Group-covariant extreme and quasiextreme channels

Laleh Memarzadeh [1,*] and Barry C. Sanders [2,†]
[1]*Department of Physics, Sharif University of Technology, Tehran 11365-9161, Iran*
[2]*Institute for Quantum Science and Technology, University of Calgary, Calgary, Alberta, Canada T2N 1N4*

(Received 5 May 2021; accepted 27 July 2022; published 13 September 2022)

Constructing all extreme instances of the set of completely positive trace-preserving (CPTP) maps, i.e., quantum channels, is a challenging and valuable open problem in quantum information theory. Here we introduce a systematic approach that, despite the lack of knowledge about the full parametrization of the set of CPTP maps on arbitrary Hilbert-spaced dimension, enables us to construct exactly those extreme channels that are covariant with respect to a finite discrete group or a compact connected Lie group. Innovative labeling of quantum channels by group representations enables us to identify the subset of group-covariant channels whose elements are group-covariant generalized-extreme channels. Furthermore, we exploit essentials of group representation theory to introduce equivalence classes for the labels and also partition the set of group-covariant channels. As a result, we show that it is enough to construct one representative of each partition. We construct Kraus operators for group-covariant generalized-extreme channels by solving systems of linear and quadratic equations for all candidates satisfying the necessary condition for being group-covariant generalized-extreme channels. Deciding whether these constructed instances are extreme or quasiextreme is accomplished by solving a system of linear equations. Proper labeling and partitioning the set of group-covariant channels leads to a novel systematic, algorithmic approach for constructing the entire subset of group-covariant extreme channels. We formalize the problem of constructing and classifying group-covariant generalized extreme channels, thereby yielding an algorithmic approach to solving, which we express as pseudocode. To illustrate the application and value of our method, we solve for explicit examples of group-covariant extreme channels. With unbounded computational resources to execute our algorithm, our method always delivers a description of an extreme channel for any finite-dimensional Hilbert space and furthermore guarantees a description of a group-covariant extreme channel for any dimension and for any finite-discrete or compact connected Lie group if such an extreme channel exists.

DOI: 10.1103/PhysRevResearch.4.033206

I. INTRODUCTION

Quantum channels, which are completely positive trace-preserving linear maps belonging to the set of linear maps on Banach spaces of operators, represent the most general allowed form of quantum dynamics [1–3], and they form a convex subset. Characterizing quantum channels is important for representing general dynamics and for modeling decoherence in quantum systems [4]. Based on such characterizations, efficient simulation of quantum dynamics becomes feasible. Another importance of characterizing the whole set of quantum channels is to describe general quantum communication channels and to analyze rates of reliable classical and quantum information that can be communicated through quantum channels [5,6].

As full characterization of convex sets is feasible by just knowing the extreme points of a convex set, full channel characterization can be accomplished by determining only the small subset comprising all extreme channels. The conundrum is that extreme channels can be determined by knowing the convex set and vice versa. For quantum channels acting on d-dimensional Hilbert space, necessary and sufficient conditions for a channel to be extreme are described in [7] using similar arguments to those presented in [8]. For qubit channels, full characterization of extreme points is known [9–12]. On the other hand, for Hilbert-space dimension $d > 2$ (qubit channels), neither extreme channels nor full characterizations of the set of quantum channels are known

Here we advance the understanding and characterization of extreme channels by treating channels restricted by symmetries, which are specified by finite discrete or compact connected Lie groups. Compactness ensures that the connected Lie group has finite-dimensional representations. We exploit this symmetry to construct exact forms of extreme channels. Specifically, for any d, we develop an algorithmic approach to derive exactly a subset of extreme points of the set of channels that have certain specified symmetries.

Only special types of channels have been fully characterized to date: a set of qubit channels including all extreme points [9–12] and some extreme points for the set of unital

*memarzadeh@sharif.edu
†sandersb@ucalgary.ca; http://iqst.ca/people/peoplepage.php?id=4

Published by the American Physical Society under the terms of the Creative Commons Attribution 4.0 International license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

channels [13,14], which are quantum counterparts of classical bistochastic processes. Studying unital channels is useful for investigating similarities between classical and quantum processes, such as establishing a quantum version of Birkhoff's theorem [13] and proving additivity or superadditivity of quantities relevant to the communication capacity of channels [15–17]. Fully characterizing quantum channels is quite challenging, which necessitates tackling restricted cases such as unitality.

Characterizing quantum channels has an important practical application to quantum simulation [18–20]. Quantum simulation is an important application of quantum computing and typically is studied for Hamiltonian-generated unitary evolution [19,21–25], but, as general evolution is described by quantum channels, a fully developed theory of quantum simulation could be based on simulating quantum channels. Whereas Hamiltonian simulation exploits notions such as the Solovay-Kitaev theorem [26,27] for gate decomposition and sparseness of Hamiltonians [22,23], direct quantum simulation of channels is challenging. However, some progress has been made by exploiting knowledge of extreme channels. Decomposing the single-qubit channel has been explored theoretically [28] and experimentally [29] and exploits properties of extreme channels. Extreme channels are valuable as well for qudit-channel decomposition [30], including for dimension-altering channels [31], and for decomposing m-qubit to n-qubit channels [32]. This latter result [32] emphasizes the importance of their constructive approach to channel decomposition, which guarantees success of the channel-decomposition procedure. Those authors contrast their constructive approach to the qudit-channel decomposition approach [30], which is provably not guaranteed to succeed based on an insufficient number of parameters. These theoretical [28,31,32] and experimental [29] advances point to the importance of determining extreme channels to make quantum-channel simulation efficient or at least tractable.

Solving for extreme channels is currently restricted to particular examples of unital channels, whereas our goal is to establish a systematic, algorithmic approach to constructing extreme channels that are group-covariant [33], whether unital or not. Beginning with the name of a finite-discrete group or a compact connected Lie group and d, we exploit the possibility of being able to look up all inequivalent irreducible representations (irreps) of the group in order to be able to construct all possible inequivalent d-dimensional representations of the group by direct sums of inequivalent group irreps. For any two d-dimensional inequivalent representations of the group and any inequivalent group irrep with dimension less than or equal to d, we solve a set of linear equations, obtained by applying the group-covariance constraint, to construct Kraus representations of corresponding group-covariant generalized extreme channels. Our method exploits the full power of representation theory; if we employed the obvious brute-force approach instead, we would be solving a set of linear equations for all pairs of d-dimensional representations and for all d^2-dimensional representations of the given group, which is an uncountably infinite number of candidates; instead, provided the representation theory for the group is known, our approach yields only a finite number of candidates, making our approach feasible algorithmically. Once we have identified these candidates, we then test if the obtained generalized extreme channel satisfies the constraint for being extreme. This constraint is expressed as a system of linear equations whose solution reveals whether the obtained generalized extreme channel is extreme or quasiextreme.

Our systematic, algorithmic approach is described by a pseudocode that we define for this purpose. Typically, pseudocode serves as a convenient way of representing the logical flow of a program for implementation on a standard, i.e., Turing-like, computer, but our pseudocode is quite different, serving as a representation of the logical flow for our mathematical approach. Thus, we make it clear that, formally, our pseudocode applies to a real-number model of computing; this model enables us to be rigorous with respect to the logic of our systematic, algorithmic approach to solving group-covariant extreme and quasiextreme channels.

We begin by presenting a full background to our work in Sec. II, including state of the art and methods, and then we proceed to describe our approach in Sec. III. Our results are presented and fully explained in Sec. IV, followed by a discussion of these results in Sec. V. Finally, we conclude in Sec. VI including an outlook on outstanding problems and potential future work.

II. BACKGROUND

In this section, we summarize the pertinent literature and provide basic concepts required for subsequent sections. We begin by discussing quantum channels in Sec. II A, including their Kraus and Choi representations. Section II B is devoted to group-covariant channels and the constraints on the Kraus operators of group-covariant quantum channels.

A. Quantum channels

In this subsection, first we review the definition of quantum channels. We then review Kraus representation and Choi matrix representations for quantum channels. Following that, we recall the definition of specific subsets of quantum channels, namely extreme channels, generalized-extreme channels, and quasiextreme channels.

1. Channel representation

For $\mathscr{H}$ a complex finite-dimensional Hilbert space and $\mathcal{L}(\mathscr{H})$ the space of linear operators acting on $\mathscr{H}$, density operators $\{\rho\}$ are positive trace-class operators on $\mathscr{H}$, i.e., they belong to the subset of $\mathcal{L}(\mathscr{H})$ denoted by

$$\mathcal{T}(\mathscr{H}) = \{\rho \in \mathcal{L}(\mathscr{H}) | \rho \geqslant 0, \mathrm{tr}(\rho) = 1\}. \tag{1}$$

For our purposes, the trace of the density operator is unity. A quantum channel is any completely positive trace-preserving map $\Phi : \mathcal{T}(\mathscr{H}) \to \mathcal{T}(\mathscr{H})$.

For $\mathscr{H}$ restricted to finite dimension d, i.e.,

$$d := \dim \mathscr{H}, \tag{2}$$

every quantum channel can be expressed as

$$\Phi(\bullet) = \sum_{k=1}^{K} A_k \bullet A_k^\dagger, \ \bullet \in \mathcal{T}(\mathscr{H}), \ K \leqslant d^2. \tag{3}$$

This expression is subject to the trace-preserving constraint

$$\Xi = \mathbb{1} \tag{4}$$

for

$$\Xi := \sum_{k=1}^{K} A_k^\dagger A_k, \tag{5}$$

which can be nondiagonal. Here the nonzero linear operators $A_k \in \mathcal{L}(\mathscr{H})$ are called Kraus operators, with each Kraus operator expressible as a $d \times d$ complex matrix whose entries are

$$(A_k)_{ij} := \langle e_i | A_k | e_j \rangle \in \mathbb{C},\ i+1, j+1 \in [d] := \{1, 2, \ldots, d\} \tag{6}$$

for $\{|e_i\rangle\}$ an orthonormal basis of finite-dimensional $\mathscr{H}$, where i goes from 0 to $d-1$. In summary, Φ can be represented by the set $\{A_k\}_{k\in[K]}$ with each A_k comprising d^2 complex-valued matrix elements. Therefore, the channel is described by up to Kd^2 complex-valued parameters.

Remark 1. For $d = 0$, the only vector in Hilbert space is zero, which has zero norm. Hence there is no allowed state in this case. For $d = 1$, only one normalized state exists, which forms the normal basis for the Hilbert space. Kraus operators of this channel are proportional to the projector onto the basis of the space satisfying the trace-preserving condition.

The set of Kraus operators describing a map Φ is unique up to an isometry [27]. For a given map Φ, the minimum number of Kraus operators, called the Choi rank, equals the rank of the Choi operator

$$C_\Phi := \frac{1}{d}(\Phi \otimes \mathbb{1})(|\Psi\rangle\langle\Psi|),\ |\Psi\rangle := \frac{1}{\sqrt{d}} \sum_{i=0}^{d-1} |e_i, e_i\rangle \tag{7}$$

and

$$(C_\Phi)_{mn,pq} = \langle e_m, e_n | C_\Phi | e_p, e_q \rangle = \frac{1}{d} \langle e_m | \Phi(|e_n\rangle\langle e_q|) | e_p \rangle \tag{8}$$

is the Choi matrix. Typically, the operator (7) is called the Choi matrix, but, due to our algorithmic approach, we need to be extra careful in distinguishing operators from their matrix representations, and we denote matrices of size $m \times n$ with entries drawn from the field $\mathbb{F}$ by $\mathcal{M}_{m\times n}(\mathbb{F})$. Choi showed that a minimal set of Kraus operators can be obtained from the eigenvectors of the Choi matrix with nonzero eigenvalues [8].

2. Extreme channels

The set of quantum channels S_Φ is convex and thus has extreme points.

Definition 1. Extreme points of the convex set S_Φ are called extreme channels. Extreme channels are channels that cannot be written as a convex combination of any other two distinct channels in a nontrivial way.

We denote the set of extreme channels by $S_{\Phi_{\text{ext}}} \subset S_\Phi$. Despite the importance of extreme channels, characterization of extreme channels is unknown except for the special case of $d = 2$ [9–11]. Results beyond $d = 2$ are restricted to characterizing extreme points of the set of unital channels [$\Phi(\mathbb{1}) \equiv \mathbb{1}$], which are not necessarily extreme points of the set of all channels [13].

An important theorem on extreme channels specifies necessary and sufficient conditions for a channel to be an extreme one [8]:

Theorem 1. A channel represented by a set of Kraus operators $\{A_k\}_{k\in[K]}$ is extreme if and only if (iff) the set of operators

$$\mathcal{S} := \{A_k^\dagger A_l\}_{k,l\in[K]} \tag{9}$$

is linearly independent [7,8].

Thus, the number K of Kraus operators of an extreme channel has upper bound

$$K \leqslant d. \tag{10}$$

Therefore, the Choi rank of an extreme channel is bounded by d.

Clearly not all channels with Choi rank satisfying inequality (10) are extreme channels, but such channels are interesting as well [11,34]. The fact that other channels are interesting leads to defining two further important subsets of quantum channels. One subset is known as generalized-extreme channels and the other subset is known as quasiextreme channels, which we now define.

Definition 2. Channels with Choi rank not exceeding d are called generalized-extreme channels [11] and the set of generalized-extreme channels is denoted by $S_{\Phi_{\text{gen}}}$.

Definition 3. Generalized-extreme channels that are not extreme are called quasiextreme channels [11], and the set of quasiextreme channels is denoted by $S_{\Phi_{\text{qe}}}$.

Remark 2. Extreme and quasiextreme channels are mutually exclusive: $S_{\Phi_{\text{gen}}} \cap S_{\Phi_{\text{qe}}} = \emptyset$.

B. Group-covariant channels

In this subsection, we elaborate on a specific class of channels, namely group-covariant channels. First we recall the definition of equivalent channels. Based on this definition we explain that a group-covariant channel is a channel (3) with the additional property that the channel's action is invariant under pre- and postunitary conjugations that are described by group representations. Then we explain the constraints on the Kraus operators of group-covariant channels and how two group-covariant channels under the same group, with respect to equivalent representations of the group, are equivalent. Group-covariant channels have been studied in the context of channel capacity [35–39], extreme points of unital channels [13], and channel characterization [40–42], and complementarity and additivity properties of various covariant channels are discussed in [43].

Definition 4. A channel Φ is unitarily equivalent to channel Φ', denoted by $\Phi \sim \Phi'$, if there exist d-dimensional unitary operators U and V such that

$$\Phi \sim \Phi' : \Phi = \mathcal{T}_U \circ \Phi' \circ \mathcal{T}_V, \tag{11}$$

where $\circ$ denotes composition of maps, and

$$\mathcal{T}_Q : \mathcal{L}(\mathscr{H}) \to \mathcal{L}(\mathscr{H}) : \bullet \mapsto Q \bullet Q^\dagger,\ Q \in \mathcal{L}(\mathscr{H}). \tag{12}$$

The equivalence relation $\sim$ (11) partitions the set of all quantum channels for given Hilbert-space dimension d into

equivalence classes of channels. Any quantum channel is equivalent to itself for $U = V = \mathbb{1}$. However, for some channels, each channel is equivalent to itself even if U and V are not identity operators. These channels, with symmetric properties, are group-covariant channels defined as below.

Definition 5. For a finite discrete group or a compact connected Lie group denoted $\mathcal{G}$, with a pair of unitary representations [33]

$$D^{(1)}, D^{(2)} \in \mathcal{L}(\mathscr{H}), \tag{13}$$

a channel Φ is group-covariant with respect to representations $D^{(1)}, D^{(2)}$ if

$$\mathcal{T}_{D^{(2)}(g)} \circ \Phi \circ \mathcal{T}_{D^{(1)}(g)} = \Phi \ \forall g \in \mathcal{G}, \tag{14}$$

where $\mathcal{T}_Q$ is defined in Eq. (12).

Suppose channel Φ, represented by Kraus operators $\{A_k\}_{k\in[K]}$ (3), is group-covariant with respect to two d-dimensional unitary representations of the group $\mathcal{G}$, namely $D^{(1)}$ and $D^{(2)}$. Then Eq. (14) implies that

$$D^{(2)^\dagger}(g)A_k D^{(1)}(g) = \sum_{l=1}^{K} \Omega_{kl}(g)A_l, \ \forall k \in [K], \ \forall g \in \mathcal{G}, \tag{15}$$

subject to the trace-preserving condition (4), for Ω a unitary representation of the group $\mathcal{G}$ on any K-dimensional unitary space [35,40]. The dimension of Ω is equal to the number of Kraus operators describing the channel [40].

Remark 3. The group-covariant channel Φ (14) with respect to $D^{(1)}$ and $D^{(2)}$ is not necessarily unique, and each distinct Φ should be suitably labeled. The role of Ω (15) is to label distinct group-covariant channels with respect to $D^{(1)}$ and $D^{(2)}$, i.e., we can write Φ_Ω.

Remark 4. If Ω is a reducible representation of the group, then the group-covariant channel with Kraus operators satisfying Eq. (15) is a convex combination of other channels that are also group-covariant with respect to $D^{(1)}$ and $D^{(2)}$ [40].

Remark 5. Let Φ (3) be group-covariant with respect to representations $D^{(1)}$ and $D^{(2)}$. Let $D'^{(1)}$ and $D'^{(2)}$ be two representations of the same group. Suppose these two representations are, respectively, unitarily equivalent to $D^{(1)}$ and $D^{(2)}$ so

$$U_i D^{(i)}(g) U_i^\dagger = D'^{(i)}(g), \ \forall g \in \mathcal{G} \tag{16}$$

for $\{U_i\}_{i\in[2]}$ d-dimensional unitary operators. Then channel Φ', described by Kraus operators $\{A'_k = U_2 A_k U_1^\dagger\}_{k\in[K]}$, is group-covariant with respect to representations $D'^{(i)}$'s [40],

$$\Phi' = \mathcal{T}_{U_2} \circ \Phi \circ \mathcal{T}_{U_1^\dagger}. \tag{17}$$

That is, according to Definition 4, these channels are unitarily equivalent.

Equality (15) holds iff this equality holds for all generators of the group. Consider any finite discrete group $\mathcal{G}$ generated by a subset of $\mathcal{G}$, namely

$$S = \{g_1, g_2, \ldots, g_r\}, \ r := \mathrm{rank}(\mathcal{G}). \tag{18}$$

Then Eq. (15) is satisfied for all $g \in \mathcal{G}$ if Eq. (15) is satisfied for all $g_i \in S$. Now consider any compact connected Lie group $\mathcal{G}$ with corresponding Lie algebra

$$\mathfrak{g} \in \mathrm{span}\{T_n\}, \ n \in [\nu], \ \nu := \dim \mathfrak{g} \tag{19}$$

for T_n a generator. Then Eq. (15) can be expressed in terms of the representations of $\mathfrak{g}$ [40],

$$D^{(1)}(T_n)A_k - A_k D^{(2)}(T_n) = \sum_{l=1}^{K} \Omega_{kl}(T_n)A_l, \ \forall T_n, \tag{20}$$

where $D^{(1)}$ and $D^{(2)}$ are d-dimensional Hermitian representations of the algebra $\mathfrak{g}$, and Ω is a K-dimensional Hermitian representation of the Lie algebra $\mathfrak{g}$.

Remark 6. If $D \equiv D^{(1)} = D^{(2)}$, Eq. (20) simplifies to the commutator

$$[D(T_n), A_k] = \sum_{l=1}^{K} \Omega(T_n)A_l \ \forall T_n. \tag{21}$$

In this section, we have reviewed the main properties of quantum channels and group-covariant channels. Now we use these results for our aim of constructing group-covariant extreme points of the set of quantum channels [40].

III. APPROACH

In this section, based on the background provided in Sec. II, we introduce our systematic approach to construct group-covariant generalized-extreme channels where the corresponding group has unitary representation. In Sec. A 1, we describe the problem as a computational problem. In Sec. III B, we present a constructing subset of extreme channels that are group-covariant. Section III C is devoted to an algorithmic approach for solving the problem (in Appendix A 3 we explain our approach to pseudocode).

A. Formal problems

The purpose of this subsection is to formalize problems of constructing group-covariant extreme channels over finite-dimensional Hilbert space. Specifically, the construction of these channels is achieved by obtaining the exact Kraus representations for generalized-extreme group-covariant channels, and we deal with both finite discrete and compact connected Lie groups. Our final problem concerns deciding whether a channel is either extreme or not.

We formulate problems by specifying the inputs and outputs, and the problem in each case is to map the inputs to the outputs, although the problem formulation does not strictly use this language. Each problem is thus a task that needs to be performed to construct group-covariant extreme and quasiextreme channels.

We now formulate and explain our three problems. The first two problems concern construction of group-covariant generalized-extreme channels discussed in Sec. II A, first for finite discrete groups and then for compact connected Lie groups. For our first problem, the input comprises the name of the finite discrete group and the dimension of the Hilbert space. The output of the first problem is the set of $d \times d$ matrix representations of Kraus operators for group-covariant generalized extreme channels. Our second problem is similar to the first, with the input comprising the name of the group, but in this case a compact connected Lie group; otherwise the statement is the same in that the input includes the dimension of the Hilbert space, and the output is the set of $d \times d$ matrix

representations of Kraus operators for group-covariant generalized extreme channels. These first two problems are now given.

Problem 1. Construct all exact d-dimensional Kraus operators of all group-covariant generalized-extreme channels for any finite discrete group.

Problem 2. Construct all exact d-dimensional Kraus representations of all group-covariant generalized-extreme channels for any compact connected Lie group.

The third problem, which is a decision problem, accepts matrix representations of Kraus operators for a channels as input. The problem is to solve whether the input channel is extreme or not, and yields this answer as a single-bit output. Therefore, in our case in which the inputs are the matrix representations of Kraus operators for a generalized-extreme channel, the problem solves whether the input channel is extreme or quasiextreme

Problem 3. Decide whether a given set of d-dimensional Kraus operators of a quantum channel describes an extreme channel or not.

Now that we have three well-posed computational problems, albeit permitting real and complex numbers, we proceed to describe our approach for designing a proper procedure and its presentation as an algorithm that solves the problem.

B. Algebraic approach

Building on the formal problems posed in Sec. A 1, we explain our approach for constructing generalized-extreme group-covariant channels. First in Sec. III B 1 we transform the relation between Kraus operators of a group-covariant channel to systems of linear equations. Then in Sec. III B 2 we discuss how to construct generalized-extreme group-covariant channels. Finally, in Sec. III B 3 we explain how defining equivalent channels helps to construct generalized-extreme group-covariant channels, which only needs to be done for group-covariant channels with respect to inequivalent representations of the group. Discussions in Secs. III B 2 and III B 3 enables us to solve Eq. (15) just for the representative of each class of group-covariant generalized extreme channels, instead of employing a brute-force approach that is constructing all group-covariant channels and then deciding which one is extreme.

1. Solving a system of linear equations for Kraus operators of a group-covariant channel

In this subsubsection, we convert the relations for Kraus operators representing a group-covariant channel in the finite discrete case (15) and in the compact connected Lie group case (20) into a system of linear equations. The algorithm for solving this system of linear equations can then be solved algorithmically and is amenable to expressing in pseudocode. First, for discrete finite groups and compact connected Lie groups, we express relations between Kraus operators as systems of linear equations. In both cases, we discuss instances for which group-covariant channels do not exist with respect to particular representations $D^{(1)}$, $D^{(2)}$, and Ω.

We denote the vectorized form of a Kraus operator A_k by $\boldsymbol{A}_k$, and matrix elements convert to vector elements according to $(A_k)_{ij} = (\boldsymbol{A}_k)_{id+j}$. Concatenation of vector representations is denoted by $\square$, i.e., $\boldsymbol{A}_k \square \boldsymbol{A}_{k'}$ for the concatenation of vectors representing Kraus operations A_k and $A_{k'}$, respectively. Concatenation of a length K sequence of vectors representing Kraus operators is expressed as

$$\boldsymbol{A} := \square_{k=1}^{K} \boldsymbol{A}_k \in \mathcal{M}_{Kd^2\times 1}(\mathbb{C}), \tag{22}$$

which represents the channel as a vector comprising all of the channel's Kraus operators.

For a group-covariant channel with a finite discrete group, a vector $\boldsymbol{A}$ (22) is obtained by solving linear equations (15) for each $g_i \in S$ (18) and then imposing the trace-preserving constraint (4). We reexpress the linear equations (15) as

$$P(g_i)\boldsymbol{A} = \boldsymbol{0}\ \forall\, i \in [r] \tag{23}$$

for

$$P(g_i) = \bigoplus_{k=1}^{K} (D^{(2)^\dagger}(g_i) \otimes D^{(1)^{\mathrm{T}}}(g_i)) - \Omega(g_i) \otimes \mathbb{1}_{d^2} \in \mathcal{M}_{Kd^2\times Kd^2}(\mathbb{C}) \tag{24}$$

with $\mathbb{1}_{d^2}$ the $d^2 \times d^2$ identity matrix, and T denoting matrix transposition: $(\bullet)^{\mathrm{T}}_{i,j} = (\bullet)_{j,i}$. Equation (23) is a set of systems of homogeneous linear equations. Each of these systems of homogeneous linear equations is labeled in terms of the same three representations of the group, namely $D^{(1)}$, $D^{(2)}$ and Ω, as clearly seen in Eq. (24).

Remark 7. In the trivial case that $D^{(1)} = \mathbb{1}_d = D^{(2)}$ and $\Omega = 1$, then $P(g_i) \equiv 0$ for all i, which implies that $\boldsymbol{A}$ in Eq. (24) is unconstrained and is a vectorized version of just one single Kraus operator.

The solution $\boldsymbol{A}$ (23) belongs to $\ker(P(g_i))$ for all $i \in [r]$. If $\bigcap_{i\in[r]} \ker(P(g_i)) \equiv \{\boldsymbol{0}\}$, then the only solution to Eq. (23) is the trivial solution $\boldsymbol{A} \equiv \boldsymbol{0}$. Such a trivial case arises either if $g_i \in S$ exists such that $\det(P(g_i)) \neq 0$, which means $\ker(P(g_i)) = \{\boldsymbol{0}\}$ for some i, or else $\ker(P(g_i)) \neq \{\boldsymbol{0}\}$ for all $i \in [r]$ but their intersection is zero.

This trivial solution $\boldsymbol{A} \equiv \boldsymbol{0}$ implies nonexistence of a group-covariant channel with respect to group representations $D^{(1)}$, $D^{(2)}$, and Ω. On the other hand, if $\det(P(g_i)) = 0$ for all $g_i \in S$ and $\bigcap_{i\in[r]} \ker(P(g_i)) \neq \{\boldsymbol{0}\}$, then Eq. (23) yields a nontrivial solution $\boldsymbol{A} \in \bigcap_{i\in[r]} \ker(P(g_i))$ with its number of parameters being less than or equal to $\min_{i\in[r]}\{\mathrm{nullity}(P(g_i))\}$. This nontrivial solution $\boldsymbol{A}$ could represent valid Kraus operators of a group-covariant completely positive (CP) map with respect to given $D^{(1)}$, $D^{(2)}$, and Ω and these nontrivial solutions are candidates for solutions for a channel if the trace-preserving condition, discussed below, can be imposed successfully.

These algebraic equations and arguments can be understood geometrically as well, which provides an alternative, and valuable, insight. From Eq. (23) we know that $\boldsymbol{A}$, for each i, is in the kernel of a matrix with Kd^2 rows of Kd^2 elements per row. Each nonzero row of $P(g_i)\boldsymbol{A} = \boldsymbol{0}$ (23) defines a hyperplane in $\mathbb{C}^{Kd^2}$. Thus, the solution to Eq. (23) for each i is a flat, that is, the intersection of Kd^2 hyperplanes obtained from rows of Eq. (23) with the dimension of this intersection being $\mathrm{nullity}(P(g_i))$. According to Eq. (23), $\boldsymbol{A}$ belongs to the intersection of all r flats. If the intersection of these flats is empty, then a group-covariant channel labeled by $D^{(1)}$, $D^{(2)}$, and Ω

does not exist. If the intersection is another flat, its dimension is less than or equal to $\min_{i\in[r]}\{\text{nullity}\,(P(g_i))\}$, which equals the number of parameters in the solution $\boldsymbol{A}$. These solutions represent valid Kraus operators of a group-covariant CP map with respect to given $D^{(1)}$, $D^{(2)}$, and Ω, and next we constrain these solutions by the trace-preserving condition to obtain, if possible, solutions $\boldsymbol{A}$ for a channel.

For a channel, these Kraus operator solutions for CP maps must further satisfy the trace-preserving condition (4), which involves additional equations that restrict the parameter-space domain for Kraus operators. To impose this trace-preserving constraint, we reshape Kraus vectors $\{\boldsymbol{A}_k\}$ back to Kraus matrices $\{A_k\}$. Then we verify if $\{A_k\}$ describe a trace preserving map for a range of complex parameters in $\{A_k\}$. Each diagonal element of Ξ (4) is the sum of modulus square of complex parameters of Kraus operators

$$\langle e_i|\,\Xi\,|e_i\rangle = \Xi_{ii} = \sum_{j=1}^{d}\sum_{k=1}^{K}|(A_k)_{ij}|^2, \tag{25}$$

hence a non-negative real number for $\{|e_i\rangle\}$ the orthonormal basis of $\mathscr{H}$.

To constrain solutions $\boldsymbol{A}$ that yield CP maps, by applying the trace-preserving condition, we test if Ξ is diagonal by checking that all off-diagonal terms for this matrix are zero. If the test shows that Ξ is a d-dimensional diagonal matrix, then we have d equations that depend only on modulus square complex parameters of Kraus operators. The number of parameters varies depending on the particular case. Equations that are linear in modulus square of parameters of Kraus operators can be solved by a linear equation solver for modulus square complex parameters. The solution describes a family of group-covariant channels.

The diagonal-Ξ matrix case for imposing the trace-preserving condition admits a beautiful geometric analogy. Recalling the geometric perspective, Kraus operators of a group-covariant CP map belong to a flat in parameter space. If Ξ is diagonal, the trace-preserving constraint (4) is represented geometrically by at most d hyperspheres embedded in that flat such as a circle embedded in a plane. Thus, the intersection of these hyperspheres determines the parameter domain for which the CP map is trace-preserving, hence a channel.

If Ξ is not diagonal, then, in general, we have $d(d+1)/2$ expressions quadratic in complex parameters. Not all these $d(d+1)/2$ are necessarily mutually independent. Furthermore, the number of parameters can be fewer than the maximum of Kd^2. The off-diagonal case is harder than the diagonal case with respect to imposing the trace-preserving condition, and the geometric perspective is not as helpful, so we discuss the nondiagonal case only from an algebraic perspective. These equations can be solved algorithmically, for example using SimPy discussed in Appendix C 1, but such solvers are not guaranteed to solve nor is a solution known to exist in general.

For compact connected Lie groups, the relation between Kraus operators of the channel (20) is transformed to the set of linear equations

$$Q(T_n)\boldsymbol{A} = \boldsymbol{0}\ \forall n \in [\nu], \tag{26}$$

analogous to Eq. (23) for discrete groups, with $\boldsymbol{A}$ (22) and ν (19). In Eq. (26),

$$Q(T_n) = \bigoplus_{k=1}^{K}(D^{(1)}(T_n)\otimes \mathbb{1}_d - \mathbb{1}_d \otimes D^{(2)}(T_n)^{\mathrm{T}}) - \Omega(T_n)\otimes \mathbb{1}_{d^2} \in \mathcal{M}_{Kd^2\times Kd^2}(\mathbb{C}). \tag{27}$$

Equation (26) is a set of systems of homogeneous linear equations. Each of these systems of homogeneous linear equations is labeled in terms of the same three representations of the group, namely $D^{(1)}$, $D^{(2)}$, and Ω, as clearly seen in Eq. (27). Algebraic and geometrical descriptions of the solution to Eq. (26) are similar to those for the solution of Eq. (23). As for the finite discrete-group case above, the solution $\boldsymbol{A}$ of Eq. (26) belongs to $\bigcap_{n\in[\nu]}\ker(Q(T_n))$. If $\bigcap_{n\in[\nu]}\ker(Q(T_n)) = \{\boldsymbol{0}\}$, then the system of linear equations (26) has only a trivial solution, which means that the group-covariant CP map with respect to $D^{(1)}$, $D^{(2)}$, and Ω does not exist. If, on the other hand, $\ker(Q(T_n)) \neq \{\boldsymbol{0}\}$, then the solution of Eqs. (26) yields all Kraus operators of the group-covariant CP map. If the trace-preserving condition can be imposed successfully, then the group-covariant channel with respect to $D^{(1)}$, $D^{(2)}$, and Ω is obtained following the approach discussed above for discrete groups.

2. *Constructing generalized-extreme group-covariant channels*

In this subsubsection, we explain our approach for constructing group-covariant generalized-extreme channels. First we establish our notation and explain how we label different group-covariant channels. Then we discuss the labels of group-covariant generalized-extreme channels.

To construct a set of group-covariant channels given group $\mathcal{G}$ and Hilbert-space dimension d, one chooses two d-dimensional representations $D^{(1)}$ and $D^{(2)}$ for $\mathcal{G}$. The set of group-covariant channels with respect to $D^{(1)}$ and $D^{(2)}$ is denoted by $\mathcal{W}_{\mathcal{G},D^{(1)},D^{(2)}}$. To construct $\mathcal{W}_{\mathcal{G},D^{(1)},D^{(2)}}$, one selects each Ω from the set of unitary representations for $\mathcal{G}$ and solves the linear equations in Eq. (15) to obtain matrix descriptions of Kraus operators. As explained in Sec. III B 1, solving Eq. (15) is equivalent to solving a set of systems of homogeneous linear equations (23) for the case of finite discrete groups, which is a set of systems of homogeneous linear equations labeled by three representation of the group, namely $D^{(1)}$, $D^{(2)}$, and Ω. Hence, we label the solution to this set of systems of homogeneous linear equations by the same labels and denote this solution by $\Phi_{D^{(1)},D^{(2)},\Omega}$. This approach, involving equivalence between Eqs. (15) and (26) and labeling by $D^{(1)}$, $D^{(2)}$, and Ω, pertains as well to the compact connected Lie group case.

Trivial solutions ($\Phi \equiv 0$) are excluded and each nontrivial solution is denoted by $\Phi_{D^{(1)},D^{(2)},\Omega} \in \mathcal{W}_{\mathcal{G},D^{(1)},D^{(2)}}$. The index Ω labels distinct group-covariant channels with respect to $D^{(1)}$ and $D^{(2)}$, which are general d-dimensional representations of $\mathcal{G}$ including the reducible case. The set of all group-covariant channels is denoted by

$$\mathcal{W}_{\mathcal{G}} := \bigcup_{D^{(1)},D^{(2)}} \mathcal{W}_{\mathcal{G},D^{(1)},D^{(2)}}, \tag{28}$$

where the union is over all d-dimensional representations of $\mathcal{G}$.

The set of indices labeling distinct group-covariant channels with respect to $D^{(1)}$ and $D^{(2)}$ is denoted by

$$\mathcal{F}_{\mathcal{G},D^{(1)},D^{(2)}} := \{\Omega : \Phi_{D^{(1)},D^{(2)},\Omega} \in \mathcal{W}_{\mathcal{G},D^{(1)},D^{(2)}}\}. \tag{29}$$

Then $\mathcal{F}_{\mathcal{G}} := \bigcup_{D^{(1)},D^{(2)}} \mathcal{F}_{\mathcal{G},D^{(1)},D^{(2)}}$ is the set of all labels of group-covariant channels.

Unitary conjugation is an equivalence relation $\approx$ that partitions $\mathcal{F}_{\mathcal{G}}$ to equivalence classes, and each class is denoted by $[\Omega]$. Indeed labels of classes are inequivalent representations of the group. We show that all elements of each class $[\Omega]$ label the same group-covariant channel; hence, instead of labeling distinct group representations by a unitary representation of the group, we use a more appropriate notation, i.e., we label each group-covariant channel by an equivalence class $\Phi_{D^{(1)},D^{(2)},[\Omega]}$.

We denote the set of group-covariant generalized-extreme channels by $\mathcal{W}_{\mathcal{G},\text{gen}} \subset \mathcal{W}_{\mathcal{G}}$, which is a subset of all generalized-extreme channels, $\mathcal{W}_{\mathcal{G},\text{gen}} \subset S_{\Phi_{\text{gen}}}$, and it has two exclusive subsets, namely quasiextreme group-covariant channels and extreme group-covariant channels. This set of channels are, respectively, denoted by $\mathcal{W}_{\mathcal{G},\text{qe}} \subset \mathcal{W}_{\mathcal{G},\text{gen}}$ and $\mathcal{W}_{\mathcal{G},\text{ext}} \subset \mathcal{W}_{\mathcal{G},\text{gen}}$ with $\mathcal{W}_{\mathcal{G},\text{qe}} \bigcap \mathcal{W}_{\mathcal{G},\text{ext}} = \emptyset$. We show that the subset of $\mathcal{F}_{\mathcal{G}}/\approx$, denoted by $\mathcal{F}_{\mathcal{G},\text{gen}}$, which includes all irreducible $[\Omega]$s with dimension not exceeding d, labels all channels in $\mathcal{W}_{\mathcal{G},\text{gen}}$.

Thus, to construct group-covariant generalized-extreme channels, instead of solving Eq. (15) for all Ω which is a brute-force approach, we solve Eq. (15) for all $[\Omega] \in \mathcal{F}_{\mathcal{G},\text{gen}}$ to obtain Kraus operators of all group-covariant generalized-extreme channels. Then by testing whether $\mathcal{S}$ (9) is a set of linearly independent operators, we classify, according to Theorem 1, resultant nontrivial channels into extreme and quasiextreme classes.

3. Equivalent extreme channels

In this subsubsection, we partition the set of group-covariant channels and explain that it is enough to construct one representative of each class, which is more efficient than the direct approach of solving every extreme channel. We discuss that, if an element of a class of equivalent channels as described in Definition 4 is an extreme group-covariant channel, then all elements of that class have that property.

First we show that, according to the equivalence relation in (11), all $\Phi_{D^{(1)},D^{(2)},[\Omega]} \in \mathcal{W}_{\mathcal{G},D^{(1)},D^{(2)}}$ and all $\Phi_{D'^{(1)},D'^{(2)},[\Omega]} \in \mathcal{W}_{\mathcal{G},D'^{(1)},D'^{(2)}}$, with $D^{(i)}$ and $D'^{(i)}$ being two unitarily equivalent representations of $\mathcal{G}$ (16), are equivalent. We denote the set of group-covariant channels with respect to all representations of $\mathcal{G}$ that are equivalent to $D^{(1)}$ and $D^{(2)}$ by $\mathcal{W}_{\mathcal{G},[D^{(1)}],[D^{(2)}]}$, which is an equivalence class. Then we show that if $\Phi_{D'^{(1)},D'^{(2)},[\Omega]} \in \mathcal{W}_{\mathcal{G},[D^{(1)}],[D^{(2)}]}$ is extreme, all channels in $\mathcal{W}_{\mathcal{G},[D^{(1)}],[D^{(2)}]}$ are extreme. Hence, for $[\Omega] \in \mathcal{F}_{\mathcal{G},\text{gen}}$ (defined in Sec. III B 2) if

$$\Phi_{D'^{(1)},D'^{(2)},[\Omega]} \in \mathcal{W}_{\mathcal{G},[D^{(1)}],[D^{(2)}]} \tag{30}$$

is extreme/quasiextreme, then all channels in $\mathcal{W}_{\mathcal{G},[D^{(1)}],[D^{(2)}]}$ are extreme/quasiextreme. Hence, to construct all generalized extreme group-covariant channels, for each $[\Omega] \in \mathcal{F}_{\mathcal{G},\text{gen}}$, we construct $\Phi_{D^{(1)},D^{(2)},[\Omega]}$. Other elements of the classes are derived according to the equivalence relation among quantum channels, which is any arbitrary pre- or postunitary conjugation. This step is repeated for all inequivalent representations $D^{(1)}$ and $D^{(2)}$.

C. Algorithmic approach

In this subsection, we explain the algorithmic approach for constructing group-covariant extreme channels for finite discrete groups and compact connected Lie groups. We commence by constructing the algorithm for solving Problems 1 and 2 in Sec. III C 1. We discuss constructing an algorithm for Problem 3 in Sec. III C 2. We express the algorithm as input and output, including the type of each input and output (such as symbol, integer, real number, bit, character, and so on). In our algorithmic approach, we do not restrict ourselves to discrete mathematics; rather we permit symbols and real- and complex-number entries in the register as we are focused on an algorithmic approach to the problem but not issues of computability or complexity associated with various computational models [44]. In each procedure, we employ required functions from our libraries discussed in Appendix C.

1. Algorithm for solving Problems 1 and 2

In this subsubsection, we present our approach to developing the algorithm for solving Problems 1 and 2. Specifically, we state the input, output, and a brief description of the procedure, with the full explanation of the algorithm in the first algorithm of Sec. IV B.

a. Input: A binary number that flags the type of the group, which is either a finite discrete group or a compact connected Lie group. The name of any finite discrete group or a compact connected Lie group expressed as a character string. Examples of finite discrete group names include $\mathbb{Z}_2$, which is the cyclic group of order 2, or S_3, which is the symmetric group of degree 3. Hilbert-space dimension $d \in \mathbb{Z}^+$ is the other input.

b. Output: A finite number $C \in \mathbb{Z}^+$ is the first output representing the total number of generalized extreme group-covariant channels on d-dimensional Hilbert space, and the second output is the set of Kraus operators for all C channels.

c. Procedure: We import required functions from the libraries discussed in Appendixes C 1 and C 2 and declare necessary variables for the algorithm. Then we solve the system of linear equations (23) and (26), respectively, for finite discrete group (Problem 1) and compact connected Lie group (Problem 2). In either case, we solve the corresponding set of linear equations for each instance $D^{(1)}$, $D^{(2)}$ (inequivalent d-dimensional representations of the group) and Ω (inequivalent irreps of the group with dimension less than or equal to d).

(i) If a nontrivial symbolic solution exists for this instance, then this solution is unique, and the algorithm constructs a set of Kraus operators for this solution and imposes the trace-preserving condition on it, which as explained in Sec. III B 1 is done by first constructing Ξ (5) and second solving $\Xi = \mathbb{1}$. If Ξ is diagonal, then a linear-solver algorithm solves a system of linear equations for modulus squares of symbols for Kraus matrices. If the solution exists, it is stored and the counter C for successful case increments. Then the algorithm proceeds to the next instance. If Ξ is not diagonal, as discussed in Sec. III B 1, a solver such as Python's SimPy aims to solve sets of quadratic equations. If the solution is found within the

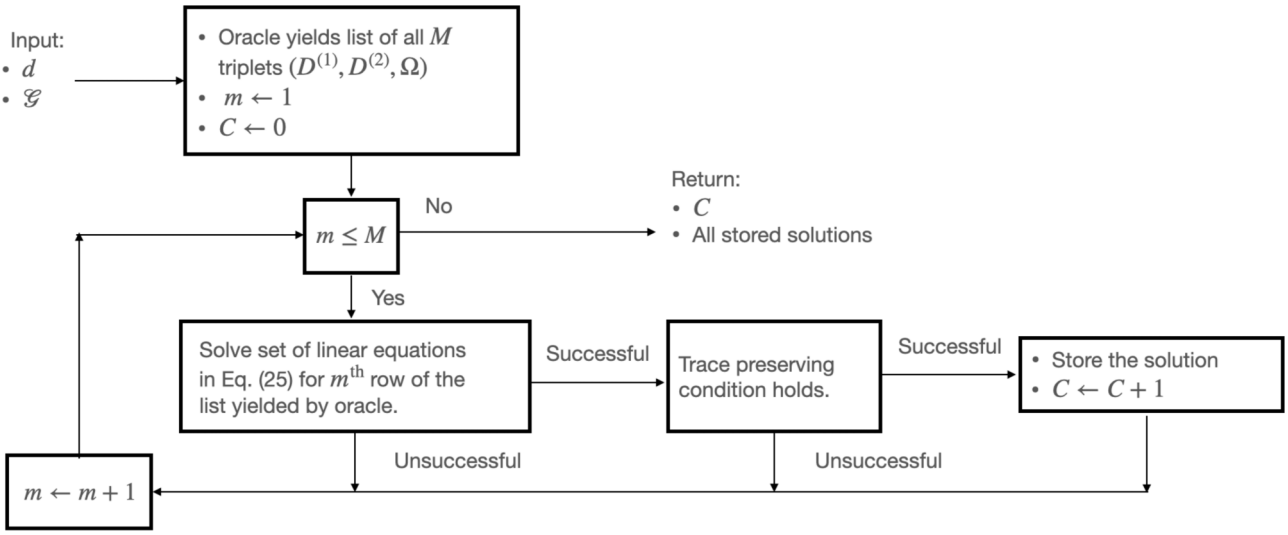

FIG. 1. Schematic representation of the algorithmic approach for solving Problem 1.

allowed time, the solution is written to the register, the counter C increments, and the algorithm proceeds to the next instance.

(ii) If the system of linear equation does not have a nontrivial solution, the algorithm goes to the next incident without incrementing the counter C.

After testing all instances, and generating a parametric solution to a system of linear equations for all instances where this solution exists, an ordered list (with ordering determined by GPROPS defined in Appendix C 2) of all sets of Kraus operators is returned. The algorithm returns finite number C in the output. If $C = 0$, then acceptable channels have not been found. Figure 1 shows a schematic representation of algorithm for solving Problem 1 which is useful for going through the details of pseudocode in Sec. IV B.

2. Algorithm for solving Problem 3

In this subsubsection, we describe inputs and outputs for the algorithm to solve Problem 3. When we present our procedure to determine whether a given channel is extreme or not.

a. Input: A list of K d-dimensional Kraus operators $\{A_{k\in[K]}\}$ with symbolic elements.

b. Output: The single-bit output is $\top$, denoting "true," if the given Kraus operators correspond to an extreme channel; otherwise the output is $\perp$, denoting "false," which means that the Kraus operators do not describe an extreme channel.

c. Procedure: We import required functions from the libraries Appendixes C 1 and C 2 and declare necessary variables for the algorithm. Then, from K d-dimensional symbolic matrices in the input A_n, $n \in [K]$, we construct a larger set of K^2 d-dimensional symbolic matrices by computing $B_{nn'} := A_n^\dagger A_{n'}$ for all $n, n' \in [K]$. In the next step, the algorithm decides whether or not matrices $B_{nn'}$ are linearly independent. Like typical algorithms for deciding linear independence, our algorithm solves a system of linear homogeneous equations. If the only solution is trivial, then the output is $\top$, which means that the input channel is an extreme channel; otherwise the output is $\perp$, which means that the input channel is not extreme.

IV. RESULTS

In this section, we present our new results. First we begin in Sec. IV A by establishing our method for constructing group-covariant generalized-extreme channels for finite discrete and compact connected Lie groups and for given finite Hilbert-space dimension. Then, in Sec. IV B, we present our pseudocode for constructing generalized extreme group-covariant channels. Finally, in Sec. IV C, we solve and elaborate on four finite discrete-group and two compact connected Lie group examples of group-covariant generalized-extreme channels as interesting in their own right but also excellent illustrations of the generality of our method.

A. Algebraic construction of group-covariant extreme channels

In this subsection, we establish a procedure for constructing group-covariant generalized-extreme channels for given finite discrete and compact connected Lie groups and also given a finite Hilbert-space dimension. We use the equivalence relation in Definition 4 to show that either all elements of an equivalence class are extreme or else none are extreme. This property of classes simplifies the procedure for constructing all generalized extreme group-covariant channels.

1. Labeling group-covariant channels

In this subsubsection, we explain how to construct the set of group-covariant generalized-extreme channel labels $\mathcal{F}_{\mathcal{G},\text{gen}}$, which is explained in Sec. III B 2. An element $[\Omega] \in \mathcal{F}_{\mathcal{G},\text{gen}}$, where $\mathcal{F}_{\mathcal{G},\text{gen}}$ is defined in Sec. III B 2 and discussed more in Sec. III B 3, labels group-covariant generalized-extreme channels, which are elements of the set $\mathcal{W}_{\mathcal{G},\text{gen}}$, defined in Sec. III B 2, with $\mathcal{G}$ the name of the group. We propose two lemmas that are needed to construct group-covariant generalized-extreme channels.

The following lemma shows that (matrix) labels Ω and Ω', which are unitarily equivalent, label the same group-covariant channel, i.e.,

$$\Phi_{D^{(1)},D^{(2)},\Omega} = \Phi_{D^{(1)},D^{(2)},\Omega'}. \tag{31}$$

Lemma 2. All unitarily equivalent representations of $\mathcal{G}$ label the same group-covariant channel.

Proof. We being by recognizing that Eq. (15) holds for Kraus operators $\{A_k\}_{k\in[K]}$ if

$$\Phi_{D^{(1)},D^{(2)},\Omega} \in \mathcal{W}_{\mathcal{G},D^{(1)},D^{(2)}} \tag{32}$$

for each $\Phi_{D^{(1)},D^{(2)},\Omega}$ described by this set $\{A_k\}_{k\in[K]}$ (3). If Ω' is a representation of $\mathcal{G}$ that is unitarily equivalent to Ω through unitary conjugation by U, i.e.,

$$\Omega'(g) = U\Omega(g)U^\dagger,\ \forall g \in \mathcal{G}, \tag{33}$$

then, for

$$A'_\ell := \sum_{k=1}^{K} U_{\ell k}A_k\ \forall \ell \in [K] \tag{34}$$

we obtain

$$D^{(2)^\dagger}(g)A'_\ell D^{(1)}(g) = \sum_{\ell'=1}^{K} \Omega'_{\ell\ell'}(g)A'_{\ell'},\ \forall g \in \mathcal{G}. \tag{35}$$

Equation (35) implies that channel $\Phi_{D^{(1)},D^{(2)},\Omega'}$, described by Kraus operators $\{A'_\ell\}_{\ell\in[K]}$, belongs to $\mathcal{W}_{\mathcal{G},D^{(1)},D^{(2)}}$. Therefore, from Eq. (34) we verify Eq. (31). ■

Equation (32) shows that the triplets $(D^{(1)}, D^{(2)}, \Omega)$ and $(D^{(1)}, D^{(2)}, \Omega')$ label the same group-covariant channel if Ω and Ω' are unitarily equivalent. Unitary conjugation is an equivalence relation that partitions the set of labels of group-covariant channels $\mathcal{F}_\mathcal{G}$ (defined in Sec. III B 2) into equivalence classes denoted by $[\Omega]$. Distinct classes are represented by inequivalent representations of $\mathcal{G}$. According to Lemma 2, all elements of each class $[\Omega]$ label the same group-covariant channel. Hence, we label group-covariant channels by equivalence classes: $\Phi_{D^{(1)},D^{(2)},[\Omega]}$. As a consequence of Lemma 2, for constructing group-covariant channels, instead of solving Eq. (15) for all Ωs, just solving Eq. (15) for inequivalent Ωs, which represent distinct classes $[\Omega]$s, suffices.

The second lemma shows a necessary condition for $[\Omega]$ to label an extreme group-covariant channel.

Lemma 3. If

$$\Phi_{D^{(1)},D^{(2)},[\Omega]} \in \mathcal{W}_{\mathcal{G},\text{ext}}, \tag{36}$$

then $[\Omega]$ is any class of irreps of $\mathcal{G}$ with dimension not exceeding d.

Proof. If Eq. (36) holds, then according to Remark 4, $[\Omega]$ is a class of irreducible representation of $\mathcal{G}$ and, due to inequality (10), the number of Kraus operators of $\Phi_{D^{(1)},D^{(2)},[\Omega]}$ does not exceed d. On the other hand, according to Eq. (15), the number K of Kraus operators of $\Phi_{D^{(1)},D^{(2)},[\Omega]}$ equals the dimension of the representation of the group denoted by Ω. Therefore, for all $\Phi_{D^{(1)},D^{(2)},[\Omega]}$, $[\Omega]$s are classes of irreducible representations of the group with dimension less than or equal to d. ■

Lemma 3 yields necessary but not sufficient conditions for $\Phi_{D^{(1)},D^{(2)},[\Omega]}$ to be extreme. If, for $\Phi_{D^{(1)},D^{(2)},[\Omega]} \in \mathcal{W}_{\mathcal{G},\text{ext}}$, Ω is an irreducible representation of the group with dimension not exceeding d, and recalling that the number of Kraus operators of $\Phi_{D^{(1)},D^{(2)},[\Omega]}$ is equal to the dimension of Ω, one concludes that the number of Kraus operators of $\Phi_{D^{(1)},D^{(2)},[\Omega]}$ does not exceed d. Therefore, the Choi rank of $\Phi_{D^{(1)},D^{(2)},[\Omega]}$ is less than or equal to d. Hence, $\Phi_{D^{(1)},D^{(2)},[\Omega]}$ is a generalized-extreme channel: $\Phi_{D^{(1)},D^{(2)},[\Omega]} \in \mathcal{W}_{\mathcal{G},\text{gen}}$. Therefore, $\mathcal{F}_{\mathcal{G},\text{gen}}$, which is the set of labels of all group-covariant generalized-extreme channels, is the set of irreps of $\mathcal{G}$ with dimension not exceeding d. Hence, solving Eq. (15) for all $[\Omega] \in \mathcal{F}_{\mathcal{G},\text{gen}}$ ensures that we construct all group-covariant generalized-extreme channels. Actually an innovation in proposing a successful algorithmic approach for constructing all group-covariant generalized extreme channels is restriction to irreps Ω with dimension not exceeding d. By testing whether or not $\mathcal{S}$ in Eq. (9) is a set of linearly independent operators, $\Phi_{D^{(1)},D^{(2)},[\Omega]} \in \mathcal{W}_{\mathcal{G},\text{gen}}$ are classified as either group-covariant extreme or group-covariant quasiextreme channels.

2. Equivalent channels

For a finite discrete group or a compact connected Lie group $\mathcal{G}$ and Hilbert-space dimension d, the set of group-covariant channels is denoted $\mathcal{W}_\mathcal{G}$ (28), where the union is over all d-dimensional representations of $\mathcal{G}$. In this subsubsection, by using the equivalence relation between quantum channels, Definition 4, we partition the set of group-covariant quantum channels. Then we show that, if an element of an equivalence class is extreme, then all channels in that class are extreme. Thus, we show that group-covariant channels with respect to equivalent representations of $\mathcal{G}$ belong to the same equivalence class, and if an element of an equivalence class is extreme/quasiextreme, then all elements of that class are extreme/quasiextreme. This observation leads us to construct group-covariant extreme channels more efficiently.

Equivalence relation $\sim$ in Eq. (11) partitions the set of quantum channels into disjoint equivalence classes $[\Phi]$. The next lemma shows that, if one of the channels in an equivalence class $[\Phi]$ is extreme, the other elements of that class are extreme as well.

Lemma 4. If channel $\Phi \in S_{\Phi_{\text{ext}}}$, then all channels belonging to $[\Phi]$ are extreme.

Proof. Let an extreme channel Φ be described by a set of Kraus operators $\{A_k\}_{k\in[K]}$. If the channel Φ', described by a set of Kraus operators $\{A'_k\}_{k\in[K]}$, belongs to the same class as Φ, then unitary operators U and V exist such that Eq. (11) holds. Therefore,

$$A'_k = \sum_m W_{km}U^\dagger A_m V^\dagger \tag{37}$$

with W being a d-dimensional unitary operator. To show that Φ' is an extreme channel, we show that operators $\{{A'_k}^\dagger A'_\ell\}$ are linearly independent (see Theorem 1). If

$$\sum_{k,\ell} X_{k\ell}{A'_k}^\dagger A'_\ell = 0, \tag{38}$$

then

$$\sum_{m,n} (W^\dagger X W)_{mn} A_m^\dagger A_n = 0 \tag{39}$$

is easy to see. As Φ is extreme, according to Theorem 1, $X = 0$. Therefore, the set of operators $\{{A'_k}^\dagger A'_\ell\}$ is linearly independent. Hence, $\Phi' \in S_{\Phi_{\text{ext}}}$. ■

Per Definition 4, if $D^{(i)}$ and $D'^{(i)}$ for $i \in [2]$ are unitarily equivalent representations of the group (16), then

Eq. (17) holds. Therefore, according to Remark 5, channels $\Phi_{D^{(1)},D^{(2)},[\Omega]} \in \mathcal{W}_{\mathcal{G},D^{(1)},D^{(2)}}$ and $\Phi'_{D'^{(1)},D'^{(2)},[\Omega]} \in \mathcal{W}_{\mathcal{G},D'^{(1)},D'^{(2)}}$ are equivalent and belong to the same class. We denote the set of all group-covariant channels with respect to representations of $\mathcal{G}$, which are unitarily equivalent to $D^{(1)}$ and $D^{(2)}$, by $\mathcal{W}_{\mathcal{G},[D^{(1)}],[D^{(2)}]}$, where labels of distinct classes $[D^{(i)}]$s are inequivalent representations of the group.

Remark 8. According to Lemma 4, if $\Phi_{D^{(1)},D^{(2)},[\Omega]} \in \mathcal{W}_{\mathcal{G},[D^{(1)}],[D^{(2)}]}$ is extreme, then all other channels in $\mathcal{W}_{\mathcal{G},[D^{(1)}],[D^{(2)}]}$ are extreme. Hence, for group-covariant generalized-extreme channels, that is for all $\Phi_{D^{(1)},D^{(2)},[\Omega]}$ with $[\Omega] \in \mathcal{F}_{\mathcal{G},\mathrm{gen}}$ (defined in Sec. III B 2), if $\Phi_{D^{(1)},D^{(2)},[\Omega]} \in \mathcal{W}_{\mathcal{G},[D^{(1)}],[D^{(2)}]}$ is extreme/quasiextreme, then all other channels in $\mathcal{W}_{\mathcal{G},[D^{(1)}],[D^{(2)}]}$ are extreme/quasiextreme. Therefore, to construct $\mathcal{W}_{\mathcal{G}}$, instead of constructing all elements of $\mathcal{W}_{\mathcal{G},D^{(1)},D^{(2)}}$ and then taking the union over d-dimensional representation of $\mathcal{G}$, we just construct one channel $\Phi_{D^{(1)},D^{(2)},[\Omega]}$ in each $\mathcal{W}_{\mathcal{G},[D^{(1)}],[D^{(2)}]}$ with $[\Omega] \in \mathcal{F}_{\mathcal{G},\mathrm{gen}}$. The other extreme/quasiextreme channels can be constructed from this representative channel by unitary conjugations before and after the action of the channel.

B. Pseudocoding the construction of generalized extreme group-covariant channels

In this subsection, we present our pseudocode results for constructing descriptions of group-covariant generalized-extreme channels. Typically construction of channel descriptions is accomplished mathematically but of course not successfully in general due to difficult in solving such generic problems. Our approach is not to write the mathematical solution in general but rather the algorithm that generates the solution on an appropriate autonomous logical machine with sufficient resources. Our approach synthesizes techniques from computing and representation theory and thus involves knowledge from both fields. Here, we express our algorithmic approach for solving Problem1, Problem 2, and Problem 3 as pseudocodes. For more details on pseudocode, notation, and library used in the pseudocode, see Appendixes A 3, B, and C.

C. Explicit examples

In this subsection, we present explicit examples for constructing group-covariant extreme or quasiextreme channels for Z_2, S_3, A_4, and D_5 as finite discrete groups [45] and SO(3) and SU(2) for compact connected Lie groups [46]. We recall from Appendix A and Sec. III B that for any given group, group-covariant generalized-extreme channels are denoted by $\Phi_{[D^{(1)}],[D^{(2)}],[\Omega]}$, where $D^{(1)}$ and $D^{(2)}$ are representative of distinct classes of d-dimensional representations of the group, and Ω is representative of class of irreducible representations of the group with dimension less than or equal to d. In making choices for Ω, we use the theorem that says that thenumber of inequivalent irreps for a finite discrete group equals the number of conjugacy classes [47].

1. Z_2-covariant extreme qubit channels

Our first example is the group Z_2, which is Abelian, hence it has just one-dimensional irreps. As the number of Kraus operators equals the dimension of the irrep, a Z_2-covariant channel has just one Kraus operator and, due to the trace-preserving constraint (4), this channel is strictly a unitary channel. Therefore, the constructed channel is certainly extreme [8].

The Z_2 group is explicitly

$$Z_2 := \langle g\rangle/\{g^2 = e\} = \{e, g\} \tag{40}$$

with e being the identity element of the group and the notation $\langle\rangle$ denoting the generating set (18). This group has two inequivalent irreducible representations, namely $\Omega^{(1)}_{\pm}(g) = \pm 1$, which are one-dimensional. Thus, we have two options for Ω in Eq. (23). For a single-qubit Z_2-covariant channel, $D^{(1)}$ and $D^{(2)}$ are two-dimensional representations of the group, and uncountably many two-dimensional representations for each of $D^{(1)}$ and $D^{(2)}$ are allowed. According to Lemma 4 and Remark 8, we just consider inequivalent two-dimensional representations. These two cases are constructed as direct sums of inequivalent irreps of Z_2, i.e., $\Omega^{(1)}_{+} \oplus \Omega^{(1)}_{\pm}$, Therefore, we have two cases for $D^{(1)}$, two cases for $D^{(2)}$, and two cases for Ω, so we have eight cases to consider.

Now we proceed to investigate the first of these eight cases. The unitary Kraus operator representing $\Phi_{[\Omega^{(1)}_{+}\oplus\Omega^{(1)}_{-}],[\Omega^{(1)}_{+}\oplus\Omega^{(1)}_{-}],[\Omega^{(1)}_{+}]}$ is represented by (3)

$$A = \mathrm{diag}(1, \mathrm{e}^{\mathrm{i}\alpha}),\ \alpha \in \mathbb{R}. \tag{41}$$

This matrix A is evidently unitary and satisfies the trace-preserving constraint (4). From the trivial linear independence of the single-element set $\mathcal{S} = \{A^{\dagger}A\}$ (9), $\Phi_{[\Omega^{(1)}_{+}\oplus\Omega^{(1)}_{-}],[\Omega^{(1)}_{+}\oplus\Omega^{(1)}_{-}],[\Omega^{(1)}_{+}]}$ must be extreme.

Now we move to the second case. The unitary Kraus operator of $\Phi_{[\Omega^{(1)}_{+}\oplus\Omega^{(1)}_{-}],[\Omega^{(1)}_{+}\oplus\Omega^{(1)}_{-}],[\Omega^{(1)}_{-}]}$ is

$$A = \begin{pmatrix} 0 & \mathrm{e}^{\mathrm{i}\beta} \\ 1 & 0 \end{pmatrix},\ \beta \in \mathbb{R}. \tag{42}$$

Evidently, this matrix is unitary, thus it is trace-preserving and extreme, as expected.

Now we consider the case of $\Phi_{[\Omega^{(1)}_{+}\oplus\Omega^{(1)}_{+}],[\Omega^{(1)}_{+}\oplus\Omega^{(1)}_{+}],[\Omega^{(1)}_{+}]}$. In this case, following from Remark 7, the group-covariant property does not constrain the single Kraus operator. Hence, the only constraint is the trace-preserving condition, which results in the map $\Phi_{[\Omega^{(1)}_{+}\oplus\Omega^{(1)}_{+}],[\Omega^{(1)}_{+}\oplus\Omega^{(1)}_{+}],[\Omega^{(1)}_{+}]}$ being a general unitary evolution. In the fourth case, $\Phi_{[\Omega^{(1)}_{+}\oplus\Omega^{(1)}_{+}],[\Omega^{(1)}_{+}\oplus\Omega^{(1)}_{+}],[\Omega^{(1)}_{-}]}$, the Kraus operator is zero, which means that such a group-covariant channel does not exist.

Finally, we consider the last four cases, which are

$$\Phi_{[\Omega^{(1)}_{+}\oplus\Omega^{(1)}_{-}],[\Omega^{(1)}_{+}\oplus\Omega^{(1)}_{+}],[\Omega^{(1)}_{+}]},\ \Phi_{[\Omega^{(1)}_{+}\oplus\Omega^{(1)}_{-}],[\Omega^{(1)}_{+}\oplus\Omega^{(1)}_{+}],[\Omega^{(1)}_{-}]},$$
$$\times\ \Phi_{[\Omega^{(1)}_{+}\oplus\Omega^{(1)}_{+}],[\Omega^{(1)}_{+}\oplus\Omega^{(1)}_{-}],[\Omega^{(1)}_{+}]},\ \Phi_{[\Omega^{(1)}_{+}\oplus\Omega^{(1)}_{+}],[\Omega^{(1)}_{+}\oplus\Omega^{(1)}_{-}],[\Omega^{(1)}_{-}]}. \tag{43}$$

All four of these maps (43) fail to satisfy the trace-preserving constraint. Thus, these cases do not occur as outputs from the algorithm.

This simple example of Z_2-covariant channels illustrates how our method leads to successful results, which are already known for the case of qubit channels [9,11]. To go beyond unitary group covariant channels, we need to consider cases of non-Abelian groups. Hence, in Sec. IV C 2, we study the permutation group.

Algorithm 1 Generate all group-covariant generalized-extreme channels up to unitary equivalence for a finite discrete group or a compact connected Lie group

```
Input:
  character GNAME                                   ▷ Name of finite discrete or compact connected Lie group 𝒢
  binary GTYPE                                      ▷ 0 for finite discrete group; 1 for compact connected Lie group
  natural HDIM                                      ▷ Hilbert-space dimension
Output:
  symjagged[ ][ ][HDIM][HDIM] KRAUS                 ▷ Set of Kraus operators with dimension HDIM
  natural COUNT                                     ▷ Number of generalized extreme GNAME-covariant channels
Procedure KROPDISCGGNAME,GTYPE,HDIM
  IMPORT SOLVECHANNEL, DELETE, DAG, ID, SYMSOLVE, PROPS, TRANSP, VECTOMATR, ZERO   ▷ Import from libraries
  Appendix C 1, Appendix C 2
  posinteger NUMIRREP
  posinteger NUMREP
  posinteger NUMGEN
  posinteger[ ] DIM
  symbol[ ][ ][HDIM][HDIM] REP
  symjagged[ ][ ][ ][ ] IRREP
  symjagged[ ][ ][HDIM][HDIM] KRAUS
  natural COUNT                                     ▷ Counts the number of generalized extreme GNAME-covariant channels
  (NUMIRREP,NUMREP,NUMGEN,DIM,IRREP,REP) ← PROPS(GNAME,GTYPE,HDIM)
  symbol[NUMREP][NUMREP][NUMIRREP][NUMGEN][ ][ ]B   ▷ Coefficient matrix in Eq. (23)/(26) (GTYPE=0/1)
  COUNT ← 1
  for M ← 0 to NUMIRREP − 1 do
    for K ← 0 to NUMREP − 1 do
      for L ← 0 to NUMREP − 1 do
        if GTYPE= 0 then
          for I ← 0 to NUMGEN − 1 do
            B[COUNT][I] ← ⊕_{j=0}^{DIM[M]−1} (DAG(GREP[K][I]) ⊗ TRANSP(GREP[L][I])) − GIRREP[M][I] ⊗ ID(HDIM^2)
          end for
        else
          for I ← 0 to NUMGEN − 1 do
            B[COUNT][I] ← ⊕_{j=0}^{DIM[M]−1} (GREP[K][I] ⊗ ID(HDIM) − ID(HDIM) ⊗ DAG(GREP[L][I])) − OMEGA[M][I]
  ⊗ ID(HDIM^2)
          end for
        end if
        symbol[DIM[M] ∗ HDIM^2] X
        X ← SYMSOLVE(B[COUNT])
        if X ≠ ZERO(DIM[M] ∗ HDIM^2) then           ▷ Tests if the covariant CP exists
          KRAUS[COUNT] ← VECTOMATR(X,DIM[M])
          KRAUS[COUNT] ← SOLVECHANNEL(KRAUS[COUNT])
          if KRAUS≠ZERO(DIM(M),HDIM,HDIM) then      ▷ Tests if the covariant channel exists
            COUNT ← COUNT + 1
          end if
          DELETE(X)                                 ▷ Delete X and clear space
        end if
      end for
    end for
  end for
  RETURN COUNT
  RETURN KRAUS
end procedure
```

2. S_3-covariant qubit and qutrit extreme and quasiextreme

In this subsubsection, we consider S_3-covariant channels with S_3 the symmetric group, which is a set of all permutations of three objects. For all Abelian groups, we always obtain unitary channels. Hence, to obtain nonunitary channels, we begin by choosing S_3, which is the smallest non-Abelian group.

The order of the group S_3 is 6, that is, $|S_3| = 6$, with $|G|$ denoting the order of the group G. It has two generators denoted by σ_1 and σ_2 such that

$$S_3 = \langle \sigma_1, \sigma_2 \rangle / \{\sigma_1^2 = \sigma_2^2 = e, \sigma_1\sigma_2\sigma_1 = \sigma_2\sigma_1\sigma_2\}. \quad (44)$$

S_3 has three inequivalent irreps, specifically two one-dimensional irreps

$$\Omega_+^{(1)}(\sigma_1) = 1 = \Omega_+^{(1)}(\sigma_2), \ \Omega_-^{(1)}(\sigma_1) = 1 = -\Omega_-^{(1)}(\sigma_2), \quad (45)$$

Algorithm 2 Test whether a given channel is extreme or not

```
Input:symbol[NUM][HDIM][HDIM] KRAUS                    ▷ Kraus operators of the channel
Output:binary ISEXTREME                                ▷ ISEXTREME is 0 for an extreme channel and 1 for a quasiextreme channel
procedure testExtKRAUS
    IMPORT DAG, SYMSOLVE, ZERO                         ▷ Import from libraries Appendix C 1 and Appendix C 2
    binary ISEXTREME
    natural N                                          ▷ Counts loops
    symbol[1][HDIM^2][NUM^2] MATCOEFF                  ▷ Matrix of coefficients
    symbol[NUM^2] SOLUSION                             ▷ Solution of system of linear equations
    symbol[NUM^2][HDIM][HDIM] SCRATCHPAD               ▷ Scratch space for calculating
    for I← 0 to NUM − 1 do
      for J← 0 to NUM − 1 do
        SCRATCHPAD[N]←DAG(KRAUS[I])×KRAUS[J]
      end for
      N← N + 1
    end for
    N← 0
    for I← 0 to HDIM^2 − 1 do
      for J← I + 1 to HDIM^2 − 1 do
        for K← 0 to NUM^2 − 1 do
          MATCOEFF[0][N][K]←SCRATCHPAD[K][I][J]
        end for
        N← N + 1
      end for
    end for
    X←SYMSOLVE(MATCOEFF)
    if X ≠ ZERO(NUM^2) then
      ISEXTREME← 1                                     ▷ ISEXTREME indicates quasiextreme channel; else extreme (0)
    end if
    RETURN ISEXTREME
end procedure
```

and one two-dimensional irrep

$$\Omega^{(2)}(\sigma_1) = \mathrm{diag}(1,-1),\ \Omega^{(2)}(\sigma_2) = \frac{1}{2}\begin{pmatrix} -1 & \sqrt{3}e^{i\phi} \\ \sqrt{3}e^{-i\phi} & 1 \end{pmatrix}, \phi \in [0, 2\pi]. \tag{46}$$

Representations of S_3 with dimension larger than 2 are indeed reducible and can be constructed as unitary conjugations of direct sums of its irreps.

a. Qubit channels: For qubit channels, $d = 2$, $D^{(1)}$ and $D^{(2)}$ are two-dimensional representations of S_3. According to Lemma 4 and Remark 8, among an uncountable number of two-dimensional representations for S_3, it is sufficient to consider just inequivalent two-dimensional reps of S_3 to solve Eq. (23). In fact, there are just four inequivalent two-dimensional representations for S_3 given by

$$\Omega^{(2)}, \Omega^{(1)}_+ \oplus \Omega^{(1)}_+,\ \Omega^{(1)}_+ \oplus \Omega^{(1)}_-,\ \Omega^{(1)}_- \oplus \Omega^{(1)}_-. \tag{47}$$

For Ω, we first note that one-dimensional irreps of the groups cannot lead to nonunitary channels. Hence, to go beyond unitary channels, we have just one option for Ω, that is, $\Omega^{(2)}$ (46). Thus, given that for each $D^{(1)}$ and $D^{(2)}$ we have four options, the total number of cases to be solved for nonunitary S_3-covariant channels is 16.

Among these 16 cases, here we just report one case, that is, $\Phi_{[\Omega^{(2)}],[\Omega^{(2)}],[\Omega^{(2)}]}$. For this case the only solution to the set of Eqs. (23) is zero. Thus, $\Phi_{[\Omega^{(2)}],[\Omega^{(2)}],[\Omega^{(2)}]}$ does not exist. Although we have only treated one the of 16 cases here, this case illustrates the feasibility of our method and the qubit case is fully understood [11], so we move on to the unexplored $d = 3$ cases instead.

b. Qutrit channels: For qutrit channels, $d = 3$, and the dimension of all irreps of S_3 satisfies $d < 3$; hence, the necessary condition in Lemma 3 is satisfied for Ω being any S_3 irrep. Among these three choices for Ω, to go beyond unitary channels, we choose $\Omega = \Omega^{(2)}$ (46) as other irreps of S_3, namely $\Omega^{(1)}_\pm$, are one-dimensional (45) and label channels that are unitary. According to Remark 8, candidates for $D^{(1)}$ and $D^{(2)}$ are all three-dimensional inequivalent representations of S_3 which we construct by direct sum of irreps of S_3. Direct sums of one-dimensional irreps yield four inequivalent irreps, namely

$$\Omega^{(1)}_+ \oplus \Omega^{(1)}_+ \oplus \Omega^{(1)}_+,\ \Omega^{(1)}_+ \oplus \Omega^{(1)}_+ \oplus \Omega^{(1)}_-,\ \Omega^{(1)}_+ \oplus \Omega^{(1)}_- \oplus \Omega^{(1)}_-,\ \times\ \Omega^{(1)}_- \oplus \Omega^{(1)}_- \oplus \Omega^{(1)}_-, \tag{48}$$

which are direct sums of one-dimensional and two-dimensional irreps and also yield two more inequivalent cases

$$\Omega^{(1)}_\pm \oplus \Omega^{(2)}. \tag{49}$$

Hence, for each $D^{(1)}$ and $D^{(2)}$, we have six candidates.

In total for strictly nonunitary S_3-covariant channels, 36 cases need to be solved. Among these 36 cases, we just solve one case as an illustration. One case suffices to confirm our

method before proceeding to test our method on different groups.

We now solve Eqs. (23) for Kraus operators with

$$D^{(1)} = D^{(2)} = \Omega_+^{(1)} \oplus \Omega^{(2)},\ \Omega = \Omega^{(2)}, \tag{50}$$

which yields the Kraus-operator family

$$A_1 = \begin{pmatrix} 0 & \alpha & 0 \\ \beta & \gamma & 0 \\ 0 & 0 & -\gamma \end{pmatrix},$$
$$A_2 = \begin{pmatrix} 0 & 0 & \alpha \\ 0 & 0 & -\gamma \\ \beta e^{-2i\phi} & -\gamma e^{-2i\phi} & 0 \end{pmatrix},\ \alpha, \beta, \gamma \in \mathbb{C}, \tag{51}$$

which represent an S_3-covariant completely positive map. By imposing the trace-preserving constraint (4), we obtain the relations

$$2|\beta|^2 = 1,\ |\alpha|^2 + 2|\gamma|^2 = 1 \tag{52}$$

between the parameters in Eq. (51). Hence, Kraus operators of the S_3-covariant channel with respect to $\Omega_+^{(1)} \oplus \Omega^{(2)}$ and labeled by $\Omega^{(2)}$, that is, $\Phi_{[\Omega_+^{(1)}\oplus\Omega^{(2)}],[\Omega_+^{(1)}\oplus\Omega^{(2)}],[\Omega^{(2)}]}$, are given in Eq. (51) with the constraint in Eq. (52).

To see which parameter values correspond to extreme channels, we construct the set $\mathcal{S}$ (9) for Kraus operators (51). The obtained set is a set of linearly independent operators for all values of parameters in Eq. (51) except for the instance

$$|\alpha|^2 = \tfrac{1}{2},\ |\gamma|^2 = \tfrac{1}{4}. \tag{53}$$

That is, the set of Kraus operators (51) with constraint (52) represents S_3-covariant extreme channels for the entire range of parameters except at the point (53), which represents an S_3-covariant quasiextreme channel.

3. A_4-covariant extreme qutrit channel

Our next example is for the alternating group A_4, which is a normal subgroup of S_4, and A_4 consists of all even permutations of a four-object set. Although S_4 is the natural next case after S_3 in Sec. IV C 2, S_4 has three generators, so we restrict to the more manageable case of A_4 here for our generalization. The group A_4 is non-Abelian and of order 12, and A_4 has two generators, which we denote by g_1 and g_2. Furthermore, A_4 has four inequivalent irreducible representations. Three of these representations are one-dimensional and are denoted by $\Omega_{+,\circ,-}^{(1)}$. The fourth case is the three-dimensional irreducible representation

$$\Omega^{(3)}(g_1) = \mathrm{diag}(1, \omega, \omega^2),\ \Omega^{(3)}(g_2) = -\frac{1}{3}\begin{pmatrix} 1 & -2\omega^2 & 2\omega \\ -2 & \omega^2 & 2\omega \\ 2 & 2\omega^2 & \omega \end{pmatrix}, \omega := e^{\frac{2i\pi}{3}}. \tag{54}$$

For a qutrit channel, $d = 3$, which determines the dimension of both $D^{(1)}$ and $D^{(2)}$. Based on Remark 5 for $D^{(1)}$ and $D^{(2)}$, we only consider inequivalent three-dimensional cases from among the uncountable number of available three-dimensional representations of A_4. Hence, for each $D^{(1)}$ and $D^{(2)}$, 11 candidates

$$\Omega_+^{(1)} \oplus \Omega_+^{(1)} \oplus \Omega_+^{(1)},\ \Omega_\circ^{(1)} \oplus \Omega_\circ^{(1)} \oplus \Omega_\circ^{(1)},\ \Omega_-^{(1)} \oplus \Omega_-^{(1)} \oplus \Omega_-^{(1)},$$
$$\Omega_+^{(1)} \oplus \Omega_+^{(1)} \oplus \Omega_\circ^{(1)},\ \Omega_+^{(1)} \oplus \Omega_\circ^{(1)} \oplus \Omega_\circ^{(1)},\ \Omega_+^{(1)} \oplus \Omega_+^{(1)} \oplus \Omega_-^{(1)},$$
$$\Omega_+^{(1)} \oplus \Omega_-^{(1)} \oplus \Omega_-^{(1)},\ \Omega_\circ^{(1)} \oplus \Omega_\circ^{(1)} \oplus \Omega_-^{(1)},\ \Omega_\circ^{(1)} \oplus \Omega_-^{(1)} \oplus \Omega_-^{(1)},$$
$$\Omega_+^{(1)} \oplus \Omega_\circ^{(1)} \oplus \Omega_-^{(1)},\ \Omega^{(3)} \tag{55}$$

exist. To select a proper representation for Ω, we note that its dimension should be less than or equal to 3. Therefore, according to Lemma 3, all irreps of A_4 satisfy the necessary condition for labeling an A_4-covariant generalized-extreme channel. The three one-dimensional irreps of A_4 label unitary channels, if they exist. Hence, the only candidate for Ω to label a nonunitary A_4-covariant generalized-extreme channel is $\Omega^{(3)}$. Thus, for nonunitary A_4-covariant generalized-extreme channel in total, there are 121 cases to study due to 11 choices we have for each $D^{(1)}$ and $D^{(2)}$. Among these, we focus on one of them just to illustrate how our method works.

Solving Eq. (23) for channel $\Phi_{[\Omega^{(3)}],[\Omega^{(3)}],[\Omega^{(3)}]}$ yields Kraus operators

$$A_1 = \frac{1}{\sqrt{2}}\mathrm{diag}(0, 1, -1),\ A_2 = \frac{1}{\sqrt{2}}\begin{pmatrix} 0 & -1 & 0 \\ 0 & 0 & 0 \\ \omega & 0 & 0 \end{pmatrix},$$
$$A_3 = \frac{1}{\sqrt{2}}\begin{pmatrix} 0 & 0 & 1 \\ -\omega & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \tag{56}$$

which is a set of constant-valued matrices rather than a parameter family (51). These Kraus operators satisfy the trace-preserving condition (4). Also the set of operators $\mathcal{S} = \{A_i^\dagger A_j\}$ with $i, j = 1, 2, 3$ is linearly independent. Hence, $\Phi_{[\Omega^{(3)}],[\Omega^{(3)}],[\Omega^{(3)}]}$ is an extreme quantum channel.

4. D_5-covariant extreme qutrit channel

For S_3 in Sec. IV C 2, we obtain Kraus operators (51) of a family of extreme channels. Instead of extending to S_4, for tractability reasons we extend to A_4, which is a restriction of S_4 to even permutations. That yields only constant matrices (56) as the solution. Now we consider another "small" group, dihedral group D_5, which is the group of symmetries for a regular pentagon. By studying D_5, we can see whether the S_3-covariant and A_4-covariant extreme channels are also extreme channels for a D_5-covariant channel or not despite $D_5 \not\leq S_3$ and $D_5 \not\leq A_4$.

The group

$$D_5 = \langle g_1, g_2\rangle / \{g_1^2 = g_2^5 = (g_1 g_2)^2 = e\} \tag{57}$$

is of order 10 with two generators, namely g_1 and g_2, which generate reflection and rotation, respectively. D_5 has four inequivalent irreps. Two of the inequivalent irreps are one-dimensional,

$$\Omega_+^{(1)}(g_1) = 1 = \Omega_+^{(1)}(g_2) = 1,\ \Omega_-^{(1)}(g_1) = -1 = -\Omega_-^{(1)}(g_2), \tag{58}$$

and the other two are two-dimensional,

$$\Omega_+^{(2)}(g_1) = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix},$$
$$\Omega_+^{(2)}(g_2) = \begin{pmatrix} \cos\omega & -\sin\omega \\ \sin\omega & \cos\omega \end{pmatrix},$$
$$\Omega_-^{(2)}(g_1) = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix},$$
$$\Omega_-^{(2)}(g_2) = \begin{pmatrix} \cos 2\omega & -\sin 2\omega \\ \sin 2\omega & \cos 2\omega \end{pmatrix}. \tag{59}$$

for $\omega := e^{\frac{2i\pi}{5}}$. Indeed, representations of D_5 in dimensions other than 1 and 2 are reducible and are constructed by unitary conjugation of direct sums of its irreps.

In this example, we are interested in qutrit D_5-covariant generalized-extreme channels. Hence, $d = 3$, which determines the dimension of $D^{(1)}$ and $D^{(2)}$. According to Remark 8, among the uncountable number of three-dimensional representations of D_5, we only consider inequivalent three-dimensional representations for each $D^{(1)}$ and $D^{(2)}$. These eight cases are

$$\begin{aligned}&\Omega_+^{(1)} \oplus \Omega_+^{(1)} \oplus \Omega_+^{(1)},\ \Omega_+^{(1)} \oplus \Omega_+^{(1)} \oplus \Omega_-^{(1)},\ \Omega_+^{(1)} \oplus \Omega_-^{(1)} \oplus \Omega_-^{(1)},\\ &\times \Omega_-^{(1)} \oplus \Omega_-^{(1)} \oplus \Omega_-^{(1)},\ \Omega_+^{(1)} \oplus \Omega_+^{(2)},\ \Omega_+^{(1)} \oplus \Omega_-^{(2)},\\ &\times \Omega_-^{(1)} \oplus \Omega_+^{(2)}, \Omega_-^{(1)} \oplus \Omega_-^{(2)}.\end{aligned} \tag{60}$$

As the dimension of all irreps of D_5 satisfies the necessary condition in Lemma 3, they are all acceptable candidates to label D_5-covariant generalized-extreme channels. However, only irreps of dimension greater than 1 can label nonunitary maps. Hence, for nonunitary D_5-covariant generalized-extreme channels, the two candidates for Ω are $\Omega_+^{(2)}$ and $\Omega_-^{(2)}$. Therefore, for nonunitary D_5-covariant generalized-extreme channels, we have 128 cases to study.

Remark 9. The number of nonunitary group-covariant generalized-extreme channel candidates for group D_5 exceeds the number for A_4 in Sec. IV C 3 despite $|A_4| > |D_5|$ and despite the number of inequivalent irreps for A_4 being equal to the number of irreps for D_5.

Among all these cases, we just consider two cases as an example, namely

$$\Phi_{[\Omega_+^{(1)}\oplus\Omega_+^{(2)}],[\Omega_+^{(1)}\oplus\Omega_+^{(2)}],[\Omega_+^{(2)}]},\ \Phi_{[\Omega_+^{(1)}\oplus\Omega_-^{(2)}],[\Omega_+^{(1)}\oplus\Omega_-^{(2)}],[\Omega_-^{(2)}]}. \tag{61}$$

Solving Eq. (23) for these instances yields the same Kraus operators,

$$A_1 = \frac{1}{\sqrt{2}}\begin{pmatrix} 0 & \sqrt{2} & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix},\ A_2 = \frac{1}{\sqrt{2}}\begin{pmatrix} 0 & 0 & \sqrt{2} \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix} \tag{62}$$

for both cases. Although $D_5 \not\leq S_3$, by comparing Eqs. (62) and (51) we see that the Kraus operators for the D_5-covariant channels discussed in this example are a special case of the family of S_3-covariant extreme channels with Kraus operators given in Eq. (51).

5. *SO*(3)-*covariant channels*

Thus far, examples of group-covariant extreme channels have been studied for finite discrete groups. In this example, we focus on SO(3)-group covariant channels with SO(3) generated by the Lie algebra $\mathfrak{so}(3)$. The generating set for $\mathfrak{so}(3)$ comprises the ladder operators $L_\pm$ and the Cartan operator L_z, and $\{|l, m\rangle\}$ are the weight states labeled by Casimir and Cartan operators, L^2 and L_z, respectively [46]. The Casimir invariant has spectrum $\{l(l+1); l \in \mathbb{N}\}$ for $\mathbb{N}$ denoting the set of natural numbers discussed in Appendix B. Thus, unitary irreps of $\mathfrak{so}(3)$ have odd dimensions $\{2l+1; l \in \mathbb{N}\}$ and are given by

$$\begin{aligned}\Omega^{(2l+1)}(L_\pm) &= \sum_{m=-l}^{l} C_{lm}^{\pm}\, |l, m\pm 1\rangle\langle l, m|,\ \Omega^{(2l+1)}(L_z)\\ &= \sum_{m=-l}^{l} m\, |l, m\rangle\langle l, m|\end{aligned} \tag{63}$$

for the Clebsch-Gordan coefficient

$$C_{lm}^{\pm} := \sqrt{l(l+1) - m(m\pm 1)}. \tag{64}$$

The case $l = 0$ yields the trivial representation for $\Omega^{(1)}$ in Eq. (63).

a. Qudit SO(3)-covariant generalized-extreme channels: For qudit channels, the dimension of both $D^{(1)}$ and $D^{(2)}$ is d. According to Remark 5, among uncountable d-dimensional unitary representations of SO(3), we only consider one of many inequivalent representations of SO(3). Only a finite number of inequivalent representations is possible as the number of inequivalent representations equals the number of partitions of d into odd-number components, such as 1, 1, 1, and 3 for $d = 3$. According to Lemma 3, all SO(3) irreps with dimension less than or equal to d label a generalized-extreme channel. Hence, for Ω, we have $[\frac{d+1}{2}]$ candidates.

b. Special case: Among the many candidates for $D^{(1)}$, $D^{(2)}$, and Ω, we choose one candidate as an example of an SO(3)-covariant generalized-extreme channel. Our arbitrary choice for study is $\Phi_{[\Omega^{(d)}],[\Omega^{(d)}],[\Omega^{(d)}]}$

In this example, $D^{(1)}$ and $D^{(2)}$ are chosen to be $\Omega^{(d)}$, which is an SO(3) irrep. As SO(3) irreps are odd-dimensional, bearing in mind that the dimension of $D^{(1)}$ and $D^{(2)}$ equals the Hilbert-space dimension d, this example is valid just for odd d. For the case of $\Phi_{[\Omega^{(d)}],[\Omega^{(d)}],[\Omega^{(d)}]}$, following Remark 6, Eq. (20) simplifies to the commutators (21)

$$\begin{aligned}[\Omega^{(d)}(L_z), A_m] &= mA_m,\ [\Omega^{(d)}(L_\pm), A_m]\\ &= C_{lm}^{\pm} A_{m\pm 1},\ l \equiv \frac{d-1}{2},\end{aligned} \tag{65}$$

which yields Kraus operators of $\Phi_{[\Omega^{(d)}],[\Omega^{(d)}],[\Omega^{(d)}]}$ being rank-l irreducible spherical tensors as candidates for generalized extreme channels. Furthermore, as in this example $D^{(1)}$ is chosen to be an irrep of SO(3), Ξ as defined in Eq. (5) is proportional to the identity [40]. To see if the trace-preserving condition (4) is satisfied, and to determine if $\Phi_{[\Omega^{(d)}],[\Omega^{(d)}],[\Omega^{(d)}]}$ is extreme or quasiextreme, we restrict our attention to the low-dimensional cases of $d = 3$ and 5 for which the problem is tractable.

c. Qutrit SO(3)-covariant extreme channels: We consider the special case of odd-dimensional Hilbert space for $d = 3$. The number 3 is partitioned into (3), (2,1), and (1,1,1), which can be represented by Young diagrams [48] $\square\square\square$, $\begin{smallmatrix}\square\square\\ \square\end{smallmatrix}$, and $\begin{smallmatrix}\square\\ \square\\ \square\end{smallmatrix}$, respectively. For our case, only partitions of $d = 3$ into

odd-number components $\ydiagram{3}$ and $\ydiagram{1,1,1}$ are required. Thus, only two candidates for each $D^{(1)}$ and $D^{(2)}$ are possible, namely three-dimensional representations

$$\Omega^{\ydiagram{1}} \oplus \Omega^{\ydiagram{1}} \oplus \Omega^{\ydiagram{1}},\ \Omega^{\ydiagram{3}}. \tag{66}$$

For Ω two candidates arise, namely a one-dimensional irrep necessarily labeling a unitary channel and a three-dimensional irrep. Hence, the case we study here,

$$\Phi_{[\Omega^{\ydiagram{3}}],[\Omega^{\ydiagram{3}}],[\Omega^{\ydiagram{3}}]}, \tag{67}$$

is one among eight possibilities. According to Eq. (65), Kraus operators are irreducible spherical tensors of rank 1, namely

$$A_1 = \begin{pmatrix} 0 & -a & 0 \\ 0 & 0 & -a \\ 0 & 0 & 0 \end{pmatrix} = -A_{-1}^{\dagger},\ A_0 = \mathrm{diag}\,(a, 0, -a),\ a \in \mathbb{C}. \tag{68}$$

It is straightforward to show that Ξ, defined in Eq. (5), is given by

$$\Xi = \sum_{k=-1}^{1} A_k^{\dagger} A_k = 2|a|^2 \mathbb{1}, \tag{69}$$

which satisfies the trace-preserving condition (4) for $|a| = \frac{1}{\sqrt{2}}$. Furthermore, the set of operators $\mathcal{S}$ (9) in this example is a set of linearly independent operators; hence, according to Theorem 1, the qutrit channel $\Phi_{[\Omega^{\ydiagram{3}}],[\Omega^{\ydiagram{3}}],[\Omega^{\ydiagram{3}}]}$ described by Kraus operators (68) with $|a| = \frac{1}{\sqrt{2}}$ is extreme.

d. Ququint SO(3)-covariant extreme channels: For ququint channels, the three candidates for each $D^{(1)}$ and $D^{(2)}$ are

$$\Omega^{\ydiagram{1}} \oplus \Omega^{\ydiagram{1}} \oplus \Omega^{\ydiagram{1}} \oplus \Omega^{\ydiagram{1}} \oplus \Omega^{\ydiagram{1}},\ \Omega^{\ydiagram{1}} \oplus \Omega^{\ydiagram{1}} \oplus \Omega^{\ydiagram{3}},\ \Omega^{\ydiagram{5}}. \tag{70}$$

For Ω there are three candidates: one-, three-, and five-dimensional irreps of $\mathfrak{so}(3)$, namely $\Omega^{\ydiagram{1}}$, $\Omega^{\ydiagram{3}}$, and $\Omega^{\ydiagram{5}}$. Among all 27 cases, we focus on the one case $\Phi_{[\Omega^{\ydiagram{5}}],[\Omega^{\ydiagram{5}}],[\Omega^{\ydiagram{5}}]}$. Following the same steps as for qutrit SO(3)-covariant channels, but here for $d = 5$, then $\Phi_{[\Omega^{\ydiagram{5}}],[\Omega^{\ydiagram{5}}],[\Omega^{\ydiagram{5}}]}$ is described by five Kraus operators that are irreducible rank-2 spherical tensors

$$A_2 = \frac{1}{2}\begin{pmatrix} 0 & 0 & 2a & 0 & 0 \\ 0 & 0 & 0 & \sqrt{6}a & 0 \\ 0 & 0 & 0 & 0 & 2a \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

$$A_1 = \frac{1}{2}\begin{pmatrix} 0 & -\sqrt{6}a & 0 & 0 & 0 \\ 0 & 0 & -a & 0 & 0 \\ 0 & 0 & 0 & a & 0 \\ 0 & 0 & 0 & 0 & \sqrt{6}a \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{71}$$

and

$$A_0 = \frac{a}{2}\,\mathrm{diag}\,(2, -1, -2, -1, 2),\ A_{-1} = -A_1^{\mathrm{T}},\ A_{-2} = A_2^{\mathrm{T}},\ a \in \mathbb{C}, \tag{72}$$

which satisfy the trace-preserving condition (4) for $|a| = \sqrt{2/7}$. The operators in $\mathcal{S}$ (9) are linearly independent; hence, the ququint channel $\Phi_{[\Omega^{\ydiagram{5}}],[\Omega^{\ydiagram{5}}],[\Omega^{\ydiagram{5}}]}$ is extreme.

6. *SU(2)-covariant channels*

In this subsubsection, we consider SU(2)-covariant channels. As $\mathfrak{su}(2) \simeq \mathfrak{so}(3)$ discussed in Sec. IV C 5, much of the analysis here is similar, but SU(2) admits both even- and odd-dimensional irreps whereas SO(3) only admits odd-dimensional irreps. The ladder and Cartan operators are $J_{\pm}$ and J_z, respectively, and the Casimir invariant is J^2 with spectrum $\{j(j+1); 2j \in \mathbb{N}\}$. Unitary irreps of $\mathfrak{su}(2)$ are given by

$$\Omega^{(2j+1)}(J_{\pm}) = \sum_{m=-j}^{j} C_{jm}^{\pm} |j, m \pm 1\rangle \langle j, m|,\ \Omega^{(2j+1)}(J_z) = \sum_{m=-j}^{j} m\, |j, m\rangle \langle j, m| \tag{73}$$

with $\{|j, m\rangle\}$ the weight states labeled by Casimir and Cartan operators, respectively, and

$$C_{jm}^{\pm} = \sqrt{j(j+1) - m(m \pm 1)} \tag{74}$$

the Clebsch-Gordan coefficients. As in Sec. IV C 5, the case $j = 0$ yields the trivial representation for $\Omega^{(1)}$ in Eq. (73).

a. Qudit SU(2)-covariant generalized-extreme channels: For qudit channels, the dimension of both $D^{(1)}$ and $D^{(2)}$ is d. According to Remark 5, among uncountable d-dimensional unitary representations of SU(2), we only consider one of many inequivalent representations of SU(2). Similar to the case of SO(3)-covariant qudit channels, the number of inequivalent d-dimensional representations of SU(2) is finite. For SO(3) the total number equals the number of partitions of d into odd-number components, but for SU(2), it equals the number of partitions of d into even- and odd-number components because irreps of $\mathfrak{su}(2)$ have both even- and odd-dimensional irreps. As an instance $\ydiagram{3}$, $\ydiagram{2,1}$, and $\ydiagram{1,1,1}$ are partitions for $d = 3$. According to Lemma 3, all SU(2)-covariant generalized-extreme channels are labeled by irreps of $\mathfrak{su}(2)$ with dimension less than or equal to d. Hence, for Ω, we have d candidates.

b. Two cases: For d-dimensional Hilbert space $\mathscr{H}_d$ among all candidates, we focus on two cases to illustrate salient points. In the first case, we clarify how the special case studied for SO(3)-covariant channels is also SU(2)-covariant channels. In the second case, we present SU(2)-covariance, which we prove to be extreme for any dimension d.

Case 1. Among all possible candidates, we pick $\Phi_{[\Omega^{(d)}],[\Omega^{(d)}],[\Omega^{(d)}]}$. For this case, following Remark 6, Eq. (20) simplifies to the commutators (21)

$$[\Omega^{(d)}(J_z), A_m] = mA_m,\ [\Omega^{(d)}(J_{\pm}), A_m] = C_{jm}^{\pm} A_{m\pm 1}. \tag{75}$$

For odd values of d, that is, for $j \in \mathbb{N}$, Eq. (65) yields Kraus operators of $\Phi_{[\Omega^{(d)}],[\Omega^{(d)}],[\Omega^{(d)}]}$ being rank-j irreducible spherical tensors, which is exactly the same as what we have for SO(3)-covariant channels $\Phi_{[\Omega^{(d)}],[\Omega^{(d)}],[\Omega^{(d)}]}$. When d is even, that is, $2j \in \mathbb{N}$, Eq. (75) yields $A_{-j \leqslant m \leqslant j} = 0$. Thus, if among all candidates we restrict our attention to $\Phi_{[\Omega^{(d)}],[\Omega^{(d)}],[\Omega^{(d)}]}$, we

get nothing more than the case we studied for SO(3)-covariant channels on odd-dimensional Hilbert space.

Case 2. Among the candidates, we pick $\Phi_{[\Omega^{(d-1)}\oplus\Omega^{(1)}],[\Omega^{(d-1)}\oplus\Omega^{(1)}],[\Omega^{(d-1)}]}$ for $\Omega^{(d-1)}$ defined in (73) with $j=\frac{d-2}{2}$. Following Remark 6, Eq. (20) simplifies to the commutators (21)

$$[\Omega^{(d-1)}(J_z)\oplus\Omega^{(1)}(J_z),A_m] = mA_m,\ [\Omega^{(d-1)}(J_\pm)\oplus\Omega^{(1)}(J_\pm),A_m]=C^{\pm}_{jm}A_{m\pm1}. \quad (76)$$

The Kraus-operator solution to Eq. (76) is [49]

$$A_m=\frac{1}{\sqrt{d-2}}\,|j=(d-2)/2,m\rangle\langle e|+(-1)^{\frac{d-2}{2}-m}\,|e\rangle \times\langle j=(d-2)/2,-m|, \quad (77)$$

represented in the orthonormal basis of Hilbert space

$$\mathscr{H}=\mathscr{H}_{d-1}\oplus\mathscr{H}_1,\ \mathscr{H}_{d-1} = \mathrm{span}\{|j=(d-1)/2,m\rangle\},\ \mathscr{H}_1=\mathrm{span}\{|e\rangle\}. \quad (78)$$

The trace-preserving condition (4) is evidently satisfied:

$$\Xi=\sum_{m=-j}^{j}A_m^\dagger A_m=\mathbb{1}_{d-1}\oplus|e\rangle\langle e|=\mathbb{1}_d. \quad (79)$$

To show that $\Phi_{[\Omega^{(d-1)}\oplus\Omega^{(1)}],[\Omega^{(d-1)}\oplus\Omega^{(1)}],[\Omega^{(d-1)}]}$ is extreme, we prove that the operators in $\mathcal{S}$ (9) are linearly independent. Hence, we assume that

$$\sum_{m,n=-j}^{j}\alpha_{mn}A_m^\dagger A_n=0,\ \alpha_{mn}\in\mathbb{C}, \quad (80)$$

and our task is to prove that $\alpha_{mn}\equiv0$ for all m and n. By replacing $A_m^\dagger$ and A_n from Eq. (77), we have

$$\sum_{m,n=-j}^{j}\alpha_{mn}A_m^\dagger A_n=\sum_{m=-j}^{j}\frac{\alpha_{mm}}{d-1}\,|e\rangle\langle e| + \sum_{m,n=-j}^{j}\alpha_{mn}(-1)^{m+n}\,|j,-m\rangle\langle j,-n| =0. \quad (81)$$

Therefore, $\alpha_{mn}=0$ for all m and n, which implies that the channel $\Phi_{[\Omega^{(d-1)}\oplus\Omega^{(1)}],[\Omega^{(d-1)}\oplus\Omega^{(1)}],[\Omega^{(d-1)}]}$ described by Kraus operators A_m (77) is an extreme channel for all d. We now illustrate explicitly for three subexamples.

c. Qubit channel: Now we consider $d=2$, hence $j=\frac{1}{2}$, for qubits and we continue the reasoning above. Channel $\Phi_{[\Omega^{(1)}\oplus\Omega^{(1)}],[\Omega^{(1)}\oplus\Omega^{(1)}],[\Omega^{(1)}]}$ is labeled by one-dimensional irrep of SU(2). Hence, it has just one Kraus operator and is a unitary channel. Following Eq. (77), its single Kraus operator is given by

$$A_0=|0,0\rangle\langle e|+|e\rangle\langle 0,0|, \quad (82)$$

which is one of the extreme channels discussed in Z_2-covariant channels in Sec. IV C 1 and is already discussed in the literature [8,34].

d. Qutrit channel: Continuing with cases of the second example, this time for qutrits, we let $d=3$ and thus set $j=\frac{1}{2}$. According to Eq. (77), the extreme channel $\Phi_{[\Omega^{(2)}\oplus\Omega^{(1)}],[\Omega^{(2)}\oplus\Omega^{(1)}],[\Omega^{(2)}]}$ is described by a pair of Kraus operators

$$A_{\pm1/2}=\frac{1}{\sqrt{2}}\,|1/2,\pm1/2\rangle\langle e|\pm|e\rangle\langle 1/2,\mp1/2|. \quad (83)$$

e. Ququart channel: Continuing with the second example, for $d=4$ we set $j=1$ and the number of Kraus operators $K=3$. According to Eq. (77), Kraus operators of the extreme ququart channel $\Phi_{[\Omega^{(3)}\oplus\Omega^{(1)}],[\Omega^{(3)}\oplus\Omega^{(1)}],[\Omega^{(3)}]}$ are

$$A_{\pm1}=\tfrac{1}{3}\,|1,\pm1\rangle\langle e|+|e\rangle\langle 1,\mp1|, \quad A_0=\tfrac{1}{3}\,|1,0\rangle\langle e|-|e\rangle\langle 1,0|. \quad (84)$$

These illustrations show how our formulation applies for all d.

V. DISCUSSION

Now we summarize and discuss our results. We begin by discussing group-covariant generalized-extreme channels. Then we explain the nature and importance of our pseudocode method and results. Finally, we discuss our explicit examples and their generalizations.

Our first results concerned establishing the mathematical framework for constructing group-covariant extreme and quasiextreme channels. Our approaches leverages off the well-understood case of qubit channels, but only a few extreme cases are known for dimension $d>2$ without providing insight into how to extend beyond these examples. The full problem of characterizing and constructing extreme channels is too daunting, so we restrict our attention to a subset of extreme channels that are group-covariant for the group either being a finite discrete group or else a compact connected Lie group. By studying this subset, we make some of the hard problems concerning constructing extreme and quasiextreme channels tractable, and, furthermore, group-covariant channels are useful and valuable in their own right.

Although previous results concern specific examples of generalized-extreme channels, we introduce a systematic method for constructing all generalized-extreme channels if they are covariant with respect to finite discrete or compact connected Lie groups. Our method labels every group-covariant channel with three unitary representations of the channel: the channel is group-covariant with respect to the first two labels, and the third label removes multiplicity. Multiplicity is removed by uniquely labeling each channel adding a label whose purpose is to distinguish between different channels that are identically labeled with respect to the first two unitary representations. After uniquely labeling channels, we prove that any group-covariant channel is a generalized-extreme channel if and only if its third label is a group irrep whose dimension does not exceed the Hilbert-space dimension. With these results, we have set the stage for a systematic method, which we formalize as algorithms expressed in pseudocode.

As algorithms and pseudocode are not common in studies of quantum information theory, we carefully developed the relevant concepts and explained how our pseudocode works in formally presenting algorithms. Importantly, our algorithms

are built not a Turing-type computational model but rather on a computational model involving complex-number arithmetic at the foundational level. Our choice of computational model reflects that our results are aimed not at solving generalized-extreme channels using computers but rather formalizing the problems and their solutions as logical steps for mathematical physics.

Thus, our systematic method for solving generalized-extreme channels is about constructing solutions for every conceivable group-covariant channel with groups of particular classes, namely finite discrete groups and compact connected Lie groups. Specifically, we establish a procedure to construct group-covariant generalized-extreme channels methodically given the name of the group and the Hilbert-space dimension. To this end, we formalize existing knowledge about groups in terms of a (hypothetical but plausible) library, which is the repository of information about groups and is called upon in our pseudocode. Our systematic method guarantees, for any Hilbert-space dimension and any valid group name, an output comprising the set of all group-covariant generalized-extreme channels, which, in the empty-set case, implies nonexistence of any group-covariant generalized-extreme channels for that group at that dimension. Furthermore, we present pseudocode for deciding whether a given channel is extreme or quasiextreme. Therefore, starting with a given dimension and a group name, and employing our first and second algorithms expressed in pseudocode, we show how to methodically construct the entire set of generalized extreme channels with an additional label conveying whether this generalized-extreme channel is extreme or quasiextreme.

We then present examples showing the application of our systematic method to these instances. These examples illustrate our methods to elucidate how our techniques work and furthermore validate our approach by showing that known results, specifically for two-dimensional $\mathbb{Z}_2$-covariant channels, are obtained using our technique. We also obtain results for new group-covariant examples, which show interesting results. For example, for three-dimensional S_3-covariant channels we obtain a continuously parametrized family of extreme channels instead of a finite number of extreme channels. Another intriguing result we find is that the extreme channel output is never empty for the group SU(2) and for any dimension greater than or equal to 2.

Furthermore, we observe that, as the group becomes larger, the number of candidates for nonunitary generalized extreme channels tends to increase, but, importantly, exceptions exist to this rule of thumb such as that shown in our results for the D_5 case. This general rule and its exceptions highlight the importance of using pseudocode for constructing group-covariant generalized-extreme channels, as studying this problem analytically is infeasible. Another important point raised in the last example concerning SU(2)-covariance shows that our algorithm yields a nonempty set of extreme channels for all Hilbert-space dimensions without exception.

VI. CONCLUSIONS

We have addressed the problem of constructing the set of extreme channels for d-dimensional Hilbert space. This well-known problem is hard because the set of channels at Hilbert-space dimension $d > 2$ has not been parametrized. Therefore, the detailed structure of the set of channels and a parametrized description of the boundary of the set are unknown, which makes it impossible currently to construct directly the extreme points of the set of channels. By considering symmetry, we can construct a subset of extreme channels, and convex combinations of these extreme channels enable parametrization of an important subset of channels.

Here we have restricted our attention to a subset of extreme channels that are covariant with respect to a finite discrete group or a compact connected Lie group. By exploiting knowledge about group and representation theory, we are able to develop a systematic approach to construct those extreme and quasiextreme channels that are group-covariant. Our systematic approach is represented by pseudocode, which makes our procedure especially clear. We present a variety of elucidating and instructive examples, which includes the proof that an extreme group-covariant channel exists for every choice of Hilbert-space dimension.

Our results extend significantly knowledge of extreme channels by going beyond the usual restriction to unital channels for $d > 2$. Furthermore, our approach reveals that the problem of constructing group-covariant generalized-extreme channels reduces to the well-studied problems of solving a system of linear and quadratic equations.

At this stage, our theory does not yet consider tensor products of Hilbert spaces. Our results could be generalized in nontrivial ways by dealing with tensor products, perhaps restricting to the same groups considered here to study group-covariant extreme channels involving this added structure. Incorporating tensor-product structure could be useful for solving problems of correlated channels [50,51].

One interesting application of our algorithmic approach could be to the problem of searching for increasingly large Holevo-capacity additivity violations; regarding the conjecture that Holevo capacity of quantum channels is additive [52], Hastings disproves this conjecture by introducing a counterexample but leaves open how to find all channels that are nonadditive and how large of an additivity violation is possible [53]. Here we suggest an algorithmic approach, based on our methods, to addressing this problem of increasing the violation.

Now we introduce additivity violation as a superadditivity computational problem of discovering channels with greater additivity violation compared to what is known now. Additivity violation of Holevo capacity for a pair of quantum channels $\Phi_{1,2}$ is quantified by

$$\begin{aligned} v(\Phi_i, \Phi_2) &:= \chi(\Phi_1 \otimes \Phi_2) - \chi(\Phi_1) - \chi(\Phi_2),\ \chi(\Phi) \\ &= \sup_{\{p_i,\rho_i\}} \left[\mathscr{S}\left(\sum_i p_i \Phi(\rho_i) - \sum_i p_i \mathscr{S}(\Phi(\rho_i)) \right) \right] \end{aligned} \tag{85}$$

if v is positive, with $\rho_i \in \mathcal{T}(\mathscr{H})$, $p_i > 0$, $\forall i$, $\sum_i p_i = 1$, and $\mathscr{S}(\rho)$ the von Neumann entropy of ρ. An algorithm for computing v (85) would accept the descriptions of two channels $\Phi_{1,2}$ and yield v as output: a value of v greater than the best-known v to date would be flagged as a success. Our algorithm yields descriptions of generalized extreme channels

as outputs, so the algorithm for computing v would call our algorithm as an oracle to obtain pairs of channels, which could be for different Hilbert-space dimension d and different group names $\{\mathcal{G}\}$.

Using this approach, we compute the additivity violation v (85) for pairs of channels drawn from the set of group-covariant generalized extreme channels. This set is smaller than the set of all channel pairs that should be searched for obtaining the largest possible v, but this restricted search is a good start to search algorithmically for the largest v over all possible channel pairs.

We emphasize some caveats on our approach to discovering increasingly large v algorithmically. As our algorithm is designed for the Blum-Shub-Smale machine [54], adapting to a Turing type of discrete computer is needed: this adaptation is achieved by working with floating numbers with a consequence that computations are then approximate rather than exact. Second, our algorithm does not generate all possible pairs of channels but rather just a restricted set of channels, so our additivity-violation algorithm would not be performed exhaustively over all pairs of channels, but if an algorithm could be devised to generate all possible channel pairs, that algorithm would supplant our own generalized extreme channel generator and then permit an exhaustive search. We also emphasize that our algorithm could be infeasible on current computers.

Another promising direction to follow would be to extend beyond pseudocodes to writing actual computer code and implementing on a computer to solve for new group-covariant extreme channels. Our results could be used to construct circuits for simulating extreme channels, which could be useful for quantum-channel simulation theoretically [12,28,32] and experimentally [29,55–57]. Finally, our theory could help to study complexity considerations associated with quantum-circuit simulation of channel-construction problems [12,28,30,32].

ACKNOWLEDGMENTS

L.M. acknowledges financial support by Sharif University of Technology, Office of Vice President for research under Grant No. G930209, and hospitality by the University of Calgary, where parts of this work were completed. B.C.S. appreciates financial support from NSERC. Both L.M. and B.C.S. acknowledge financial support from the American Physical Society through their International Research Travel Award Program, which supported a research visit to the University of Calgary. We acknowledge valuable discussions with Camila Suarez Viltres concerning pseudocode and data-type formalities.

APPENDIX A: FORMAL APPROACH

In this Appendix, we review the formal approach and main ingredients for specifying a formal problem, as well as basics for writing a formal problem as a pseudocode. We first discuss, in Appendix A 1, how we specify problems formally as such formal specifications are needed for our algorithmic approach, and we make clear our model for solving formal problems. Second we explain pseudocode, data types, and the notation we use. Next we provide background on the role of a library in writing algorithms to solve the problem. Finally, we explain the structure of any algorithm and its essentials.

1. Formal problems

In this subsection, we provide principles for specifying formal problems. Then we explain our model for stating and solving formal problems, with our model described in computational terms. A problem should be stated in a form that the reader can understand, which requires that the language for the problem be clear and the terms employed are understood by the intended audience, which can be sentient (e.g., a human) or autonomous (e.g., a computer). In our case, we would like our hard channel problems to be so clear that a machine can understand and potentially solve the problem. For this goal of autonomous solving to be met, we adopt principles from computing, which requires us to specify various structures such as data types, which could be integer or real or complex numbers or symbols, as we explain below. A formal problem has well-defined arithmetic operations such as addition and multiplication. The formal statement is lucid regarding input and output and whether the solution is obtained deterministically or probabilistically and whether exact or approximate. The procedure to map input to output is specified in terms of universal primitive operations. Library functions are permitted; such library functions offer known algorithms that solve specific problems and are in a library because these algorithms are popular for multiple applications.

Typically the Turing machine [58], or perhaps its extension to quantum Turing machines [59,60], would suffice as the starting point for the foundational model on which to base formal problems and their algorithmic solutions. However, we are not devising ways to solve group-covariant channels on a computer per se but rather seeking to formalize the mathematical problems. As these quantum channels are defined over the complex number field, we choose as our starting point the Blum-Shub-Smale machine [54], which modifies the Turing machine to allow for an uncountable alphabet corresponding to real numbers. Although we are employing complex numbers, rather than real numbers, the Blum-Shub-Smale machine can be extended to complex numbers by treating complex numbers as pairs of real numbers with complex conjugation and multiplication rules suitably incorporated. In addition to the alphabet including the full uncountable set of complex numbers, our alphabet also includes standard alphanumeric characters.

Various types of problems can be specified, such as search, optimization, function inversion, and decision. In our work, we deal with just two types of computational problems, namely function problems and decision problems. A function problem maps each input to some output according to a set of rules, such as exponentiating or taking a square root. Of course a well-posed function problem can fail to have a solution that occurs from dividing by zero. A decision problem is a special case of a function problem in that the output can only be binary, such as "true," denoted by $\top$, or "false," which is denoted by $\perp$.

A problem statement requires specification of a clear task and needs well-defined inputs and outputs including their data

types. The algorithm for solving the problem is expressed as inputs, outputs, and a step-by-step procedure to map the input to the output with these steps specified in terms of operations that are themselves constructed from a universal set of primitives, and some operations can be drawn from the library whereas others are explicit logical steps in terms of the computational model.

2. Essentials of algorithms

An algorithm is designed to solve a specific, well-posed problem, in terms of instructions that the machine can understand. Specifically, the algorithm accepts inputs, processes these inputs by a procedure expressed as a sequence of instructions, and yields the desired output. Below we describe what inputs, outputs, and procedures are as we use this terminology and employ these principles throughout our work.

a. Inputs: The first component of an algorithm is its inputs. We require, in our approach to algorithms, the value of the input and also type. Type refers to the nature of this input, which implies rules such as arithmetic or concatenation. In some programming languages, such as Python, to which we refer often, type is not needed but then an interpreter is required; as we are focused on an algorithmic approach but not on actual programming or implementations of computation, we prefer to keep all types explicitly stated.

b. Outputs: The outputs of the algorithm are the solutions of the problem. In our approach, we specify type of each output to be clear, and the format is the same as for inputs. Outputs can be exact or approximate solutions and obtained deterministically or probabilistically. Flags can be binary outputs that indicate whether a valid solution was found or not.

c. Procedures: The algorithmic procedure's purpose is to map the inputs to the outputs in a logical way following steps that the machine can understand. Procedures can be represented by flow charts or by pseudocode; we prefer using pseudocode, which includes both instructions and comments following $\triangleright$ symbols that explain briefly the instructions. Procedure statements include explicit declaration of variables, including their types, and return statements for sending variable values to outputs and end statements that terminate the procedure. We typically do not include initialization statements as we assume all numbers and numerical arrays are initialized to 0 or arrays of zeros and symbolic variables are initialized to blank $\flat$.

3. Pseudocode

In this subsection, we explain pseudocode. We have thus far explained formal problems and essentials of algorithms, and now we explain the transition from algorithms to pseudocode.

Pseudocode appears similar to how a formal program looks, but pseudocode is not meant to be compiled. Rather pseudocode formalizes the logic of how we are solving the problem, and it is an alternative to a flow-chart representation. We employ pseudocode to ensure that our algorithms are complete. Breaking what could be one large algorithm into multiple subalgorithms ensures that what would be a complicated algorithm is modularized and thus fathomable.

Our pseudocode for each algorithm and subalgorithms (treated as an algorithm) is introduced by first presenting the description of the algorithm and then presenting the name of the algorithm. Then we present sequentially the input and then the output and then the procedure. The procedure is named on formal line 1 of each pseudocode and is followed by the formal name of the algorithm and then its arguments in parentheses.

In the procedure, the pseudocode begins with importing functions from the library or other algorithms. Variables that arise during the procedure are declared as they arise. Requisite variables are returned for output before the procedure ends.

APPENDIX B: NOTATION AND DATA TYPES

In this Appendix, we establish concepts and notation for those data types [61] that are pertinent to our study and used in our pseudocode. Our two key additions to the typical notation in physics pseudocode are the use of multidimensional arrays with hyperrows and using a jagged structure. We describe these new notational results at the end of this Appendix.

We write data types in lower-case `typewriter` font and allow for both static and dynamic arrays.

Names of variables are expressed in small-capital CAMELCASE with our pseudocode presentation and notational convention closely following the qudit benchmarking case [62]. As our model includes real and complex numbers, we use these data types in that context rather than real and complex data types representing floating point numbers as in languages such as Fortran.

Now we proceed to introduce and explain specific data types. In some cases we introduce new data types for use in our studies; although some terms and types are new, these novelties are based on established concepts.

a. Numbers: First we define numbers, which can be a countable set such as integers $\mathbb{Z}$, a finite subset of integers such as $[n]$ (6) for $n \in \mathbb{Z}^+$ (positive integers denoted `posinteger`) and natural numbers $\mathbb{N}$ (`natural`) comprising positive integers and zero or the uncountably infinite numbers such as real $\mathbb{R}$ (`real`) and complex $\mathbb{C}$ (`complex`) numbers or their floating-point representations. A binary digit (bit) has type `binary` with two values 0 and 1. All numbers are subject to arithmetic operations such as addition/subtraction $\pm$, multiplication $*$, and exponentiation $\wedge$, with these operations and their character representations regarded as being defined deep in the computer architecture stack [63] in a natural way for each kind of number.

b. Non-numerical: Variables are represented by symbols (denoted `symbol`), which serve akin to variables in symbolic expressions. These symbolic variables are expressed as an alphabetical string of Latin and Greek letters in our model. The type `string` is a concatenation of alphanumeric characters, brackets, and punctuation marks, but not arithmetic operation symbols, to produce a literal constant such as "SO(3)" to denote the special orthogonal group of dimension 3.

c. Array: An array is a list of elements drawn from the same data type (e.g., `natural`, `symbol`) and can be multidimensional. We introduce the term "hyperrow" here to refer to coordinates (labeled by indices) of the array, with a hyperrow of first order referring to an array's row, a hyperrow of second

order referring to a column, a hyperrow of third order referring to, say, a sheet (following Microsoft Excel terminology), and so on. A one-dimensional array is a vector whose coordinates are row numbers. A matrix is a two-dimensional array, with coordinates given by rows and columns. This concept can be generalized to multidimensional arrays with multiple arrays. An array need not have the same number of columns for each row: instead the number of columns could vary for each row, in which case this array is "jagged." In our pseudocode, we declare arrays following the common convention, that is, expressing first the type of array and its elements data type, second the dimension of the array represented by the number of brackets, and finally the name of the array. For example,

$$\texttt{symjagged}[\,][\,]\ \textsc{arrayOne} \tag{B1}$$

declares a two-dimensional symbolic jagged array called ARRAYONE. As another example,

$$\texttt{complexarray}[2][\,]\ \textsc{arrayTwo} \tag{B2}$$

declares a two-dimensional complex array called ARRAYTWO with its row having two elements and the number of elements in its column not determined at the declaration step. We address array elements using the notation ARRAY[row number][column number] for an element of the two-dimensional array ARRAY with row number and column number. This notation is extended in a straightforward way for higher-dimensional arrays. Generalized arithmetic operations used for arrays include tensor product $\otimes$ and direct sum $\oplus$.

Previously, two-dimensional arrays have been treated in physics pseudocode, but higher-dimensional arrays, although employed in programming, have not previously been introduced in physics pseudocode. Here we produce a straightforward but useful notation for higher-dimensional arrays, exemplified by the following case. A three-dimensional array, denoted ARRAY, with elements being of any type, say, `symbol` for symbol, is declared as `symbol`[SIZE0][SIZE1][SIZE2]ARRAY, where SIZE0, SIZE1, and SIZE2, respectively, are the sizes of the first-order hyperrow, second-order hyperrow, and third-order hyperrow, respectively, for ARRAY. By convention, the index of each array starts from zero. Then ARRAY[I][J][K] is an element of ARRAY with sheet (third-order hyperrow) number K, column (second-order hyperrow) number J, and row (first-order hyperrow) number I, and this notation convention extends in an obvious way to arrays of higher dimensions.

Sometimes we need to work with a lower-dimensional array from a higher-dimensional array, which we explain here by example. Given three-dimensional array ARRAY defined above, we construct the "projected" one-dimensional array ONEDARRAY $\leftarrow$ ARRAY[I][J], which is a one-dimensional array extracted from ARRAY with SIZE2. The Kth element of ONEDARRAY, ONEDARRAY[K], is ARRAY[I][J][K]. Similarly, TWODARRAY $\leftarrow$ ARRAY[I] is a two-dimensional array extracted from ARRAY with the first hyperrow of size SIZE1 and the second hyperrow of size SIZE2.

We find necessary the use of jagged arrays, which are used in programming but have not been introduced into pseudocode within physics papers yet, so we do so here. Hence, we define a multidimensional jagged array as an array whose order n hyperrow has a size that depends on the index of the hyperrow of order $n-1$. For example, if ARRAY is a three-dimensional jagged array with elements of type symbol, its declaration is `symjagged`[SIZE0][SIZE1][ ]ARRAY, and ARRAY[I][J] is a one-dimensional array whose size depends on the chosen indices I and J.

APPENDIX C: LIBRARY

In this Appendix, we paraphrase the concept and role of libraries [64] in solving computational problems algorithmically. Then we briefly describe the functions we use for solving group-covariant channels in Secs. III C and IV B. In Appendix C 1 we explain the common functions available in public libraries that are required in our pseudocode. Then, in Appendix C 2 we discuss a convenient library of our new functions used in our pseudocode. A library is vital in computer programming and more generally in software development [64]. Libraries appear in a variety of forms, but our use of libraries here is restricted to subroutines, which are a packaged sequence of instructions that perform a well-defined task [63]. Such library elements are known as functions, and we use this terminology throughout. A function is called by invoking the name of the function and passing parameters and then receiving new parameter values after execution.

1. Functions available in a public library

In this subsection, we describe the common well-known functions that are useful for our algorithm. Specifically, we describe the matrix functions that prove useful later, and then we proceed to describe the function we use to solve simultaneously sets of equations. Finally, we describe the functions that identify symbols in expressions and simplifies algebraic expressions. Our approach to introducing library functions is to make use of concepts existing in the literature or from actual libraries such as Python's SymPy [65] or NumPy [66] or alternatives such as NAG [67].

As solving linear equations is vital to our analysis, and array manipulation is germane to solving such systems of equations, we introduce here basic array library functions that prove to be useful in our algorithm for solving generalized group-covariant channels. The zero and identity matrices are especially useful, so we introduce library functions ZERO and ID for creating zero and identity matrices of specific size, and they can be two-dimensional arrays or beyond to multidimensional cases. If a matrix has complex entries, the transpose and the Hermitian conjugate of a matrix can also be valuable, for which we use the functions TRANSP and DAG, respectively. Another important function for our purposes is RESHAPE, which converts an array of some dimension to an array of another dimension, with the easiest nontrivial examples being conversion of a vector to a matrix and vice versa. Although SymPy includes an ordering option, we only ever use one ordering here, so we do not include this input option.

Finally, we consider useful functions for solving systems of equations. The first of these library functions is SOLVE, which solves a given system of equations in terms of symbols, which are imported or given as input. Thus, SOLVE yields symbolic solutions to this set of equations in terms of specified allowed symbols. Of course these expressions can be com-

plicated, so rules are applied to simplify these expressions, and this simplifying function is called SIMPLIFY. The role of SIMPLIFY is to simplify algebraic expressions by recognizing and simplifying common or specified arithmetic expressions for symbols being both numbers and arithmetic operations, for example the replacements $5+2$ by 7 or replacement of $1+2x+x^2-(1+x)^2$ by 0 with the end result of SIMPLIFY still being a symbol, albeit simpler in form. Lastly, inspired by a SymPy library function FREE_SYM, we introduce another equivalent function in our library, which we call SYMIDENTIFIER; this function returns all symbols found in expression entries of a given multidimensional array.

2. New functions for the computer library

In this Appendix, we discuss the computer library of our new functions, which augments the set of known functions in public libraries discussed in Appendix C 1. We briefly describe each new built-in function that is used in our algorithms in Sec. IV B, and then we present the detailed information about these functions, including types of input and output. These functions are presented as their name in CAMELCASE followed by a plain description of the function, then their inputs and outputs.

(1) Oracle for group/algebra properties: Our library includes one oracular function, accepting a binary input that determines the type of the group, which is either a finite discrete group or a compact connected Lie group. Next the input of this function is a name of a finite discrete group or a compact connected Lie group. This oracular function also accepts an integer input corresponding to Hilbert-space dimension d. This function returns pertinent information about the properties of the input finite discrete group or the input compact connected Lie group, depending on the type of the group determined by the first input of the function. This oracular function yields six outputs. Depending on the type of the group in the input, the first output is the number of inequivalent irreps of the finite discrete group or the number of inequivalent irreps of Lie algebra corresponding to the compact connected Lie group with dimension less than or equal to d. The second output is the number of inequivalent representations of the finite discrete group or the number of inequivalent representations of the Lie algebra corresponding to the compact connected Lie group, with respect to the group type in the input, for given d. The third output is the rank of the finite discrete group or the number of generators of the Lie algebra corresponding to the compact connected Lie group. The fourth output is the dimension of the inequivalent irreps of the finite discrete group or the dimension of all inequivalent irreps of the compact connected Lie algebra corresponding to the compact connected Lie group, with dimension less than or equal to d. Next we have the fifth output, which is all inequivalent unitary irreps of the finite discrete group or all inequivalent unitary irreps of the Lie algebra corresponding to the compact connected Lie group with dimension less than or equal to d. Finally, the sixth output is all unitary representations of the finite discrete group or all unitary representations of the Lie algebra corresponding to the compact connected Lie group, for the given d.

• PROPS▷ Properties and representations of a discrete group or a compact connected Lie group GNAME

INPUT:

`binary[ ]` GTYPE▷ Flag: 0 for finite discrete groups and 1 for compact connected Lie groups.

`character[ ]` GNAME▷ Name of finite discrete group or compact connected Lie group.

`posinteger` HDIM ▷ Hilbert-space dimension.

OUTPUT:

`posinteger` NUMIRREP▷ Number of inequivalent irreps of the GNAME for GTYPE=0 and number of inequivalent irreps of algebra generating GNAME for GTYPE=1 with dimension less than or equal to HDIM.

`posinteger` NUMREP▷ Number of inequivalent reps of the GNAME for GTYPE=0 and number of inequivalent irreps of algebra generating GNAME for GTYPE=1 with dimension HDIM.

`posinteger` NUMGEN▷ rank of GNAME for GTYPE=0 and number of generators of the Lie algebra generating GNAME for GTYPE=1.

`posinteger[ ]` DIM▷ Dimension of NUMIRREP inequivalent irreps of GNAME with dimension less than or equal to HDIM for GTYPE=0 and dimension of NUMIRREP inequivalent irreps of Lie algebra generating GNAME with dimension less than or equal to HDIM.

`symjagged[ ][ ][ ][ ]` IRREP ▷ For GTYPE=0, entries of the first hyperrow label the NUMIRREP inequivalent irreps of GNAME with dimension less than or equal to HDIM. Entries of the second hyperrow label NUMGEN generators of GNAME. The third and fourth hyperrows are matrix elements for each irrep of GNAME with dimension less than or equal to HDIM. For GTYPE=1 entries of the first hyperrow label the NUMIRREP inequivalent irreps of Lie algebra generating GNAME with dimension less than or equal to HDIM. Entries of the second hyperrow label NUMGEN generators of the Lie algebra generating GNAME. The third and fourth hyperrows are matrix elements for each irrep of Lie algebra generating GNAME with dimension less than or equal to HDIM.

`symbol[ ][ ][ ][ ]` REP ▷ For GTYPE $=0/1$, entries of the first hyperrow label the NUMREP inequivalent reps of GNAME/Lie algebra generating GNAME with dimension HDIM. Entries of the second hyperrow label NUMGEN generators of GNAME/Lie algebra generating GNAME. The third and fourth hyperrows are matrix elements for each rep of GNAME/Lie algebra generating GNAME with dimension HDIM.

(2) Imposing a trace-preserving condition on completely positive maps: Our library requires a function for accepting N number of $d\times d$ Kraus operators of a completely positive map and returning N number of $d\times d$ Kraus operators of the same completely positive map that satisfy the trace-preserving condition in Eq. (4). Given the set of Kraus operators, this function constructs Ξ in Eq. (5) and solves the matrix equation in Eq. (4) for parameters in Kraus operators of the completely positive map. If the solution exists, it employs the solution to simplify the Kraus operators and returns 1 for TP, indicating that the trace-preserving condition is satisfied in addition to the set of Kraus operators satisfying the trace-preserving condition (4). If the solution does not exist, the

function returns 0 for TP, indicating that the solution does not exist.

• SOLVECHANNEL▷ Apply the trace-preserving condition to the input CP map to yield either a channel or no solution.

INPUT:

`symbol`[N][D][D] KRAUS▷ ND × D Kraus matrices for CP map.

OUTPUT:

`binary`[ ] TP▷ Flag: 0 for no solution and 1 for existence of the solution.

`symbol`[N][D][D] KRAUS▷ ND × D Kraus matrices for channel.

PROCEDURE:

(a) `symbol`[D][D] XI▷ Represents Ξ (5).

(b) `symbol`[$\frac{\mathrm{D(D-1)}}{2}$] EQS▷ Constrained expressions due to the trace-preserving condition.

(c) `symbol`[ ] SYMBOLS▷ Yields parameters obtained from KRAUS.

(d) `symbol`[ ] SOL▷ Relations between parameters in KRAUS due to the trace-preserving constraint; same size as SYMBOLS.

(e) `binary` FLAG▷ TRUE if solutions exist.

(f) Compute XI▷ From Eq. (5)

(g) For all I ∈ [D], EQS[I]← XI[I][I] − 1▷ Apply trace-preserving condition for diagonal elements.

(h) For I from D + 1 to $\frac{\mathrm{D(D-1)}}{2}$, EQS gets upper-diagonal elements of XI▷ Apply trace-preserving condition for upper-diagonal elements.

(i) SYMBOLS← SYMIDENTIFIER(KRAUS) ▷ Extracts symbols from KRAUS.

(j) SOL← SOLVE(EQS,PARAMS) ▷ Assigns EQS=0, then solves for PARAMS; FLAG←FALSE if a solution is not found.

(k) If FLAG, proceed to the next step.

(l) RETURN KRAUS← SIMPLIFY(KRAUS,SOL) ▷ Employ expressions in SOL to simplify KRAUS

(3) Solving systems of linear equations: The next function solves systems of homogeneous linear equations symbolically; this system is $A_n\boldsymbol{x} = \mathbf{0}$ for $(d_1 \times d_2)$-dimensional symbolic matrices A_ns with $n \in [N]$. Solving this system of equations is accomplished by computing the intersection between kernels of $\{A_n\}$, which is accomplished by solving a system of linear homogeneous equations. The matrix of coefficients in this system of coupled linear equations is an $Nd_1 \times d_2$ matrix constructed by stacking N instances of A_n matrices. This system of homogeneous linear equations is solved by standard methods such as singular value decomposition.

• SYMSOLVE▷ Solves symbolic $\boldsymbol{x}$ for $A_n\boldsymbol{x} = \mathbf{0}\forall n \in [N]$, with each A_n a K × L symbolic matrix, and SIMPLIFY employed to simplify all algebraic expressions.

INPUT: `symbol`[N][K][L] A

OUTPUT: `symbol`[L] X

(4) Reshape: Our library incorporates a function that reshapes the input symbolic vector into a square matrix, where the input vector has a length that is a squared integer d^2 and the matrix has size $d \times d$. Reshaping is accomplished by writing in order each element of the vector into each element of the first row of the matrix until that row is full. Then we continue by writing the next elements of the vector into the next row of the matrix until that row is filled. This procedure is complete when all rows of the vector are written into all elements of the square matrix, and we have ensured that our matrix is exactly the right size for this transcription from a length d^2 vector to work properly.

• RESHAPE▷ Reshape symbolic VECTOR Appendix C 1 but here only for a vector of squared-integer length to a square matrix, converted according to the rule that the first row fills the matrix, then the second row, and so on.

INPUT: `symbol`[DIM^2] VECTOR

OUTPUT: `symbol`[DIM][DIM] MATRIX

(5) Convert a vector to multiple matrices: Our library furthermore requires a function that converts a given vector to a set of square matrices, which generalizes the previous function that maps a vector to a single square matrix. The input is a symbolic vector of size $Kd^2 \times 1$ yielding K number of $d \times d$ matrices at the output. This function chops an input vector of size Kd^2 into K length d^2 vectors and then reshapes each of these vectors into a square $d \times d$ matrix.

• VECTOMATR▷ Chops a length-KD^2 symbolic vector into K length D^2 vectors and RESHAPE each vector to a square D-dimensional symbolic.

INPUT:

`symbol`[K ∗ D^2] VECTOR,

`posinteger` K

OUTPUT: `symbol`[K][D][D] MATRIXSEQ

[1] G. Ludwig, Attempt of an axiomatic foundation of quantum mechanics and more general theories. III, Commun. Math. Phys. **9**, 1 (1968).

[2] K. Hellwig and K. Kraus, Pure operations and measurements, Commun. Math. Phys. **11**, 214 (1969).

[3] S. Doplicher, R. Hagg, and J. E. Roberts, Local observables and particle statistics I, Commun. Math. Phys. **23**, 199 (1971).

[4] A. S. Holevo, *Statistical Structure of Quantum Theory*, Lecture Notes in Physics Monographs Vol. 67 (Springer-Verlag, Berlin, 2001).

[5] C. E. Shannon, A mathematical theory of communication, Bell Syst. Tech. J. **27**, 379 (1948).

[6] C. E. Shannon, A mathematical theory of communication, Bell Syst. Tech. J. **27**, 623 (1948).

[7] L. J. Landau and R. F. Streater, On birkhoff's theorem for doubly stochastic completely positive maps of matrix algebras, Lin. Alg. Appl. **193**, 107 (1993).

[8] M.-D. Choi, Completely positive linear maps on complex matrices, Lin. Alg. Appl. **10**, 285 (1975).

[9] A. Fujiwara and P. Algoet, Affine parameterization of quantum channels, in *Proceedings of the 1998 IEEE International Symposium on Information Theory* (IEEE, Piscataway, NJ, 1998), p. 87.

[10] A. Fujiwara and P. Algoet, One-to-one parametrization of quantum channels, Phys. Rev. A **59**, 3290 (1999).

[11] M. B. Ruskai, S. Szarek, and E. Werner, An analysis of completely positive trace-preserving maps on m_2, Lin. Alg. Appl. **347**, 159 (2002).

[12] D. Braun, O. Giraud, I. Nechita, C. Pellegrini, and M. Žnidarič, A universal set of qubit quantum channels, J. Phys. A **47**, 135302 (2014).

[13] C. B. Mendl and M. M. Wolf, Unital quantum channels–convex structure and revivals of birkhoff's theorem, Commun. Math. Phys. **289**, 1057 (2009).

[14] U. Haagerup, M. Musat, and M. B. Ruskai, Extreme points and factorizability for new classes of unital quantum channels, Ann. Henri Poincaré **22**, 3455 (2021).

[15] D. Pérez-García, M. M. Wolf, D. Petz, and M. B. Ruskai, Contractivity of positive and trace-preserving maps under lp norms, J. Math. Phys. **47**, 083506 (2006).

[16] C. King, Additivity for unital qubit channels, J. Math. Phys. **43**, 4641 (2002).

[17] M. Fukuda, Simplification of additivity conjecture in quantum information theory, Quantum Inf. Proc. **6**, 179 (2007).

[18] R. P. Feynman, Simulating physics with computers, Int. J. Theor. Phys. **21**, 467 (1982).

[19] S. Lloyd, Universal quantum simulators, Science **273**, 1073 (1996).

[20] I. Buluta and F. Nori, Quantum simulators, Science **326**, 108 (2009).

[21] D. Aharonov and A. Ta-Shma, Adiabatic quantum state generation and statistical zero knowledge, in *Proceedings of the Thirty-Fifth Annual ACM Symposium on Theory of Computing*, STOC '03 (Association for Computing Machinery, New York, 2003), pp. 20–29.

[22] D. W. Berry, G. Ahokas, R. Cleve, and B. C. Sanders, Efficient quantum algorithms for simulating sparse Hamiltonians, Commun. Math. Phys. **270**, 359 (2007).

[23] A. M. Childs, On the relationship between continuous- and discrete-time quantum walk, Commun. Math. Phys. **294**, 581 (2010).

[24] N. Wiebe, D. W. Berry, P. Høyer, and B. C. Sanders, Simulating quantum dynamics on a quantum computer, J. Phys. A **44**, 445308 (2011).

[25] B. C. Sanders, Efficient algorithms for universal quantum simulation, in *Reversible Computation. RC 2013*, Lecture Notes in Computer Science: Programming and Software Engineering, Vol. 7948, edited by Gerhard W. Dueck and D. Michael Miller (Springer, 2013), pp. 1–10.

[26] C. M. Dawson and M. A. Nielsen, The Solovay-Kitaev algorithm, Quantum Inf. Comput. **6**, 81 (2006).

[27] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, 2010).

[28] D.-S. Wang, D. W. Berry, M. C. de Oliveira, and B. C Sanders, Solovay-Kitaev Decomposition Strategy for Single-Qubit Channels, Phys. Rev. Lett. **111**, 130504 (2013).

[29] H. Lu, C. Liu, D.-S. Wang, L.-K. Chen, Z.-D. Li, X.-C. Yao, L. Li, N.-L. Liu, C.-Z. Peng, B. C. Sanders, Y.-A. Chen, and J.-W. Pan, Experimental quantum channel simulation, Phys. Rev. A **95**, 042310 (2017).

[30] D.-S. Wang and B. C. Sanders, Quantum circuit design for accurate simulation of qudit channels, New J. Phys. **17**, 043004 (2015).

[31] D.-S. Wang, Convex decomposition of dimension-altering quantum channels, Int. J. Quantum Inform. **14**, 1650045 (2016).

[32] R. Iten, R. Colbeck, and M. Christandl, Quantum circuits for quantum channels, Phys. Rev. A **95**, 052316 (2017).

[33] H. Scutaru, Some remarks on covariant completely positive linear maps on c*-algebras, Rep. Math. Phys. **16**, 79 (1979).

[34] M. B Ruskai, Some open problems in quantum information theory, arXiv:0708.1902.

[35] A. S. Holevo, Remarks on the classical capacity of quantum channel, arXiv:quant-ph/0212025.

[36] R. König and S. Wehner, A Strong Converse for Classical Channel Coding Using Entangled Inputs, Phys. Rev. Lett. **103**, 070504 (2009).

[37] N. Datta, M. Tomamichel, and M. M. Wilde, On the second-order asymptotics for entanglement-assisted communication, Quantum Inf. Proc. **15**, 2569 (2016).

[38] M. M. Wilde, M. Tomamichel, and M. Berta, Converse bounds for private communication over quantum channels, IEEE Trans. Inf. Theor. **63**, 1792 (2017).

[39] S. Das, S. Bäuml, and M. M. Wilde, Entanglement and secret-key-agreement capacities of bipartite quantum interactions and read-only memory devices, Phys. Rev. A **101**, 012344 (2020).

[40] V. Karimipour, A. Mani, and L. Memarzadeh, Characterization of qutrit channels in terms of their covariance and symmetry properties, Phys. Rev. A **84**, 012321 (2011).

[41] M. Mozrzymas, M. Studziński, and N. Datta, Structure of irreducibly covariant quantum channels for finite groups, J. Math. Phys. **58**, 052204 (2017).

[42] K. Siudzińska and D. Chruściński, Quantum channels irreducibly covariant with respect to the finite group generated by the weyl operators, J. Math. Phys. **59**, 033508 (2018).

[43] N. Datta, M. Fukuda, and A. S. Holevo, Complementarity and additivity for covariant channels, Quantum Inf. Proc. **5**, 179 (2006).

[44] S. Arora and B. Barak, *Computational Complexity: A Modern Approach* (Cambridge University Press, Cambridge, 2009).

[45] H. S. M. Coxeter and W. O. J. Moser, *Generators and Relations for Discrete Groups* (Springer-Verlag, Berlin, 1972), Vol. 14.

[46] A. Barut and R. Raczka, *Theory of Group Representations and Applications* (World Scientific, Singapore, 1986).

[47] I. N. Herstein, *Topics in Algebra*, 2nd ed. (John Wiley & Sons, New York, 1975).

[48] A. Young, On quantitative substitutional analysis, Proc. London Math. Soc. **s1-33**, 97 (1900).

[49] V. Karimipour and L. Memarzadeh, Matrix product representations for all valence bond states, Phys. Rev. B **77**, 094416 (2008).

[50] L. M. Duan and G. C. Guo, Reducing decoherence in quantum-computer memory with all quantum bits coupling to the same environment, Phys. Rev. A **57**, 737 (1998).

[51] C. Macchiavello and G. M. Palma, Entanglement-enhanced information transmission over a quantum channel with correlated noise, Phys. Rev. A **65**, 050301(R) (2002).

[52] A. S. Holevo, The additivity problem in quantum information theory, in *Proceedings of the International Congress of Mathematicians* (European Mathematical Society, 2006), pp. 999–1018.

[53] M. B. Hastings, Superadditivity of communication capacity using entangled inputs, Nat. Phys. **5**, 255 (2009).

[54] L. Blum, M. Shub, and S. Smale, On a theory of computation and complexity over the real numbers: Np-completeness, recursive functions and universal machines, Bull. Am. Math. Soc. **21**, 1 (1989).

[55] Y.-C. Jeong, J.-C. Lee, and Y.-H. Kim, Experimental implementation of a fully controllable depolarizing quantum operation, Phys. Rev. A **87**, 014301 (2013).

[56] A. Shaham and H. S. Eisenberg, Realizing a variable isotropic depolarizer, Opt. Lett. **37**, 2643 (2012).

[57] F. Sciarrino, C. Sias, M. Ricci, and F. De Martini, Realization of universal optimal quantum machines by projective operators and stochastic maps, Phys. Rev. A **70**, 052305 (2004).

[58] A. M. Turing, On computable numbers, with an application to the entscheidungsproblem, Proc. London Math. Soc. **s2-42**, 230 (1937).

[59] A. Chi-Chih Yao, Quantum circuit complexity, in *Proceedings of the 1993 IEEE 34th Annual Foundations of Computer Science, SFCS '93* (IEEE Computer Society, USA, 1993), pp. 352–361.

[60] A. Molina and J. Watrous, Proc. R. Soc. A **475**, 20180767 (2019).

[61] D. L. Parnas, J. E. Shore, and D. Weiss, Abstract types defined as classes of variables, in *Proceedings of the 1976 Conference on Data: Abstraction, Definition and Structure* (Association for Computing Machinery, New York, 1976), pp. 149–154.

[62] M. Jafarzadeh, Y.-D. Wu, Y. R. Sanders, and B. C. Sanders, Randomized benchmarking for qudit clifford gates, New J. Phys. **22**, 063014 (2020).

[63] D. E. Knuth, *The Art of Computer Programming*, 3rd ed. (Addison-Wesley Longman, Reading, MA, 1997), Vol. 1.

[64] M. Wilkes, D. Wheeler, and S. Gill, *The Preparation of Programs for an Electronic Digital Computer: With Special Reference to the EDSAC and the Use of a Library of Subroutines* (Addison-Wesley, Boston, 1951).

[65] A. Meurer, C. P. Smith, M. Paprocki, O. Čertík, S. B. Kirpichev, M. Rocklin, A. Kumar, S. Ivanov, J. K. Moore, S. Singh, T. Rathnayake, S. Vig, B. E. Granger, R. P. Muller, F. Bonazzi, H. Gupta, S. Vats, F. Johansson, F. Pedregosa, M. J. Curry *et al.*, SymPy: Symbolic Computing in Python, PeerJ Comput. Sci. **3**, e103 (2017).

[66] T. Oliphant, *Guide to NumPy* (CreateSpace Independent Publishing Platform, 2015).

[67] The NAG Library, The Numerical Algorithms Group (NAG), Oxford, United Kingdom, www.nag.com.