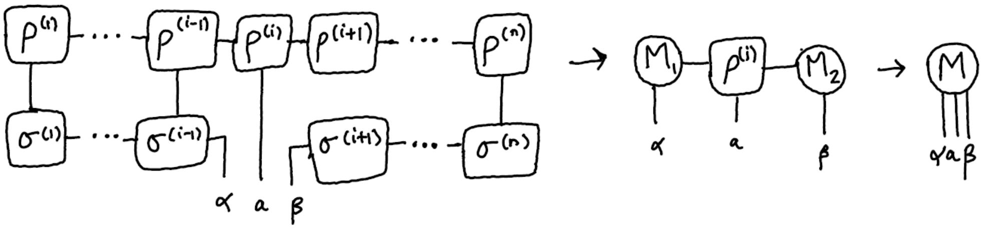
[8] The factor of d^{-n} comes from the normalization of the phase space operators (22)

which grows in size horizontally through the tensor network until all contractions are completed, at which point it contains no external legs and holds the value $\langle\rho|\sigma\rangle$. The contractions in the diagram take $dD_\rho D_\sigma + dD_\rho^2 D_\sigma + dD_\rho D_\sigma^2 + dD_\sigma D_\rho = \mathcal{O}(dD_\rho D_\sigma(D_\rho + D_\sigma))$ steps to compute. The entire calculation requires n such contractions, and will take $O(ndD_\rho D_\sigma(D_\rho + D_\sigma))$ time.

The derivative of $d(\rho, \sigma)$ with respect to the matrix element $\sigma^{(i)a}_{\alpha\beta}$ requires the computation of the derivative of $\langle\rho|\sigma\rangle$ with respect to $\sigma^{(i)a}_{\alpha\beta}$, which is given by the following tensor:



While this tensor as a function of α, a, and β, can also be computed in linear time in d using a similar contraction scheme, we can do even better by precomputing as much of it as we can before using it in the optimization procedure. This is done by contracting all of the tensors before and after $\rho^{(i)}$ into tensors M_1 and M_2 using the same procedure for calculating overlaps above, and then performing the final $\mathcal{O}(dD_\rho D_\sigma)$ contractions to group the three remaining tensors into one:



References

[1] Victor Veitch, S A Hamed Mousavian, Daniel Gottesman, and Joseph Emerson. The resource theory of stabilizer quantum computation. *New Journal of Physics*, 16(1):013009, Jan 2014.

[2] D. Gross. Hudson's theorem for finite-dimensional quantum systems. *Journal of Mathematical Physics*, 47(12):122107, Dec 2006.

[3] Ciyou Zhu, Richard H. Byrd, Peihuang Lu, and Jorge Nocedal. Algorithm 778: L-bfgs-b: Fortran subroutines for large-scale bound-constrained optimization. *ACM Trans. Math. Softw.*, 23(4):550–560, December 1997.

[4] Richardh Byrd, Peihuang Lu, Jorge Nocedal, and Ciyou Zhu. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16, 02 2003.