The Quantum Schur Transform: I. Efficient Qudit Circuits

D. Bacon
Dept. of Computer Science and Engineering, Univ. of Washington, Seattle, WA, USA
Santa Fe Institute, Santa Fe, NM 87501 USA
Institute for Quantum Information, California Institute of Technology, Pasadena, CA, USA and
Dept. of Physics, California Institute of Technology, Pasadena, CA, USA

I. Chuang
Center for Bits and Atoms, Massachusetts Institute of Technology, Cambridge, MA, USA
Dept. of Electrical Engineering and Computer Science,
Massachusetts Institute of Technology, Cambridge, MA, USA and
Dept. of Physics, Massachusetts Institute of Technology, Cambridge, MA, USA

A. Harrow
Center for Bits and Atoms, Massachusetts Institute of Technology, Cambridge, MA, USA
Dept. of Physics, Massachusetts Institute of Technology, Cambridge, MA, USA and
Dept. of Computer Science, Univ. of Bristol, Bristol, U.K.
(Dated: February 1, 2008)

We present an efficient family of quantum circuits for a fundamental primitive in quantum information theory, the Schur transform. The Schur transform on n d dimensional quantum systems is a transform between a standard computational basis to a labelling related to the representation theory of the symmetric and unitary groups. If we desire to implement the Schur transform to an accuracy of ϵ, then our circuit construction uses a number of gates which is polynomial in n, d and $\log(\epsilon^{-1})$. The important insights we use to perform this construction are the selection of the appropriate subgroup adapted basis and the Wigner-Eckart theorem. Our efficient circuit construction renders numerous protocols in quantum information theory computationally tractable and is an important new efficient quantum circuit family which goes significantly beyond the standard paradigm of the quantum Fourier transform.

I. INTRODUCTION

The last decade has seen the development and expansion of a robust theory of quantum information[1–6]. The basic goal of this new work has been the identification and quantification of different information resources in situations where the laws of quantum theory are applied to the physical carriers of information. Quantum information theory has made great progress in understanding the optimal rates of the manipulation and transmission of quantum information. Despite this success, however, much of the work in quantum information theory may not be of practical value. This is because most of the work in quantum information theory has focused on protocols which allow for unbounded quantum computational resources. Thus while the transforms in the quantum information protocols are well defined, whether these transforms can be implemented with quantum circuits whose size scales efficiently with the size of the quantum information problem is often left unaddressed. An analogous situation arises classically, for example, in the theory of classical error correcting codes. On the one hand, we would like the classical error correcting code to attain some characteristic efficiency for communicating over a noisy channel. On the other hand, we would also like to design codes whose encoding and decoding does not significantly lag our communication. In order to be of practical value a classical error correcting code must use computational resources which scale at a reasonable rate. While the goal of performing classical coding tasks in polynomial or even linear time has long been studied, quantum information theory results have typically ignored questions of efficiency. For example, random quantum coding results (such as [7–10]) require an exponential number of bits to describe, and like classical random coding techniques, do not yield efficient algorithms. There are a few important exceptions. Some quantum coding tasks, such as Schumacher compression[3, 11], are essentially equivalent to classical circuits, and as such can be performed efficiently on a quantum computer by carefully modifying an efficient classical algorithm to run reversibly and to deal properly with ancilla systems[12]. Another example, which illustrates some of the challenges involved, is Ref. [13]'s efficient implementation of entanglement concentration[5]. Quantum key distribution[14] not only runs efficiently, but can be implemented with entirely, or almost entirely, single-qubit operations and classical computation. Fault tolerant quantum computing[15] usually seeks to perform error correction with as few gates as possible, although with

teleportation-based techniques[16, 17] computational efficiency may not be quite as critical to the threshold rate. Finally, some randomized quantum code constructions have been given efficient constructions using classical derandomization techniques in [18]. In this paper we present an efficient family of quantum circuits for a transform used ubiquitously[19–30] in quantum information protocols, the Schur transform. Our efficient construction of the Schur transform adds to the above list a powerful new tool for finding algorithms that implement quantum communication tasks.

The Schur transform is a unitary transform on n d-dimensional quantum systems (n qudits). The basis change corresponding to the Schur transform goes from a standard computational basis on the n qudits to a labelling related to the representation theory of the symmetric and unitary groups; much like the Fourier transform, it thus transforms from a local to a more global, collective basis, which captures symmetries of the system. In this article we show how to efficiently implement the Schur transform as a quantum circuit. The size of the circuit we construct is polynomial in the number of qudits, n, the dimension of the individual quantum systems, d, and the log of accuracy to which we implement the transform, $\log(\epsilon^{-1})$. Our efficient quantum circuit for the Schur transform makes possible efficient quantum circuits for numerous quantum information tasks: optimal spectrum estimation[19, 20], universal entanglement concentration[22], universal compression with optimal overflow exponent[23, 24], encoding into decoherence-free subsystems[26–29], optimal hypothesis testing[25], and quantum and classical communication without shared reference frames[30]. The central role of the Schur transform in all of these protocols is due to the fact that the symmetries of independent and identically distributed quantum states are naturally treated by the representation theory of the symmetric and unitary groups. Thus in addition to making practical these quantum information protocols, the Schur transform is an interesting new unitary transformation with an interpretation relating to these symmetries.

There are many difficulties in designing an efficient quantum circuit for the Schur transform which we overcome in this paper. The first difficulty is in efficiently representing the basis used in the Schur transform, the Schur basis. A second difficulty comes in the actual circuit construction. In particular we would like to construct the Schur transform from a series of Clebsch-Gordan transforms. However, it is not at all obvious how to efficiently implement these Clebsch-Gordan transforms, nor is obvious that such a cascade can perform the complete Schur transform.

Our resolution to these problems begins by our selection of certain subgroup-adapted bases for the Schur basis. In particular we use the Gel'fand-Zetlin basis[31] and the Young-Yamanouchi basis (sometimes called Young's orthogonal basis)[32]. We note that subgroup-adapted bases are also used in constructing efficient quantum circuits for Fourier transforms over nonabelian finite groups[33]. However, we should emphasize that the Schur transform is not a Fourier transform over a finite group, although connections between such transforms and the Schur transform exist, and are discussed in part II of this paper. By choosing the Gel'fand-Zetlin basis and the Young-Yamanouchi basis, we are able to show that the Schur transform can be constructed from a cascade of Clebsch-Gordan transforms. Further, the use of the Gel'fand-Zetlin basis, combined with the Wigner-Eckart theorem, allows us to efficiently implement the Clebsch-Gordan transform. In particular the Wigner-Eckart theorem allows us to recursively express the d dimensional Clebsch-Gordan transform in terms of the $d-1$ dimensional Clebsch-Gordan transform and small, efficiently implementable, unitary transforms. This produces an efficient recursive construction of the Clebsch-Gordan transform. Without the recursive structure we exploit, a naive circuit construction would seem to require $n^{O(d^2)}$ gates. Our recursive exploitation of the Wigner-Eckart theorem allows us to implement the Clebsch-Gordan transform to accuracy ϵ using $\mathrm{poly}(d, \log n, \log 1/\epsilon)$ gates. The total size of our circuit construction for the Schur transform is $n\,\mathrm{poly}(d, \log n, \log 1/\epsilon)$.

The outline of the paper is as follows. In Section II we introduce the Schur transform, along with basic concepts from representation theory, and review the numerous applications of the Schur transform in quantum information theory. In Section III we introduce the basis labelling scheme used in the Schur transformation using the concept of a subgroup-adapted basis. Once we have a concrete Schur basis defined, we describe the Clebsch-Gordan transform and explain how to use it to give an efficient circuit for the Schur transform in Sec. IV. Finally, we complete the algorithm in Sec. V by constructing an efficient circuit for the Clebsch-Gordan transform.

II. THE SCHUR TRANSFORM AND ITS APPLICATIONS

Consider a system of n d-dimensional quantum systems: n qudits. Fix a standard computational basis $|i\rangle$, $i = 1 \ldots d$ for the state space of each qudit: $\mathbb{C}^d$. A basis for the system $(\mathbb{C}^d)^{\otimes n}$ is then $|i_1\rangle \otimes |i_2\rangle \otimes \cdots \otimes |i_n\rangle = |i_1, i_2, \ldots, i_n\rangle$ where $i_k = 1 \ldots d$. The Schur transform is a unitary transform on the standard basis

$|i_1, i_2, \ldots, i_n\rangle$. After the Schur transform, the standard computational basis is relabeled as $|\lambda\rangle|p\rangle|q\rangle$ (symbols which we later define). In this section we review the basic representation theory necessary to understand the Schur basis and also review the applications of this transformation to different protocols in quantum information theory.

A. Representation theory background

The Schur transform is related to the representations of two groups on $(\mathbb{C}^d)^{\otimes n}$, a representation of the symmetric group and a representation of the unitary group. We first recall the basics of representation theory before introducing these representations. For a more detailed description of representation theory, the reader should consult [34] for general facts about group theory and representation theory or [35] for representations of Lie groups. See also [36] for a more introductory and informal approach to Lie groups and their representations.

Representations: For a complex vector space V, define $\mathrm{End}(V)$ to be set of linear maps from V to itself (endomorphisms). A representation of a group $\mathcal{G}$ is a vector space V together with a homomorphism from $\mathcal{G}$ to $\mathrm{End}(V)$, i.e. a function $\mathbf{R} : \mathcal{G} \to \mathrm{End}(V)$ such that $\mathbf{R}(g_1)\mathbf{R}(g_2) = \mathbf{R}(g_1 g_2)$. If $\mathbf{R}(g)$ is a unitary operator for all g, then we say $\mathbf{R}$ is a unitary representation. Furthermore, we say a representation $(\mathbf{R}, V)$ is finite dimensional if V is a finite dimensional vector space. In this paper, we will always consider complex finite dimensional, unitary representations and use the generic term 'representation' to refer to complex, finite dimensional, unitary representations. Also, when clear from the context, we will denote a representation $(\mathbf{R}, V)$ simply by the representation space V.

The reason we consider only complex, finite dimensional, unitary representations is so that we can use them in quantum computing. If $d = \dim V$, then a d-dimensional quantum system can hold a unit vector in a representation V. A group element $g \in \mathcal{G}$ corresponds to a unitary rotation $\mathbf{R}(g)$, which can in principle be performed by a quantum computer.

Homomorphisms: For any two vector spaces V_1 and V_2, define $\mathrm{Hom}(V_1, V_2)$ to be the set of linear transformations from V_1 to V_2. If $\mathcal{G}$ acts on V_1 and V_2 with representation matrices $\mathbf{R}_1$ and $\mathbf{R}_2$ respectively, then the canonical action of $\mathcal{G}$ on $\mathrm{Hom}(V_1, V_2)$ is given by the map from M to $\mathbf{R}_2(g) M \mathbf{R}_1(g)^{-1}$ for any $M \in \mathrm{Hom}(V_1, V_2)$. For any representation $(\mathbf{R}, V)$ define $V^{\mathcal{G}}$ to be the space of $\mathcal{G}$-invariant vectors of V: i.e. $V^{\mathcal{G}} := \{|v\rangle \in V : \mathbf{R}(g)|v\rangle = |v\rangle \; \forall g \in \mathcal{G}\}$. Of particular interest is the space $\mathrm{Hom}(V_1, V_2)^{\mathcal{G}}$, which can be thought of as the linear maps from V_1 to V_2 which commute with the action of $\mathcal{G}$. If $\mathrm{Hom}(V_1, V_2)^{\mathcal{G}}$ contains any invertible maps (or equivalently, any unitary maps) then we say that $(\mathbf{R}_1, V_1)$ and $(\mathbf{R}_2, V_2)$ are *equivalent* representations and write

$$V_1 \overset{\mathcal{G}}{\cong} V_2.$$

This means that there exists a unitary change of basis $U : V_1 \to V_2$ such that for any $g \in \mathcal{G}$, $U\mathbf{R}_1(g)U^\dagger = \mathbf{R}_2(g)$.

Dual representations: Recall that the *dual* of a vector space V is the set of linear maps from V to $\mathbb{C}$ and is denoted V^*. Usually if vectors in V are denoted by kets (e.g. $|v\rangle$) then vectors in V^* are denoted by bras (e.g. $\langle v|$). If we fix a basis $\{|v_1\rangle, |v_2\rangle, \ldots\}$ for V then the transpose is a linear map from V to V^* given by $|v_i\rangle \to \langle v_i|$. Now, for a representation $(\mathbf{R}, V)$ we can define the *dual representation* $(\mathbf{R}^*, V^*)$ by $\mathbf{R}^*(g)\langle v^*| := \langle v^*|\mathbf{R}(g^{-1})$. If we think of $\mathbf{R}^*$ as a representation on V (using the transpose map to relate V and V^*), then it is given by $\mathbf{R}^*(g) = (\mathbf{R}(g^{-1}))^T$. When $\mathbf{R}$ is a unitary representation, this is the same as the *conjugate representation* $\mathbf{R}(g)^*$, where here * denotes the entrywise complex conjugate. One can readily verify that the dual and conjugate representations are indeed representations and that $\mathrm{Hom}(V_1, V_2) \overset{\mathcal{G}}{\cong} V_1^* \otimes V_2$.

Irreducible representations: Generically the unitary operators of a representation may be specified (and manipulated on a quantum computer) in an arbitrary orthonormal basis. The added structure of being a representation, however, implies that there are particular bases which are more fundamental to expressing the action of the group. We say a representation $(\mathbf{R}, V)$ is irreducible (and call it an irreducible representaiton, or *irrep*) if the only subspaces of V which are invariant under $\mathbf{R}$ are the empty subspace $\{0\}$ and the entire space V. For finite groups, any finite-dimensional complex representation is reducible; meaning it is decomposable into a direct sum of irreps. For Lie groups, we need additional conditions, such as demanding that the representation $\mathbf{R}(g)$ be *rational*; i.e. its matrix elements are polynomial functions of the matrix elements g_{ij} and $(\det g)^{-1}$. We say a representation of a Lie group is *polynomial* if its matrix elements are polynomial functions only of the g_{ij}.

Isotypic decomposition: Let $\hat{\mathcal{G}}$ be a complete set of inequivalent irreps of $\mathcal{G}$. Then for any reducible representation $(\mathbf{R}, V)$ there is a basis under which the action of $\mathbf{R}(g)$ can be expressed as

$$\mathbf{R}(g) \cong \bigoplus_{\lambda \in \hat{\mathcal{G}}} \bigoplus_{j=1}^{n_\lambda} \mathbf{r}_\lambda(g) = \bigoplus_{\lambda \in \hat{\mathcal{G}}} \mathbf{r}_\lambda(g) \otimes \mathbf{I}_{n_\lambda} \tag{1}$$

where $\lambda \in \hat{\mathcal{G}}$ labels an irrep $(\mathbf{r}_\lambda, V_\lambda)$ and n_λ is the multiplicity of the irrep λ in the representation V. Here we use $\cong$ to indicate that there exists a unitary change of basis relating the left-hand size to the right-hand side.[1] Under this change of basis we obtain a similar decomposition of the representation space V (known as the *isotypic decomposition*):

$$V \overset{\mathcal{G}}{\cong} \bigoplus_{\lambda \in \hat{\mathcal{G}}} V_\lambda \otimes \mathbb{C}^{n_\lambda}. \tag{2}$$

Thus while generically we may be given a representation in some arbitrary basis, the structure of being a representation picks out a particular basis under which the action of the representation is not just block diagonal but also maximally block diagonal: a direct sum of irreps.

Moreover, the multiplicity space $\mathbb{C}^{n_\lambda}$ in Eq. (2) has the structure of $\mathrm{Hom}(V_\lambda, V)^{\mathcal{G}}$. This means that for any representation $(\mathbf{R}, V)$, Eq. (2) can be restated as

$$V \overset{\mathcal{G}}{\cong} \bigoplus_{\lambda \in \hat{\mathcal{G}}} V_\lambda \otimes \mathrm{Hom}(V_\lambda, V)^{\mathcal{G}}. \tag{3}$$

Since $\mathcal{G}$ acts trivially on $\mathrm{Hom}(V_\lambda, V)^{\mathcal{G}}$, Eq. (1) remains the same. As with the other results in this chapter, a proof of Eq. (3) can be found in [35], or other standard texts on representation theory.

The value of Eq. (3) is that the unitary mapping from the right-hand side (RHS) to the left-hand side (LHS) has a simple explicit expression: it corresponds to the canonical map $\varphi : A \otimes \mathrm{Hom}(A, B) \to B$ given by $\varphi(a \otimes f) = f(a)$. Of course, this doesn't tell us how to describe $\mathrm{Hom}(V_\lambda, V)^{\mathcal{G}}$, or how to specify an orthonormal basis for the space, but we will later find this form of the decomposition useful.

B. The Schur Transform

We now turn to the two representations relevant to the Schur transform. Recall that the symmetric group $\mathcal{S}_n$ of degree n, is the group of all permutations of n objects. Then we have the following natural representation of the symmetric group on the space $(\mathbb{C}^d)^{\otimes n}$:

$$\mathbf{P}(s)|i_1\rangle \otimes |i_2\rangle \otimes \cdots \otimes |i_n\rangle = |i_{s^{-1}(1)}\rangle \otimes |i_{s^{-1}(2)}\rangle \otimes \cdots \otimes |i_{s^{-1}(n)}\rangle \tag{4}$$

where $s \in \mathcal{S}_n$ is a permutation and $s(i)$ is the label describing the action of s on label i. For example, if we are considering $\mathcal{S}_3$ and the permutation we are considering is the transposition $s = (12)$, then $\mathbf{P}(s)|i_1, i_2, i_3\rangle = |i_2, i_1, i_3\rangle$. $(\mathbf{P}, (\mathbb{C}^d)^{\otimes n})$ is the representation of the symmetric group which will be relevant to the Schur transform. Note that $\mathbf{P}$ obviously depends on n, but also has an implicit dependence on d.

Now we turn to the representation of the unitary group. Let $\mathcal{U}_d$ denote the group of $d \times d$ unitary operators. Then there is a representation of $\mathcal{U}_d$ given by the n-fold product action as

$$\mathbf{Q}(U)|i_1\rangle \otimes |i_2\rangle \otimes \cdots \otimes |i_n\rangle = U|i_1\rangle \otimes U|i_2\rangle \otimes \cdots \otimes U|i_n\rangle \tag{5}$$

for any $U \in \mathcal{U}_d$. More compactly, we could write that $\mathbf{Q}(U) = U^{\otimes n}$. $(\mathbf{Q}, (\mathbb{C}^d)^{\otimes n})$ is the representation of the unitary group which will be relevant to the Schur transform.

---

[1] We only need to use $\overset{\mathcal{G}}{\cong}$ when relating representation spaces. In Eq. (1) and other similar isomorphisms, we instead explicitly specify the dependence of both sides on $g \in \mathcal{G}$.

Since both $\mathbf{P}(s)$ and $\mathbf{Q}(U)$ meet our above criteria for reducibility, they can be decomposed into a direct sum of irreps as in Eq. (1),

$$\begin{aligned}\mathbf{P}(s) &\overset{\mathcal{S}_n}{\cong} \bigoplus_\alpha \mathbf{I}_{n_\alpha} \otimes \mathbf{p}_\alpha(s) \\ \mathbf{Q}(U) &\overset{\mathcal{U}_d}{\cong} \bigoplus_\beta \mathbf{I}_{m_\beta} \otimes \mathbf{q}_\beta(U)\end{aligned} \tag{6}$$

where n_α (m_β) is the multiplicity of the αth (βth) irrep $\mathbf{p}_\alpha(s)$ ($\mathbf{q}_\beta(U)$) in the representation $\mathbf{P}(s)$ ($\mathbf{Q}(U)$). At this point there is not necessarily any relation between the two different unitary transforms implementing the isomorphisms in Eq. (6). However, further structure in this decomposition follows from the fact that $\mathbf{P}(s)$ commutes with $\mathbf{Q}(U)$: $\mathbf{P}(s)\mathbf{Q}(U) = \mathbf{Q}(U)\mathbf{P}(s)$. This implies, via Schur's Lemma, that the action of the irreps of $\mathbf{P}(s)$ must act on the multiplicity labels of the irreps $\mathbf{Q}(U)$ and vice versa. Thus, the simultaneous action of $\mathbf{P}$ and $\mathbf{Q}$ on $(\mathbb{C}^d)^{\otimes n}$ decomposes as

$$\mathbf{Q}(U)\mathbf{P}(s) \overset{\mathcal{U}_d\times\mathcal{S}_n}{\cong} \bigoplus_\alpha \bigoplus_\beta \mathbf{I}_{m_{\alpha,\beta}} \otimes \mathbf{q}_\beta(U) \otimes \mathbf{p}_\alpha(s) \tag{7}$$

where $m_{\alpha,\beta}$ can be thought of as the multiplicity of the irrep $\mathbf{p}_\alpha(s) \otimes \mathbf{q}_\beta(U)$ of the group $\mathcal{U}_d \times \mathcal{S}_n$.

Not only do $\mathbf{P}$ and $\mathbf{Q}$ commute, but the algebras they generate (i.e. $\mathcal{A} := \mathbf{P}(\mathbb{C}[\mathcal{S}_n]) = \mathrm{Span}\{\mathbf{P}(s) : s \in \mathcal{S}_n\}$ and $\mathcal{B} := \mathbf{Q}(\mathbb{C}[\mathcal{U}_d]) = \mathrm{Span}\{\mathbf{Q}(U) : U \in \mathcal{U}_d\}$) *centralize* each other[35], meaning that $\mathcal{B}$ is the set of operators in $\mathrm{End}((\mathbb{C}^d)^{\otimes n})$ commuting with $\mathcal{A}$ and vice versa, $\mathcal{A}$ is the set of operators in $\mathrm{End}((\mathbb{C}^d)^{\otimes n})$ commuting with $\mathcal{B}$. This means that the multiplicities $m_{\alpha,\beta}$ are either zero or one, and that each α and β appears at most once. Thus Eq. (7) can be further simplified to

$$\mathbf{Q}(U)\mathbf{P}(s) \overset{\mathcal{U}_d\times\mathcal{S}_n}{\cong} \bigoplus_\lambda \mathbf{q}_\lambda(U) \otimes \mathbf{p}_\lambda(s) \tag{8}$$

where λ runs over some unspecified set.

Finally, Schur duality (or Schur-Weyl duality)[35, 37] provides a simple characterization of the range of λ in Eq. (8) and shows how the decompositions are related for different values of n and d. To define Schur duality, we will first need to specify the irreps of $\mathcal{S}_n$ and $\mathcal{U}_d$.

Let $\mathcal{I}_{d,n} = \{\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_d) | \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d \geq 0 \text{ and } \sum_{i=1}^d \lambda_i = n\}$ denote partitions of n into $\leq d$ parts. We consider two partitions $(\lambda_1, \ldots, \lambda_d)$ and $(\lambda_1, \ldots, \lambda_d, 0, \ldots, 0)$ equivalent if they differ only by trailing zeroes; according to this principle, $\mathcal{I}_n := \mathcal{I}_{n,n}$ contains all the partitions of n. Partitions label irreps of $\mathcal{S}_n$ and $\mathcal{U}_d$ as follows: if we let d vary, then $\mathcal{I}_{d,n}$ labels irreps of $\mathcal{S}_n$, and if we let n vary, then $\mathcal{I}_{d,n}$ labels polynomial irreps of $\mathcal{U}_d$. Call these $(\mathbf{p}_\lambda, \mathcal{P}_\lambda)$ and $(\mathbf{q}_\lambda^d, \mathcal{Q}_\lambda^d)$ respectively, for $\lambda \in \mathcal{I}_{d,n}$. We need the superscript d because the same partition λ can label different irreps for different $\mathcal{U}_d$; on the other hand the $\mathcal{S}_n$-irrep $\mathcal{P}_\lambda$ is uniquely labeled by λ since $n = \sum_i \lambda_i$.

For the case of n qudits, Schur duality states that there exists a basis (which we label $|\lambda\rangle|q_\lambda\rangle|p_\lambda\rangle_{\mathrm{Sch}}$ and call the *Schur basis*) which simultaneously decomposes the action of $\mathbf{P}(s)$ and $\mathbf{Q}(U)$ into irreps:

$$\begin{aligned}\mathbf{Q}(U)|\lambda\rangle|q_\lambda\rangle|p_\lambda\rangle_{\mathrm{Sch}} &= |\lambda\rangle(\mathbf{q}_\lambda^d(U)|q_\lambda\rangle)|p_\lambda\rangle_{\mathrm{Sch}} \\ \mathbf{P}(s)|\lambda\rangle|q_\lambda\rangle|p_\lambda\rangle_{\mathrm{Sch}} &= |\lambda\rangle|q_\lambda\rangle(\mathbf{p}_\lambda(s)|p_\lambda\rangle)_{\mathrm{Sch}}\end{aligned} \tag{9}$$

and that the common representation space $(\mathbb{C}^d)^{\otimes n}$ decomposes as

$$(\mathbb{C}^d)^{\otimes n} \overset{\mathcal{U}_d\times\mathcal{S}_n}{\cong} \bigoplus_{\lambda\in\mathcal{I}_{d,n}} \mathcal{Q}_\lambda^d \otimes \mathcal{P}_\lambda. \tag{10}$$

The Schur basis can be expressed as superpositions over the standard computational basis states $|i_1, i_2, \ldots, i_n\rangle$ as

$$|\lambda, q_\lambda, p_\lambda\rangle_{\mathrm{Sch}} = \sum_{i_1,i_2,\ldots,i_n} [\mathbf{U}_{\mathrm{Sch}}]_{i_1,i_2,\ldots,i_n}^{\lambda,q_\lambda,p_\lambda} |i_1 i_2 \ldots i_n\rangle, \tag{11}$$

where $\mathbf{U}_{\mathrm{Sch}}$ is the unitary transformation implementing the isomorphism in Eq. (10). Thus, for any $U \in \mathcal{U}_d$ and any $s \in \mathcal{S}_n$,

$$\mathbf{U}_{\mathrm{Sch}}\mathbf{Q}(U)\mathbf{P}(s)\mathbf{U}_{\mathrm{Sch}}^\dagger = \sum_{\lambda\in\mathcal{I}_{d,n}} |\lambda\rangle\langle\lambda| \otimes \mathbf{q}_\lambda^d(U) \otimes \mathbf{p}_\lambda(s). \tag{12}$$

If we now think of $\mathbf{U}_{\mathrm{Sch}}$ as a quantum circuit, it will map the Schur basis state $|\lambda, q_\lambda, p_\lambda\rangle_{\mathrm{Sch}}$ to the computational basis state $|\lambda, q_\lambda, p_\lambda\rangle$ with λ, q_λ, and p_λ expressed as bit strings. The dimensions of the irreps $\mathbf{p}_\lambda$ and $\mathbf{q}^d_\lambda$ vary with λ, so we will need to pad the $|q_\lambda, p_\lambda\rangle$ registers when they are expressed as bit strings. We will label the padded basis as $|\lambda\rangle|q\rangle|p\rangle$, explicitly dropping the λ dependence. Later in the paper we will show how to do this padding efficiently with only a logarithmic spatial overhead. We will refer to the transform from the computational basis $|i_1, i_2, \ldots, i_n\rangle$ to the basis of three strings $|\lambda\rangle|q\rangle|p\rangle$ as the Schur transform. The Schur transform is shown schematically in Fig. 1. Notice that just as the standard computational basis $|i\rangle$ is arbitrary up to a unitary transform, the bases for $\mathcal{Q}^d_\lambda$ and $\mathcal{P}_\lambda$ are also both arbitrary up to a unitary transform, though we will later choose particular bases for $\mathcal{Q}^d_\lambda$ and $\mathcal{P}_\lambda$.

Example of the Schur transform—Let $d = 2$. Then for $n = 2$ there are two valid partitions, $\lambda_1 = 2, \lambda_2 = 0$ and $\lambda_1 = \lambda_2 = 1$. Here the Schur transform corresponds to the change of basis from the standard basis to the singlet and triplet basis: $|\lambda = (1,1), q_\lambda = 0, p_\lambda = 0\rangle_{\mathrm{Sch}} = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle)$, $|\lambda = (2,0), q_\lambda = +1, p_\lambda = 0\rangle_{\mathrm{Sch}} = |00\rangle$, $|\lambda = (2,0), q_\lambda = 0, p_\lambda = 0\rangle_{\mathrm{Sch}} = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle)$, and $|\lambda = (2,0), q_\lambda = -1, p_\lambda = 0\rangle_{\mathrm{Sch}} = |11\rangle$. Abstractly, then, the Schur transform then corresponds to a transformation

$$\mathbf{U}_{\mathrm{Sch}} = \begin{array}{c} |\lambda = (1,1), q_\lambda = 0, p_\lambda = 0\rangle_{\mathrm{Sch}} \\ |\lambda = (2,0), q_\lambda = +1, p_\lambda = 0\rangle_{\mathrm{Sch}} \\ |\lambda = (2,0), q_\lambda = 0, p_\lambda = 0\rangle_{\mathrm{Sch}} \\ |\lambda = (2,0), q_\lambda = -1, p_\lambda = 0\rangle_{\mathrm{Sch}} \end{array} \overbrace{\begin{array}{c} \\ \left[\begin{array}{cccc} 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 \\ 1 & 0 & 0 & 0 \\ 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 0 & 1 \end{array}\right] \end{array}}^{|00\rangle\ |01\rangle\ |10\rangle\ |11\rangle} \tag{13}$$

It is easy to verify that the $\lambda = (1,1)$ subspace transforms as a one dimensional irrep of $\mathcal{U}_2$ and as the alternating sign irrep of $\mathcal{S}_2$ and that the $\lambda = (2,0)$ subspace transforms as a three dimensional irrep of $\mathcal{U}_2$ and as the trivial irrep of $\mathcal{S}_2$. Notice that the labeling scheme for the standard computational basis uses 2 qubits while the labeling scheme for the Schur basis uses more qubits (one such labeling assigns one qubit to $|\lambda\rangle$, none to $|p\rangle$ and two qubits to $|q\rangle$). Thus we see how padding will be necessary to directly implement the Schur transform.

To see a more complicated example of the Schur basis, let $d = 2$ and $n = 3$. There are again two valid partitions, $\lambda = (3,0)$ and $\lambda = (2,1)$. The first of these partitions labels to the trivial irrep of $\mathcal{S}_3$ and a 4 dimensional irrep of $\mathcal{U}_3$. The corresponding Schur basis vectors can be expressed as

$$\begin{aligned} |\lambda = (3,0), q_\lambda = +3/2, p_\lambda = 0\rangle_{\mathrm{Sch}} &= |000\rangle \\ |\lambda = (3,0), q_\lambda = +1/2, p_\lambda = 0\rangle_{\mathrm{Sch}} &= \frac{1}{\sqrt{3}}\left(|001\rangle + |010\rangle + |100\rangle\right) \\ |\lambda = (3,0), q_\lambda = -1/2, p_\lambda = 0\rangle_{\mathrm{Sch}} &= \frac{1}{\sqrt{3}}\left(|011\rangle + |101\rangle + |110\rangle\right) \\ |\lambda = (3,0), q_\lambda = -3/2, p_\lambda = 0\rangle_{\mathrm{Sch}} &= |111\rangle. \end{aligned} \tag{14}$$

The second of these partitions labels a two dimensional irrep of $\mathcal{S}_3$ and a two dimensional irrep of $\mathcal{U}_2$. Its Schur basis states can be expressed as

$$\begin{aligned} |\lambda = (2,1), q_\lambda = +1/2, p_\lambda = 0\rangle_{\mathrm{Sch}} &= \frac{1}{\sqrt{2}}\left(|100\rangle - |010\rangle\right) \\ |\lambda = (2,1), q_\lambda = -1/2, p_\lambda = 0\rangle_{\mathrm{Sch}} &= \frac{1}{\sqrt{2}}\left(|101\rangle - |011\rangle\right) \\ |\lambda = (2,1), q_\lambda = +1/2, p_\lambda = 1\rangle_{\mathrm{Sch}} &= \sqrt{\frac{2}{3}}|001\rangle - \frac{|010\rangle + |100\rangle}{\sqrt{6}} \\ |\lambda = (2,1), q_\lambda = -1/2, p_\lambda = 1\rangle_{\mathrm{Sch}} &= \sqrt{\frac{2}{3}}|110\rangle - \frac{|101\rangle + |011\rangle}{\sqrt{6}}. \end{aligned} \tag{15}$$

We can easily verify that Eqns. (14) and (15) indeed transform under $\mathcal{U}_2$ and $\mathcal{S}_3$ the way we expect; not so easy however is generalizing this basis to any n and d, let alone coming up with a natural circuit relating this basis to the computational basis. However, note that p_λ determines whether the first two qubits are in a singlet or a triplet state. This gives a hint of a recursive structure that we will exploit in Sec. III to describe Schur bases for any choice of n and d, and in Sec. IV to construct an efficient recursive algorithm for the Schur transform.

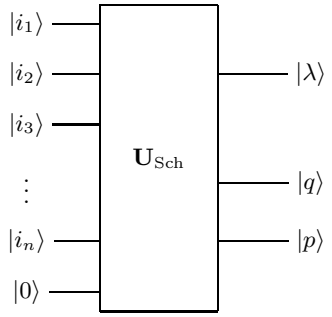
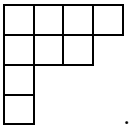


FIG. 1: The Schur transform. Notice how the direct sum over λ in Eq. (10) becomes a tensor product between the $|\lambda\rangle$ register and the $|q\rangle$ and $|p\rangle$ registers. Since the number of qubits needed for $|q\rangle$ and $|p\rangle$ vary with λ, we need slightly more spatial resources, which are here denoted by the ancilla input $|0\rangle$.

C. Constructing $\mathcal{Q}_\lambda^d$ and $\mathcal{P}_\lambda$ using Schur duality

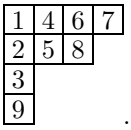
So far we have said little about the form of $\mathcal{Q}_\lambda^d$ and $\mathcal{P}_\lambda$, other than that they are indexed by partitions. It turns out that Schur duality gives a straightforward description of the irreps of $\mathcal{U}_d$ and $\mathcal{S}_n$. We will not use this explicit description to construct the Schur transform, but it is still helpful for understanding the irreps $\mathcal{Q}_\lambda^d$ and $\mathcal{P}_\lambda$. As with the rest of this section, proofs and further details can be found in [35].

We begin by expressing $\lambda \in \mathcal{I}_{d,n}$ as a Young diagram in which there are up to d rows with λ_i boxes in row i. For example, to the partition $(4,3,1,1)$ we associate the diagram



. (16)

Now we define a Young tableau T of shape λ to be a way of filling the n boxes of λ with the integers $1,\ldots,n$, using each number once and so that integers increase from left to right and from top to bottom. For example, one valid Young tableau with shape $(4,3,1,1)$ is



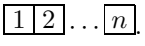
.

For any Young tableau T, define $\mathrm{Row}(T)$ to be set of permutations obtained by permuting the integers within each row of T; similarly define $\mathrm{Col}(T)$ to be the permutations that leave each integer in the same column of T. Now we define the *Young symmetrizer* $\Pi_{\lambda:T}$ to be an operator acting on $(\mathbb{C}^d)^{\otimes n}$ as follows:

$$\Pi_{\lambda:T} := \frac{\dim \mathcal{P}_\lambda}{n!} \left(\sum_{c \in \mathrm{Col}(T)} \mathrm{sgn}(c) \mathbf{P}(c) \right) \left(\sum_{r \in \mathrm{Row}(T)} \mathbf{P}(r) \right). \tag{17}$$

It can be shown that the Young symmetrizer $\Pi_{\lambda:T}$ is a projection operator whose support is a subspace isomorphic to $\mathcal{Q}_\lambda^d$. In particular $\mathbf{U}_{\mathrm{Sch}} \Pi_{\lambda:T} \mathbf{U}_{\mathrm{Sch}}^\dagger = |\lambda\rangle\langle\lambda| \otimes |y(T)\rangle\langle y(T)| \otimes \mathbf{I}_{\mathcal{Q}_\lambda^d}$ for some unit vector $|y(T)\rangle \in \mathcal{P}_\lambda$. Moreover, these vectors $|y(T)\rangle$ form a basis known as Young's natural basis, though the $|y(T)\rangle$ are not orthogonal, so we will usually not work with them in quantum circuits.

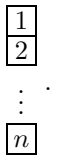
Using Young symmetrizers, we can now explore some more general examples of $\mathcal{Q}_\lambda^d$ and $\mathcal{P}_\lambda$. If $\lambda = (n)$, then the only valid tableau is



.

The corresponding $\mathcal{S}_n$-irrep $\mathcal{P}_{(n)}$ is trivial and the $\mathcal{U}_d$-irrep is given by the action of $\mathbf{Q}$ on the totally symmetric subspace of $(\mathbb{C}^d)^{\otimes n}$, i.e. $\{|v\rangle : \mathbf{P}(s)|v\rangle = |v\rangle \forall s \in \mathcal{S}_n\}$. On the other hand, if $\lambda = (1^n)$, meaning $(1,1,\ldots,1)$

(n times), then the only valid tableau is



The $\mathcal{S}_n$-irrep $\mathcal{P}_{(1^n)}$ is still one-dimensional, but now corresponds to the sign irrep of $\mathcal{S}_n$, mapping s to $\mathrm{sgn}(s)$. The $\mathcal{U}_d$-irrep $\mathcal{Q}^d_{(1^n)}$ is equivalent to the totally antisymmetric subspace of $(\mathbb{C}^d)^{\otimes n}$, i.e. $\{|v\rangle : \mathbf{P}(s)|v\rangle = \mathrm{sgn}(s)|v\rangle \forall s \in \mathcal{S}_n\}$. Note that if $d > n$, then this subspace is zero-dimensional, corresponding to the restriction that irreps of $\mathcal{U}_d$ are indexed only by partitions with $\leq d$ rows.

Other explicit examples of $\mathcal{U}_d$ and $\mathcal{S}_n$ irreps are presented from a particle physics perspective in [38]. We also give more examples in Sec. III B, when we introduce explicit bases for $\mathcal{Q}^d_\lambda$ and $\mathcal{P}_\lambda$.

D. Applications of the Schur Transform

The Schur transform is useful in a surprisingly large number of quantum information protocols. Here we, review these applications, with particular attention to the use of the Schur transform circuit in each protocol. We emphasize again that our construction of the Schur transform simultaneously makes all of these tasks computationally efficient.

1. Spectrum and state estimation

Suppose we are given many copies of an unknown mixed quantum state, $\boldsymbol{\rho}^{\otimes n}$. An important task is to obtain an estimate for the spectrum of $\boldsymbol{\rho}$ from these n copies. An asymptotically good estimate (in the sense of large deviation rate) for the spectrum of $\boldsymbol{\rho}$ can be obtained by applying the Schur transform, measuring λ and taking the spectrum estimate to be $(\lambda_1/n, \ldots, \lambda_d/n)$[19, 21]. Thus an efficient implementation of the Schur transform will efficiently implement the spectrum estimating protocol (note that it is efficient in d, not in $\log(d)$). Estimating $\boldsymbol{\rho}$ reduces to measuring $|\lambda\rangle$ and $|q\rangle$, but optimal estimators have only been explicitly constructed for the case of $d = 2$[20]. Further, optimal quantum hypothesis testing can be obtained by a similar protocol[25].

2. Universal distortion-free entanglement concentration

Let $|\psi\rangle_{AB}$ be a bipartite partially entangled state shared between two parties, A and B. Suppose we are given many copies of $|\psi\rangle_{AB}$ and we want to transform these states into copies of a maximally entangled state using only local operations and classical communication. Further, suppose that we wish this protocol to work when neither A nor B know the state $|\psi\rangle_{AB}$. Such a scheme is called a universal (meaning it works with unknown states $|\psi\rangle_{AB}$) entanglement concentration protocol, as opposed to the original entanglement concentration protocol described by Bennett *et.al.*[5]. Further we also would like the scheme to produce perfect maximally entangled states, i.e. to be distortion free. Universal distortion-free entanglement concentration can be performed[22] by both parties performing Schur transforms on their n halves of $|\psi\rangle_{AB}$, measuring their $|\lambda\rangle$, discarding $|q\rangle$ and retaining $|p\rangle$. The two parties will now share a maximally entangled state of varying dimension depending on what λ was measured. This dimension asymptotes to 2^{nH}, where H is the entropy of one of the parties' reduced mixed states.

3. Universal Compression with Optimal Overflow Exponent

Measuring $|\lambda\rangle$ weakly so as to cause little disturbance, together with appropriate relabeling, comprises a universal compression algorithm with optimal overflow exponent (rate of decrease of the probability that the algorithm will output a state that is much too large)[23, 24].

4. Encoding and decoding into decoherence-free subsystems

Further applications of the Schur transform include encoding into decoherence-free subsystems[26–29]. Decoherence-free subsystems are subspaces of a system's Hilbert space which are immune to decoherence due to a symmetry of the system-environment interaction. For the case where the environment couples identically to all systems, information can be protected from decoherence by encoding into the $|p_\lambda\rangle$ basis. We can use the inverse Schur transform (which, as a circuit can be implemented by reversing the order of all gate elements and replacing them with their inverses) to perform this encoding: simply feed in the appropriate $|\lambda\rangle$ with the state to be encoded into the $|p\rangle$ register and any state into the $|q\rangle$ register into the inverse Schur transform. Decoding can similarly be performed using the Schur transform.

5. Communication without a shared reference frame

An application of the concepts of decoherence-free subsystems comes about when two parties wish to communicate (in either a classical or quantum manner) when the parties do not share a reference frame. The effect of not sharing a reference frame is the same as the effect of collective decoherence (the same random unitary rotation has been applied to each subsystem). Thus encoding information into the $|p\rangle$ register will allow this information to be communicated in spite of the fact that the two parties do not share a reference frame[30]. Just as with decoherence-free subsystems, this encoding and decoding can be done with the Schur transform.

III. SUBGROUP ADAPTED BASES AND THE SCHUR BASIS

In the last section, we defined the Schur transform in a way that left the basis almost completely arbitrary. To construct a quantum circuit for the Schur transform, we will need to explicitly specify the Schur basis. Since we want the Schur basis to be of the form $|\lambda, q, p\rangle$, our task reduces to specifying orthonormal bases for $\mathcal{Q}_\lambda^d$ and $\mathcal{P}_\lambda$. We will call these bases Q_λ^d and P_λ, respectively.

We will choose Q_λ^d and P_λ to both be a type of basis known as a *subgroup-adapted* basis. In Sec. III A we describe the general theory of subgroup-adapted bases, and in Sec. III B, we will describe subgroup-adapted bases for $\mathcal{Q}_\lambda^d$ and $\mathcal{P}_\lambda$. As we will later see, the properties of these bases are intimately related to the structure of the algorithms that work with them. In this section, we will show how the bases can be stored on a quantum computer with a small amount of padding, and in the following sections we will show how the subgroup-adapted bases described here enable efficient implementations of Clebsch-Gordan and Schur duality transforms.

A. Subgroup Adapted Bases

Here we review the basic idea of a subgroup adapted basis. We assume that all groups we talk about are finite or compact Lie groups. Suppose $(\mathbf{r}, V)$ is an irrep of a group $\mathcal{G}$ and $\mathcal{H}$ is a proper subgroup of $\mathcal{G}$. We will construct a basis for V via the representations of $\mathcal{H}$.

Begin by restricting the input of $\mathbf{r}$ to $\mathcal{H}$ to obtain a representation of $\mathcal{H}$, which we call $(\mathbf{r}|_\mathcal{H}, V{\downarrow_\mathcal{H}})$. Note that unlike V, $V{\downarrow_\mathcal{H}}$ may be reducible. In fact, if we let $(\mathbf{r}'_\alpha, V'_\alpha)$ denote the irreps of $\mathcal{H}$, then $V{\downarrow_\mathcal{H}}$ will decompose under the action of $\mathcal{H}$ as

$$V{\downarrow_\mathcal{H}} \overset{\mathcal{H}}{\cong} \bigoplus_{\alpha \in \hat{\mathcal{H}}} \mathbb{C}^{n_\alpha} \otimes V'_\alpha \tag{18}$$

or equivalently, $\mathbf{r}|_\mathcal{H}$ decomposes as

$$\mathbf{r}(h) = \mathbf{r}{\downarrow_\mathcal{H}}(h) \cong \bigoplus_{\alpha \in \hat{\mathcal{H}}} \mathbf{I}_{n_\alpha} \otimes \mathbf{r}'_\alpha(h) \tag{19}$$

where $\hat{\mathcal{H}}$ runs over a complete set of inequivalent irreps of $\mathcal{H}$ and n_α is the *branching multiplicity* of the irrep labeled by α. Note that since $\mathbf{r}$ is a unitary representation, the subspaces corresponding to different

irreps of $\mathcal{H}$ are orthogonal. Thus, the problem of finding an orthonormal basis for V now reduces to the problem of (1) finding an orthonormal basis for each irrep of $\mathcal{H}$, V'_α and (2) finding orthonormal bases for the multiplicity spaces $\mathbb{C}^{n_\alpha}$. The case when all the n_α are either 0 or 1 is known as *multiplicity-free branching*. When this occurs, we only need to determine which irreps occur in the decomposition of V, and find bases for them.

Now consider a group $\mathcal{G}$ along with a tower of subgroups $\mathcal{G} = \mathcal{G}_1 \supset \mathcal{G}_2 \supset \cdots \supset \mathcal{G}_{k-1} \supset \mathcal{G}_k = \{e\}$ where $\{e\}$ is the trivial subgroup consisting of only the identity element. For each $\mathcal{G}_i$, denote its irreps by V^i_α, for $\alpha \in \hat{\mathcal{G}}_i$. Any irrep $V^1_{\alpha_1}$ of $\mathcal{G} = \mathcal{G}_1$ decomposes under restriction to $\mathcal{G}_2$ into $\mathcal{G}_2$-irreps: say that $V^2_{\alpha_2}$ appears n_{α_1,α_2} times. We can then look at these irreps of $\mathcal{G}_2$, consider their restriction to $\mathcal{G}_3$ and decompose them into different irreps of $\mathcal{G}_3$. Carrying on in such a manner down this tower of subgroups will yield a labeling for subspaces corresponding to each of these restrictions. Moreover, if we choose orthonormal bases for the multiplicity spaces, this will induce an orthonormal basis for $\mathcal{G}$. This basis is known as a *subgroup-adapted basis* and basis vectors have the form $|\alpha_2, m_2, \alpha_3, m_3, \ldots, \alpha_n, m_n\rangle$, where $|m_i\rangle$ is a basis vector for the ($n_{\alpha_{i-1},\alpha_i}$-dimensional) multiplicity space of $V^i_{\alpha_i}$ in $V^{i-1}_{\alpha_{i-1}}$.

If the branching for each $\mathcal{G}_{i+1} \subset \mathcal{G}_i$ is multiplicity-free, then we say that the tower of subgroups is *canonical*. In this case, the subgroup adapted basis takes the particularly simple form of $|\alpha_2, \ldots, \alpha_n\rangle$, where each $\alpha_i \in \hat{\mathcal{G}}_i$ and α_{i+1} appears in the decomposition of $V_{\alpha_i}\downarrow_{\mathcal{G}_{i+1}}$. Often we include the original irrep label $\alpha = \alpha_1$ as well: $|\alpha_1, \alpha_2, \ldots, \alpha_k\rangle$. This means that there exists a basis whose vectors are completely determined (up to an arbitrary choice of phase) by which irreps of $G_1, \ldots, G_k$ they transform according to. Notice that a basis for the irrep V_α does not consist of all possible irrep labels α_i, but instead only those which can appear under the restriction which defines the basis.

The simple recursive structure of subgroup adapted bases makes them well-suited to performing explicit computations. Thus, for example, subgroup adapted bases play a major role in efficient quantum circuits for the Fourier transform over many nonabelian groups[33].

B. Explicit orthonormal bases for $\mathcal{Q}^d_\lambda$ and $\mathcal{P}_\lambda$

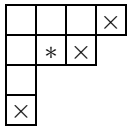
In this section we describe canonical towers of subgroups for $\mathcal{U}_d$ and $\mathcal{S}_n$, which give rise to subgroup-adapted bases for the irreps $\mathcal{Q}^d_\lambda$ and $\mathcal{P}_\lambda$. These bases go by many names: for $\mathcal{U}_d$ (and other Lie groups) the basis is called the Gel'fand-Zetlin basis (following [31]) and we denote it by Q^d_λ, while for $\mathcal{S}_n$ it is called the Young-Yamanouchi basis, or sometimes Young's orthogonal basis (see [32] for a good review of its properties) and is denoted P_λ. The constructions and corresponding branching rules are quite simple, but for proofs we again refer the reader to [35].

The Gel'fand-Zetlin basis for $\mathcal{Q}^d_\lambda$ — For $\mathcal{U}_d$, it turns out that the chain of subgroups $\{1\} = \mathcal{U}_0 \subset \mathcal{U}_1 \subset \ldots \subset \mathcal{U}_{d-1} \subset \mathcal{U}_d$ is a canonical tower. For $c < d$, the subgroup $\mathcal{U}_c$ is embedded in $\mathcal{U}_d$ by $\mathcal{U}_c := \{u \in \mathcal{U}_d : u|i\rangle = |i\rangle \text{ for } i = c+1, \ldots, d\}$. In other words, it corresponds to matrices of the form

$$U \oplus I_{d-c} := \left(\begin{array}{c|c} u & 0 \\ \hline 0 & I_{d-c} \end{array}\right), \tag{20}$$

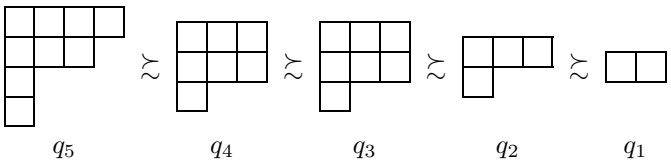
where u is a $c \times c$ unitary matrix.

Since the branching from $\mathcal{U}_d$ to $\mathcal{U}_{d-1}$ is multiplicity-free, we obtain a subgroup-adapted basis Q^d_λ, which is known as the *Gel'fand-Zetlin* (GZ) basis. Our only free choice in a GZ basis is the initial choice of basis $|1\rangle, \ldots, |d\rangle$ for $\mathbb{C}^d$ which determines the canonical tower of subgroups $\mathcal{U}_1 \subset \ldots \subset \mathcal{U}_d$. Once we have chosen this basis, specifying Q^d_λ reduces to knowing which irreps $\mathcal{Q}^{d-1}_\mu$ appear in the decomposition of $\mathcal{Q}^d_\lambda\downarrow_{\mathcal{U}_{d-1}}$. Recall that the irreps of $\mathcal{U}_d$ are labeled by elements of $\mathcal{I}_{d,n}$ with n arbitrary. This set can be denoted by $\mathbb{Z}^d_{++} := \cup_n \mathcal{I}_{d,n} = \{\lambda \in \mathbb{Z}^d : \lambda_1 \geq \ldots \geq \lambda_d \geq 0\}$. For $\mu \in \mathbb{Z}^{d-1}_{++}, \lambda \in \mathbb{Z}^d_{++}$, we say that μ *interlaces* λ and write $\mu \precsim \lambda$ whenever $\lambda_1 \geq \mu_1 \geq \lambda_2 \ldots \geq \lambda_{d-1} \geq \mu_{d-1} \geq \lambda_d$. In terms of Young diagrams, this means that μ is a valid partition (i.e. a nonnegative, nonincreasing sequence) obtained from removing zero or one boxes from each column of λ. For example, if $\lambda = (4, 3, 1, 1)$ (as in Eq. (16)), then $\mu \precsim \lambda$ can be obtained by removing any subset of the marked boxes below, although if the box marked $*$ on the second line is removed, then the other marked box on that line must also be removed.



(21)

Thus a basis vector in Q^d_λ corresponds to a sequence of partitions $q = (q_d = \lambda, \ldots, q_1)$ such that $q_1 \precsim q_2 \precsim \ldots \precsim q_d$ and $q_j \in \mathbb{Z}^j_{++}$ for $j = 1, \ldots, d$. Again using $\lambda = (4,3,1,1)$ as an example, and choosing $d = 5$ (any $d \geq 4$ is possible), we might have the sequence



(22)

Observe that it is possible in some steps not to remove any boxes, as long as q_j has no more than j rows.

In order to work with the Gel'fand-Zetlin basis vectors on a quantum computer, we will need an efficient method to write them down. Typically, we think of d as constant and express our resource use in terms of n. Then an element of $\mathcal{I}_{d,n}$ can be expressed with $d \log(n+1)$ bits, since it consists of d integers between 0 and n. (This is a crude upper bound on $|\mathcal{I}_{d,n}| = \binom{n+d-1}{d-1}$, but for constant d it is good enough for our purposes.) A Gel'fand-Zetlin basis vector then requires no more than $d^2 \log(n+1)$ bits, since it can be expressed as d partitions of integers no greater than n into $\leq d$ parts. Here, we assume that all partitions have arisen from a decomposition of $(\mathbb{C}^d)^{\otimes n}$, so that no Young diagram has more than n boxes. Unless otherwise specified, our algorithms will use this encoding of the GZ basis vectors.

It is also possible to express GZ basis vectors in a more visually appealing way by writing numbers in the boxes of a Young diagram. If $q_1 \precsim \ldots \precsim q_d$ is a chain of partitions, then we write the number j in each box contained in q_j but not q_{j-1} (with $q_0 = (0)$). For example, the sequence in Eq. (22) would be denoted

1	1	2	5
2	3	3	
3			
5			

. (23)

Equivalently, any method of filling a Young diagram with numbers from $1, \ldots, d$ corresponds to a valid chain of irreps as long as the numbers are nondecreasing from left to right and are strictly increasing from top to bottom. The resulting diagram is known as a *semi-standard Young tableau* and gives another way of encoding a GZ basis vector; this time using $n \log d$ bits. (It turns out the actual dimension of $\mathcal{Q}^d_\lambda$ is $\left[\prod_{1 \leq i < j \leq d} (\lambda_i - \lambda_j + j - i)\right] / \left[\prod_{m=1}^d m!\right]$, and later in this section we will give an algorithm for efficiently encoding a GZ basis vector in the optimal $\lceil \log \dim \mathcal{Q}^d_\lambda \rceil$ qubits. However, this is not necessary for most applications.)

Example: irreps of $\mathcal{U}_2$— To ground the above discussion in an example more familiar to physicists, we show how the GZ basis for $\mathcal{U}_2$ irreps corresponds to states of definite angular momentum along one axis. An irrep of $\mathcal{U}_2$ is labeled by two integers (λ_1, λ_2) such that $\lambda_1 + \lambda_2 = n$ and $\lambda_1 \geq \lambda_2 \geq 0$. A GZ basis vector for $\mathcal{Q}^2_\lambda$ has $\lambda_2 + m$ 1's in the first row, followed by $\lambda_1 - (\lambda_2 + m)$ 2's in the first row and λ_2 2's in the second row, where m ranges from 0 to $\lambda_1 - \lambda_2$. This arrangement is necessary to satisfy the constraint that numbers are strictly increasing from top to bottom and are nondecreasing from left to right. Since the GZ basis vectors are completely specified by m, we can label the vector $|(\lambda_1, \lambda_2); (\lambda_2 + m)\rangle \in Q^2_\lambda$ simply by $|m\rangle$. For example, $\lambda = (9,4)$ and $m = 2$ would look like

1	1	1	1	1	1	2	2	2	
2	2	2	2						

. (24)

Now observe that $\dim \mathcal{Q}^2_\lambda = \lambda_1 - \lambda_2 + 1$, a fact which is consistent with having angular momentum $J = (\lambda_1 - \lambda_2)/2$. We claim that m corresponds to the Z component of angular momentum (specifically, the Z component of angular momentum is $m - J = m - (\lambda_1 - \lambda_2)/2$). To see this, first note that $\mathcal{U}_1$ acts on a GZ basis vector $|m\rangle$ according to the representation $x \to x^{\lambda_2 + m}$, for $x \in \mathcal{U}_1$; equivalently $\mathbf{q}^2_\lambda \left(\left(\begin{smallmatrix} x & 0 \\ 0 & 1 \end{smallmatrix} \right) \right) |m\rangle = x^{\lambda_2 + m} |m\rangle$. Since $\mathbf{q}^2_\lambda(yI_2)|m\rangle = y^n |m\rangle = y^{\lambda_1 + \lambda_2} |m\rangle$, we can find the action of $e^{i\theta\sigma_z} = \left(\begin{smallmatrix} e^{2i\theta} & 0 \\ 0 & 1 \end{smallmatrix} \right) \left(\begin{smallmatrix} e^{-i\theta} & 0 \\ 0 & e^{-i\theta} \end{smallmatrix} \right)$ on $|m\rangle$. We do this by combining the above arguments to find that $\mathbf{q}^2_\lambda(e^{i\theta\sigma_z})|m\rangle = e^{2i\theta(\lambda_2 + m)} e^{-i\theta(\lambda_1 + \lambda_2)} |m\rangle = e^{2i\theta(m - J)} |m\rangle$. Thus we obtain the desired action of a Z rotation on a particle with total angular momentum J and Z-component of angular momentum m.

Example: The defining irrep of $\mathcal{U}_d$— The simplest nontrivial irrep of $\mathcal{U}_d$ is its action on $\mathbb{C}^d$. This corresponds to the partition (1), so we say that $(\mathbf{q}^d_{(1)}, \mathcal{Q}^d_{(1)})$ is the *defining irrep* of $\mathcal{U}_d$ with $\mathcal{Q}^d_{(1)} = \mathbb{C}^d$

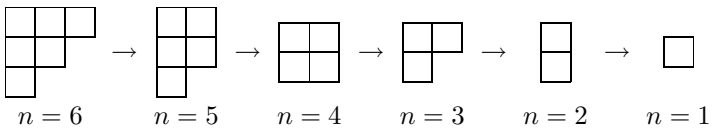
and $\mathbf{q}^d_{(1)}(U) = U$. Let $|1\rangle, \ldots, |d\rangle$ be an orthonormal basis for $\mathbb{C}^d$ corresponding to the canonical tower of subgroups $\mathcal{U}_1 \subset \cdots \subset \mathcal{U}_d$. It turns out that this is already a GZ basis. To see this, note that $\mathcal{Q}^d_{(1)}{\downarrow_{\mathcal{U}_{d-1}}} \overset{\mathcal{U}_{d-1}}{\cong} \mathcal{Q}^{d-1}_{(0)} \oplus \mathcal{Q}^{d-1}_{(1)}$. This is because $|d\rangle$ generates $\mathcal{Q}^{d-1}_{(0)}$, a trivial irrep of $\mathcal{U}_{d-1}$; and $|1\rangle, \ldots, |d-1\rangle$ generate $\mathcal{Q}^{d-1}_{(1)}$, a defining irrep of $\mathcal{U}_{d-1}$. Another way to say this is that $|j\rangle$ is acted on according to the trivial irrep of $\mathcal{U}_1, \ldots, \mathcal{U}_{j-1}$ and according to the defining irrep of $\mathcal{U}_j, \ldots, \mathcal{U}_d$. Thus $|j\rangle$ corresponds to the chain of partitions $\{(0)^{j-1}, (1)^{d-j+1}\}$. We will return to this example many times in the rest of the paper.

The Young-Yamanouchi basis for $\mathcal{P}_\lambda$ — The situation for $\mathcal{S}_n$ is quite similar. Our chain of subgroups is $\{e\} = \mathcal{S}_1 \subset \mathcal{S}_2 \subset \ldots \subset \mathcal{S}_n$, where for $m < n$ we define $\mathcal{S}_m \subset \mathcal{S}_n$ to be the permutations in $\mathcal{S}_n$ which leave the last $n - m$ elements fixed. For example, if $n = 3$, then $\mathcal{S}_3 = \{e, (12), (23), (13), (123), (321)\}$, $\mathcal{S}_2 = \{e, (12)\}$, and $\mathcal{S}_1 = \{e\}$. Recall that the irreps of $\mathcal{S}_n$ can be labeled by $\mathcal{I}_n = \mathcal{I}_{n,n}$: the partitions of n into $\leq n$ parts.

Again, the branching from $\mathcal{S}_n$ to $\mathcal{S}_{n-1}$ is multiplicity-free, so to determine an orthonormal basis P_λ for the space $\mathcal{P}_\lambda$ we need only know which irreps occur in the decomposition of $\mathcal{P}_\lambda{\downarrow_{\mathcal{S}_{n-1}}}$. It turns out that the branching rule is given by finding all ways to remove one box from λ while leaving a valid partition. Denote the set of such partitions $\lambda - \Box$. Formally, $\lambda - \Box := \mathcal{I}_n \cap \{\lambda - e_j : j = 1, \ldots, n\}$, where e_j is the unit vector in $\mathbb{Z}^n$ with a one in the j^{th} position and zeroes elsewhere. Thus, the general branching rule is

$$\mathcal{P}_\lambda{\downarrow_{\mathcal{S}_{n-1}}} \overset{\mathcal{S}_{n-1}}{\cong} \bigoplus_{\mu \in \lambda - \Box} \mathcal{P}_\mu. \tag{25}$$

For example, if $\lambda = (3, 2, 1)$, we might have the chain of partitions:



(26)

Again, we can concisely label this chain by writing the number j in the box that is removed when restricting from $\mathcal{S}_j$ to $\mathcal{S}_{j-1}$. The above example would then be

1	3	6
2	4	
5		

. (27)

Note that the valid methods of filling a Young diagram are slightly different than for the $\mathcal{U}_d$ case. Now we use each integer in $1, \ldots, n$ exactly once such that the numbers are increasing from left to right and from top to bottom. The resulting tableau is called a *standard Young tableau*. (The same filling scheme appeared in the description of Young's natural representation in Sec. II C, but the resulting basis states are of course quite different.)

This gives rise to a straightforward, but inefficient, method of writing an element of P_λ using $\log n!$ bits. However, for applications such as data compression[23, 24] we will need an encoding which gives us closer to the optimal $\log P_\lambda$ bits. First we note an exact (and efficiently computable) expression for $|P_\lambda| = \dim \mathcal{P}_\lambda$:

$$\dim \mathcal{P}_\lambda = \frac{n!}{\lambda_1 + d - 1! \lambda_2 + d - 2! \cdots \lambda_d!} \prod_{1 \leq i < j \leq d} (\lambda_i - \lambda_j + j - i). \tag{28}$$

Now we would like to efficiently and reversibly map an element of P_λ (thought of as a chain of partitions $p = (p_n = \lambda, \ldots, p_1 = (1)) \in P_\lambda$, with $p_j \in p_{j+1} - \Box$) to an integer in $[|P_\lambda|] := \{1, \ldots, |P_\lambda|\}$. We will construct this bijection $f_n : P_\lambda \to [|P_\lambda|]$ by defining an ordering on P_λ and setting $f_n(p) := |\{p' \in P_\lambda : p' \leq p\}|$. First fix an arbitrary, but easily computable, (total) ordering on partitions in $\mathcal{I}_n$ for each n; for example, lexicographical order. This induces an ordering on P_λ if we rank a basis vector $p \in P_\lambda$ first according to p_{n-1}, using the order on partitions we have chosen, then according to p_{n-2} and so on. We skip p_n, since it is always equal to λ. In other words, for $p, p' \in P_\lambda$, $p > p'$ if $p_{n-1} > p'_{n-1}$ *or* $p_{n-1} = p'_{n-1}$ and $p_{n-2} > p'_{n-2}$ *or* $p_{n-1} = p'_{n-1}$, $p_{n-2} = p'_{n-2}$ and $p_{n-3} > p'_{n-3}$, and so on. Thus $f_n : P_\lambda \to [|P_\lambda|]$ can be easily verified to be

$$f_n(p) = f_n(p_1, \ldots, p_n) := 1 + \sum_{k=2}^{n} \sum_{\substack{\mu \in p_k - \Box \\ \mu < p_{k-1}}} \dim \mathcal{P}_\mu. \tag{29}$$

Thus f_n is an injective map from P_λ to $[|P_\lambda|]$. Moreover, since there are $O(n^2)$ terms in Eq. (29) and Eq. (28) gives an efficient way to calculate each $|P_\lambda|$, this mapping can be performed in time polynomial in n.

IV. THE CLEBSCH-GORDAN TRANSFORM AND EFFICIENT CIRCUITS FOR THE SCHUR TRANSFORM

In this section, we describe an efficient circuit for the Schur transform $\mathbf{U}_{\mathrm{Sch}}$. To do so, we first describe the Clebsch-Gordan transform, which decomposes a Kronecker product of $\mathcal{U}_d$-irreps $\mathcal{Q}^d_\mu \otimes \mathcal{Q}^d_\nu$ into a direct sum of other $\mathcal{U}_d$-irreps. We defer the algorithm for the Clebsch-Gordan transform to Sec. V, but give a description of its properties in Sec. IV A. Then in Sec. IV B, we show how to construct the Schur transform by cascading a series of Clebsch-Gordan transforms and performing reversible classical manipulations of P_λ and Q^d_λ.

A. The Clebsch-Gordan Series and Transform

Suppose we have two irreps of $\mathcal{U}_d$ given by the partitions μ and ν: $\mathcal{Q}^d_\mu$ and $\mathcal{Q}^d_\nu$. The tensor product of these irreps $\mathcal{Q}^d_\mu \otimes \mathcal{Q}^d_\nu$ (with representation matrices $\mathbf{q}^d_\mu(U) \otimes \mathbf{q}^d_\nu(U)$ for $U \in \mathcal{U}_d$) is a new representation of $\mathcal{U}_d$. This new representation will generally be reducible into irreps of $\mathcal{U}_d$; following Eq. (3) we have

$$\mathcal{Q}^d_\mu \otimes \mathcal{Q}^d_\nu \cong \bigoplus_{\lambda \in \mathbb{Z}^d_{++}} \mathcal{Q}^d_\lambda \otimes \mathrm{Hom}(\mathcal{Q}^d_\lambda, \mathcal{Q}^d_\mu \otimes \mathcal{Q}^d_\nu)^{\mathcal{U}_d} \tag{30}$$

This decomposition is referred to as the Clebsch-Gordan series. Setting $N^\lambda_{\mu\nu} = \dim \mathrm{Hom}(\mathcal{Q}^d_\lambda, \mathcal{Q}^d_\mu \otimes \mathcal{Q}^d_\nu)^{\mathcal{U}_d}$, we obtain the corresponding decomposition of the representation matrices as

$$\mathbf{q}^d_\mu(U) \otimes \mathbf{q}^d_\nu(U) \cong \bigoplus_{\lambda \in \mathbb{Z}^d_{++}} \mathbf{q}^d_\lambda(U) \otimes I_{N^\lambda_{\mu\nu}} \tag{31}$$

The unitary matrix which maps the LHS of Eq. (31) to the RHS is known as the Clebsch-Gordan transform and we denote it $\mathbf{U}^{\mu,\nu}_{\mathrm{CG}}$. It maps vectors of the form $|q_\mu\rangle|q_\nu\rangle \in \mathcal{Q}^d_\mu \otimes \mathcal{Q}^d_\nu$ to superpositions of vectors of the form $|\lambda\rangle|q_\lambda\rangle|\alpha\rangle$, where $\lambda \in \mathbb{Z}^d_{++}$, $|q_\lambda\rangle \in \mathcal{Q}^d_\lambda$ and $\alpha \in \mathrm{Hom}(\mathcal{Q}^d_\lambda, \mathcal{Q}^d_\mu \otimes \mathcal{Q}^d_\nu)^{\mathcal{U}_d}$.

The multiplicity space $\mathrm{Hom}(\mathcal{Q}^d_\lambda, \mathcal{Q}^d_\mu \otimes \mathcal{Q}^d_\nu)^{\mathcal{U}_d}$ plays a crucial role in the CG transform. In particular, the inverse CG transform $(\mathbf{U}^{\mu,\nu}_{\mathrm{CG}})^\dagger$ is given simply by

$$(\mathbf{U}^{\mu,\nu}_{\mathrm{CG}})^\dagger |q_\lambda\rangle|\alpha\rangle = \alpha|q_\lambda\rangle. \tag{32}$$

Note that on the LHS, we interpret $|\alpha\rangle$ as a vector in the multiplicity space $\mathrm{Hom}(\mathcal{Q}^d_\lambda, \mathcal{Q}^d_\mu \otimes \mathcal{Q}^d_\nu)^{\mathcal{U}_d}$, and on the RHS we treat α as an operator. These are normalized such that $|\alpha\rangle$ is a unit vector if and only if α is an isometry.

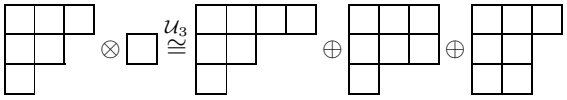
We now specialize to the case of tensoring in the defining irrep $\mathcal{Q}^d_{(1)}$, for which the CG transform is particularly simple. Recall that for $\lambda \in \mathbb{Z}^d_{++}$ and $1 \le j \le d$ we have $\lambda + e_j = (\lambda_1, \ldots, \lambda_{j-1}, \lambda_j + 1, \lambda_{j+1}, \ldots, \lambda_d)$. This is not always a valid partition, i.e. if $\lambda' = \lambda + e_j$, the condition $\lambda'_{j-1} \ge \lambda'_j$ might not hold. Recall that the valid partitions (of any integer) are given by the set $\mathbb{Z}^d_{++}$. Then the Clebsch-Gordan series we are interested in is given by

$$\mathcal{Q}^d_\lambda \otimes \mathcal{Q}^d_{(1)} \overset{\mathcal{U}_d}{\cong} \bigoplus_{\substack{j=1,\ldots d \\ \lambda + e_j \in \mathbb{Z}^d_{++}}} \mathcal{Q}^d_{\lambda + e_j} \tag{33}$$

This is the "add a single box" prescription for tensoring in a defining representation of $\mathcal{U}_d$: we add a single box to a Young diagram and if the new Young diagram is a valid Young diagram (i.e. corresponds to a valid partition), then this irrep appears in the Clebsch-Gordan series. For example if $\lambda = (3, 2, 1)$ then

$$\mathcal{Q}^3_{(3,2,1)} \otimes \mathcal{Q}^3_{(1)} \overset{\mathcal{U}_3}{\cong} \mathcal{Q}^3_{(4,2,1)} \oplus \mathcal{Q}^3_{(3,3,1)} \oplus \mathcal{Q}^3_{(3,2,2)} \tag{34}$$

or in Young diagram form



(35)

Note that if we had $d > 3$, then the partition $(3, 2, 1, 1)$ would also appear.

We now seek to define the CG transform as a quantum circuit. We specialize to the case where one of the input irreps is the defining irrep, but allow the other irrep to be specified by a quantum input. The resulting CG transform is defined as:

$$\mathbf{U}_{\mathrm{CG}} = \sum_{\lambda \in \mathbb{Z}^d_{++}} |\lambda\rangle\langle\lambda| \otimes \mathbf{U}^{\lambda,(1)}_{\mathrm{CG}}. \tag{36}$$

This takes as input a state of the form $|\lambda\rangle|q\rangle|i\rangle$, for $\lambda \in \mathbb{Z}^d_{++}$, $|q\rangle \in Q^d_\lambda$ and $i \in [d]$. The output is a superposition over vectors $|\lambda\rangle|\lambda'\rangle|q'\rangle$, where $\lambda' = \lambda + e_j \in \mathbb{Z}^d_{++}$, $j \in [d]$ and $|q'\rangle \in Q^d_{\lambda'}$. Equivalently, we could output $|\lambda\rangle|j\rangle|q'\rangle$ or $|j\rangle|\lambda'\rangle|q'\rangle$, since (λ, λ'), (λ, j) and (λ', j) are all trivially related via reversible classical circuits.

To better understand the input space of $\mathbf{U}_{\mathrm{CG}}$, we introduce the *model representation* $\mathcal{Q}^d_* := \bigoplus_{\lambda \in \mathbb{Z}^d_{++}} \mathcal{Q}^d_\lambda$, with corresponding matrix $\mathbf{q}^d_*(U) = \sum_\lambda |\lambda\rangle\langle\lambda| \otimes \mathbf{q}^d_\lambda(U)$. The model representation (also sometimes called the *Schwinger representation*) is infinite dimensional and contains each irrep once.[2] Its basis vectors are of the form $|\lambda, q\rangle$ for $\lambda \in \mathbb{Z}^d_{++}$ and $|q\rangle \in Q^d_\lambda$. Since $\mathcal{Q}^d_*$ is infinite-dimensional, we cannot store it on a quantum computer and in this paper work only with representations $\mathcal{Q}^d_\lambda$ with $|\lambda| \leq n$; nevertheless $\mathcal{Q}^d_*$ is a useful abstraction.

Thus $\mathbf{U}_{\mathrm{CG}}$ decomposes $\mathcal{Q}^d_* \otimes \mathcal{Q}^d_{(1)}$ into irreps. There are two important things to notice about this version of the CG transform. First is that it operates simultaneously on different input irreps. Second is that different input irreps must remain orthogonal, so in order to to maintain unitarity $\mathbf{U}_{\mathrm{CG}}$ needs to keep the information of which irrep we started with. However, since $\lambda' = \lambda + e_j$, this information requires only storing some $j \in [d]$. Thus, $\mathbf{U}_{\mathrm{CG}}$ is a map from $\mathcal{Q}^d_* \otimes \mathbb{C}^d$ to $\mathcal{Q}^d_* \otimes \mathbb{C}^d$, where the $\mathbb{C}^d$ in the input is the defining representation and the $\mathbb{C}^d$ in the output tracks which irrep we started with.

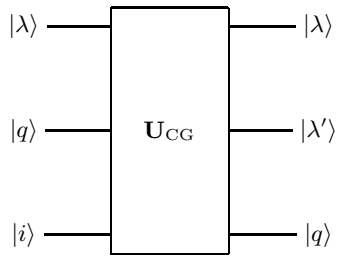


FIG. 2: Schematic of the Clebsch-Gordan transform. Equivalently, we could replace either the λ output or the λ' output with j.

B. Constructing the Schur Transform from Clebsch-Gordan Transforms

We now describe how to construct the Schur transform out of a series of Clebsch-Gordan transforms. Suppose we start with an input vector $|i_1, \ldots, i_n\rangle \in (\mathbb{C}^d)^{\otimes n}$, corresponding to the $\mathcal{U}_d$-representation $(\mathcal{Q}^d_{(1)})^{\otimes n}$. According to Schur duality (Eq. (10)), to perform the Schur transform it suffices to decompose $(\mathcal{Q}^d_{(1)})^{\otimes n}$ into $\mathcal{U}_d$-irreps. This is because Schur duality means that the multiplicity space of $\mathcal{Q}^d_\lambda$ must be isomorphic to $\mathcal{P}_\lambda$. In other words, if we show that

$$(\mathcal{Q}^d_{(1)})^{\otimes n} \overset{\mathcal{U}_d}{\cong} \bigoplus_{\lambda \in \mathbb{Z}^d_{++}} \mathcal{Q}^d_\lambda \otimes \mathcal{P}'_\lambda, \tag{37}$$

then we must have $\mathcal{P}'_\lambda \overset{\mathcal{S}_n}{\cong} \mathcal{P}_\lambda$ when $\lambda \in \mathcal{I}_{d,n}$ and $\mathcal{P}'_\lambda = \{0\}$ otherwise.

---

[2] By contrast, $L^2(\mathcal{U}_d)$, which will we not use, contains $\mathcal{Q}^d_\lambda$ with multiplicity $\dim \mathcal{Q}^d_\lambda$

To perform the $\mathcal{U}_d$-irrep decomposition of Eq. (37), we simply combine each of $|i_1\rangle, \ldots, |i_n\rangle$ using the CG transform, one at a time. We start by inputting $\left|\lambda^{(1)}\right\rangle = |(1)\rangle$, $|i_1\rangle$ and $|i_2\rangle$ into $\mathbf{U}_{\mathrm{CG}}$ which outputs $\left|\lambda^{(1)}\right\rangle$ and a superposition of different values of $\left|\lambda^{(2)}\right\rangle$ and $|q_2\rangle$. Here $\lambda^{(2)}$ can be either $(2,0)$ or $(1,1)$ and $|q_2\rangle \in Q^d_{\lambda^{(2)}}$. Continuing, we apply $\mathbf{U}_{\mathrm{CG}}$ to $\left|\lambda^{(2)}\right\rangle|q_2\rangle|i_3\rangle$, and output a superposition of vectors of the form $\left|\lambda^{(2)}\right\rangle\left|\lambda^{(3)}\right\rangle|q_3\rangle$, with $\lambda^{(3)} \in \mathcal{I}_{d,3}$ and $|q_3\rangle \in Q^d_{\lambda^{(3)}}$. Each time we are combining an arbitrary irrep $\lambda^{(k)}$ and an associated basis vector $|q_k\rangle \in Q^d_{\lambda^{(k)}}$, together with a vector from the defining irrep $|i_{k+1}\rangle$. This is repeated for $k = 1, \ldots, n-1$ and the resulting circuit is depicted in Fig. 3.

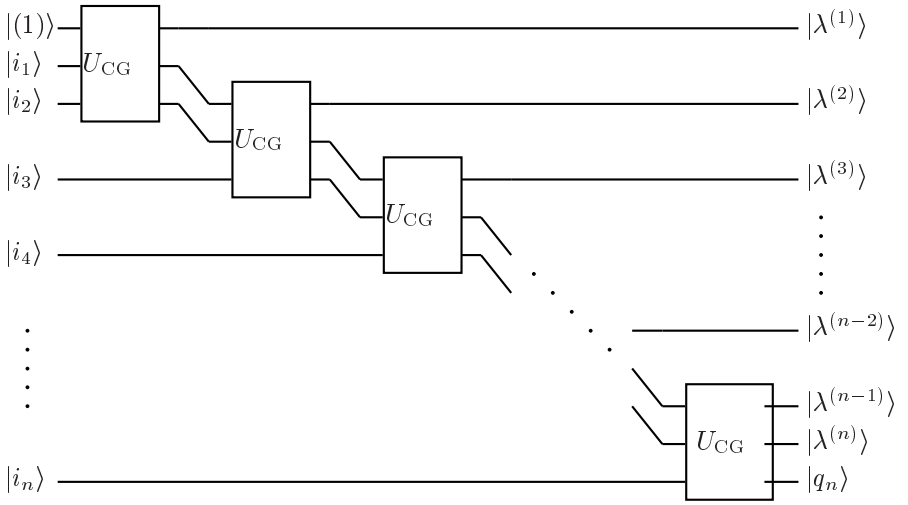


FIG. 3: Cascading Clebsch-Gordan transforms to produce the Schur transform. Not shown are any ancilla inputs to the Clebsch-Gordan transforms. The structure of inputs and outputs of the Clebsch-Gordan transforms are the same as in Fig. 2.

Finally, we are left with a superposition of states of the form $\left|\lambda^{(1)}, \ldots, \lambda^{(n)}\right\rangle|q_n\rangle$, where $|q_n\rangle \in Q^d_{\lambda^{(n)}}$, $\lambda^{(k)} \in \mathcal{I}_{d,k}$ and each $\lambda^{(k)}$ is obtained by adding a single box to $\lambda^{(k-1)}$; i.e. $\lambda^{(k)} = \lambda^{(k-1)} + e_{j_k}$ for some $j_k \in [d]$. If we define $\lambda = \lambda^{(n)}$ and $|q\rangle = |q_n\rangle$, then we have the decomposition of Eq. (37) with $\mathcal{P}'_\lambda$ spanned by the vectors $\left|\lambda^{(1)}, \ldots, \lambda^{(n-1)}\right\rangle$ satisfying the constraints described above. But this is precisely the Young-Yamanouchi basis P_λ that we have defined in Sec. III! Since the first k qudits transform under $\mathcal{U}_d$ according to $\mathcal{Q}^d_{\lambda^{(k)}}$, Schur duality implies that they also transform under $\mathcal{S}_k$ according to $\mathcal{P}_{\lambda^{(k)}}$. Thus we set $|p\rangle = \left|\lambda^{(1)}, \ldots, \lambda^{(n-1)}\right\rangle$ (optionally compressing to $\lceil \log |P_\lambda| \rceil$ qubits using the techniques described in the last section) and obtain the desired $|\lambda\rangle|q\rangle|p\rangle$. As a check on this result, note that each $\lambda^{(k)}$ is invariant under $\mathbf{Q}(\mathcal{U}_d)$ since $U^{\otimes n}$ acts on the first k qubits simply as $U^{\otimes k}$.

If we choose not to perform the poly(n) steps to optimally compress $\left|\lambda^{(1)}, \ldots, \lambda^{(n-1)}\right\rangle$, we could instead have our circuit output the equivalent $|j_1, \ldots, j_{n-1}\rangle$, which requires only $n \log d$ qubits and asymptotically no extra running time.

We can now appreciate the similarity between the $\mathcal{U}_d$ CG "add a box" prescription and the $\mathcal{S}_{n-1} \subset \mathcal{S}_n$ branching rule of "remove a box." Schur duality implies that the representations $\mathcal{Q}^d_{\lambda'}$ that are obtained by decomposing $\mathcal{Q}^d_\lambda \otimes \mathcal{Q}^d_{(1)}$ are the same as the $\mathcal{S}_n$-irreps $\mathcal{P}_{\lambda'}$ that include $\mathcal{P}_\lambda$ when restricted to $\mathcal{S}_{n-1}$.

Define $T_{\mathrm{CG}}(n, d, \epsilon)$ to be the time complexity (in terms of number of gates) of performing a single $\mathcal{U}_d$ CG transform to accuracy ϵ on Young diagrams with $\leq n$ boxes. Then the total complexity for the Schur transform is $n \cdot (T_{\mathrm{CG}}(n, d, \epsilon/n) + O(1))$, possibly plus a poly$(n)$ factor for compressing the $\mathcal{P}_\lambda$ register to $\lceil \log \dim \mathcal{P}_\lambda \rceil$ qubits (as is required for applications such as data compression and entanglement concentration, cf. Sec. II D). In the next section we will show that $T_{\mathrm{CG}}(n, d, \epsilon)$ is poly$(\log n, d, \log 1/\epsilon)$, but first we give a step-by-step description of the algorithm for the Schur transform.

Algorithm: Schur transform (plus optional compression)
Inputs: (1) Classical registers d and n. (2) An n qudit quantum register $|i_1, \ldots, i_n\rangle$.
Outputs: Quantum registers $|\lambda\rangle|q\rangle|p\rangle$, with $\lambda \in \mathcal{I}_{d,n}$, $q \in Q^d_\lambda$ and $p \in P_\lambda$.
Runtime: $n \cdot (T_{\mathrm{CG}}(n, d, \epsilon/n) + O(1))$ to achieve accuracy ϵ.
(Optionally plus poly(n) to compress the $\mathcal{P}_\lambda$ register to $\lceil \log \dim \mathcal{P}_\lambda \rceil$ qubits.)

Procedure:
1. Initialize $\left|\lambda^{(1)}\right\rangle := |(1)\rangle$ and $|q_1\rangle = |i_1\rangle$.
2. For $k = 1, \ldots, n-1$:
3. Apply $\mathbf{U}_{\mathrm{CG}}$ to $\left|\lambda^{(k)}\right\rangle|q_k\rangle|i_{k+1}\rangle$ to obtain output $|j_k\rangle\left|\lambda^{(k+1)}\right\rangle|q_{k+1}\rangle$, where $\lambda^{(k+1)} = \lambda^{(k)} + e_{j_k}$.
4. Output $|\lambda\rangle := \left|\lambda^{(n)}\right\rangle$, $|q\rangle := |q_n\rangle$ and $|p\rangle := |j_1, \ldots, j_{n-1}\rangle$.
5. (Optionally use Eq. (29) to reversibly map $|j_1, \ldots, j_{n-1}\rangle$ to an integer $p \in [\dim \mathcal{P}_\lambda]$.)

This algorithm will be made efficient in the next section, where we efficiently construct the CG transform for $\mathcal{U}_d$, proving that $T_{\mathrm{CG}}(n, d, \epsilon) = \mathrm{poly}(\log n, d, \log 1/\epsilon)$.

V. EFFICIENT CIRCUITS FOR THE CLEBSCH-GORDAN TRANSFORM

We now turn to the actual construction of the circuit for the Clebsch-Gordan transform described in Sec. IV A. To get a feel for the what will be necessary, we start by giving a circuit for the CG transform that is efficient when d is constant; i.e. it has complexity $n^{O(d^2)}$, which is $\mathrm{poly}(n)$ for any constant value of d.

First recall that $\dim \mathcal{Q}^d_\lambda \leq (n+1)^{d^2}$. Thus, controlled on λ, we want to construct a unitary transform on a D-dimensional system for $D = \max_{\lambda \in \mathcal{I}_{d,n}} \dim \mathcal{Q}^d_\lambda = \mathrm{poly}(n)$. There are classical algorithms[39] to compute matrix elements of $\mathbf{U}_{\mathrm{CG}}$ to an accuracy ϵ_1 in time $\mathrm{poly}(D)\,\mathrm{poly}\log(1/\epsilon_1)$. Once we have calculated all the relevant matrix elements (of which there are only polynomially many), we can (again in time $\mathrm{poly}(D)\,\mathrm{poly}\log(1/\epsilon)$) decompose $\mathbf{U}_{\mathrm{CG}}$ into $D^2\,\mathrm{poly}\log(D)$ elementary one and two-qubit operations[40–43]. These can in turn be approximated to accuracy ϵ_2 by products of unitary operators from a fixed finite set (such as Clifford operators and a $\pi/8$ rotation) with a further overhead of $\mathrm{poly}\log(1/\epsilon_2)$[44, 45]. We can either assume the relevant classical computations (such as decomposing the $D \times D$ matrix into elementary gates) are performed coherently on a quantum computer, or as part of a polynomial-time classical Turing machine which outputs the quantum circuit. In any case, the total complexity is $\mathrm{poly}(n, \log 1/\epsilon)$ if the desired final accuracy is ϵ and d is held constant.

The goal of this section is to reduce this running time to $\mathrm{poly}(n, d, \log(1/\epsilon))$; in fact, we will achieve circuits of size $\mathrm{poly}(d, \log n, \log(1/\epsilon))$. To do so, we will reduce the $\mathcal{U}_d$ CG transform to two components; first, a $\mathcal{U}_{d-1}$ CG transform, and second, a $d \times d$ unitary matrix whose entries can be computed classically in $\mathrm{poly}(d, \log n, 1/\epsilon)$ steps. After computing all d^2 entries, the second component can then be implemented with $\mathrm{poly}(d, \log 1/\epsilon)$ gates according to the above arguments.

This reduction from the $\mathcal{U}_d$ CG transform to the $\mathcal{U}_{d-1}$ CG transform is a special case of the Wigner-Eckart Theorem, which we review in Sec. V A. Then, following [39, 46], we use the Wigner-Eckart Theorem to give an efficient recursive construction for $\mathbf{U}_{\mathrm{CG}}$ in Sec. V B. Putting everything together, we obtain a quantum circuit for the Schur transform that is accurate to within ϵ and runs in time $n \cdot \mathrm{poly}(\log n, d, \log 1/\epsilon)$, optionally plus an additional $\mathrm{poly}(n)$ time to compress the $|p\rangle$ register.

A. The Wigner-Eckart Theorem and Clebsch-Gordan transform

In this section, we introduce the concept of an irreducible tensor operator, which we use to state and prove the Wigner-Eckart Theorem. Here we will find that the CG transform is a key part of the Wigner-Eckart Theorem, while in the next section we will turn this around and use the Wigner-Eckart Theorem to give a recursive decomposition of the CG transform.

Suppose $(\mathbf{r}_1, V_1)$ and $(\mathbf{r}_2, V_2)$ are representations of $\mathcal{U}_d$. Recall that $\mathrm{Hom}(V_1, V_2)$ is a representation of $\mathcal{U}_d$ under the map $T \to \mathbf{r}_2(U) T \mathbf{r}_1(U)^{-1}$ for $T \in \mathrm{Hom}(V_1, V_2)$. If $\boldsymbol{T} = \{T_1, T_2, \ldots\} \subset \mathrm{Hom}(V_1, V_2)$ is a basis for a $\mathcal{U}_d$-invariant subspace of $\mathrm{Hom}(V_1, V_2)$, then we call $\boldsymbol{T}$ a *tensor operator*. Note that a tensor operator $\mathbf{T}$ is a collection of operators $\{T_i\}$ indexed by i, just as a tensor (or vector) is a collection of scalars labeled by some index. For example, the Pauli matrices $\{\sigma_x, \sigma_y, \sigma_z\} \subset \mathrm{Hom}(\mathbb{C}^2, \mathbb{C}^2)$ comprise a tensor operator, since conjugation by $\mathcal{U}_2$ preserves the subspace that they span.

Since $\mathrm{Hom}(V_1, V_2)$ is a representation of $\mathcal{U}_d$, it can be decomposed into irreps. If $\boldsymbol{T}$ is a basis for one of these irreps, then we call it an *irreducible tensor operator*. For example, the Pauli matrices mentioned above comprise an irreducible tensor operator, corresponding to the three-dimensional irrep $\mathcal{Q}^2_{(2)}$. Formally, we say that $\boldsymbol{T}^\nu = \{T^\nu_{q_\nu}\}_{q_\nu \in Q^d_\nu} \subset \mathrm{Hom}(V_1, V_2)$ is an irreducible tensor operator (corresponding to the irrep $\mathcal{Q}^d_\nu$) if for

all $U \in \mathcal{U}_d$ we have

$$\mathbf{r}_2(U) T^{\nu}_{q_\nu} \mathbf{r}_1(U)^{-1} = \sum_{q'_\nu \in Q^d_\nu} \langle q'_\nu | \mathbf{q}^d_\nu(U) | q_\nu \rangle T^{\nu}_{q'_\nu}. \tag{38}$$

Now assume that V_1 and V_2 are irreducible (say $V_1 = \mathcal{Q}^d_\mu$ and $V_2 = \mathcal{Q}^d_\lambda$), since if they are not, we could always decompose $\mathrm{Hom}(V_1, V_2)$ into a direct sum of homomorphisms from an irrep in V_1 to an irrep in V_2. We can decompose $\mathrm{Hom}(\mathcal{Q}^d_\mu, \mathcal{Q}^d_\lambda)$ into irreps using Eq. (3) and the identity $\mathrm{Hom}(A, B) \cong A^* \otimes B$ as follows:

$$\begin{aligned}
\mathrm{Hom}(\mathcal{Q}^d_\mu, \mathcal{Q}^d_\lambda) &\overset{\mathcal{U}_d}{\cong} \bigoplus_{\nu \in \mathbb{Z}^d_{++}} \mathcal{Q}^d_\nu \otimes \mathrm{Hom}(\mathcal{Q}^d_\nu, \mathrm{Hom}(\mathcal{Q}^d_\mu, \mathcal{Q}^d_\lambda))^{\mathcal{U}_d} \\
&\overset{\mathcal{U}_d}{\cong} \bigoplus_{\nu \in \mathbb{Z}^d_{++}} \mathcal{Q}^d_\nu \otimes \mathrm{Hom}(\mathcal{Q}^d_\nu, (\mathcal{Q}^d_\mu)^* \otimes \mathcal{Q}^d_\lambda)^{\mathcal{U}_d} \\
&\overset{\mathcal{U}_d}{\cong} \bigoplus_{\nu \in \mathbb{Z}^d_{++}} \mathcal{Q}^d_\nu \otimes \left((\mathcal{Q}^d_\mu)^* \otimes (\mathcal{Q}^d_\nu)^* \otimes \mathcal{Q}^d_\lambda\right)^{\mathcal{U}_d} \\
&\overset{\mathcal{U}_d}{\cong} \bigoplus_{\nu \in \mathbb{Z}^d_{++}} \mathcal{Q}^d_\nu \otimes \mathrm{Hom}(\mathcal{Q}^d_\mu \otimes \mathcal{Q}^d_\nu, \mathcal{Q}^d_\lambda)^{\mathcal{U}_d}
\end{aligned} \tag{39}$$

Now consider a particular irreducible tensor operator $\mathbf{T}^\nu \subset \mathrm{Hom}(\mathcal{Q}^d_\mu, \mathcal{Q}^d_\lambda)$ with components $T^\nu_{q_\nu}$ where q_ν ranges over Q^d_ν. We can define a linear operator $\hat{T} : \mathcal{Q}^d_\mu \otimes \mathcal{Q}^d_\nu \to \mathcal{Q}^d_\lambda$ by letting

$$\hat{T} |q_\mu\rangle |q_\nu\rangle := T^\nu_{q_\nu} |q_\mu\rangle \tag{40}$$

for all $q_\mu \in Q^d_\mu, q_\nu \in Q^d_\nu$ and extending it to the rest of $\mathcal{Q}^d_\mu \otimes \mathcal{Q}^d_\nu$ by linearity. By construction, $\hat{T} \in \mathrm{Hom}(\mathcal{Q}^d_\mu \otimes \mathcal{Q}^d_\nu, \mathcal{Q}^d_\lambda)$, but we claim that in addition $\hat{T}$ is invariant under the action of $\mathcal{U}_d$; i.e. that it lies in $\mathrm{Hom}(\mathcal{Q}^d_\mu \otimes \mathcal{Q}^d_\nu, \mathcal{Q}^d_\lambda)^{\mathcal{U}_d}$. To see this, apply Eqns. (38) and (40) to show that for any $U \in \mathcal{U}_d$, $q_\mu \in Q^d_\mu$ and $q_\nu \in Q^d_\nu$, we have

$$\begin{aligned}
\mathbf{q}^d_\lambda(U) \hat{T} \left[\mathbf{q}^d_\mu(U)^{-1} \otimes \mathbf{q}^d_\nu(U)^{-1}\right] |q_\mu\rangle |q_\nu\rangle &= \sum_{q'_\nu \in Q^d_\nu} \langle q'_\nu | \mathbf{q}^d_\nu(U)^{-1} | q_\nu \rangle \mathbf{q}^d_\lambda(U) T^\nu_{q'_\nu} \mathbf{q}^d_\mu(U)^{-1} |q_\mu\rangle \\
&= \sum_{q'_\nu, q''_\nu \in Q^d_\nu} \langle q''_\nu | \mathbf{q}^d_\nu(U) | q'_\nu \rangle \langle q'_\nu | \mathbf{q}^d_\nu(U)^{-1} | q_\nu \rangle T^\nu_{q''_\nu} |q_\mu\rangle \\
&= T^\nu_{q_\nu} |q_\mu\rangle = \hat{T} |q_\mu\rangle |q_\nu\rangle.
\end{aligned} \tag{41}$$

Now, fix an orthonormal basis for $\mathrm{Hom}(\mathcal{Q}^d_\mu \otimes \mathcal{Q}^d_\nu, \mathcal{Q}^d_\lambda)^{\mathcal{U}_d}$ and call it $M^\lambda_{\mu,\nu}$. Then we can expand $\hat{T}$ in this basis as

$$\hat{T} = \sum_{\alpha \in M^\lambda_{\mu,\nu}} \hat{T}_\alpha \cdot \alpha, \tag{42}$$

where the $\hat{T}_\alpha$ are scalars. Thus

$$\langle q_\lambda | T^\nu_{q_\nu} | q_\mu \rangle = \sum_{\alpha \in M^\lambda_{\mu,\nu}} \hat{T}_\alpha \langle q_\lambda | \alpha | q_\mu, q_\nu \rangle. \tag{43}$$

This last expression $\langle q_\lambda | \alpha | q_\mu, q_\nu \rangle$ bears a striking resemblance to the CG transform. Indeed, note that the multiplicity space $\mathrm{Hom}(\mathcal{Q}^d_\lambda, \mathcal{Q}^d_\mu \otimes \mathcal{Q}^d_\nu)^{\mathcal{U}_d}$ from Eq. (30) is the dual of $\mathrm{Hom}(\mathcal{Q}^d_\mu \otimes \mathcal{Q}^d_\nu, \mathcal{Q}^d_\lambda)^{\mathcal{U}_d}$ (which contains α), meaning that we can map between the two by taking the transpose. In fact, taking the conjugate transpose of Eq. (32) gives $\langle q_\lambda | \alpha = \langle q_\lambda, \alpha^\dagger | \mathbf{U}^{\mu,\nu}_{\mathrm{CG}}$. Thus

$$\langle q_\lambda | \alpha | q_\mu, q_\nu \rangle = \langle q_\lambda, \alpha^\dagger | \mathbf{U}^{\mu,\nu}_{\mathrm{CG}} | q_\mu, q_\nu \rangle. \tag{44}$$

The arguments in the last few paragraphs constitute a proof of the Wigner-Eckart theorem[47], which is stated as follows:

Theorem 1 (Wigner-Eckart) *For any irreducible tensor operator* $\mathbf{T}^\nu = \{T^\nu_{q_\nu}\}_{q_\nu \in Q^d_\nu} \subset \mathrm{Hom}(\mathcal{Q}^d_\mu, \mathcal{Q}^d_\lambda)$, *there exist* $\hat{T}_\alpha \in \mathbb{C}$ *for each* $\alpha \in M^\lambda_{\mu,\nu}$ *such that for all* $|q_\mu\rangle \in \mathcal{Q}^d_\mu$, $|q_\nu\rangle \in \mathcal{Q}^d_\nu$ *and* $|q_\lambda\rangle \in \mathcal{Q}^d_\lambda$:

$$\langle q_\lambda | T^\nu_{q_\nu} | q_\mu \rangle = \sum_{\alpha \in M^\lambda_{\mu,\nu}} \hat{T}_\alpha \langle q_\lambda, \alpha^\dagger \big| \mathbf{U}^{\mu,\nu}_{CG} | q_\mu, q_\nu \rangle. \tag{45}$$

Thus, the action of tensor operators can be related to a component $\hat{T}_\alpha$ that is invariant under $\mathcal{U}_d$ and a component that is equivalent to the CG transform. We will use this in the next section to derive an efficient quantum circuit for the CG transform.

B. A recursive construction of the Clebsch-Gordan transform

In this section we show how the $\mathcal{U}_d$ CG transform (which here we call $\mathbf{U}^{[d]}_{\mathrm{CG}}$) can be efficiently reduced to the $\mathcal{U}_{d-1}$ CG transform (which we call $\mathbf{U}^{[d-1]}_{\mathrm{CG}}$). Our strategy, following [46], will be to express $\mathbf{U}^{[d]}_{\mathrm{CG}}$ in terms of $\mathcal{U}_{d-1}$ tensor operators and then use the Wigner-Eckart Theorem to express it in terms of $\mathbf{U}^{[d-1]}_{\mathrm{CG}}$. After we have explained this as a relation among operators, we describe a quantum circuit for $\mathbf{U}^{[d]}_{\mathrm{CG}}$ that uses $\mathbf{U}^{[d-1]}_{\mathrm{CG}}$ as a subroutine.

First, we express $\mathbf{U}^{[d]}_{\mathrm{CG}}$ as a $\mathcal{U}_d$ tensor operator. For $\mu \in \mathbb{Z}^d_{++}$, $|q\rangle \in Q^d_\mu$ and $i \in [d]$, we can expand $\mathbf{U}^{[d]}_{\mathrm{CG}} |\mu\rangle|q\rangle|i\rangle$ as

$$\mathbf{U}^{[d]}_{\mathrm{CG}} |\mu\rangle|q\rangle|i\rangle = |\mu\rangle \sum_{\substack{j \in [d] \text{ s.t.} \\ \mu + e_j \in \mathbb{Z}^d_{++}}} \sum_{q' \in Q^d_{\mu+e_j}} C^{\mu,j}_{q,i,q'} |\mu + e_j\rangle |q'\rangle. \tag{46}$$

for some coefficients $C^{\mu,j}_{q,i,q'} \in \mathbb{C}$. Now define operators $T^{\mu,j}_i : \mathcal{Q}^d_\mu \to \mathcal{Q}^d_{\mu+e_j}$ by

$$T^{\mu,j}_i = \sum_{q \in Q^d_\mu} \sum_{q' \in Q^d_{\mu+e_j}} C^{\mu,j}_{q,i,q'} |q'\rangle\langle q|, \tag{47}$$

so that $\mathbf{U}^{[d]}_{\mathrm{CG}}$ decomposes as

$$\mathbf{U}^{[d]}_{\mathrm{CG}} |\mu\rangle|q\rangle|i\rangle = |\mu\rangle \sum_{\substack{j \in [d] \text{ s.t.} \\ \mu + e_j \in \mathbb{Z}^d_{++}}} |\mu + e_j\rangle T^{\mu,j}_i |q\rangle. \tag{48}$$

Thus $\mathbf{U}^{[d]}_{\mathrm{CG}}$ can be understood in terms of the maps $T^{\mu,j}_i$, which are irreducible tensor operators in $\mathrm{Hom}(\mathcal{Q}^d_\mu, \mathcal{Q}^d_{\mu+e_j})$ corresponding to the irrep $\mathcal{Q}^d_{(1)}$. (This is unlike the notation of the last section in which the superscript denoted the irrep corresponding to the tensor operator.)

The plan for the rest of the section is to decompose the $T^{\mu,j}_i$ operators under the action of $\mathcal{U}_{d-1}$, so that we can apply the Wigner-Eckart theorem. This involves decomposing three different $\mathcal{U}_d$ irreps into $\mathcal{U}_{d-1}$ irreps: the input space $\mathcal{Q}^d_\mu$, the output space $\mathcal{Q}^d_{\mu+e_j}$ and the space $\mathcal{Q}^d_{(1)}$ corresponding to the subscript i. Once we have done so, the Wigner-Eckart Theorem gives an expression for $T^{\mu,j}_i$ (and hence for $\mathbf{U}^{[d]}_{\mathrm{CG}}$) in terms of $\mathbf{U}^{[d-1]}_{\mathrm{CG}}$ and a small number of coefficients, known as *reduced Wigner coefficients*. These coefficients can be readily calculated, and in the next section we cite a formula from [46] for doing so.

First, we examine the decomposition of $\mathcal{Q}^d_{(1)}$, the $\mathcal{U}_d$-irrep according to which the $T^{\mu,j}_i$ transform. Recall that $\mathcal{Q}^d_{(1)} \overset{\mathcal{U}_{d-1}}{\cong} \mathcal{Q}^{d-1}_{(0)} \oplus \mathcal{Q}^{d-1}_{(1)}$. In terms of the tensor operator we have defined, this means that $T^{\mu,j}_d$ is an irreducible $\mathcal{U}_{d-1}$ tensor operator corresponding to the trivial irrep $\mathcal{Q}^{d-1}_{(0)}$ and $\{T^{\mu,j}_1, \ldots, T^{\mu,j}_{d-1}\}$ comprise an irreducible $\mathcal{U}_{d-1}$ tensor operator corresponding to the defining irrep $\mathcal{Q}^{d-1}_{(1)}$.

Next, we would like to decompose $\mathrm{Hom}(\mathcal{Q}^d_\mu, \mathcal{Q}^d_{\mu+e_j})$ into maps between irreps of $\mathcal{U}_{d-1}$. This is slightly more complicated, but can be derived from the $\mathcal{U}_{d-1} \subset \mathcal{U}_d$ branching rule introduced in Sec. III B. Recall that $\mathcal{Q}^d_\mu \overset{\mathcal{U}_{d-1}}{\cong} \bigoplus_{\mu' \precsim \mu} \mathcal{Q}^{d-1}_{\mu'}$, and similarly $\mathcal{Q}^d_{\mu+e_j} \overset{\mathcal{U}_{d-1}}{\cong} \bigoplus_{\mu'' \precsim \mu+e_j} \mathcal{Q}^{d-1}_{\mu''}$. This is the moment that we anticipated

in Sec. III B when we chose our set of basis vectors Q_μ^d to respect these decompositions. As a result, a vector $|q\rangle \in Q_\mu^d$ can be expanded as $q = (q_{d-1}, q_{d-2}, \ldots, q_1) = (\mu', q_{(d-2)})$ with $q_{d-1} = \mu' \in \mathbb{Z}_{++}^{d-1}$, $\mu' \precsim \mu$ and $\left|q_{(d-2)}\right\rangle = |q_{d-2}, \ldots, q_1\rangle \in Q_{\mu'}^{d-1}$. In other words, we will separate vectors in Q_μ^d into a $\mathcal{U}_{d-1}$ irrep label $\mu' \in \mathbb{Z}_{++}^{d-1}$ and a basis vector from $\mathcal{Q}_{\mu'}^{d-1}$.

This describes how to decompose the spaces $\mathcal{Q}_\mu^d$ and $\mathcal{Q}_{\mu+e_j}^d$. To extend this to decomposition of $\mathrm{Hom}(\mathcal{Q}_\mu^d, \mathcal{Q}_{\mu+e_j}^d)$, we use the canonical isomorphism $\mathrm{Hom}(\bigoplus_x A_x, \bigoplus_y B_y) \cong \bigoplus_{x,y} \mathrm{Hom}(A_x, B_y)$, which holds for any sets of vector spaces $\{A_x\}$ and $\{B_y\}$. Thus

$$\mathrm{Hom}(\mathcal{Q}_\mu^d, \mathcal{Q}_{\mu+e_j}^d) \overset{\mathcal{U}_{d-1}}{\cong} \bigoplus_{\mu' \precsim \mu} \bigoplus_{\mu'' \precsim \mu+e_j} \mathrm{Hom}(\mathcal{Q}_{\mu'}^{d-1}, \mathcal{Q}_{\mu''}^{d-1}). \tag{49a}$$

Sometimes we will find it convenient to denote the $\mathcal{Q}_{\mu'}^{d-1}$ subspace of $\mathcal{Q}_\mu^d$ by $\mathcal{Q}_{\mu'}^{d-1} \subset \mathcal{Q}_\mu^d$, so that Eq. (49a) becomes

$$\mathrm{Hom}(\mathcal{Q}_\mu^d, \mathcal{Q}_{\mu+e_j}^d) \overset{\mathcal{U}_{d-1}}{\cong} \bigoplus_{\mu' \precsim \mu} \bigoplus_{\mu'' \precsim \mu+e_j} \mathrm{Hom}(\mathcal{Q}_{\mu'}^{d-1} \subset \mathcal{Q}_\mu^d, \mathcal{Q}_{\mu''}^{d-1} \subset \mathcal{Q}_{\mu+e_j}^d). \tag{49b}$$

According to Eq. (49) (either version), we can decompose $T_i^{\mu,j}$ as

$$T_i^{\mu,j} = \sum_{\mu' \precsim \mu} \sum_{\mu'' \precsim \mu+e_j} |\mu''\rangle\langle\mu'| \otimes T_i^{\mu,j,\mu',\mu''}. \tag{50}$$

Here $T_i^{\mu,j,\mu',\mu''} \in \mathrm{Hom}(\mathcal{Q}_{\mu'}^{d-1} \subset \mathcal{Q}_\mu^d, \mathcal{Q}_{\mu''}^{d-1} \subset \mathcal{Q}_{\mu+e_j}^d)$ and we have implicitly decomposed $|q\rangle \in Q_\mu^d$ into $|\mu'\rangle\left|q_{(d-2)}\right\rangle$.

The next step is to decompose the representions in Eq. (49) into irreducible components. In fact, we are not interested in the entire space $\mathrm{Hom}(\mathcal{Q}_{\mu'}^{d-1}, \mathcal{Q}_{\mu''}^{d-1})$, but only the part that is equivalent to $\mathcal{Q}_{(1)}^{d-1}$ or $\mathcal{Q}_{(0)}^{d-1}$, depending on whether $i \in [d-1]$ or $i = d$ (since $T_i^{\mu,j,\mu',\mu''}$ transforms according to $\mathcal{Q}_{(1)}^{d-1}$ if $i \in \{1, \ldots, d-1\}$ and according to $\mathcal{Q}_{(0)}^{d-1}$ if $i = d$). This knowledge of how $T_i^{\mu,j,\mu',\mu''}$ transforms under $\mathcal{U}_{d-1}$ will give us two crucial simplifications: first, we can greatly reduce the range of μ'' for which $T_i^{\mu,j,\mu',\mu''}$ is nonzero, and second, we can apply the Wigner-Eckart theorem to describe $T_i^{\mu,j,\mu',\mu''}$ in terms of $\mathbf{U}_{\mathrm{CG}}^{[d-1]}$.

The simplest case is $\mathcal{Q}_{(0)}^{d-1}$, when $i = d$: according to Schur's Lemma the invariant component of $\mathrm{Hom}(\mathcal{Q}_{\mu'}^{d-1}, \mathcal{Q}_{\mu''}^{d-1})$ is zero if $\mu' \neq \mu''$ and consists of the matrices proportional to $I_{\mathcal{Q}_{\mu'}^{d-1}}$ if $\mu' = \mu''$. In other words $T_d^{\mu,j,\mu',\mu''} = 0$ unless $\mu' = \mu''$, in which case $T_d^{\mu,j,\mu',\mu'} := \hat{T}^{\mu,j,\mu',0} I_{\mathcal{Q}_{\mu'}^{d-1}}$ for some scalar $\hat{T}^{\mu,j,\mu',0}$. (The final superscript 0 will later be convenient when we want a single notation to encompass both the $i = d$ and the $i \in \{1, \ldots, d-1\}$ cases.)

The $\mathcal{Q}_{(1)}^{d-1}$ case, which occurs when $i \in \{1, \ldots, d-1\}$, is more interesting. We will simplify the $T_i^{\mu,j,\mu',\mu''}$ operators (for $i = 1, \ldots, d-1$) in two stages: first using the branching rules from Sec. III B to reduce the number of nonzero terms and then by applying the Wigner-Eckart theorem to find an exact expression for them. Begin by recalling from Eq. (39) that the multiplicity of $\mathcal{Q}_{(1)}^{d-1}$ in the isotypic decomposition of $\mathrm{Hom}(\mathcal{Q}_{\mu'}^{d-1}, \mathcal{Q}_{\mu''}^{d-1})$ is given by $\dim \mathrm{Hom}(\mathcal{Q}_{\mu'}^{d-1} \otimes \mathcal{Q}_{(1)}^{d-1}, \mathcal{Q}_{\mu''}^{d-1})^{\mathcal{U}_{d-1}}$. According to the $\mathcal{U}_{d-1}$ CG "add a box" prescription (Eq. (33)), this is one if $\mu' \in \mu'' - \square$ and zero otherwise. Thus if $i \in [d-1]$, then $T_i^{\mu,j,\mu',\mu''}$ is zero unless $\mu'' = \mu' + e_{j'}$ for some $j' \in [d-1]$. Since we need not consider all possible μ'', we can define $T_i^{\mu,j,\mu',j'} := T_i^{\mu,j,\mu',\mu'+e_{j'}}$. This notation can be readily extended to cover the case when $i = d$; define $e_0 = 0$, so that the only nonzero operators for $i = d$ are of the form $T_d^{\mu,j,\mu',0} := T_d^{\mu,j,\mu',\mu'} = \hat{T}^{\mu,j,\mu',0} I_{\mathcal{Q}_{\mu'}^{d-1}}$. Thus, we can replace Eq. (50) with

$$T_i^{\mu,j} = \sum_{\mu' \precsim \mu} \sum_{j'=0}^{d-1} |\mu' + e_{j'}\rangle\langle\mu'| \otimes T_i^{\mu,j,\mu',\mu'+e_{j'}}. \tag{51}$$

Now we show how to apply the Wigner-Eckart theorem to the $i \in [d-1]$ case. The operators $T_i^{\mu,j,\mu',j'}$ map $\mathcal{Q}_{\mu'}^{d-1}$ to $\mathcal{Q}_{\mu'+e_{j'}}^{d-1}$ and comprise an irreducible $\mathcal{U}_{d-1}$ tensor operator corresponding to the irrep $\mathcal{Q}_{(1)}^{d-1}$.

This means we can apply the Wigner-Eckart theorem and since the multiplicity of $\mathcal{Q}^{d-1}_{\mu'+e_{j'}}$ in $\mathcal{Q}^{d-1}_{\mu'} \otimes \mathcal{Q}^{d-1}_{(1)}$ is one, the sum over the multiplicity label α has only a single term. The theorem implies the existence of a set of scalars $\hat{T}^{\mu,j,\mu',j'}$ such that for any $|q\rangle \in Q^{d-1}_{\mu'}$ and $|q'\rangle \in Q^{d-1}_{\mu'+e_{j'}}$,

$$\langle q'|T_i^{\mu,j,\mu',j'}|q\rangle = \hat{T}^{\mu,j,\mu',j'}\langle \mu', \mu'+e_{j'}, q'|\mathbf{U}_{\mathrm{CG}}^{[d-1]}|\mu', q, i\rangle. \tag{52}$$

Sometimes the matrix elements of $\mathbf{U}_{\mathrm{CG}}$ or $T_i^{\mu,j,\mu',j'}$ are called *Wigner coefficients* and the $\hat{T}^{\mu,j,\mu',j'}$ are known as *reduced Wigner coefficients*.

Let us now try to interpret these equations operationally. Eq. (48) reduces the $\mathcal{U}_d$ CG transform to a $\mathcal{U}_d$ tensor operator, Eq. (51) decomposes this tensor operator into d^2 different $\mathcal{U}_{d-1}$ tensor operators (weighted by the $\hat{T}^{\mu,j,\mu',j'}$ coefficients) and Eq. (52) turns this into a $\mathcal{U}_{d-1}$ CG transform followed by a $d \times d$ unitary matrix. The coefficients for this matrix are the $\hat{T}^{\mu,j,\mu',j'}$, which we will see in the next section can be efficiently computed by conditioning on μ and μ'.

Now we spell this recursion out in more detail. Suppose we wish to apply $\mathbf{U}_{\mathrm{CG}}^{[d]}$ to $|\mu\rangle|q\rangle|i\rangle = |\mu\rangle|\mu'\rangle\big|q_{(d-2)}\big\rangle|i\rangle$, for some $i \in \{1, \ldots, d-1\}$. Then Eq. (52) indicates that we should first apply $\mathbf{U}_{\mathrm{CG}}^{[d-1]}$ to $|\mu'\rangle\big|q_{(d-2)}\big\rangle|i\rangle$ to obtain output that is a superposition over states $|\mu'+e_{j'}\rangle|j'\rangle\left|q'_{(d-2)}\right\rangle$ for $j' \in \{1, \ldots, d-1\}$ and $\left|q'_{(d-2)}\right\rangle \in Q^{d-1}_{\mu'+e_{j'}}$. Then, controlled by μ and μ', we want to map the $(d-1)$-dimensional $|j'\rangle$ register into the d-dimensional $|j\rangle$ register, which will then tell us the output irrep $\mathcal{Q}^d_{\mu+e_j}$. According to Eq. (52), the coefficients of this $d \times (d-1)$ matrix are given by the reduced Wigner coefficients $\hat{T}^{\mu,j,\mu',j'}$, so we will denote the overall matrix $\hat{T}^{[d]}_{\mu,\mu'} := \sum_{j,j'} \hat{T}^{\mu,j,\mu'+e_{j'},j'}|j\rangle\langle j'|$.[3] The resulting circuit is depicted in Fig. 4: a $\mathcal{U}_{d-1}$ CG transform is followed by the $\hat{T}^{[d]}$ operator, which is defined to be

$$\hat{T}^{[d]} = \sum_{\mu' \precsim \mu} \sum_{j,j'} \hat{T}^{\mu,j,\mu',j'}|\mu\rangle\langle\mu| \otimes |\mu+e_j\rangle\langle\mu'| \otimes |\mu'+e_{j'}\rangle\langle\mu'+e_{j'}|. \tag{53}$$

Then Fig. 5 shows how $\hat{T}^{[d]}$ can be expressed as a $d \times (d-1)$ matrix $\hat{T}^{[d]}_{\mu,\mu'}$ that is controlled by μ and μ'. In fact, once we consider the $i = d$ case in the next paragraph, we will find that $\hat{T}^{[d]}_{\mu,\mu'}$ is actually a $d \times d$ unitary matrix. In the next section, we will then show how the individual reduced Wigner coefficients $\hat{T}^{\mu,j,\mu',j'}$ can be efficiently computed, so that ultimately $\hat{T}^{[d]}_{\mu,\mu'}$ can be implemented in time $\mathrm{poly}(d, \log 1/\epsilon)$.

Now we turn to the case of $i = d$. The circuit is much simpler, but we also need to explain how it works in coherent superposition with the $i \in [d-1]$ case. Since $i = d$ corresponds to the trivial representation of $\mathcal{U}_{d-1}$, the $\mathbf{U}_{\mathrm{CG}}^{[d-1]}$ operation is not performed. Instead, $|\mu'\rangle$ and $\big|q_{(d-2)}\big\rangle$ are left untouched and the $|i\rangle = |d\rangle$ register is relabeled as a $|j'\rangle = |0\rangle$ register. We can combine this relabeling operation with $\mathbf{U}_{\mathrm{CG}}^{[d-1]}$ in the $i \in [d-1]$ case by defining

$$\tilde{U}_{\mathrm{CG}}^{[d-1]} := \left(|0\rangle\langle d| \otimes \sum_{\mu' \in \mathbb{Z}^{d-1}_{++}} |\mu'\rangle\langle\mu'|\right) \otimes I_{\mathcal{Q}^{d-1}_{\mu'}} + \mathbf{U}_{\mathrm{CG}}^{[d-1]}. \tag{54}$$

This ends up mapping $i \in \{1, \ldots, d\}$ to $j' \in \{0, \ldots, d-1\}$ while mapping $\mathcal{Q}^{d-1}_{\mu'}$ to $\mathcal{Q}^{d-1}_{\mu'+e_{j'}}$. Now we can interpret the sum on j' in the above definitions of $\hat{T}^{[d]}$ and $\hat{T}^{[d]}_{\mu,\mu'}$ as ranging over $\{0, \ldots, d-1\}$, so that $\hat{T}^{[d]}_{\mu,\mu'}$ is a $d \times d$ unitary matrix. We thus obtain the circuit in Fig. 4 with the implementation of $\hat{T}^{[d]}$ depicted in Fig. 5.

We have now reduced the problem of performing the CG transform $\mathbf{U}_{\mathrm{CG}}^{[d]}$ to the problem of computing reduced Wigner coefficients $\hat{T}^{\mu,j,\mu',j'}$.

---

[3] The reason why $\mu' + e_{j'}$ appears in the superscript rather than μ' is that after applying $\hat{T}^{[d]}_{\mu,\mu'}$ we want to keep a record of $\mu' + e_{j'}$ rather than of μ'. This is further illustrated in Fig. 5.

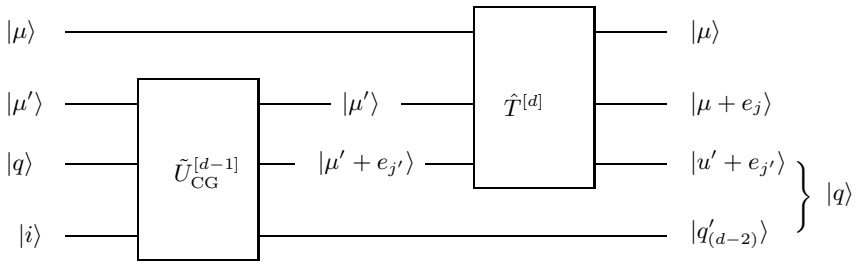


FIG. 4: The $\mathcal{U}_d$ CG transform, $\mathbf{U}_{\rm CG}^{[d]}$, is decomposed into a $\mathcal{U}_{d-1}$ CG transform $\tilde{U}_{\rm CG}^{[d-1]}$ (see Eq. (54)) and a reduced Wigner operator $\hat{T}^{[d]}$. In Fig. 5 we show how to reduce the reduced Wigner operator to a $d \times d$ matrix conditioned on μ and $\mu' + e_{j'}$.

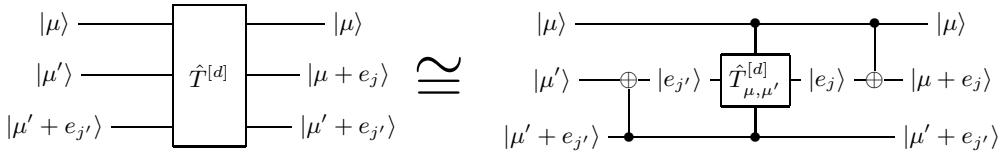


FIG. 5: The reduced Wigner transform $\hat{T}^{[d]}$ can be expressed as a $d \times d$ rotation whose coefficients are controlled by μ and $\mu' + e_{j'}$.

C. Efficient Circuit for the Reduced Wigner Operator

The method of Biedenharn and Louck[46] allows us to compute reduced Wigner coefficients for the cases we are interested in. This will allow us to construct an efficient circuit to implement the controlled-$\hat{T}$ operator to accuracy ϵ using an overhead which scales like $\mathrm{poly}(\log n, d, \log(\epsilon^{-1}))$.

To compute $\hat{T}^{\mu,j,\mu',j'}$, we first introduce the vectors $\tilde{\mu} := \mu + \sum_{j=1}^{d}(d-j)e_j$ and $\tilde{\mu}' := \mu' + \sum_{j=1}^{d-1}(d-1-j)e_j$. Also define $S(j-j')$ to be 1 if $j \geq j'$ and -1 if $j < j'$. Then according to Eq. (38) in Ref [46],

$$\hat{T}^{\mu,j,\mu',j'} = \begin{cases} S(j-j')\left[\frac{\prod_{s\in[d-1]\setminus j}(\tilde{\mu}_j - \tilde{\mu}'_s)\prod_{t\in[d]\setminus j'}(\tilde{\mu}'_{j'} - \tilde{\mu}_t + 1)}{\prod_{s\in[d]\setminus j}(\tilde{\mu}'_j - \tilde{\mu}'_s)\prod_{t\in[d-1]\setminus j'}(\tilde{\mu}'_{j'} - \tilde{\mu}'_t + 1)}\right]^{\frac{1}{2}} & \text{if } j' \in \{1,\ldots,d-1\}. \\ S(j-d)\left[\frac{\prod_{s\in[d-1]\setminus j}(\tilde{\mu}_j - \tilde{\mu}'_s)}{\prod_{s\in[d]\setminus j}(\tilde{\mu}'_j - \tilde{\mu}'_s)}\right]^{\frac{1}{2}} & \text{if } j' = 0. \end{cases} \tag{55}$$

The elements of the partitions here are of size $O(n)$, so the total computation necessary is $\mathrm{poly}(d, \log n)$. Now how do we implement the $\hat{T}^{[d]}$ transform given this expression?

As in the introduction to this section, note that any unitary gate of dimension d can be implemented using a number of two qubit gates polynomial in d[41–43]. The method of this construction is to take a unitary gate of dimension d with *known* matrix elements and then convert this into a series of unitary gates which act non-trivially only on two states. These two state gates can then be constructed using the methods described in [42]. In order to modify this for our work, we calculate, to the specified accuracy ϵ, the elements of the $\hat{T}^{[d]}$ operator, conditional on the μ and $\mu' + e_{j'}$ inputs, perform the decomposition into two qubit gates as described in [41, 42] *online*, and then, conditional on this calculation perform the appropriate controlled two-qubit gates onto the space where $\hat{T}^{[d]}$ will act. Finally this classical computation must be undone to reset any garbage bits created during the classical computation. To produce an accuracy ϵ we need a classical computation of size $\mathrm{poly}(\log(1/\epsilon))$ since we can perform the appropriate controlled rotations with bitwise accuracy.

Putting everything together as depicted in figures 4 and 5 gives a $\mathrm{poly}(d, \log n, \log 1/\epsilon)$ algorithm to reduce $\mathbf{U}_{\rm CG}^{[d]}$ to $\mathbf{U}_{\rm CG}^{[d-1]}$. Naturally this can be applied d times to yield a $\mathrm{poly}(d, \log n, \log 1/\epsilon)$ algorithm for $\mathbf{U}_{\rm CG}^{[d]}$. (We can end the recursion either at $d = 2$, using the construction in [48], or at $d = 1$, where the CG transform simply consists of the map $\mu \to \mu + 1$ for $\mu \in \mathbb{Z}$, or even at $d = 0$, where the CG transform is completely trivial.) We summarize the CG algorithm as follows.

Algorithm: Clebsch-Gordan transform

Inputs: (1) Classical registers d and n. (2) Quantum registers $|\lambda\rangle$ (in any superposition over different $\lambda \in \mathcal{I}_{d,n}$), $|q\rangle \in \mathcal{Q}_\lambda^d$ (expressed as a superposition of GZ basis elements) and $|i\rangle \in \mathbb{C}^d$.
Outputs: (1) Quantum registers $|\lambda\rangle$ (equal to the input), $|j\rangle \in \mathbb{C}^d$ (satisfying $\lambda + e_j \in \mathcal{I}_{d,n+1}$) and $|q'\rangle \in \mathcal{Q}_{\lambda+e_j}^d$.
Runtime: $d^3 \operatorname{poly}(\log n, \log 1/\epsilon)$ to achieve accuracy ϵ.
Procedure:
1. If $d = 1$
2. Then output $|j\rangle := |i\rangle = |1\rangle$ and $|q'\rangle := |q\rangle = |1\rangle$ (i.e. do nothing).
3. Else
4. Unpack $|q\rangle$ into $|\mu'\rangle\left|q_{(d-2)}\right\rangle$, such that $\mu' \in \mathcal{I}_{d,m}$, $m \leq n$, $\mu' \precsim \mu$ and $\left|q_{(d-2)}\right\rangle \in \mathcal{Q}_{\mu'}^{d-1}$.
5. If $i < d$
6. Then perform the CG transform with inputs $(d-1, m, |\mu'\rangle, \left|q_{(d-2)}\right\rangle, |i\rangle)$ and outputs $(|\mu'\rangle, |j'\rangle, \left|q'_{(d-2)}\right\rangle)$.
7. Else (if $i = d$)
8. Replace $|i\rangle = |d\rangle$ with $|j'\rangle := |0\rangle$ and set $\left|q'_{(d-2)}\right\rangle := \left|q'_{(d-2)}\right\rangle$.
9. End. (Now $i \in \{1, \ldots, d\}$ has been replaced by $j \in \{0, \ldots, d-1\}$.)
10. Map $|\mu'\rangle|j'\rangle$ to $|\mu' + e_{j'}\rangle|j'\rangle$.
11. Conditioned on μ and $\mu' + e'_j$, calculate the gate sequence necessary to implement $\hat{T}^{[d]}$, which inputs $|j'\rangle$ and outputs $|j\rangle$.
12. Execute this gate sequence, implementing $\hat{T}^{[d]}$.
13. Undo the computation from **11**.
14. Combine $|\mu' + e_{j'}\rangle$ and $\left|q'_{(d-2)}\right\rangle$ to form $|q'\rangle$.
15. End.

Finally, in Sec. IV we described how n CG transforms can be used to perform the Schur transform, so that $\mathbf{U}_{\text{Sch}}$ can be implemented in time $n \cdot \operatorname{poly}(d, \log n, \log 1/\epsilon)$, optionally plus an additional $\operatorname{poly}(n)$ time to compress the $|p\rangle$ register.

VI. CONCLUSION

We have taken on the challenge of implementing a circuit which performs the Schur transform. This transform, used ubiquitously[19–30] in quantum information theory, represents an important new transformation for quantum information science. The key ingredients in the construction of this circuit were the relationship between Wigner operators and reduced Wigner operators and an efficient classical algorithm for the calculation of the matrix elements of the reduced Wigner operators. This extends our construction from [48] where we constructed the Schur transform for n qubits ($d = 2$). Our construction has a running time which is polynomial in dimension, d, number of qudits, n, and accuracy, $\log(1/\epsilon$. We have thus made practical the large set of quantum information protocols whose computational efficiency has, prior to our work, been uncertain.

For some applications, it is not necessary to perform the full Schur transform, but instead to only be able to perform a projective measurement onto the different Schur subspaces. In part II, we consider a quantum circuit, based on Kitaev's phase estimation algorithm[49], for this task. We further generalize this algorithm to a circuit which is applicable to any nonabelian finite group. Our algorithm is efficient if there exists an efficient quantum circuit for the Fourier transform over this group[33, 50] and represents an ideal way to efficiently deal with situations where quantum states possess symmetries corresponding to some finite group. Further in part II we discuss relationships between the Schur transform and the Fourier transform over the symmetric group.

Finally, we will conclude with some open problems suggested by our construction of the Schur transform. The first interesting question which arises from our work is whether Clebsch-Gordan transforms for other groups can be efficiently constructed. We suspect that for many finite groups, even when dealing with representations which are of dimension d, that their Clebsch-Gordan transforms can be constructed using circuits of size polynomial in $\log(d)$. Our intuition for this claim comes from the construction of quantum Fourier transforms over finite groups[33, 50]. A second question is that while the Schur transform is used frequently in quantum information theory, it has thus far not seen used in the field of quantum algorithms. Kuperberg's[51] subexponential algorithm for the dihedral hidden subgroup problem makes use of the effect

of the Clebsch-Gordan series for the dihedral group. Does the Clebsch-Gordan series for $\mathcal{U}_d$ produce any similar speedup for the appropriately defined hidden subgroup problem on $\mathcal{U}_d$?

Acknowledgments: This work was partially funded by the NSF Institute for Quantum Information under grant number EIA-0086048. AWH acknowledges partial support from the NSA and ARDA under ARO contract DAAD19-01-1-06.

---

[1] C. H. Bennett and S. J. Wiesner. Communication via one- and two-particle opeartors on einstein-podolsky-rosen states. *Phys. Rev. Lett.*, 69:2881–2884, 1992.

[2] C. H. Bennett, G. Brassard, C. Crépeau, R. Jozsa, A. Peres, and W. K. Wootters. Teleporting an unknown quantum state via dual classical and einstein-podolsky-rosen channels. *Phys. Rev. Lett.*, 70:1895–1899, 1993.

[3] B. Schumacher. Quantum coding. *Phys. Rev. A*, 51:2738–2747, 1995.

[4] C. H. Bennett, D. P. DiVincenzo, J. A. Smolin, and W. K. Wooters. Mixed state entanglement and quantum error correction. *Phys. Rev. A*, 52:3824–3851, 1996.

[5] C. H. Bennett, H. J. Bernstein, S. Popescu, and B. Schumacher. Concentrating partial entanglement by local operations. *Phys. Rev. A*, 53:2046–2052, 1996.

[6] I. Devetak and P. W. Shor. The capacity of a quantum channel for simultaneous transmission of classical and quantum information. quant-ph/0311131, 2003.

[7] A. S. Holevo. The capacity of the quantum channel with general signal states. *IEEE Trans. Inf. Theory*, 44, 1998. quant-ph/9611023.

[8] B. Schumacher and M. D. Westmoreland. Sending classical information via noisy quantum channels. *Phys. Rev. A*, 56, 1997.

[9] C. H. Bennett, P. Hayden, D. W. Leung, P. W. Shor, and A. J. Winter. Remote preparation of quantum states. *IEEE Trans. Inf. Theory*, 51(1):56–74, 2005. quant-ph/0307100.

[10] I. Devetak and A.J. Winter. Relating quantum privacy and quantum coherence: an operational approach. *Phys. Rev. Lett.*, 93, 2004. quant-ph/0307053.

[11] R. Jozsa and B. Schumacher. A new proof of the quantum noiseless coding theorem. *J. Mod. Opt.*, 41:2343, 1994.

[12] R. Cleve and D.P. DiVincenzo. Schumacher's quantum data compression as a quantum computation. *Phys. Rev. A*, 54(4):2636–2650, 1996. quant-ph/9603009.

[13] P. Kaye and M. Mosca. Quantum networks for concentrating entanglement. *J. Phys. A*, 34:6939–6948, 2001. quant-ph/0101009.

[14] C. H. Bennett and G. Brassard. Quantum cryptography: Public key distribution and coin tossing. In *Proceedings of IEEE International Conference on Computers, Systems, and Signal Processing*, pages 175–179. IEEE, 1984.

[15] P. W. Shor. Fault tolerant quantum computation. In *Proceedings of the 37th Symposium on the Foundations of Computer Science*, pages 56–65, Los Alamitos, CA, 1996. IEEE.

[16] D. Gottesman and I. L. Chuang. Demonstrating the viability of universal quantum computation using teleportation and single-qubit operations. *Nature*, 402:390–393, 1999.

[17] E. Knill. Fault-tolerant postselected quantum computation: Schemes. quant-ph/0402171, 2004.

[18] A. Ambainis and A. Smith. Small pseudo-random families of matrices: Derandomizing approximate quantum encryption. In K. Jansen, S. Khanna, J.D.P. Rolim, and D. Ron, editors, *APPROX-RANDOM*, volume 3122 of *Lecture Notes in Computer Science*, pages 249–260. Springer, 2004. quant-ph/0404075.

[19] M. Keyl and R. F. Werner. Estimating the spectrum of a density operator. *Phys. Rev. A*, 64:052311, 2001. quant-ph/0102027.

[20] R. Gill and S. Massar. State estimation for large ensembles. *Phys. Rev. A*, 61:042312, 2002. quant-ph/9902063.

[21] G. Vidal, J.I. Latorre, P. Pascual, and R. Tarrach. Optimal minimal measurements of mixed states. *Phys. Rev. A*, 60:126, 1999. quant-ph/9812068.

[22] M. Hayashi and K. Matsumoto. Universal distortion-free entanglement concentration, 2002. quant-ph/0209030.

[23] M. Hayashi and K. Matsumoto. Quantum universal variable-length source coding. *Phys. Rev. A*, 66(2):022311, 2002. quant-ph/0202001.

[24] M. Hayashi and K. Matsumoto. Simple construction of quantum universal variable-length source coding. *Quantum Inform. Compu.*, 2:519–529, 2002. quant-ph/0209124.

[25] M. Hayashi. Optimal sequence of quantum measurements in the sense of stein's lemma in quantum hypothesis testing. *J. Phys. A*, 35:10759–10773, 2002. quant-ph/0208020.

[26] P. Zanardi and M. Rasetti. Error avoiding quantum codes. *Mod. Phys. Lett. B*, 11(25):1085–1093, 1997.

[27] E. Knill, R. Laflamme, and L. Viola. Theory of quantum error correction for general noise. *Phys. Rev. Lett.*, 84:2525–2528, 2000.

[28] J. Kempe, D. Bacon, D. A. Lidar, and K. B. Whaley. Theory of decoherence-free fault-tolerant quantum computation. *Phys. Rev. A*, 63:042307–1–042307–29, 2001. quant-ph/0004064.

[29] D. Bacon. *Decoherence, Control, and Symmetry in Quantum Computers*. PhD thesis, University of Calfornia at Berkeley, Berkeley, CA, 2001. quant-ph/0305025.
[30] S.D. Bartlett, T. Rudolph, and R.W. Spekkens. Classical and quantum communication without a shared reference frame. *Phys. Rev. Lett.*, 91:027901, 2003.
[31] I.M. Gelfand and M.L. Zetlin. Matrix elements for the unitary groups. *Dokl. Akad. Nauk.*, 71:825, 1950.
[32] G. D. James and A. Kerber. *The representation theory of the symmetric group*. Addison-Wesley, Reading, Mass., 1981.
[33] C. Moore, D. Rockmore, and A. Russell. Generic quantum fourier transforms. quant-ph/0304064, 2003.
[34] M. Artin. *Algebra*. Prentice Hall, New Jersey, 1995.
[35] R. Goodman and N.R. Wallach. *Representations and Invariants of the Classical Groups*. Cambridge University Press, 1998.
[36] W. Fulton and J. Harris. *Representation Theory – A First Course*. Springer–Verlag, 1991.
[37] J. Chen, J. Ping, and F. Want. *Group Representation Theory for Physicists*. World Scientific, New Jersey, 2002.
[38] H. Georgi. *Lie Algebras in Particle Physics*. Perseus Books Group, 1999.
[39] J. D. Louck. Recent progress toward a theory of tensor operators in unitary groups. *Am. J. Phys.*, 38(1):3, 1970.
[40] V.V. Shende, S.S. Bullock, and I.L. Markov. Synthesis of quantum logic circuits, 2004. quant-ph/0406176.
[41] M. Reck, A. Zeilinger, H.J. Bernstein, and P. Bertani. Experimental realization of any discrete unitary operator. *Phys. Rev. Lett.*, 73:58–61, 1994.
[42] A. Barenco. A universal two-bit gate for quantum computation. *Proc. Roy. Soc. London Ser. A*, 449:679–683, 1995.
[43] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quatum Information*. Cambridge University Press, New York, 2000.
[44] C.M. Dawson and M.A. Nielsen. The solovay-kitaev algorithm. quant-ph/0505030, 2005.
[45] A. Yu Kitaev, A. H. Shen, and M. N. Vyalyi. *Classical and Quantum Computation*, volume 47 of *Graduate Studies in Mathematics*. AMS, 2002.
[46] L. C. Biedenharn and J. D. Louck. A pattern calculus for tensor operators in the unitary groups. *Commun. Math. Phys.*, 8:89–131, 1968.
[47] A. Messiah. *Quantum Mechanics, Vol. 2*, chapter Representation of Irreducible Tensor Operators: Wigner-Eckart Theorem, pages 573–575. North-Holland, Amsterdam, Netherlands, 1962.
[48] D. Bacon, I. Chuang, and A. Harrow. Efficient quantum circuits for quantum information theory. quant-ph/0407082, 2004.
[49] A. Kitaev. Quantum measurements and the abelian stabilizer problem. quant-ph/9511026.
[50] R. Beals. Quantum computation of fourier transforms over symmetric groups. In *Proceedings of the 29th Annual ACM Symposium on Theory of Computing*, pages 48–53, New York, NY, May 1997. ACM Press.
[51] G. Kuperberg. A subexponential-time quantum algorithm for the dihedral hidden subgroup. quant-ph/0302112, 2003.