QUDITS FOR EFFICIENTLY GENERATING HIGHER MOMENTS OF HERMITIAN OPERATORS

Mahadevan Subramanian, Undergraduate at Department of Physics, IIT Bombay *

Guide: Professor Sai Vinjanampathy, Department of Physics, IIT Bombay

December 8, 2021

Contents

---

*email: mahadevan_s@iitb.ac.in, roll no: 190260027

ABSTRACT

The task of efficiently generating higher Hermitian moments for the case of LCU Hermitian operators turns out to scale exponentially due to the number of unique unitaries in the higher powers. This can however be dealt with far more efficiently using qubitization as discussed in [26]. The conditions for this however are not general enough for guaranteeing scalability so we combine these with a co-design approach using 4-level qudits to efficiently make arbitrary controlled gates [38] motivated by the approach in [32]. In addition to this we prove that that the error scaling is linear and additionally explore the quantum error correction codes for usage in qudits.

1   Introduction

Since the conception of quantum computing, the thought of using qubits seemed like the most obvious due to the already existing algorithmic ideals made for bit based computers. There are however certain works revolving around using more than two levels of a quantum system to store information called qudits and as we shall see in this report they have various interesting uses.

Over the many years of development quantum computers have posed various challenges in implementing them. Noisy intermediate-scale quantum (NISQ) devices refer to quantum devices with 50 to a few hundred qubits that are not completely fault tolerant (hence noisy) and are still at an intermediate-scale in terms of number of qubits. We are currently in this era of quantum technology where we deal with NISQ devices and so a lot of algorithms are designed keeping this specific idea in mind. Variational quantum algorithms largely hinge on the variational principle in quantum mechanics and boil down to searching the state that best approximates the ground state of a Hamiltonian by parametrizing the wavefunction and varying the parameters. The variational quantum algorithm is hybrid algorithm in the sense that it runs using both a quantum computer and a classical one to aid by running optimization methods on the classical computer and using the quantum computer to construct the wavefunction and find the energy at any step of the algorithm.

In NISQ devices, variational quantum algorithms [5] have become one of the most capable form of algorithms that can be executed due to the hybrid nature of the algorithm. This hybrid nature however can be seen as somewhat restrictive as the circuit is executed a various number of times and additionally the general VQA task is NP-Hard [6]. To circumvent this one can use a modified version of the VQA known as the Quantum Assisted Eigensolver [3]. This task essentially seperates the classical and quantum parts of the algorithm and finally performs all the optimization on a classical computer that under certain conditions has efficient convergence. As described in section 2 a crucial part of this is to make use of a certain ansatz one of which requires a way to efficiently calculate a quantity of form H^k where H is Hermitian.

Higher moments of Hermitian operators have uses in imaginary time evolution [1] and many tasks such as the HHL Algorithm [18] rely on Hamiltonian simulation subroutines. Additionally finding these higher moments aid in many-body energy estimations using Lanczos expansion theory [20, 19]. In [35] this method is demonstrated on a 25-qubit IBM quantum computer where complexity is transferred to the task of finding $\langle H^k \rangle$ that are central to certain non-perturbative approximation schemes in many-body theory.

We start off by describing two methods in approaching this task: the Quantum SWAP test explored in section 3 and Block encoding explored in section 4. The SWAP test, while useful, doesn't offer tremendous scalability and state preparation for arbitrary density matrices is a task that one may have to deal with in that direction. However we see that Block encoding in a way where one can efficiently find an encoding for H^k using the iterate approach [26] boils down to efficiently creating controlled gates.

We then proceed to build up the theory surrounding error scaling in quantum computers in section 5. Following which in section 6 we explore a method for arbitrary controlled qubit gates using 4-level qudits and demonstrate that this has linear scaling in errors. In section 7 we delve into quantum error correction codes including applying them to qudits.

We reach our results that we can make use of qudits to create iterates efficiently which we can then use to efficiently implement higher moments of Hermitian matrices. We discuss all these in section 8. These approaches are motivated by the co-design approaches that are also discussed in [32] and using this approach we present a viable direction in quantum computation using qudits.

2 Quantum assisted eigensolver

2.1 Variational quantum eigensolvers

Before one can dive into quantum assisted eigensolvers, we must first understand what a Variational Quantum Algorithm (VQE) is. A VQA workflow as described in [5] can be boiled down to the following components

1. The objective function O that encodes the problem to be solved.

2. A parametrized quantum circuit which has some set of parameters $\vec{\theta}$ which are to be tuned in a way to minimize O.

3. A measurement scheme that is used to evaluate O by measuring expectation values $\langle H \rangle$ and do basis changes.

4. A classical optimizer which minimizes O by variationally updating $\vec{\theta}$.

The parametrized quantum circuit is defined using a circuit ansatz which can be defined in accordance to the system for which the VQA is being applied. The *hardware-efficient* ansatz [21] aims at providing a low depth ansatz which is constructed with the constraints of the kind of connectivity between the qubits in the given processor.
The algorithm for VQE can be written as follows:

Algorithm 1: Variational Quantum Eigensolver

Map the quantum Hamiltonian to a qubit Hamiltonian H
Set d as depth of circuit for trial state preparation
Choose a set of $\{\theta_i\}$ as parameters for rotations applied
Choose a number of samples S for the feedback loop and one S_f for final estimation
Choose the maximal number of control updates k_L
while E_f *has not converged* **do**
    begin procedure quantum feedback loop
        for $k = 1$ *to* k_L **do**
            Prepare trial states using θ_k and evaluate $\langle H \rangle$ with S samples
            Update and store the controls θ_k
        end
        Evaluate $E_f = \langle H \rangle$ with S_f samples using the best controls
    end
    Increase d, k_L, S, S_f
end
return E_f

The updation is done on the basis of optimizing the cost function. There are various optimization approaches [8] such as stochastic gradient descent, simultaneous perturbation stochastic approximation (used in [21]) or even using neural networks in combination with SGD.
There however are issues with VQAs, one of which being the vanishing gradient problem (also called the barren plateau problem) which arises as circuit depth or hardware noise increases [27]. The letter [3] aims to discuss a new algorithm which takes a new approach on the hybrid nature of variational quantum algorithms.
This new approach doesn't tune the parameters of the quantum circuit however performs measurements over a fixed circuit and then finally does classical post processing. There are however a fair number of intricacies involved in both constructing the quantum circuit and efficiently performing the classical post processing. We now proceed to study these in detail.

2.2 Details of setup & algorithm

The following is based on the discussions provided in letter [3]. We start off with the assumption that the Hamiltonian is given in the form of linear combination of unitaries $U_i \in SU(2^N)$, $\beta_i \in \mathbb{C}$, $n \in \mathbb{N}$, and $i \in \{1, 2, \ldots, n\}$

$$H = \sum_{i=1}^{n} \beta_i U_i \tag{1}$$

We now pick a basis of m states $\{|\phi_j\rangle\}_{j=1}^{m}$ that satisfies the condition that $\langle \phi_j | \phi_k \rangle = 1$ iff $j = k$ and $|\phi_j\rangle = V^j |0^{\otimes N}\rangle$. This will be the ansatz for the basis and now we express

$$|\psi(\boldsymbol{\alpha})\rangle = \sum_{j=1}^{m} \alpha_j |\phi_j\rangle, \quad \boldsymbol{\alpha} \in \mathbb{C}^m \tag{2}$$

The task at hand is to find $\boldsymbol{\alpha}$ such that it minimizes $\langle H(\boldsymbol{\alpha}) \rangle = \langle \psi(\boldsymbol{\alpha}) | H | \psi(\boldsymbol{\alpha}) \rangle$. Now we can more concretely define the optimization by declaring matrices D and E as follows

$$D_{j,k} = \sum_i \beta_i \langle \phi_j | U_i | \phi_k \rangle, \quad E_{j,k} = \langle \phi_j | \phi_k \rangle \tag{3}$$

This lets us write $\langle H(\boldsymbol{\alpha}) \rangle = \sum_{j,k} \alpha_j^* D_{j,k} \alpha_k = \boldsymbol{\alpha}^\dagger D \boldsymbol{\alpha}$ and since we wish $\psi(\boldsymbol{\alpha})$ to be normalized this gives us the constraint $\sum_{j,k} \alpha_j^* \langle \phi_j | \phi_k \rangle \alpha_k = 1$ which can be all finally expressed as the following optimization problem

$$\text{minimize } \boldsymbol{\alpha}^\dagger D \boldsymbol{\alpha} \quad \text{subject to } \boldsymbol{\alpha}^\dagger E \boldsymbol{\alpha} = 1 \tag{4}$$

Note that if we happened to know the entries in the matrices D and E this reduces to a quadratic constraint quadratic problem that can be completely solved classically. The steps of this algorithm can be written as follows

1. Select the ansatz and prepare the required states.
2. Using these states evaluate the entries in the D and E matrices.
3. Perform QCQP on a classical computer using the evaluated D and E.

2.3 Choosing the ansatz

QCQP in its most general form is NP Hard [2]. However the specific QCQP is reformulated using Lagrangian relaxation to be made convex and also [3] proves that local optimality will be the global optimal solution. The exact details of this will not be explored in this report since our main aim is in dealing with creating the ansatz.

Choice of ansatz greatly affects the performance of the ansatz specifically in the expressibility of the chosen basis. In [4] an ansatz for the Krylov subspace based algorithm chooses the $V = H$ and so $|\phi_j\rangle = H^j |0^{\otimes N}\rangle$. This choice naturally arises from doing time evolution of a state as this can be used to approximate the quantum imaginary time evolution. We define the Krylov subspace as

$$Kr_K = \text{span}\{|\psi\rangle, H|\psi\rangle, \ldots, H^K |\psi\rangle\} \tag{5}$$

However given that quantum computers have unitary circuits the task of non unitary operators is not easy to carry out [29] and may even require more measurements than capable on NISQ devices. There are however existing methods to carry out the task and we will now proceed to explore these methods. For the sake of [4] the analysis pertains to using the $\mathbb{CS}_K$ ansatz.

Definition 2.1 ($\mathbb{CS}_K$ Ansatz). We take $\mathbb{U} = \{U_i\}_{i=1}^r$ and a positive integer K and some state $|\psi\rangle$. Using this we define a set of K moment states $\mathbb{S}_K = \{U_{i_K} \ldots U_{i_2} U_{i_1} |\psi\rangle\}_i$ for $U_{i_1} \in \mathbb{U}$. We define $\mathbb{CS}_K = \cup_{j=0}^K \mathbb{S}_j$. Using this we can construct states as

$$|\xi(\boldsymbol{\alpha})\rangle^{(K)} = \sum_{|\xi_i\rangle \in \mathbb{CS}_K} \alpha_i |\xi_i\rangle \tag{6}$$

We can see that for the LCU description of the Hamiltonian we can see that this would span the same space as the Krylov subspace. In this report we analyze methods to simply work to generate the higher moments of Hermitian matrices.

3 Quantum SWAP test

The main reference for this section is from [11]. Linear functionals such as expectation values of operators are quite commonly found as measurable quantities represented for a density matrix ρ and operator A as $\text{Tr}(A\rho)$. The letter [11] presents methods for both linear and nonlinear functionals such as even finding values such as $\text{Tr}(\rho_1 \rho_2)$ where ρ_1 and ρ_2 are two density matrices. The setup above define $|0\rangle$ and $|1\rangle$ by wave packets arranged in different directions defined by the geometry of a Mach-Zender interferometer. Here mirrors, beam splitters and relative phase action can be defined as quantum gates. We can define the gates being applied as follows where U is the gate as applied in figure 1 and the ϕ is a phase gate that adds phase to the $|0\rangle$ term.

$$\mathbf{U} = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \otimes U + \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \otimes I \tag{7}$$

$$\mathbf{U}_H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \otimes I \tag{8}$$

$$\mathbf{U}_\phi = \begin{pmatrix} e^{\iota\phi} & 0 \\ 0 & 1 \end{pmatrix} \otimes I \tag{9}$$

4
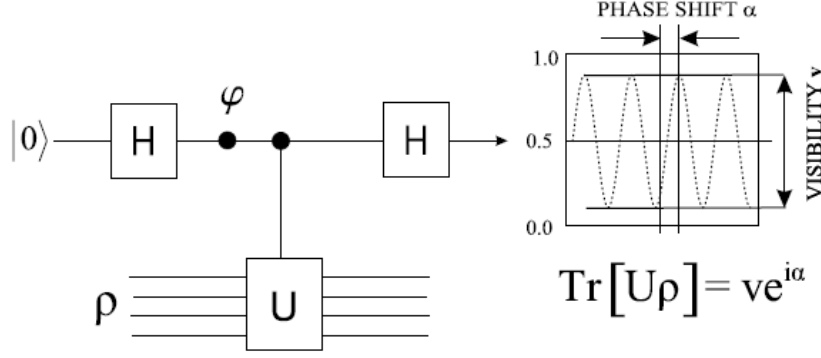
Figure 1: Shown above is the circuit for the estimation of $\text{Tr}(U\rho)$. Figure taken from [11]

We can see that density matrix evolves as follows

$$\rho_{out} = \mathbf{U}_H \mathbf{U} \mathbf{U}_H \mathbf{U}_\phi \rho \mathbf{U}_H^\dagger \mathbf{U}_\phi^\dagger \mathbf{U}^\dagger \mathbf{U}_H^\dagger \tag{10}$$

$$\rho_{out} = \frac{1}{4}\left[\begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \otimes U\rho U^\dagger + \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \otimes \rho + e^{\iota\phi}\begin{pmatrix} 1 & 1 \\ -1 & -1 \end{pmatrix} \otimes \rho U^\dagger + e^{-\iota\phi}\begin{pmatrix} 1 & -1 \\ 1 & -1 \end{pmatrix} \otimes U\rho\right] \tag{11}$$

We note that $\text{Tr}(\rho U^\dagger) = \text{Tr}(U\rho)^*$ and so we get that the intensity along $|0\rangle$ of the control qubit is

$$I \propto 1 + |\text{Tr}(U\rho)|\cos(\phi - \arg(\text{Tr}(U\rho))) \tag{12}$$

Hence we can see that now how we can find the value $\text{Tr}(U\rho) = ve^{\iota\alpha}$ where v is visibility in the interferometer and α is the phase shift ϕ where intensity is maximized. The formalism here has been taken from [34]. We now note the
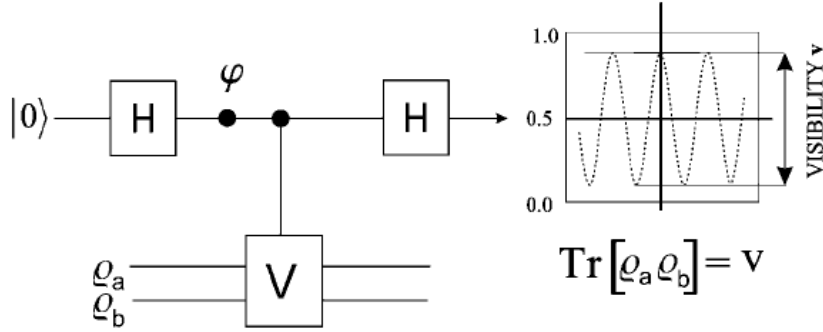
Figure 2: Shown above is the figure for measuring quantities such as $\text{Tr}(\rho_a\rho_b)$. Figure taken from [11]

circuit in the figure 2. The V operator here is a simple SWAP gate but we can define a generalization as follows

$$V^{(k)}|\phi_1\rangle|\phi_2\rangle\ldots|\phi_k\rangle = |\phi_k\rangle|\phi_1\rangle\ldots|\phi_{k-1}\rangle \tag{13}$$

We can see that the controlled version of this gate can be made using $k-1$ cascaded Fredkin gates. We can see that $\text{Tr}(V(\rho_a \otimes \rho_b)) = \text{Tr}(\rho_a\rho_b)$ can be easily checked for V being the SWAP operation and this can also be used to estimate overlaps between pure states since if $\rho_a = |\alpha\rangle\langle\alpha|$ and $\rho_a = |\beta\rangle\langle\beta|$, $\text{Tr}(\rho_a\rho_b) = |\langle\alpha|\beta\rangle|^2$. Since these are real values the phase $\alpha = 0$ and we simply need visibility in which case $v = 2P_0 - 1$ where P_0 is the probability to measure $|0\rangle$ for the control qubit and if α then $v\cos(\alpha) = 2P_0 - 1$ as we would get from equation 12.

We can now feed in the same quantum state to multiple channels and change the swap gate to $V^{k,m}$ where this is defined as shown in equation 13 but each $|\phi_i\rangle$ is composed of m qubits. This will give us the following visibility

$$v = \text{Tr}(V^{k,m}\rho^{\otimes k}) = Tr(\rho^k) = \sum_{i=1}^{m}\lambda_i^k \tag{14}$$

5

We can make use of this for our main aim of evaluating higher moments of Hermitian matrices. Suppose that we construct the following density matrix

$$\rho_H = \frac{-\lambda I + H}{\sum_{i=1}^m \lambda_i - m\lambda} \tag{15}$$

Here λ is the smallest eigenvalue of H (which is taken as m dimensional) and λ_i are all the eigenvalues. We can ignore the cases of H having all eigenvalues equal and so we can now see that ρ_H is actually the density matrix for some state. Using the visibility obtained in equation 14 we could do as follows

$$v = \text{Tr}(V^{k,m}\rho_H^{\otimes k}) = \text{Tr}\left(\left(\frac{-\lambda I + H}{\sum_{i=1}^m \lambda_i - m\lambda}\right)^k\right) \tag{16}$$

As we can see this can help in calculating certain values related to the actual Hamiltonian however there are a few problems with this

1. The circuit scales in width linearly with k calling for large number of registers for holding the state.

2. Preparing the density matrix ρ_H cannot be guaranteed to be efficient given that it requires us to know the Hamiltonian very well and so in that way may not scale favorably.

It however must be noted that there is a recent work that circumvents the disadvantage of width [36]. In this work an efficient version of the entanglement spectroscopy Hadamard test and two copy test is discussed. Qubit resets are made use of in the algorithm where instead of preparing multiple copies of the state ρ for different registers, the circuit is modified in a way where the same qubits are recycled and the state is prepared in them. This keeps the circuit depth linear but introduces a linear dependence on state preparation which again isn't a necessarily efficient task.

4 Block encoding

4.1 Introduction

By nature of quantum circuits, if one wishes to execute any operation on a state $\hat{U} : \mathcal{H} \to \mathcal{H}$ in some Hilbert space $\mathcal{H}$, the operation $\hat{U}$ is required to be unitary since it represents the evolution of a quantum state [31]. One may however in practice find this as somewhat restrictive if one requires to apply non unitary operations on some quantum state. For the quantum assisted eigensolver [3] described in the section 2, the Krylov Subspace ansatz can be constructed using Hermitian operators.

The HHL algorithm [18] marked the beginning of the sub-field of algorithms that use Hamiltonian simulation subroutines to solve linear algebraic problems. The majority of the analysis pertaining to the further improvements to this algorithm deal with sparse matrices. We can see how the block encoding can help us deal with the problem of Hamiltonian simulation.

Definition 4.1 (Block encoding). Given $\hat{H} : \mathcal{H}_S \to \mathcal{H}_s$ with $\|\hat{H}\| \leq 1$, and $\hat{G}|0\rangle_a = |G\rangle_a \in \mathcal{H}_a$, we define an encoding for $\hat{H}$ as the unitary $\hat{U} : \mathcal{H}_a \otimes \mathcal{H}_s \to \mathcal{H}_a \otimes \mathcal{H}_s$. This encoding satisfies the property $(\langle G|_a \otimes \hat{\mathcal{I}}_s)\hat{U}(|G\rangle_a \otimes \hat{\mathcal{I}}_s)) = \hat{H}$ and we assume that there is query access to inverses and controlled versions of $\hat{U}, \hat{G}$.

Another definition is of a (α, a, ϵ) encoding as $\|A - \alpha(\langle G|_a \otimes \hat{\mathcal{I}}_s)\hat{U}(|G\rangle_a \otimes \hat{\mathcal{I}}_s))\| < \epsilon$ where $\alpha > \|A\|_F$ that essentially removes the constraint of $\|\hat{H}\| \leq 1$ put in place in the above definition.

Lemma 4.1 (Product of block encoded matrices). If U is an (α, a, δ) encoding of an s qubit operator A and V is an (β, b, ϵ) encoding of an s qubit operator B, $(\mathcal{I}_b \otimes U)(\mathcal{I}_a \otimes V)$ is an $(\alpha\beta, a + b, \alpha\epsilon + \beta\delta)$ encoding of AB.

Proof.

$$\|AB - \alpha\beta(\langle 0|^{\otimes a+b} \otimes I)(I_b \otimes U)(I_a \otimes V)(|0\rangle^{\otimes a+b} \otimes I)\|$$
$$= \|AB - \alpha(\langle 0|^{\otimes a} \otimes I)U(|0\rangle^{\otimes a} \otimes I)\beta(\langle 0|^{\otimes b} \otimes I)V(|0\rangle^{\otimes b} \otimes I)\|$$
$$= \|AB - \tilde{A}B + \tilde{A}B - \tilde{A}\tilde{B}\|$$
$$\leq \|A - \tilde{A}\|\beta + \alpha\|B - \tilde{B}\| \leq \alpha\epsilon + \beta\delta$$

$\square$

The proof has been taken from [9] which discusses a lot of the applications of block encoding. As we can see, error scales linearly but to ensure that the subspaces $\mathcal{H}_a$ and $\mathcal{H}_b$ do not mix, we need linear growth in ancillas.

4.2 Explicit encoding for LCU

We take the example of the Hamiltonian being a linear combination of unitaries. In this case the higher powers of the Hamiltonian would contain exponentially scaling number of unitaries in combination making it very poorly scalable if implemented by Trotterization.

$$\hat{H} = \sum_{j=1}^{d} \alpha_j \hat{U}_j, \qquad \|\hat{H}\| \le \|\vec{\alpha}\|_1 = \sum_{j=1}^{d} |\alpha_j| \tag{17}$$

The algorithm assumes that we are provided α_j as a list of d numbers. Without a loss of generality we absorb all phases into $\hat{U}_j$ and hence take all $\alpha_j \ge 0$. The upper bound is the spectral norm and for a certain choice of decomposition is a tight bound. The following are the oracles we define

$$\hat{G} = \sum_{j=1}^{d} \sqrt{\frac{\alpha_j}{\|\vec{\alpha}\|_1}} |j\rangle\langle 0|_a, \qquad \hat{U} = \sum_{j=1}^{d} |j\rangle\langle j|_a \otimes \hat{U}_j \tag{18}$$

We can check that $\langle G|\hat{U}|G\rangle = \dfrac{\hat{H}}{\|\vec{\alpha}\|_1}$ where $|G\rangle = \hat{G}|0\rangle$. The state preparation $\hat{G}$ is implemented with $\mathcal{O}(d)$ gates and $\hat{U}$ is implemented with $\mathcal{O}(dC)$ assuming each U_j is composed of $\mathcal{O}(C)$ primitive gates.

What if we want to get something like $\hat{H}^2$? It would be tempting to try $\hat{U}^2$ directly so we can see

$$\hat{U}^2 = \sum_{j=1}^{d} |j\rangle\langle j|_a \otimes \hat{U}_j^2 \tag{19}$$

In combination with $\hat{G}$ define earlier, we see that

$$\langle G|\hat{U}^2|G\rangle = \sum_{j=1}^{d} \frac{\alpha_j}{\|\vec{\alpha}\|_1} \hat{U}_j^2 \ne \frac{\hat{H}^2}{\|\vec{\alpha}\|_1} \tag{20}$$

It seems that this approach simply would not provide the right expansion for powers of $\hat{H}$. For this purpose we move on to the iterate approach described in [26].

4.3 Quantum signal processor

We wish to apply a non unitary operator which is $\hat{H}$ on a state and hence require it to be embedded in some larger Hilbert space to accomplish this. This was the entire purpose of block encoding. Given some block encoding $\hat{U}$ for $\hat{H}$ we can write

$$\hat{U}|G\rangle_a|\psi\rangle_s = |G\rangle_a \hat{H}|\psi\rangle_s + \sqrt{1 - \|\hat{H}|\psi\rangle\|^2}|G_\psi^\perp\rangle_{a+s} \tag{21}$$

For some eigenvector $\hat{H}|\lambda\rangle = \lambda|\lambda\rangle$ we can define the subspace $\mathcal{H}_\lambda = \text{span}\{|G_\lambda\rangle, \hat{U}|G_\lambda\rangle\}$ where

$$\hat{U}|G\rangle|\lambda\rangle = \hat{U}|G_\lambda\rangle = \lambda|G_\lambda\rangle + \sqrt{1 - |\lambda|^2}|G_\lambda^\perp\rangle \tag{22}$$

The inability to produce higher moments by repeating $\hat{U}$ was due to leakage out of $\mathcal{H}_\lambda$ as said in [26].

We can replace $\hat{U}$ with a unitary iterate $\hat{W}$ which satisfies $\langle G|\hat{W}|G\rangle = \hat{H}$ but it performs disjoint rotations in SU(2) for each subspace $\mathcal{H}_\lambda$ hence no mixing.

$$|G_\lambda^\perp\rangle = \frac{(\hat{W} - \lambda)|G_\lambda\rangle}{\sqrt{1 - |\lambda|^2}} \tag{23}$$

$$\hat{X}_\lambda|G_\lambda\rangle = |G_\lambda^\perp\rangle, \quad \hat{Y}_\lambda|G_\lambda\rangle = \iota|G_\lambda^\perp\rangle, \quad \hat{Z}_\lambda|G_\lambda\rangle = |G_\lambda\rangle \tag{24}$$

For each eigenvalue the iterate acts as per the following

$$\hat{W} = \begin{matrix} \lambda|G_\lambda\rangle\langle G_\lambda| & -\sqrt{1 - |\lambda|^2}|G_\lambda\rangle\langle G_\lambda^\perp| \\ +\sqrt{1 + |\lambda|^2}|G_\lambda^\perp\rangle\langle G_\lambda| & +\lambda|G_\lambda^\perp\rangle\langle G_\lambda^\perp| \end{matrix} \tag{25}$$

More concretely it is simply the sum of all these for all λ so we can represent this as $W = \bigoplus_\lambda e^{-\iota \hat{Y}_\lambda \theta_\lambda}$ where $\theta_\lambda = \cos^{-1}(\lambda)$.

We can check that this indeed works. Let $|\psi\rangle = \sum_\lambda a_\lambda |\lambda\rangle$ then we can see that $|G\rangle |\psi\rangle = \sum_\lambda a_\lambda |G_\lambda\rangle$. On applying the iterate once $\hat{W} |G\rangle |\psi\rangle = \sum_\lambda a_\lambda \lambda |G_\lambda\rangle = |G\rangle \hat{H} |\psi\rangle$ hence verifying this approach. We must note that thanks to the way we defined the iterate,

$$\hat{W}^n = \bigoplus_\lambda e^{-\iota \hat{Y}_\lambda n \theta_\lambda} \tag{26}$$

This means that $\hat{W}^n |G\rangle |\psi\rangle = |G\rangle f_n(\hat{H}) |\psi\rangle$ where $f_n(\theta)$ is the nth Chebyshev polynomial ($f_n(\cos(\theta)) = \cos(n\theta)$). While not giving the exact power, one generate higher moments using Chebyshev polynomials as a basis since they are a complete functional basis.

Qubitization refers to the act of encoding $\hat{H}$ into an iterate which functions as we described earlier making use of an overhead cost along with an already existing encoding.

Lemma 4.2 (Conditions on Qubitization). For all encodings $\hat{U}$ of $\hat{H}$, a unitary S can be used to create an iterate $\hat{W} = ((2|G\rangle\langle G| - \hat{\mathcal{I}}_a) \otimes \hat{\mathcal{I}}_s)\hat{S}\hat{U}$ if and only if

$$\langle G|_a \hat{S}\hat{U} |G\rangle_a = \hat{H} \text{ and } \langle G|_a \hat{S}\hat{U}\hat{S}\hat{U} |G\rangle_a = \hat{\mathcal{I}} \tag{27}$$

The proof can be arrived at by verifying all the conditions as per the definition of $\hat{W}$ and is written in *Low & Chuang, 2019* [26].

The interesting property however is that the second condition implies that $\hat{S}\hat{U}$ is a reflection when controlled by input state $|G\rangle$ and a Grover iterate happens to be a product of two reflections, the start and target subspaces.

Lemma 4.3 (Existence of Qubitization). For all unitaries $\hat{U}$ that encode $\hat{H}$, there exists a quantum circuit $\hat{U}'$, which uses one extra qubit and queries a controlled-$\hat{U}$ and controlled-$\hat{U}^\dagger$ once to implement an encoding that satisfies the properties in the previous lemma.

Proof. Let $\hat{V}_1 = |0\rangle\langle 0| \otimes \hat{\mathcal{I}} + |1\rangle\langle 1| \otimes \hat{H}$ and $\hat{V}_2 = |0\rangle\langle 0| \otimes \hat{U} + |1\rangle\langle 1| \otimes \hat{\mathcal{I}}$. Define $\hat{U}' = \hat{V}_1 \hat{V}_2 = |0\rangle\langle 0| \otimes \hat{U} + |1\rangle\langle 1| \otimes \hat{U}^\dagger$, $\hat{S} = (|0\rangle\langle 1| + |1\rangle\langle 0|) \otimes \hat{\mathcal{I}}_{a+s}$ and finally $|G'\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) |G\rangle$.

It's easy to verify that the conditions hold under this choice of operators. $\hat{S}\hat{U}'\hat{S}$ turns out to be $\hat{U}'^\dagger$ hence the second condition holds. $\square$

Hence one can construct the iterate using an already existing encoding with minimal overhead. However this leads us to requiring controlled version of the block encoding $\hat{U}$. The general problem of constructing circuits for controlled versions of any arbitrary gate is not an easy or efficiently scalable task. For this purpose we now study on how qudits can be used here to get scalable circuits both in depth and width for arbitrary control as shown in [38].

5 Errors in circuits

In any physical implementation of a quantum computer, errors are bound to spring up in one form or another. Here we take note of the theory surrounding circuits that are performed on mixed states rather than the regular pure state circuits that define a majority of quantum circuits. This helps in having a more general approach as it lets us have circuits with intermediate measurements and having noisy gates and mathematically quantify their differences.

5.1 Distinguishing states

Distinguishing between states here is done between two density matrices and the formalism is taken from [7]. We first define quantum operations to act on density matrices as follows

$$E(\rho) = \sum_i K_i \rho K_i^\dagger \tag{28}$$

Here all K_i are some unitaries and so we can understand $E(\rho)$ to act as a combination of all these operations. These maps are referred to as completely positive trace preserving maps and are a subset of superoperators that are $d^2 \times d^2$ matrices acting on the vector space of $d \times d$ matrices. To distinguish between two quantum states we define the a few distance measures. The first being the trace distance $d_1(\rho, \sigma)$ and fidelity $F(\rho, \sigma)$ as follows

$$d_1(\rho, \sigma) = \|\rho - \sigma\|_1, \quad F(\rho, \sigma) = \text{Tr}\left[\sqrt{\sqrt{\rho}\sigma\sqrt{\rho}}\right]^2 \tag{29}$$

The trace distance is interpreted as the maximum probability of correctly guessing whether ρ or σ was prepared or more concretely $p_{\text{correct}} = \frac{1}{2} + \frac{1}{4} \|\rho - \sigma\|_1$. As we can see for states that are difficult to distinguish this distance is very small. Here we note what occurs for if we take some N samples where N is very large

$$\|\rho^{\otimes N} - \sigma^{\otimes N}\|_1 \approx 2 - ce^{-D(\rho,\sigma)N} \tag{30}$$

Here $D(\rho,\sigma)$ is known as the quantum Chernoff bound and when the states are very close it is close to $1 - F(\rho,\sigma)$ and it satisfies

$$\frac{1 - F(\rho,\sigma)}{2} \leq D(\rho,\sigma) \leq 1 - F(\rho,\sigma) \tag{31}$$

5.2 Distinguishing operators

To distinguish between two operators, the metric used is the Diamond distance that is defined below for a general map from $n \times n$ density matrices to $m \times m$ density matrices.

Definition 5.1 (Diamond norm & distance). Given $\Phi : M_n(\mathbb{C}) \to M_m(\mathbb{C})$ which is a linear map and $X \in M_{n^2}(\mathbb{C})$ where $M_n(\mathbb{C})$ is the set of $n \times n$ complex valued matrices.

$$\|\Phi\|_\diamond := \max_{X; \|X\|_1 \leq 1} \|(\Phi \otimes \mathbb{I}_n)X\|_1, \tag{32}$$

Here $\|A\|_1 = \text{Tr}\sqrt{A^\dagger A}$. Using the above equation we define the diamond distance for two CPTP maps Φ_1 and Φ_2 as follows for density matrices ρ

$$d_\diamond(\Phi_1, \Phi_2) = \|\Phi_1 - \Phi_2\|_\diamond := \max_\rho \|(\Phi_1 \otimes \mathbb{I}_n)\rho - (\Phi_2 \otimes \mathbb{I}_n)\rho\|_1, \tag{33}$$

We also have the 1-norm defined similar to those of general matrices as

$$\|T\|_1 = \sup_{X \neq 0} \frac{\|T(X)\|_1}{\|X\|_1} \tag{34}$$

Here X is a matrix belonging to the set of complex matrices that are in the domain of T. The reason we require a notion of a diamond norm is that the 1-norm of operators are not stable with respect to tensoring with the identity, as pointed out in [1] a simple counter example is $T : |i\rangle\langle j| \to |j\rangle\langle i|$ $(i, j = 0, 1)$. Clearly $\|T\|_1 \leq 1$ however $\|T \otimes I_2\| \geq 2$ as can be checked by picking an $X = \sum_{i,j} |i, i\rangle\langle j, j|$. We now generalize the notion of what is a quantum gate as follows

Definition 5.2 (Quantum gate). A gate g of order (m, n) represents a general quantum gate which is a CPTP map from a n-qubit density matrix to a m-qubit density matrix. Its action is written as $g \circ \rho$

Unitary gates are a special case of a quantum gate. Using the notation defined we write the shorthand of $U \circ \rho = U^\dagger \rho U$. Using these gates one can bring a generalized notion of what a quantum circuit is.

Definition 5.3 (Quantum circuit). A quantum circuit $\mathcal{G}$ is a directed acyclic graph where each gate is a node such that the edges coming in and out of it are as per the defined order of the gate.

The topological sorts of this DAG will represent the orders in which we can apply these gates. The following are all proven in [1].

Lemma 5.1 (Properties of diamond norm). The following properties hold for a quantum gate $T : \mathbf{L}(\mathcal{N}) \to \mathbf{L}(\mathcal{M})$.

1. $\|T\|_\diamond = \|T \otimes I_\mathcal{G}\|_1 \geq \|T\|_1$ where $\dim \mathcal{G} \leq \dim \mathcal{N}$

2. $\|T \circ \rho\|_\diamond \leq \|T\|_\diamond \|\rho\|_1$

3. $\|T \circ R\|_\diamond \leq \|T\|_\diamond \|R\|_\diamond$

4. $\|T \otimes R\|_\diamond = \|T\|_\diamond \|R\|_\diamond$

5. $\|T\|_\diamond = 1$ if it is physically allowed.

Proof. The following are all proven in [1]. □

Lemma 5.2. Let T_1, T_2 and T_1', T_2' be super-operators such that they all have diamond norm less than 1 and $d_\diamond(T_i, T_i') \leq \epsilon_i$. Then it follows that $d_\diamond(T_2 \circ T_1, T_2' \circ T_1') \leq \epsilon_1 + \epsilon_2$.

Proof. Write $T_2' \circ T_1' - T_2 \circ T_1 = T_2' \circ (T_1' - T_1) + (T_2' - T_2) \circ T_1$ and then using triangle inequality along with norm properties in lemma 5.1.5, we get the errors to add up. $\square$

We will be making use of these properties to extend the proofs of scalable error propogation in qudits.

5.3 The world in qudits

We will first define any pure n qudit (with q levels) state as follows

$$|\psi\rangle = \sum_{\mathbf{j} \in (\mathbb{Z}/q\mathbb{Z})^n} z_{\mathbf{j}} |\mathbf{j}\rangle \tag{35}$$

Here we have all the states belonging to the ring of integers modulo q represented by $\mathbb{Z}/q\mathbb{Z}$. We define a new Pauli operator for these qudits as follows

$$X = \sum_{k \in \mathbb{Z}/q\mathbb{Z}} |k+1\rangle \langle k| \tag{36}$$

$$Z = \sum_{k \in \mathbb{Z}/q\mathbb{Z}} \omega^k |k\rangle \langle k| \tag{37}$$

Let $\mathbf{r}, \mathbf{s} \in (\mathbb{Z}/q\mathbb{Z})^n$ be vectors where dot product and addition is done over the ring for each component. We can upto global phase, declare q^{2n} Pauli operators as follows

$$X^{\mathbf{r}} Z^{\mathbf{s}} = \bigotimes_{i=1}^{n} X^{r_i} Z^{s_i} = \sum_{\mathbf{k} \in (\mathbb{Z}/q\mathbb{Z})^n} \omega^{\mathbf{k} \cdot \mathbf{s}} |\mathbf{k} + \mathbf{r}\rangle \langle \mathbf{k}| \tag{38}$$

For sake of completeness using the gate definitions [13] we define CNOT and SWAP but adapt those using the convention used in [28]

$$\mathrm{CNOT}_{ab} = \sum_{j,k \in \mathbb{Z}/q\mathbb{Z}} |j\rangle \langle j|_a \otimes |k+j\rangle \langle k| \tag{39}$$

$$\mathrm{SWAP}_{ab} = \sum_{j,k \in \mathbb{Z}/q\mathbb{Z}} |k\rangle \langle j|_a \otimes |j\rangle \langle k| \tag{40}$$

Now we define the Clifford group for qudits using our set of Pauli operations

Definition 5.4 (Clifford group for qudits). A unitary operation U is part of the Clifford group iff $U(X^{\mathbf{r}} Z^{\mathbf{s}})U \propto X^{\mathbf{r}'} Z^{\mathbf{s}'}$ where $\mathbf{r}, \mathbf{r}', \mathbf{s}, \mathbf{s}' \in (\mathbb{Z}/q\mathbb{Z})^n$. Here $\propto$ means that they are equal up to a global phase.

We now build toward defining an error probability tensor [28] using the Kraus operators constructed from the Pauli operations as shown in equation 38.

$$\mathcal{F}(\rho) = \sum_{\mathbf{r}, \mathbf{s} \in (\mathbb{Z}/q\mathbb{Z})^n} f_{\mathbf{r},\mathbf{s}} (X^{\mathbf{r}} Z^{\mathbf{s}}) \rho (X^{\mathbf{r}} Z^{\mathbf{s}})^{\dagger} \tag{41}$$

Here $\sum_{\mathbf{r},\mathbf{s}} f_{\mathbf{r},\mathbf{s}}$. We can additionally define $\mathcal{E}(\rho) = \sum_{\mathbf{r},\mathbf{s} \in (\mathbb{Z}/q\mathbb{Z})^n} p_{\mathbf{r},\mathbf{s}} (X^{\mathbf{r}} Z^{\mathbf{s}}) \rho (X^{\mathbf{r}} Z^{\mathbf{s}})^{\dagger}$. To understand error propagation for the Clifford group we define a $\mathcal{E}' = \mathcal{F} \circ \mathcal{E}$. We can now understand the new channel to be defined using tensorial equations.

$$\mathcal{E}'(\rho) = \sum_{\mathbf{r}, \mathbf{s} \in (\mathbb{Z}/q\mathbb{Z})^n} p'_{\mathbf{r},\mathbf{s}} (X^{\mathbf{r}} Z^{\mathbf{s}}) \rho (X^{\mathbf{r}} Z^{\mathbf{s}})^{\dagger} \tag{42}$$

We then define the following

$$p'_{\mathbf{r},\mathbf{s}} = \sum_{\mathbf{k},\mathbf{l} \in (\mathbb{Z}/q\mathbb{Z})^n} f_{\mathbf{r}-\mathbf{k},\mathbf{s}-\mathbf{l}} p_{\mathbf{k},\mathbf{l}} = \sum_{\mathbf{k},\mathbf{l} \in (\mathbb{Z}/q\mathbb{Z})^n} F_{\mathbf{r}}{}^{\mathbf{k}}{}_{\mathbf{s}}{}^{\mathbf{l}} p_{\mathbf{k},\mathbf{l}} \tag{43}$$

Here we define $F_{\mathbf{r}}{}^{\mathbf{k}}{}_{\mathbf{s}}{}^{\mathbf{l}} = f_{\mathbf{r}-\mathbf{k},\mathbf{s}-\mathbf{l}}$ as a tensor with $2n$ covariant and $2n$ contravariant indices. This gives us a general model for understanding error propagation in Clifford group circuits.

6 Arbitrary controlled gates using qudits

This section is based on the discussions provided in [38]. We extend the model provided for cascaded control gates.
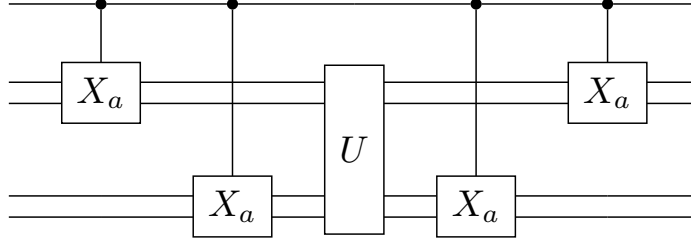
Figure 3: The channels with two lines represent 4 level qudits. Controlled X_a here has a normal qubit as control.

6.1 Using 4 level qudits

We first define the internal swap gate called X_a as follows

$$X_a = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \tag{44}$$

$$X_a |0\rangle = |2\rangle, \quad X_a |1\rangle = |3\rangle \tag{45}$$
$$X_a |2\rangle = |0\rangle, \quad X_a |3\rangle = |1\rangle \tag{46}$$

The main method described here is to use controlled X_a gates where the qubit is controlling a 4 level qudit. We also will assume that the gate U is applied on the qubit subspace of $|0\rangle$ and $|1\rangle$. Using this we can construct circuits as shown in figure 3.

Initially and finally only the bottom levels $|0\rangle$ and $|1\rangle$ are populated in all the qudits. If the swap operation is done the information in the lower states is transferred to the higher states on which the gate U doesn't act hence the output has the operation only applied if the control qubit is $|0\rangle$.

We can see that using a linear number of extra gates we can use this for implementing the controlled version of any multi-qubit unitary gate even if the operation by itself is unknown. This can also be easily extended to multiple qubits used as the control and also to extend multiple gates being controlled by the same register of qubits.

We can extend this to making controlled unitary with m controls simply by making the controlled X_a to be controlled by those m qubits. It is easy to see that this is equivalent to control U with m qubits control. We define X_{am} as the X_a gate being controlled by m qubits. Additionally we can note that due to the initial and final states having no population
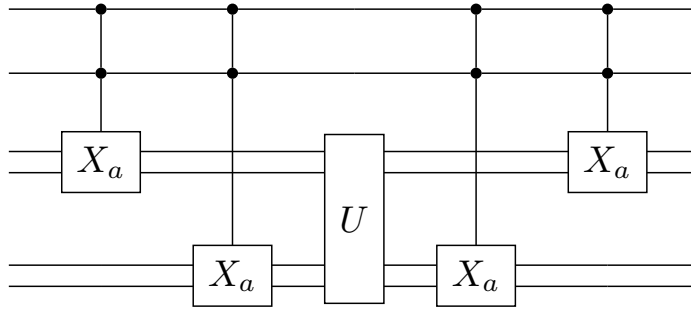

Figure 4: The channels with two lines represent 4 level qudits. Controlled X_a here has two normal qubits as control.

in the higher states we can append gates controlled by an independent register of qubits without any mixing up or loss of information.

6.2 Creating noisy gates

Lets define $X_{a,\epsilon}$ as X_a with some noise such that $\|X_{a,\epsilon} - X_a\|_\diamond \leq \epsilon$. Using lemma 5.2 if we have a qudit channel with n qudits being controlled by m qubits just using a single controlled-U (the case of figure 3). The error will add

11

up as

$$d_\diamond(cU', cU) \leq \sum_i d_\diamond(cX_a', cX_a) + d_\diamond(U, U) + \sum_{i=1}^{n} d_\diamond(cX_a', cX_a)$$
$$\leq 2n d_\diamond(cX_a', cX_a)$$
$$= \mathcal{O}(n\epsilon)$$

This implies that the scaling is linear due to the nature of the diamond norm being defined for CPTP maps on general matrices and not being limite to qubits. Suppose we wish to use some l control gates in succession such that the qubits controlling them are m in number each and not all control qubits are common in between the gates

$$d_\diamond(cU_1' \circ CU_2' \ldots cU_l', cU_1 \circ CU_2 \ldots cU_l)$$
$$\leq \sum_{i=1,k=0}^{n,l-1} d_\diamond(cX_{am}', cX_{am}) + \sum_{i=1}^{l} d_\diamond(U_i, U_i) +$$
$$\sum_{i=1,k=0}^{n,l-1} d_\diamond(cX_{am}', cX_{am}) = \mathcal{O}(nl\epsilon)$$

We can see that using the properties of the diamond norm the errors will add up linearly showing that the given model for extension to qudits is reliable in error scaling.

The main reason however that we were able to comfortably extend the inequalities laid out for these norms is mainly because we made us of a quantum gate which merely transforms from one kind of density matrix to another and there is no real constraint asking it to be a qubit density matrix. Hence the nature of adding up errors will very smoothly transfer here.

7 Quantum error correction

When using qubits, one can generalize a single qubit to have two kinds of errors. The first one being the bit flip occurring by applying X on the state and the other being the phase filp occurring by applying Z on the state. We have already defined channels using Kraus superoperators and we can see that for the single qubit gate X and Z can combine to give arbitrary errors.

A very common notion in classical error correction is to introduce redundancy to help in detecting errors. One of these redundancies is to repeat the bits and so the probability that all the bits have some error will eventually tend to zero however there are significantly better methods than to use repeaters such as partity check bits and such. One cannot however clone a quantum state thanks to the No-Cloning theorem [31]. However this is where the notion of a logical qubit comes into play. We can define a logical qubit as $|0_L\rangle = |000\rangle$ and $|1_L\rangle = |111\rangle$ and while this isn't cloning the state, a single qubit state can be converted to a single logical qubit here easily by just 2 CNOTs. We now proceed to generalize this notion using error correcting codes.

7.1 Stabilizer codes

The definition of a stabilizer code is given as follows

Definition 7.1 (Stabilizer code). A stabilizer code on q-ary qudits is written as $[[n, k, d]]_q$ where n is the number of qudits that the states are over, k is $\log_q(m)$ where m is the number of messages that we are encoding and d is minimum the hamming distance between these messages. The number of stabilizers would be $n - k$.

Definition 7.2 (Stabilizers). For a $[[n, k, d]]_q$ code, a stabilizer is an operation $\hat{P}$ that any state that is a valid message for this code is an eigenvector of $\hat{P}$ with eigenvalue $= 1$. The number of independent stabilizers of any code is $n - k$ in number. All stabilizers must belong to the group of Pauli operations of the n qudit system and any two stabilizers P_i and P_j must commute.

The definition by itself is extremely general but for the sake of this section we will set $q = 2$ and so in this case we will omit that subscript. We can take certain examples starting with the earliest example of a quantum error correcting code: the Shor's code. The Shor's code is a $[[9, 1, 3]]$ code, meaning that one logical qubit is made of 9 real qubits. The specialty of this code is that it can correct any arbitrary error of a single qubit meaning that if one of these 9 qubits has some error, that would have to be represented as a combination of bit and phase flips, we can correct that error.
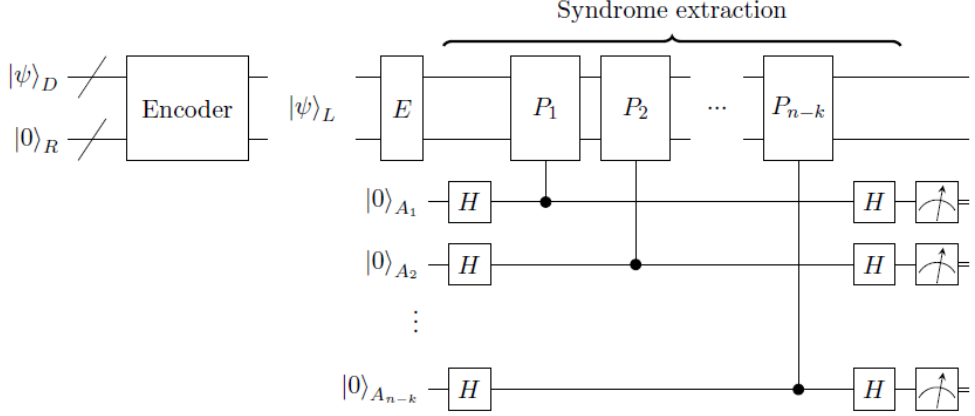
Figure 5: A general scheme for error correction using stabilizer codes. Here P_1 to P_{n-k} represent the stabilizers and the ancillary qubits $|0\rangle_{A_1}$ to $|0\rangle_{A_{n-k}}$ are measured to extract the error. Figure taken from [33].

The logical qubits are given as

$$|0_L\rangle = \frac{(|000\rangle + |111\rangle)(|000\rangle + |111\rangle)(|000\rangle + |111\rangle)}{2\sqrt{2}} \tag{47}$$

$$|1_L\rangle = \frac{(|000\rangle - |111\rangle)(|000\rangle - |111\rangle)(|000\rangle - |111\rangle)}{2\sqrt{2}} \tag{48}$$

The idea behind building it up this way is that this directly combines the seperate 3 qubit codes for detecting phase flips and bit flips seperately in a way. The bit flip detecting code has stabilizers as $Z_1 Z_2$ and $Z_2 Z_3$ since if each have eigenvalue equal to 1 this implies that the state is purely composed of linear combination of $|000\rangle$ and $|111\rangle$. In case any of the bits flip the eigenvalue no longer is 1 however we can see that two simultaneous errors can also be treated as 1 error which is where this code begins to fail.

Similarly for the phase flip the distinguishing states are $|+\rangle$ and $|-\rangle$ as they differ only in phase. Hence we can keep the logical qubits as $|0_L\rangle = |+++\rangle$ and $|1_L\rangle = |---\rangle$ and then the stabilizers become $X_1 X_2$ and $X_2 X_3$. On concatenating these codes together where essentially each qubit in the bit flip code is encoded as the phase flip logical qubit we get the Shor's code as we can see from the previously defined logical qubits.

There are however better codes such as the Steane code that is $[[7,1,3]]$ and the 5-qubit code that is $[[5,1,3]]$ that are each capable of detecting arbitrary single qubit errors. In practice the Steane code, despite not being completely optimal in size, is more often used due to it belonging to a class of codes referred to as CSS (Calderbank-Shor-Steane) codes. CSS codes are a special set of Stabilizer codes that are derived from classical codes and have certain special properties. The Steane code is the first among the Hamming code family $[[2^r - 1, 2^r - 1 - 2r, 3]]$ where $r \geq 3$ and is also a color code that is a type of Topological stabilizer code [25].

Definition 7.3 (Topological stabilizer code)**.** Topological stabilizer codes are a class of stabilizer codes with geometrically local generators. This means that the physical qubits can be placed on a manifold and so the stabilizers only have local support.

In this report we will not be going in detail for topological codes but will be mainly analyzing the $[[5,1,3]]$ code. The 5-qubit code is the smallest code that is capable of correcting a single error (we can check optimality of this via the bounds in the following subsection). It has the following stabilizers in the qubit case

$$XZZXI, \quad IXZZX, \quad XIXZZ, \quad ZXIXZ \tag{49}$$

 In figure 6, the circuit for syndrome measurement is depicted. The logical qubits here are defined as follows where P_1 to P_4 are the stabilizers of the 5-qubit code.

$$|0_L\rangle = \frac{1}{4}(1 + P_1)(1 + P_2)(1 + P_3)(1 + P_4)\,|00000\rangle \tag{50}$$

$$|1_L\rangle = \frac{1}{4}(1 + P_1)(1 + P_2)(1 + P_3)(1 + P_4)\,|11111\rangle \tag{51}$$
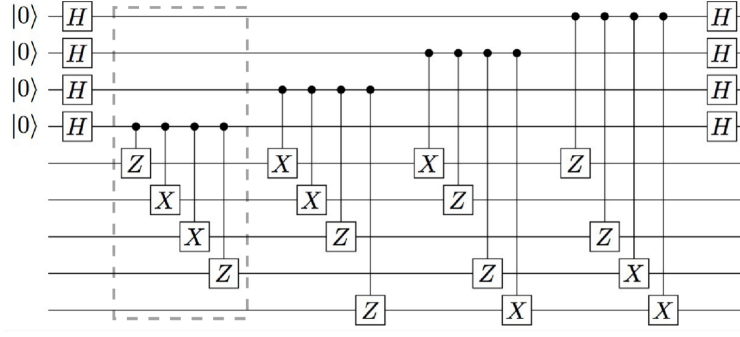
13

Figure 6: Circuit for syndrome measurement of the 5-qubit code. The Ancillas are the top 4 qubits and the bottom 5 carry the logical qubit.

7.2 Bounds

There are several bounds to the kind of stabilizer codes that one can define and these often arise purely from classical coding theory [17]. Here we list down a few of the ones relevant here.

Theorem 7.1 (Quantum Hamming bound). A $[[n, k, 2t + 1]]_q$ code that is capable of correcting t errors has the following hold

$$\sum_{j=0}^{t} 3^j \binom{n}{j} q^k \leq q^n \tag{52}$$

As n, k and t get large, this takes on an asymptomatic form. Here $H(x)$ is Hamming entropy.

$$\frac{k}{n} \leq 1 - \frac{t}{n} \log_q 3 - H\left(\frac{t}{n}\right) \qquad H(x) = -x \log_q x - (1 - x) \log_q (1 - x) \tag{53}$$

Proof. In the Hamming space of all possible q-ary strings of size n, all codewords have a sphere which of radius t which contains states that can be error corrected to this codeword. The distance t is signified as the hamming distance up to which it appropriately detects the error hence t number of errors. As there are $\binom{n}{j}$ number of j errors and each error has three different possibilities of σ_x, σ_y and σ_z, we get the above condition for the spheres to not overlap. Adapted from proof presented in [14]. $\qquad \square$

The difference between the classical and quantum Hamming bounds is the 3^j factor since in the quantum case one can compose the errors in 3 different ways whereas in the classical case it is treated simply as the q-ary string differing at that point.

Theorem 7.2 (Gilbert-Varshamov bound). A $[[n, k, 2t + 1]]_q$ code that is capable of correcting t errors has the following hold

$$\sum_{j=0}^{2t} 3^j \binom{n}{j} q^k \geq q^n \tag{54}$$

As n, k and t get large, this takes on an asymptomatic form.

$$\frac{k}{n} \geq 1 - 2\frac{t}{n} \log_q 3 - H\left(\frac{2t}{n}\right) \tag{55}$$

The proof of Gilber-Varshamov bound is an extension of the Hamming bound proof and is hence ommitted from here.

Theorem 7.3 (Knill-Laflamme bound). For a general $[[n, k, d]]_q$ code we have the following hold

$$n - k \geq 2(d - 1) \tag{56}$$

Proof. Suppose we choose any $d - 1$ qudits and then remove them. The remaining $n - d + 1$ qudits should contain enough information to reconstruct not only the q^k possible codewords but also the state of the missing qudits. The missing qudits can be any we can choose those that maximize entropy which result in the above bound [23]. $\qquad \square$

We can see that the $[[5, 1, 3]]_q$ code is optimal to correct single errors from this. These codes for which the lower bound is satisfied are known as quantum MDS (maximum distance seperable) codes [15].

7.3 Concatenating codes

The $[[5, 1, 3]]_q$ encodes a single logical qudit to 5 qudits. We could instead define a new logical qudit that encodes a single logical qudit to 5 of the previous logical qudits hence 25 qudits.

We can concatenate the codes in this manner and increase the distance of the codes. On concatenating m times we get 5^m codes and since the original minimum distance is 3, on concatenating the minimum has to be 3^m.

Given that d increases we have prospects of having an increased t errors to be corrected we can guess that concatenation has benefits in terms of scaling and so we do a rough analysis for the same. We have assumed that all the codes
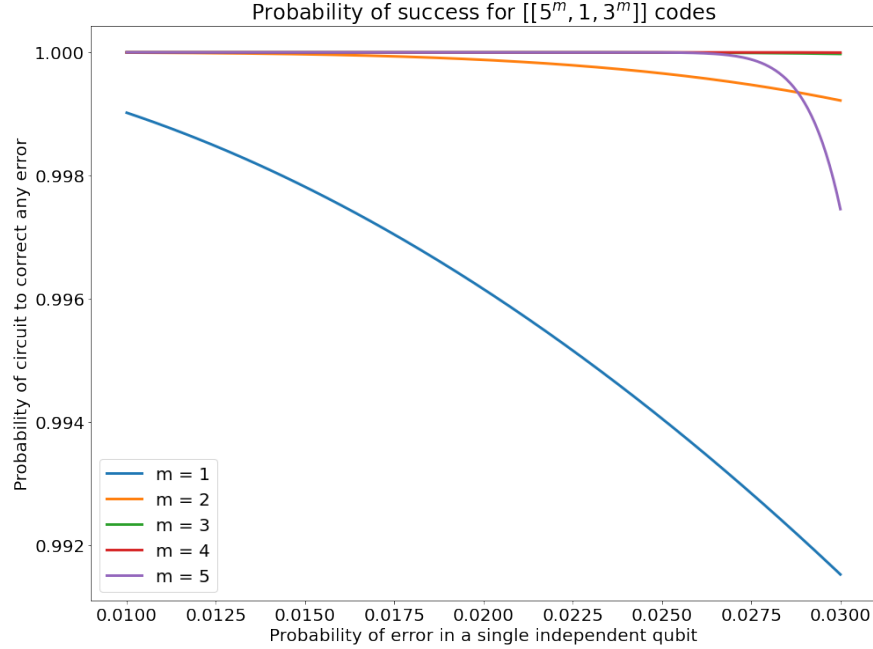
Figure 7: Shown above is the graph showing the probability of success for the independent qubit probability of error ranging form 0.01 to 0.03 and we can see that the concatenated codes perform significantly better than the regular code however there is a sharp dip for $m = 5$. One must note however that before this dip the values are tremendously close to 1 for $m = 2, 3, 4, 5$ as compared to the $m = 1$ case.

$[[5^m, 1, 3^m]]_q$ are of form where they can correct $(3^m - 1)/2$. Now we assume that each qudit has a probability p of failing which means that this particular qudit has a single error act on it.

Hence the number of errors would be decided by a binary distribution where the probability of j qubits failing is $\binom{n}{j} p^j (1 - p)^{n-j}$. The probability of success of any code is

$$p_{success} = \sum_{j=0}^{(3^m - 1)/2} \binom{n}{j} p^j (1 - p)^{n-j}$$

Interestingly concatenating the codes does not benefit the error correcting capabilities since even though we can correct more errors, if the probability of failure is not very small, the number of qudits presents overpowers the fact that we can correct more errors and this happens to such an extent that even if the qubits are 10% error prone, $[[5, 1, 3]]_q$ is our best choice.

We can concatenate any code $[[n, k, d]]$ by encoding each of the n qudits as a logical qudit of a $[[n', 1, d']]$ code. Since the distance is a minimum and the concatenate encodes a single to another string each of which has a minimum distance, the overall code has the distances multiply. Hence the new code is $[[n'n, k, d'd]]$.

The code in question for demonstrating something capable of protecting entangled logical qudits would boil down to a $[[20, 2, 6]]_q$ code. This can correct 2 arbitrary errors which cannot be said to be optimal as a distance of 7 is possible (using Hamming bound).

This shows that concatenation is on a whole not optimal but is surely more easy to construct circuits for.
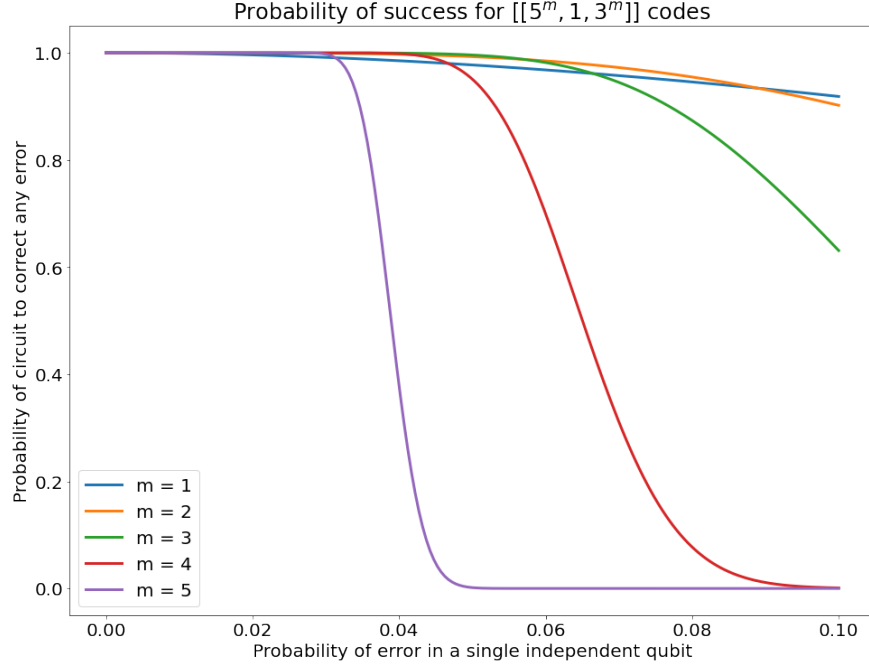
Figure 8: As we can see here the fall in success foreshadowing the dip in the previous figure manifests itself well before even the 0.1 mark. One can see here that the shear number of qudits ends up overwhelming the error correction and puts the odds against itself and these codes all sharply dip and by 0.1 error probability, the $m = 1$ outperforms all the others to a very notable extent.

7.4 Error codes in qudits

We have so far checked that the error scaling remains the same even in qudits as it does in qubits using the properties of the diamond norm. However we are yet to describe exact codes that are capable of error correcting in qudits. While we have described error correcting codes in a general sense in the bounds, the $[[5, 1, 3]]$ code described was for qubits. In [16], various results on qudits are discussed including the use of quantum error correcting codes for qudits. However a lot of these results deal with restricting the number of levels q to be a prime number. Before we generally move into error codes for qudits we first look into the proof given in [10] for showing existence of $[[5, 1, 3]]_q$.
The necessary and sufficient encoding condition for a QECC is given here as

$$\langle i_{\text{Encode}}|A^\dagger B|j_{\text{Encode}}\rangle = \lambda_{A,B}\delta_{ij} \tag{57}$$

Here $|i_{\text{Encode}}\rangle$ and $|j_{\text{Encode}}\rangle$ are encoded states and $\lambda_{A,B}$ is a factor dependent purely on A, B and not on i or j. The character values of any representation chosen for the finite additive group $\mathbb{Z}_q$, $\chi : \mathbb{Z}_q \to \mathbb{C}$ will have $\sum_{m \in \mathbb{Z}_N} \chi(m) = q$ if it is the trivial representation and $= 0$ otherwise. This condition can be written as

$$\sum_{m=0}^{q-1} \omega_q^{mk} = \begin{cases} q & \text{if } k = 0 \mod N \\ 0 & \text{otherwise} \end{cases} \tag{58}$$

Where ω_q is the q-th complex root of unity. Using this we define the encoding

$$|k_L\rangle = \frac{1}{q^{3/2}} \sum_{l,m,n=0}^{q-1} \omega_q^{k(l+m+n)+ln} |l + m + k, l + n, m + n, l, m\rangle \tag{59}$$

Here $|k_L\rangle$ represents the encoded/logical qudit and k goes from 0 to $q - 1$. We then define $E_{i,\alpha}$ as the operator that performs the error operation E_α on the ith qudit for $i = 1$ to 5. Choosing different combinations of i, j one can prove that the following holds

$$\langle k_L|E_{i,\alpha}^\dagger E_{j,\beta}|k_L'\rangle = \delta_{k,k'}\Lambda_{i,\alpha,j,\beta} \tag{60}$$

Clearly this satisfies the condition for a QECC. The values of the term $\Lambda_{i,\alpha,j,\beta}$ are all calculated in [10]. This tells us that using this form of the mapping we can define a $[[5, 1, 3]]_4$ code for error correction in circuits for creating arbitrary

16

controlled gates as described in section 6.

Another interesting result in qudits is the existence of optimal MDS codes for all prime power q described in [15].

Theorem 7.4. If q is an arbitrary prime power, then for all $3 \leq n \leq q$ and $1 \leq d \leq n/2 + 1$ there exists quantum MDS codes $[[n, n - 2d + 2, d]]_q$. Additionally for some $2 \leq d \leq q$ and some s there exists quantum MDS codes $[[q^2 - s . q^2 - 2d + 2 - s, d]]$

The proof of this can be found in detail in [15] and has been omitted from this report. This theorem strengthens the case for qudits as it tells us that there are certain optimal codes that can only be made using qudits that cannot be using qubits. More specifically the case we are concerned with, of $q = 4$ is a prime power and so we can take advantage of this strength in physical implementations hence making this fairly viable.

8 Conclusion & discussion

In this report we have explored a new avenue for quantum computing in abandoning the restrictions of sticking to two levels. While section 6 describes the use of qudits mainly to add onto qubit based circuits, we can see that using the formalism defined for qudits and the error correcting capabilities ([15, 10, 16]) there is a lot more that can be achieved with qudits. The main aim here was to illustrate the iterate approach described in [26] and how we could guarantee efficiency by using 4-level qudits for arbitrary controlled qubit gates [38] in conjunction to this, hence allowing us to efficiently generate higher moments of any LCU Hamiltonian. In section 4.2 we construct the LCU Block encoding and this can be efficiently constructed using $\mathcal{O}(dC)$ and further if the iterate can be constructed using the qudit approach, we can efficiently construct the iterate oracle in $\mathcal{O}(dC + kN)$ gates where k is some constant and N is the dimension of the Block encoding.

This approach can be used in the context of various application including imaginary time evolution [1] and even for the variational quantum eigensolver [3]. Further these methods also aid in general estimation algorithms such as the transition amplitude estimation algorithm (TAEA) described in [32] using Block encoding and cements the error scalability of the technique.

There is yet to be complete bench marking of qudit based quantum computation with the experimental implementations ([22, 12, 30]) usually not focusing on error correction protocols. However there has been interesting discussion regarding qudit repeater lines in [28] used along with error correcting codes. Superconducting qubits having been implemented on anharmonic oscillator models make a prime candidate for the step to using more levels under careful parameter selection as shown in some qutrit implementations [37].

The beauty of error correcting codes in qudits is illustrated in [15] where optimality may be fairly practical using qudits. A large issue with error correction in quantum computers has been the scale of redundancy but perhaps this illustrates that one can get better codes for larger numbers of qudits instead of sticking to qubits and would be interesting to compare these with results such as those presented in [24] that rely on making 4 copies of the circuit.

Acknowledgements

References

[1] Dorit Aharonov, Alexei Kitaev, and Noam Nisan. "Quantum Circuits with Mixed States". In: *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing*. STOC '98. Dallas, Texas, USA: Association for Computing Machinery, 1998, pp. 20–30. ISBN: 0897919629. DOI: 10.1145/276698.276708. URL: https://doi.org/10.1145/276698.276708.

[2] Kinjal Basu, Ankan Saha, and Shaunak Chatterjee. *Large-Scale Quadratically Constrained Quadratic Program via Low-Discrepancy Sequences*. 2017. arXiv: 1710.01163 [stat.ML].

[3] Kishor Bharti. *Quantum Assisted Eigensolver*. 2020. arXiv: 2009.11001 [quant-ph].

[4] Kishor Bharti and Tobias Haug. "Iterative quantum-assisted eigensolver". In: *Phys. Rev. A* 104 (5 Nov. 2021), p. L050401. DOI: 10.1103/PhysRevA.104.L050401. URL: https://link.aps.org/doi/10.1103/PhysRevA.104.L050401.

[5] Kishor Bharti et al. *Noisy intermediate-scale quantum (NISQ) algorithms*. 2021. arXiv: 2101.08448 [quant-ph].

[6] Lennart Bittel and Martin Kliesch. "Training Variational Quantum Algorithms Is NP-Hard". In: *Physical Review Letters* 127.12 (Sept. 2021). ISSN: 1079-7114. DOI: 10.1103/physrevlett.127.120502. URL: http://dx.doi.org/10.1103/PhysRevLett.127.120502.

[7] Robin Blume-Kohout. "How distinguishable are two quantum processes? a.k.a. What is the error rate of a quantum gate?" In: *APS March Meeting Abstracts*. Vol. 2017. APS Meeting Abstracts. Mar. 2017, B52.006.

[8] M. Cerezo et al. "Variational quantum algorithms". In: *Nature Reviews Physics* 3.9 (Aug. 2021), pp. 625–644. ISSN: 2522-5820. DOI: 10.1038/s42254-021-00348-9. URL: http://dx.doi.org/10.1038/s42254-021-00348-9.

[9] Shantanav Chakraborty, András Gilyén, and Stacey Jeffery. "The power of block-encoded matrix powers: improved regression techniques via faster Hamiltonian simulation". In: July 2019. DOI: 10.4230/LIPIcs.ICALP.2019.33.

[10] H. F. Chau. "Five quantum register error correction code for higher spin systems". In: *Physical Review A* 56.1 (July 1997), R1–R4. ISSN: 1094-1622. DOI: 10.1103/physreva.56.r1. URL: http://dx.doi.org/10.1103/PhysRevA.56.R1.

[11] Artur K. Ekert et al. "Direct Estimations of Linear and Nonlinear Functionals of a Quantum State". In: *Physical Review Letters* 88.21 (May 2002). ISSN: 1079-7114. DOI: 10.1103/physrevlett.88.217901. URL: http://dx.doi.org/10.1103/PhysRevLett.88.217901.

[12] Z. Gedik et al. "Computational speed-up with a single qudit". In: *Scientific Reports* 5.1 (Oct. 2015), p. 14671. ISSN: 2045-2322. DOI: 10.1038/srep14671. URL: https://doi.org/10.1038/srep14671.

[13] Vlad Gheorghiu. "Standard form of qudit stabilizer groups". In: *Physics Letters A* 378.5-6 (Jan. 2014), pp. 505–509. ISSN: 0375-9601. DOI: 10.1016/j.physleta.2013.12.009. URL: http://dx.doi.org/10.1016/j.physleta.2013.12.009.

[14] Daniel Gottesman. "Stabilizer codes and quantum error correction". In: (May 1997). arXiv: quant-ph/9705052.

[15] Markus Grassl, Thomas Beth, and Martin Rötteler. "On optimal quantum codes". In: *International Journal of Quantum Information* 02.01 (Mar. 2004), pp. 55–64. ISSN: 1793-6918. DOI: 10.1142/s0219749904000079. URL: http://dx.doi.org/10.1142/S0219749904000079.

[16] Lane Gunderman. *Some Results on Qudit Quantum Error-Correction*. 2020. URL: http://hdl.handle.net/10012/15600.

[17] Venkatesan Guruswami. *15-859V: Introduction to Coding Theory, Spring 2010*. 2010. URL: https://www.cs.cmu.edu/~venkatg/teaching/codingtheory/.

[18] Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd. "Quantum Algorithm for Linear Systems of Equations". In: *Physical Review Letters* 103.15 (Oct. 2009). ISSN: 1079-7114. DOI: 10.1103/physrevlett.103.150502. URL: http://dx.doi.org/10.1103/PhysRevLett.103.150502.

[19] Lloyd C. L. Hollenberg. "Plaquette expansion in lattice Hamiltonian models". In: *Phys. Rev. D* 47 (4 Feb. 1993), pp. 1640–1644. DOI: 10.1103/PhysRevD.47.1640. URL: https://link.aps.org/doi/10.1103/PhysRevD.47.1640.

[20] Lloyd C. L. Hollenberg and N. S. Witte. "Analytic solution for the ground-state energy of the extensive many-body problem". In: *Phys. Rev. B* 54 (23 Dec. 1996), pp. 16309–16312. DOI: 10.1103/PhysRevB.54.16309. URL: https://link.aps.org/doi/10.1103/PhysRevB.54.16309.

[21] Abhinav Kandala et al. "Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets". In: *Nature* 549.7671 (Sept. 2017), pp. 242–246. ISSN: 1476-4687. DOI: 10.1038/nature23879. URL: http://dx.doi.org/10.1038/nature23879.

[22] E.O. Kiktenko et al. "Single qudit realization of the Deutsch algorithm using superconducting many-level quantum circuits". In: *Physics Letters A* 379.22-23 (July 2015), pp. 1409–1413. ISSN: 0375-9601. DOI: 10.1016/j.physleta.2015.03.023. URL: http://dx.doi.org/10.1016/j.physleta.2015.03.023.

[23] Emanuel Knill, Raymond Laflamme, and Lorenza Viola. "Theory of Quantum Error Correction for General Noise". In: *Physical Review Letters* 84.11 (Mar. 2000), pp. 2525–2528. ISSN: 1079-7114. DOI: 10.1103/physrevlett.84.2525. URL: http://dx.doi.org/10.1103/PhysRevLett.84.2525.

[24] Bálint Koczor. "Exponential Error Suppression for Near-Term Quantum Devices". In: *Physical Review X* 11.3 (Sept. 2021). ISSN: 2160-3308. DOI: 10.1103/physrevx.11.031057. URL: http://dx.doi.org/10.1103/PhysRevX.11.031057.

[25] Aleksander Kubica, Beni Yoshida, and Fernando Pastawski. "Unfolding the color code". In: *New Journal of Physics* 17.8 (Aug. 2015), p. 083026. ISSN: 1367-2630. DOI: 10.1088/1367-2630/17/8/083026. URL: http://dx.doi.org/10.1088/1367-2630/17/8/083026.

[26] Guang Hao Low and Isaac L. Chuang. "Hamiltonian Simulation by Qubitization". In: *Quantum* 3 (July 2019), p. 163. ISSN: 2521-327X. DOI: 10.22331/q-2019-07-12-163. URL: https://doi.org/10.22331/q-2019-07-12-163.

[27] Jarrod R. McClean et al. "Barren plateaus in quantum neural network training landscapes". In: *Nature Communications* 9.1 (Nov. 2018), p. 4812. ISSN: 2041-1723. DOI: 10.1038/s41467-018-07090-4. URL: https://doi.org/10.1038/s41467-018-07090-4.

[28] Daniel Miller et al. "Propagation of generalized Pauli errors in qudit Clifford circuits". In: *Physical Review A* 98.5 (Nov. 2018). ISSN: 2469-9934. DOI: 10.1103/physreva.98.052316. URL: http://dx.doi.org/10.1103/PhysRevA.98.052316.

[29] Mario Motta et al. "Determining eigenstates and thermal states on a quantum computer using quantum imaginary time evolution". In: *Nature Physics* 16.2 (Nov. 2019), pp. 205–210. ISSN: 1745-2481. DOI: 10.1038/s41567-019-0704-4. URL: http://dx.doi.org/10.1038/s41567-019-0704-4.

[30] Matthew Neeley et al. "Emulation of a Quantum Spin with a Superconducting Phase Qudit". In: *Science* 325.5941 (2009), pp. 722–725. DOI: 10.1126/science.1173440. eprint: https://www.science.org/doi/pdf/10.1126/science.1173440. URL: https://www.science.org/doi/abs/10.1126/science.1173440.

[31] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. 10th. USA: Cambridge University Press, 2011. ISBN: 1107002176.

[32] Guru-Vamsi Policharla and Sai Vinjanampathy. "Algorithmic Primitives for Quantum-Assisted Quantum Control". In: *Phys. Rev. Lett.* 127 (22 Nov. 2021), p. 220504. DOI: 10.1103/PhysRevLett.127.220504. URL: https://link.aps.org/doi/10.1103/PhysRevLett.127.220504.

[33] Joschka Roffe. "Quantum error correction: an introductory guide". In: *Contemporary Physics* 60.3 (July 2019), pp. 226–245. ISSN: 1366-5812. DOI: 10.1080/00107514.2019.1667078. URL: http://dx.doi.org/10.1080/00107514.2019.1667078.

[34] Erik Sjöqvist et al. "Geometric Phases for Mixed States in Interferometry". In: *Phys. Rev. Lett.* 85 (14 Oct. 2000), pp. 2845–2849. DOI: 10.1103/PhysRevLett.85.2845. URL: https://link.aps.org/doi/10.1103/PhysRevLett.85.2845.

[35] Harish J. Vallury et al. "Quantum computed moments correction to variational estimates". In: *Quantum* 4 (Dec. 2020), p. 373. ISSN: 2521-327X. DOI: 10.22331/q-2020-12-15-373. URL: https://doi.org/10.22331/q-2020-12-15-373.

[36] Justin Yirka and Yiğit Subaşı. "Qubit-efficient entanglement spectroscopy using qubit resets". In: *Quantum* 5 (Sept. 2021), p. 535. ISSN: 2521-327X. DOI: 10.22331/q-2021-09-02-535. URL: https://doi.org/10.22331/q-2021-09-02-535.

[37] M. A. Yurtalan et al. "Implementation of a Walsh-Hadamard Gate in a Superconducting Qutrit". In: *Physical Review Letters* 125.18 (Oct. 2020). ISSN: 1079-7114. DOI: 10.1103/physrevlett.125.180504. URL: http://dx.doi.org/10.1103/PhysRevLett.125.180504.

[38] Xiao-Qi Zhou et al. "Adding control to arbitrary unknown quantum operations". In: *Nature Communications* 2.1 (Aug. 2011), p. 413. ISSN: 2041-1723. DOI: 10.1038/ncomms1392. URL: https://doi.org/10.1038/ncomms1392.