The Quantum Schur Transform and its Applications

Joey Li

September 21, 2019

Abstract

The quantum Schur transform is a fundamental protocol in quantum information theory which performs a change of basis from a local, qudit-level description of a system to a global, symmetry-based representation. More formally, Schur-Weyl duality allows the simultaneous decomposition of n-fold tensor products of d-dimensional complex space into irreducible representations of the unitary and symmetric groups, and the Schur transform is the particular change of basis from our standard basis to the basis induced by these actions. In 2005, [1] introduced an efficient implementation of the quantum Schur transform, which allowed many quantum information protocols to become experimentally viable. In this paper, we review their work and implement the quantum Schur transform on IBM's quantum computers. In addition, we study the use of the quantum Schur transform for the specific purpose of optimal qubit purification, as first outlined in [2].

1 Addition of Angular Momenta

To understand the Schur transform, it helps to have the context of the general formalism of addition of angular momenta in quantum mechanics. In particular, in the case of qubits, or $d = 2$ in the general Schur-Weyl duality, the Schur transform corresponds exactly to addition of angular momenta. Thus, we give a brief treatment of the topic in this section, generally seeking to highlight important results and ideas more than specific proofs. The treatment we give follows some combination of [3] and [4].

1.1 The J Operator

Recall that rotations $R(\mathbf{x}, \theta)$ in $\mathbb{R}^3$ by angle θ around axis $\mathbf{x}$ are isometries of the space, and thus correspond to elements of the Lie group $SO(3)$. In particular, the condition that the determinant is positive one corresponds to the fact that orientation is preserved under rotation. From a physics standpoint, we have the intuition from classical mechanics that angular momentum is an operator which generates rotation. Thus, mathematically, we want the operator J corresponding to angular momentum to satisfy the relation

$$R(\mathbf{x}, \epsilon) = e^{-i\epsilon J_{\mathbf{x}}}$$

for small $\epsilon > 0$.

As it turns out, this is possible. A key fact is that $\mathfrak{su}(2) \cong \mathfrak{so}(3)$, and in fact, $SU(2)$ is the universal space for all Lie groups which have Lie algebra $\mathfrak{su}(2)$. In particular, $SU(2)$ is a double cover of $SO(3)$. We may view the rotation operators as unitaries then, and this aids our intuition for obtaining these so-called J operators: we know from linear algebra that unitary matrices of determinant one are exponentials of i times a traceless Hermitian matrix. Thus, these angular momentum operators J correspond to traceless Hermitians. As an example, in the two qubit case, we have the familiar Pauli operators $\{\sigma_x, \sigma_y, \sigma_z\}$ as a basis for this space, and they can be modified to obtain angular momentum J operators:

$$\sigma_x = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \sigma_y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, \sigma_z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

In the above discussion, we have glossed over an important point: one can check that the matrices above have the property $[J_i, J_j] = ih\epsilon_{ijk}J_k$ where ϵ_{ijk} is a structure constant. This is in fact the defining point of angular momentum operators: their commutation relations. It gives us the natural Lie bracket through which we may define a Lie algebra corresponding to these

operators. Generally, we will consider the operators J_x, J_y, J_z corresponding to the x, y, z axes, which will determine our system.

1.2 Quantized Angular Momenta

Assume our system is in a finite-dimensional Hilbert space. This is a reasonable assumption for a system of n qudits. Now we wish to characterize the possible values for angular momentum. From above, we have that the operators J_x, J_y, J_z have nontrivial commutation relations, and thus cannot be simultaneously diagonalized. We can, however, define an angular momentum operator $J^2 = J_x^2 + J_y^2 + J_z^2$ which commutes with all of these operators, and thus can be simultaneously diagonalized with one of them. In particular, recall that our J operators are Hermitian, and thus can indeed be diagonalized into orthogonal eigenspaces. By convention, we choose to simultaneously diagonalize J^2 and J_z, taking eigenvectors $|a, b\rangle$ such that $J^2 |a, b\rangle = a |a, b\rangle$ and $J_z |a, b\rangle = b |a, b\rangle$.

Now to help us analyze these eigenvectors more closely, we may define "ladder operators" J_+ and J_- by $J_+ = \frac{1}{2}(J_x + iJ_y)$ and $J_- = \frac{1}{2}(J_x - iJ_y)$. These satisfy commutation relations $[J^2, J_\pm] = 0$ and $[J_z, J_\pm] = \pm\hbar J_\pm$. Since J^2 and $J_\pm$ commute, we see immediately that $J_\pm$ preserve the eigenvalue-a eigenspaces of our basis $|a, b\rangle$. On the other hand, we have

$$\begin{aligned} J_z J_\pm |a, b\rangle &= ([J_z, J_\pm] + J_\pm J_z) |a, b\rangle \\ &= \pm\hbar J_\pm |a, b\rangle + J_\pm J_z |a, b\rangle \\ &= (b \pm \hbar) J_\pm |a, b\rangle . \end{aligned}$$

and so if $J_\pm |a, b\rangle \neq 0$, it is also an eigenvector of $|a, b\rangle$, with eigenvalue $b \pm h$.

Now since we have a finite dimensional space, we claim we must have $J_\pm |a, b\rangle = 0$ for some $b = b_{max}$. We can in fact derive an expression for b_{max} by noting the following.

We have $J_+ = (J_-)^\dagger$ by definition, and thus the operator $J_+J_- + J_-J_+$ must be positive semidefinite. However, expanding we obtain $J_+J_- + J_-J_+ = \frac{1}{2}(J^2 - J_z^2)$. Then we must have

$$\langle a, b | J^2 - J_z^2 | a, b \rangle \geq 0$$

for all $|a, b\rangle$, or $a \geq b_{max}^2$. Then for every given a, we must have a b_{max} such that $J_+ |a, b_{max}\rangle = 0$. Further, we must have $J_-J_+ |a, b_{max}\rangle = 0$, and substituting in for J_-J_+, we obtain $(J^2 - J_z^2 - \hbar J_z) |a, b_{max}\rangle = 0$, or $a = b_{max}(b_{max} + \hbar)$. We may repeat an analogous argument using b_{min} to obtain $a = b_{min}(b_{min} - \hbar)$. It follows that $b_{min} = -b_{max}$ and since the ladder operators raise the eigenvalue by $\hbar$ that $b_{max} - k\hbar = -b_{max}$ or $b_{max} = \frac{k\hbar}{2}$. Then with appropriate substitutions, we may let $j = \frac{k}{2}$ so that $a = j(j+1)\hbar^2$ and m be such that $b = m\hbar$ to reparameterize our eigenvectors into the familiar $|j, m\rangle$ notation. Then we have that j must be a half-integer, and for given j the possible m values range from $-j, -j+1, \ldots, j-1, j$. Then

$$\begin{aligned} J^2 |j, m\rangle &= j(j+1)\hbar^2 |j, m\rangle \\ J_z |j, m\rangle &= m\hbar |j, m\rangle \end{aligned}$$

We may derive equations for the matrix elements of these transformations, if we so desire, using the equations above. For example, we have

$$\begin{aligned} \langle j, m | (J_+)^\dagger J_+ | j, m \rangle &= |c_{j,m}|^2 \langle j, m+1 | j, m+1 \rangle \\ &= |c_{j,m}|^2. \end{aligned}$$

but also

$$\begin{aligned} \langle j, m | (J_+)^\dagger J_+ | j, m \rangle &= \langle j, m | J^2 - J_z^2 - hJ_z | j, m \rangle \\ &= j(j+1)\hbar^2 - m^2\hbar^2 - m\hbar^2. \end{aligned}$$

By convention, we take $c_{j,m}$ to be real and positive, and thus we obtain

$$J_+ |j, m\rangle = \hbar\sqrt{(j-m)(j+m+1)}\, |j, m+1\rangle .$$

An analogous calculation gives a formula for J_-, and we can combine the two to obtain

$$J_\pm |j, m\rangle = \hbar\sqrt{(j \mp m)(j \pm m + 1)}\, |j, m \pm 1\rangle$$

In summary, angular momentum can only take discrete values in the quantum formalism, and we generally consider simultaneous eigenstates of the J^2 and J_z operators, moving between eigenspaces by means of the $J_\pm$ ladder operators. For given j, we obtain a $(2j+1)$-dimensional irreducible representation of $SU(2)$ with basis vectors $\{|j,m\rangle \mid m = -j, -j+1, \ldots, j\}$. The exact connection between the calculations of this previous section and irreducible representations of $SU(2)$ has something to do with the fact that our angular momentum operators generate a Lie algebra and representations of Lie algebras and Lie groups are closely related.

1.3 Addition of Angular Momenta

Generally, we will be interested in multipartite systems, and thus we would like to have some method of "adding" angular momenta. For example, one might want to add the orbital angular momentum of an electron to its spin, or in our case, we may like to add the spins of many electrons, representing a multi-qubit system in our computer.

Let V_1, V_2 have basis vectors $|j_1, m_1\rangle$ and $|j_2, m_2\rangle$, respectively. Then the joint space $V_3 = V_1 \otimes V_2$ has basis vectors $|j_1, j_2; m_1, m_2\rangle$. We would like to describe these basis vectors in terms of some total angular momentum, and consequently we define $J = J_1 \otimes 1 + 1 \otimes J_2$. We can analogously define $J_x = J_{1x} \otimes 1 + 1 \otimes J_{2x}$ and so on, and we find that these total angular momentum operators satisfy the same conditions as before. Now note our operators J, J_z, J_{1z}, J_{2z} commute, and thus we can also write our states in the eigenbasis $|j_1, j_2; j, m\rangle$ which is in some sense a more global description. Now we would like to have the change of basis

$$|j_1, j_2; j, m\rangle = \sum_{m_1, m_2} |j_1, j_2; m_1, m_2\rangle\langle j_1, j_2; m_1, m_2| \, |j_1, j_2; j, m\rangle$$

where we call the $\langle j_1, j_2; m_1, m_2 | j_1, j_2; j, m\rangle$ *Clebsch Gordan coefficients*. In total, there seem to be many Clebsch Gordan coefficients, but it turns out our space of results is restricted by the fact that we only get nonzero coefficients for $m = m_1 + m_2$ and $|j_1 - j_2| \leq j \leq j_1 + j_2$. The Clebsch Gordan transform, which we will explore shortly and will use heavily in our construction of the Schur transform, relies heavily on the calculation of these coefficients. To find these coefficients, we will often use a recursive strategy, but we will defer this discussion to [4].

2 The Schur Transform

In this section, we will give a general outline of the Schur transform following the presentation of [1], but we will restrict our focus to the qubit case because it is most directly relevant to our work.

2.1 Schur Weyl Duality

Schur Weyl duality refers to the decomposition of the action of the unitary group U_d and the permutation group S_n on $(\mathbb{C}^d)^{\otimes n}$ into irreducible representations (henceforth, irreps). In particular, if we let $P(s)$ and $Q(U)$ act on $(\mathbb{C}^d)^{\otimes n}$ by

$$P(s)\,|i_1 i_2 \ldots i_n\rangle = |i_{s^{-1}(1)} \ldots i_{s^{-1}(n)}\rangle$$
$$Q(U)\,|i_1 \ldots i_n\rangle = U^{\otimes n}\,|i_1 \ldots i_n\rangle\,,$$

then for some indexing set λ of irreps, we obtain

$$(\mathbb{C}^d)^{\otimes n} = \bigoplus_\lambda q_\lambda(U) \otimes p_\lambda(s).$$

The rest of this section will give some inutition as to why this happens.

Suppose we have (R_1, V_1), (R_2, V_2) representations[1] of G. Then the vector space $\mathrm{Hom}(V_1, V_2)$ is also a representation of G under the action

$$R_2(g)(\cdot)R_1(g)^{-1}.$$

We denote by $\mathrm{Hom}(V_1, V_2)^G$ the G-covariant maps, which commute with every element $g \in G$. More formally, $f \in \mathrm{Hom}(V_1, V_2)^G$ is a map $V_1 \to V_2$ satisfying $R_2(g) f R_1(g)^{-1} = f$ for all $g \in G$. By definition, if any $f \in \mathrm{Hom}(V_1, V_2)^G$ is invertible, we have $V_1 \cong V_2$.

[1] See B.2 for basic notions in representation theory.

Given any representation (R, V) we also have associated dual representation (R^*, V^*) given by the action $\langle v| \mapsto \langle v| R(g)^{-1}$. This can also be considered a representation on V under the natural correspondence between V and V^* given by the transpose, i.e., $R^*(g)|v\rangle = (R(g)^{-1})^T |v\rangle$.

Given a reducible representation, we have the so-called isotypic decomposition [5] into irreps,

$$\begin{aligned} R(g) &\cong \bigoplus_{\lambda \in \hat{\mathcal{G}}} \bigoplus_{i=1}^{n_\lambda} r_\lambda(g) \\ &\cong \bigoplus_{\lambda \in \hat{\mathcal{G}}} r_\lambda(g) \otimes I_{n_\lambda} \end{aligned}$$

where λ is some label drawn from $\hat{\mathcal{G}}$ the set of labels of irreps of G and n_λ indicates the multiplicity of irrep r_λ. This induces the decomposition of the space as

$$V \cong \bigoplus_\lambda V_\lambda \otimes \mathbb{C}^{n_\lambda}$$

or, noting that $\mathbb{C}^{n_\lambda}$ has the same structure as $\mathrm{Hom}(V_\lambda, V)$, we have

$$V \cong \bigoplus_\lambda V_\lambda \otimes \mathrm{Hom}(V_\lambda, V).$$

This turns out to be a useful decomposition. In particular, the isomorphism from the RHS to the LHS is particularly clean, given by $v \otimes f \mapsto f(v)$ and extended by linearity everywhere.

Applying this to the specific cases of $P(s)$ and $Q(U)$ as given above, we obtain

$$\begin{aligned} P(s) &\cong \bigoplus_\alpha p_\alpha(s) \otimes I_{n_\alpha} \\ Q(U) &\cong \bigoplus_\beta q_\beta(U) \otimes I_{m_\beta}, \end{aligned}$$

but since the actions of $P(s)$ and $Q(U)$ commute, we have by Schur's Lemma that the irreps $q_\beta(U)$ act on the multiplicities I_{n_α} and likewise, so that

$$Q(U)P(s) \cong \bigoplus_\alpha \bigoplus_\beta m_{\alpha,\beta} q_\beta(U) \otimes p_\alpha(s).$$

However, it turns out that not only do $P(s)$ and $Q(U)$ commute, but these two groups centralize each other, allowing us to note that each of the $m_{\alpha,\beta}$ is zero or one, giving

$$Q(U)P(s) \cong \bigoplus_\lambda q_\lambda(U) \otimes p_\lambda(s).$$

This allows us to decompose the space as

$$(\mathbb{C}^d)^{\otimes n} \cong \bigoplus_\lambda Q_\lambda^d \otimes P_\lambda$$

as desired. The λ indexing the expression are given by partitions of n into $\leq d$ parts.

2.2 Implementation

The main result of [1] was an efficient recursive implementation of the quantum Schur transform for n qudits. However, this requires some more representation theory and quantum mechanics, and thus we will instead restrict our attention to the qubit case, which is equivalent to understanding addition of angular momentum. Thus, we summarize the result on qubits from [1].

The basic intuition is that one can successively add the angular momentum of each new qubit to our existing system using a Clebsch Gordan transform, and then cascade these Clebsch Gordan transforms to obtain a Schur transform. The addition of angular momentum follows the typical

calculation of the coefficients, with the permutation label of the irrep arising from the different pathways one might take to reach a certain total J.

In particular, if we let J be the total angular momentum of a state and m be the z-component of angular momentum, the Clebsch-Gordan transform U_{CG} takes in a state $|J, m\rangle$ along with a spin $|s\rangle$ and outputs a linear combination of the possible total angular momentums, $|J \pm \frac{1}{2}, m \pm \frac{1}{2}\rangle$, along with a permutation label. The amplitudes of these states are derived from the ladder operators as explained in [4]. Formally, U_{CG} is given by a rotation

$$\begin{bmatrix} |J'_-, m', p = -\frac{1}{2}\rangle \\ |J'_+, m', p = +\frac{1}{2}\rangle \end{bmatrix} = \begin{bmatrix} \cos\theta_{J,m'} & -\sin\theta_{J,m'} \\ \sin\theta_{J,m'} & \cos\theta_{J,m'} \end{bmatrix} \begin{bmatrix} |J, m_+\rangle\,|s = -\frac{1}{2}\rangle \\ |J, m_-\rangle\,|s = \frac{1}{2}\rangle \end{bmatrix} \tag{1}$$

where $J'_\pm = J \pm 1/2$, $m_\pm = m' \pm 1/2$, and $\cos\theta_{J,m'} = \sqrt{\frac{J+m'+1/2}{2J+1}}$. The circuit for U_{CG} is given below,

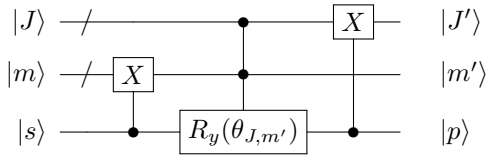

Figure 1: Circuit for the Clebsch-Gordan transform. The rotation gate implements the rotation matrix given in equation 1.

As mentioned above, the Schur transform is constructed simply by stringing together these Clebsch-Gordan transforms, as shown below in the diagram from [5],

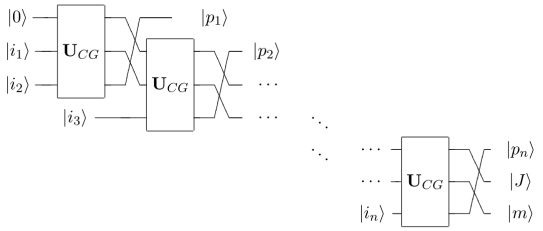

Figure 2: Circuit for the Schur transform, taken from [5] .

One of the goals of this project was to implement this circuit on IBM Q's quantum computers. The implementation is given in section 3.

2.3 Application to Optimal Single Qubit Purification

The Schur transform applies directly to the procedure for optimal single qubit purification when restricted to the case of qubits. In particular, the decomposition of $\rho^{\otimes n}$ referenced in [2] is precisely the consequence of rewriting this state in the Schur basis.

Following the suggestion of Professor Marvian, we propose a simplified version of the Schur transform for use on states of the form $\rho^{\otimes n}$ which will help optimize our circuit. Note that since $\rho^{\otimes n}$ is clearly invariant under the action of the permutation group, we have the decomposition

$$\rho^{\otimes n} = \bigoplus_j p_j \rho_j \otimes I_{d_j}.$$

In particular, the relevant information regarding the state is all stored in p_j and ρ_j, and p_j is dependent only on j and not the permutation label. Therefore, in any protocol in which we implement a Schur transform and then measure the permutation labels, we may assume by the principle of deferred measurement that the permutation labels are measured after each Clebsch Gordan transform instead, giving an equivalent circuit.

We may further simplify by noting that the permutation labels give the transitions in angular momentum, and thus the J register of the Schur transform may be replaced by a classical register

which is updated based on the measurements of the p_i values. Thus, we only need quantum registers to store the m values and each spin which is added.

Note this apparatus simplifies the protocol for optimal qubit purification significantly. We can also implement an additional simplification. Note that a restrictive step in the implementation of the protocol is the construction of the $U_{j,\alpha}$ transforms which taken an arbitrary $\rho_{j,\alpha}$ to $\rho_{j,1}$. Since we are measuring the p labels as we go, we claim we can reduce the problem of constructing these $U_{j,\alpha}$ to a subset of all $U_{j,\alpha}$ by rotating our state $\rho_{j,\alpha}$ into the standard $\rho_{j,1}$ after each measurement of the p register.

3 Building the Schur Transform in IBM Q

All code for these implementations can be found on Github at https://github.com/Octophi/pruv19. The user should have the latest version of QISKit in order to properly run these notebooks.

To understand the Schur transform, we implemented it for two and three qubits in the IBM Q interface. A more general implementation for n qubits will require a J register of roughly $\log_2 n$ qubits, a m register of roughly $\log_2 n + 1$ qubits, more general circuits for addition, and a general construction of the controlled rotation gate. This code can be found in the file 'Two and Three Qubit Schur'.

Additionally, we have calculated the matrices for the Schur basis implementations of several permutations. Since permutations reduce to their actions on irreps in the Schur basis, one might expect that they would afford relatively clean implementations, and we have worked out these implementations for the case of three qubits. One can work out what the actions of these irreps should be or look them up in [6]. Since the three qubit case is relatively simple, we also wrote a script in the file 'Matrix Calculator' to compute the change of basis for a desired permutation, which confirmed the transformation predicted by [6]. The challenge, however, is building the circuits to implement the desired actions on the irreps.

In particular, in our implementation of the three qubit Schur transform, we use five qubits in order to store all the information in the $|J\rangle$, $|m\rangle$ and $|p\rangle$ registers. This poses a challenge when constructing the physical circuits which correspond to the desired logical operations we have calculated. In the case of the (12) permutation, the desired operation is essentially equivalent to performing a Z gate on the irrep indexed by the $(2,1)$ Young tableaux, and conveniently, this corresponds with applying a CZ gate with the $|J\rangle$ register as control and the fourth qubit as our target. In the case of the (23) permutation, it is more difficult to realize our logical operator. We see that the permutation should have no impact on the subspace with $J = 3/2$, while it should apply the matrix

$$\begin{bmatrix} -1/2 & \sqrt{3}/2 \\ \sqrt{3}/2 & 1/2 \end{bmatrix}$$

to the subspace with $J = 1/2$. Physically, this means we must apply a gate which acts by the above matrix on the states $|01\rangle$ and $|10\rangle$ in the fourth and fifth qubit slots, and do so only if the top qubit is one. This would seem to be a controlled rotation, but this is somewhat complicated by the fact that we are not rotating a single qubit, but rather, between the states $|01\rangle$ and $|10\rangle$. We have tried to make the appropriate adjustment by sandwiching a controlled rotation from the J qubit to the first qubit of the $|p\rangle$ register between two CCX gates, however, this has introduced additional entanglement which is causing some factors of other permutations to appear. Currently, we are working on fixing this issue.

4 Future Directions

One immediate goal for future work would involve finishing the experimental runs for all the proposed protocols above. In particular, as of now we have not finished implementing the permutation matrices in the Schur basis. Additionally, it might be worthwhile to run these protocols on IBM's actual computers, as so far all runs have been conducted on the simulators. This may happen soon, contingent on continued work on this project and the acquiring of enough IBM Q credits to properly run the experiment.

Future theoretical work on this project would involve solidifying understanding of the proof of Schur-Weyl duality and of the efficient implementation of the Schur transform for a general

qudit scenario. From a practical standpoint, it is unlikely that the general Schur transform will be relevant in the near future, but it is an important protocol to understand theoretically.

Another potential avenue for continued work on this project would involve developing a general framework for n qubit Schur transforms using the standard gates, as well as perhaps developing a circuit for the transform adapted to the natural gates of ion trap quantum computers.

One other topic that could be interesting would be studying the practical efficiency of the qubit purification procedure of [2] relative to simpler, more naive procedures for qubit purification. We have carried out some preliminary analysis on this topic in A, but have not had the chance to actually run these experiments on IBM's computers. The value of such an experiment might be somewhat nebulous, however, as the relative quality of different protocols will surely change with the advances of new hardware for quantum computers.

References

[1] Dave Bacon, Isaac L Chuang, and Aram W Harrow. Efficient quantum circuits for schur and clebsch-gordan transforms. *Physical review letters*, 97(17):170502, 2006.

[2] J. I. Cirac, A. K. Ekert, and C. Macchiavello. Optimal purification of single qubits. *Phys. Rev. Lett.*, 82:4344–4347, 1999.

[3] Peter Woit, Woit, and Bartolini. *Quantum theory, groups and representations*. Springer, 2017.

[4] Jun John Sakurai and Eugene D Commins. *Modern quantum mechanics, revised edition*. AAPT, 1995.

[5] Dave Bacon, Isaac L Chuang, and Aram W Harrow. The quantum schur transform: I. efficient qudit circuits. *arXiv preprint quant-ph/0601001*, 2005.

[6] Jin-Quan Chen, Jialun Ping, and Fan Wang. *Group representation theory for physicists*. World Scientific Publishing Company, 2002.

[7] William Fulton and Joe Harris. *Representation theory: a first course*, volume 129. Springer Science & Business Media, 2013.

A A Simplified Approach to Qubit Purification

Suppose as in [2] that we would like to purify several copies of a mixed state $\rho = a|1\rangle\langle 1| + (1-a)\frac{1}{2}I$. While [2] has given an outline of the optimal procedure for such a task, one might ask whether its usefulness is hampered by the relative complexity of performing the Schur transform. In this section, we consider an alternative approach to qubit purification given by the simple naive procedure: tensor two copies of ρ together, measure the Swap operator on the pair, and trace over one of the qubits. Here, we seek to optimize this procedure and compare it to the [2] procedure.

This purification procedure works as follows. Since Swap is a Hermitian, unitary operator, it has eigenvalues of ± 1, and thus the operators $\frac{I+Swap}{2}$ and $\frac{I-Swap}{2}$ are projections onto the positive and negative eigenspaces of Swap, respectively. These spaces are spanned by $\{|00\rangle, |11\rangle, \frac{|01\rangle+|10\rangle}{2}\}$ and $\{\frac{|01\rangle-|10\rangle}{2}\}$, respectively. By inspection, if we project onto the negative eigenspace and trace out a qubit, our resulting state is the completely mixed state. However, if we project onto the positive eigenspace, we will gain information.

To see this, note that if we project onto the positive eigenspace, we will obtain

$$\begin{aligned}\frac{I+Swap}{2}(\rho\otimes\rho)\frac{I+Swap}{2} &= \frac{1}{4}(\rho\otimes\rho + Swap(\rho\otimes\rho) + (\rho\otimes\rho)Swap + Swap(\rho\otimes\rho)Swap)\\ &= \frac{1}{4}(2\rho\otimes\rho + Swap(\rho\otimes\rho) + (\rho\otimes\rho)Swap)\end{aligned}$$

and noting that this expression is symmetric in both qubits, we may trace over one of them, using

$$\begin{aligned}\mathrm{Tr}_B(Swap(\rho\otimes\rho)) &= \mathrm{Tr}_B\left(\sum_{i,j}(|i\rangle\langle j|\otimes|j\rangle\langle i|)(\rho\otimes\rho)\right)\\ &= \sum_{i,j}\mathrm{Tr}(|j\rangle\langle i|\rho)|i\rangle\langle j|\rho\\ &= \rho^2.\end{aligned}$$

The same holds for $\mathrm{Tr}_B((\rho\otimes\rho)Swap)$ so we obtain output

$$\frac{1}{2}\rho + \frac{1}{4}(\mathrm{Tr}_B(Swap(\rho\otimes\rho) + (\rho\otimes\rho)Swap)) = \frac{1}{2}(\rho+\rho^2).$$

This state is not yet normalized, but note that for $\rho = a|1\rangle\langle 1| + (1-a)\frac{1}{2}I$, this expression gives the final state $a|1\rangle\langle 1| + \frac{1}{2}((1-a)^2 + (1-a))\cdot\frac{1}{2}I$, which is more pure than our original state as long as $a > 0.5$.

Thus, we can ask the following question: given n copies of a mixed state $\rho = a|1\rangle\langle 1| + (1-a)\frac{1}{2}I$, what is the greatest expected fidelity we can achieve through a process of repeatedly measuring Swap operators on pairs of qubits?

In the $n = 2$ case, our expected fidelity is

$$P(+1)F(+1) + P(-1)F(-1)$$

for $F(a)$ the fidelity of an outcome a with $|1\rangle\langle 1|$ and $P(a)$ the probability of projecting onto the eigenspace corresponding to a.

The probability of projecting onto the positive eigenspace is the trace of the state above, which we may compute to be $\frac{1}{2}(1+\mathrm{Tr}(\rho^2))$. By a simple calculation, the fidelity of $\rho = a|1\rangle\langle 1| + (1-a)\frac{1}{2}I$ with $|1\rangle\langle 1|$ is $\frac{1}{2}(1+a)$, and so we may substitute in to get

$$\frac{1}{2}(1+\mathrm{Tr}(\rho^2))\cdot F(+1) + \frac{1}{2}(1-\mathrm{Tr}(\rho^2))\cdot\frac{1}{2}.$$

With a little bit more work, we can obtain the expression $\frac{1}{2}(1+\frac{4a}{3+a^2})$ for $F(+1)$, and plugging back in for expected fidelity, we obtain

$$\frac{1}{2}\left(1+\frac{1}{2}(1+a^2)\right)\cdot\frac{1}{2}\left(1+\frac{4a}{3+a^2}\right) + \frac{1}{2}\left(1-\frac{1}{2}(1+a^2)\right)\cdot\frac{1}{2} = \frac{1}{2}(1+a),$$

which is our original fidelity. Thus, in the $n = 2$ case, this process actually does not increase the average fidelity.

However, in the $n = 3$ case, we can obtain real gains. As before, measure the Swap operator on the first two mixed states, and if this projects onto the positive eigenspace, trace out over one qubit to get a more purified qubit, and if it projects onto the negative eigenspace, take the third mixed state as our estimate.

Reusing some calculations from above, we obtain the expected fidelity

$$\begin{aligned}\frac{1}{8}(3 + a^2 + 4a + (1 - a^2)(1 + a)) &= \frac{1}{8}(a^2 + 4a + 3 + 1 + a - a^2 - a^3) \\ &= \frac{1}{8}(4 + 5a - a^3),\end{aligned}$$

which is better than doing nothing, for all values of a, as expected.

It is not directly obvious how to compare the results of this simplified protocol to the [2] procedure because this procedure is necessarily influenced by the outcome we wish to achieve. For example, our strategy for maximizing expected fidelity will depend on how many purified states we wish to obtain, whereas the [2] protocol will randomly output some number of purified states.

However, we can do some basic calculations. In the case of $n = 2$ qubits, note that this purification procedure is exactly equivalent to the optimal procedure. In the case of $n = 4$ qubits, we begin by measuring Swap on the first pair of qubits. If it is unsuccessful, we are back in the $n = 2$ case, and have fidelity $\frac{1}{2}(1 + a)$. If it is successful, we may either measure Swap on the new qubit along with an unpurified qubit or on two unpurified qubits. Notice that measuring Swap on the two unpurified qubits cannot give us anything more pure, so this cannot be optimal. Measuring Swap on the new qubit along with an unpurified qubit gives expected value

$$P(1 \text{ fails}) \cdot \frac{1}{2}(1 + a) + P(1 \text{ works}, 2 \text{ fails}) \cdot \frac{1}{2}(1 + a) + P(\text{both work}) \cdot \frac{ab + 2a + 2b + 3}{2(3 + ab)}$$

where b is the value obtained from the first purification. In comparison, running the optimal purification procedure gives results as described in [2]. One could compare these two procedures by computing expected fidelity of one purified qubit and graphing the results as a function of the randomness of the initial state ρ, and we have done this in the below graph, where green shows expected fidelity from our procedure and red shows the expected fidelity from the optimal procedure.

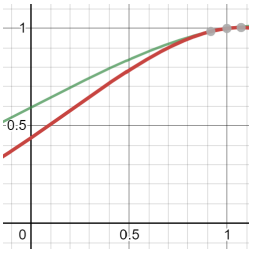

Figure 3: Comparison of Expected Fidelity of Purified Qubits Between Two Protocols

However, one must beware that this is not exactly a reasonable comparison, since in reality the optimal procedure will output two purified qubits with a nontrivial probability, and this is more valuable than can be captured in a simple expected value. A practical analysis of the efficiency of each procedure, however, would require actual experiments on a quantum computer, which we have not done here. However, the graph suggests that the expected fidelities of the procedures may not differ so much as to warrant using the optimal procedure over the simplified one, for at least cases in which we have a small number of qubits and have a clear objective for number of purified qubits.

B Basic Background

This section is by no means exhaustive, but hopefully contains enough information to roughly grasp the flow of the rest of this paper.

B.1 Quantum Notions

In contrast to AntMan's fantastical depictions of the quantum realm, mathematicians understand it as a complex Hilbert space, with the state of a system represented by a vector. Throughout this paper, we will be working with systems of n qudits, given by vector $|\psi\rangle \in (\mathbb{C}^d)^{\otimes n}$. The time evolution of such a system is governed by the Schrodinger equation

$$-ih\frac{d}{dt}|\psi(t)\rangle = H|\psi(0)\rangle$$

where H is the Hamiltonian, a Hermitian matrix specific to the system. For a time independent Hamiltonian, one can easily solve this equation by separation of variables and see as a consequence that the state vector evolves according to unitary time evolutions.

Generally, we will pick basis $|1\rangle, \ldots, |d\rangle$ for our qudit $|\psi\rangle$, which will give in some sense a standard basis. Frequently, we will default to the case $d = 2$, the case of the qubit, in which case we will deviate and let our basis be $|0\rangle$, $|1\rangle$ with the convention that

$$|0\rangle = \begin{bmatrix}1\\0\end{bmatrix}, |1\rangle = \begin{bmatrix}0\\1\end{bmatrix}.$$

A state vector, then, is simply obtained by tensoring together these individual systems to get some state of the form $|i_1 \ldots i_n\rangle$.

B.2 Representation Theory

The information in this section follows the presentation from [3] and [7].

A *representation* (ρ, V) of group G is a homomorphism $\rho : G \to GL(V)$ for some vector space V. Equivalently, it may also be considered a specification of the action of some group on V, or a left module V with action by $\mathbb{C}[G]$ the group algebra. This is a useful notion because it allows us to apply the tools of linear algebra to understand properties of groups.

Examples: Any group G always has the trivial representation $\rho(g) = 1$ for all $g \in G$. Any matrix group, such as $GL(V)$, has the natural representation on V given by typical matrix multiplication.

Two representations $(\rho_1, V_1), (\rho_2, V_2)$ are *isomorphic* if there exists an intertwiner, an invertible linear map $P \in \mathrm{Hom}(V_1, V_2)$ such that $P\rho_1(g) = \rho_2(g)P$ for all $g \in G$. This can be thought of as finding a suitable change of basis matrix.

A *subrepresentation* of (ρ, V) is a subspace $W \subset V$ such that $\rho(g)(w) \in W$ for all $g \in G$, $w \in W$. Correspondingly, an *irreducible* representation (abbreviated *irrep*) is one with no proper subrepresentations. Optimistically, one might hope that every representation can be decomposed into a direct sum of irreducible representations, and it turns out in the case of finite groups and compact Lie groups, this is exactly the case. There are a few important results that will be relevant in this discussion.

Decomposition of Representations: For any complex representation (π, V) or a finite group or compact Lie group with subrepresentation $(\pi \restriction_W, W)$, there exists another subrepresentation $(\pi \restriction_U, U)$ such that $V = W \oplus U$.

Proof. We follow the presentation of [7] for finite groups, and note that the case of compact Lie groups is analogous, with sums replaced by integrals. The key idea is introducing a positive definite Hermitian inner product H on V which is fixed under the action of G, such that $H(v, w) = H(\pi(g)v, \pi(g)w)$. Notice this is equivalent to having a unitary representation of G on V with respect to some Hermitian inner product. To construct such an inner product, simply take any Hermitian inner product H_0 and let $H(v, w) = \sum_{g \in G} H(gv, gw)$. The existence of U follows because we can find the orthogonal complement of W in V using the Hermitian inner product and it is invariant under the action of G since H is fixed under the action of G by construction. □

Notice this implies that any representation of a finite group or compact Lie group may be decomposed into a direct sum of irreps, which is sometimes referred to as Maschke's theorem in the finite group case.

Schur's Lemma: Given any irreducible complex representations (π, V) and (π', W) and map $\varphi : V \to W$ which satisfies $\varphi\pi(g) = \pi'(g)\varphi$ for all $g \in G$, we must either have that φ is the zero map or V is isomorphic to W and $\varphi = \lambda I$ for some $\lambda \in \mathbb{C}$.

Proof. Note that $\ker \varphi$ and $\operatorname{im} \varphi$ must be subrepresentations of (π, V) and (π', W), respectively. Since these are irreducible representations by assumption, we see that either φ is the zero map or φ gives an isomorphism between V and W. Since V and W are complex vector spaces, φ must have an eigenvalue λ, and since the map $\varphi' = \varphi - \lambda I$ also satisfies $\varphi'\pi(g) = \pi'(g)\varphi'$ and must have nontrivial kernel, we obtain $\varphi' = 0$, or $\varphi = \lambda I$, as desired. □

Notice that combining these two theorems gives us a sort of existence and uniqueness theorem on complex representations of a finite group or compact Lie group. In particular, for any complex representation, we obtain that there exists a unique direct sum decomposition into irreps, up to reordering.

We also get as a direct corollary that the irreducible representations of finite abelian groups are one-dimensional constant maps, which follows because any complex representation $\pi(g)$ of a finite abelian group must be an intertwiner of itself by definition.