Qudit Colour Codes and Gauge Colour Codes in All Spatial Dimensions

Fern H.E. Watson, 1, 2, * Earl T. Campbell, 3 Hussain Anwar, 4 and Dan E. Browne 1

1 Department of Physics and Astronomy, University College London, Gower Street, London WC1E 6BT, UK.
2 Department of Physics, Imperial College London, Prince Consort Road, London SW7 2AZ, UK.
3 Department of Physics and Astronomy, University of Sheffield, Sheffield, S3 7RH, UK.
4 Department of Mathematical Sciences, Brunel University, Uxbridge, Middlesex UB8 3PH, UK.

Two-level quantum systems, qubits, are not the only basis for quantum computation. Advantages exist in using qudits, d-level quantum systems, as the basic carrier of quantum information. We show that colour codes—a class of topological quantum codes with remarkable transversality properties—can be generalised to the qudit paradigm. In recent developments it was found that in three spatial dimensions a qubit colour code can support a transversal non-Clifford gate, and that in higher spatial dimensions additional non-Clifford gates can be found, saturating Bravyi and König’s bound [Phys. Rev. Lett. 110, 170503 (2013)]. Furthermore, by using gauge fixing techniques, an effective set of Clifford gates can be achieved, removing the need for state distillation. We show that the qudit colour code can support the qudit analogues of these gates, and show that in higher spatial dimensions a colour code can support a phase gate from higher levels of the Clifford hierarchy which can be proven to saturate Bravyi and König’s bound in all but a finite number of special cases. The methodology used is a generalisation of Bravyi and Haah’s method of triorthogonal matrices [Phys. Rev. A 86 052329 (2012)], which may be of independent interest. For completeness, we show explicitly that the qudit colour codes generalise to gauge colour codes, and share the many of the favourable properties of their qubit counterparts.

I. INTRODUCTION

Quantum technologies are often developed in the qubit paradigm, where the basic carrier of quantum information is a two-level quantum system. Qubits are a natural choice because binary is the language of classical technologies. However, even here, despite the prevalence of binary, its supremacy is questionable. Indeed, Donald Knuth has advocated the use of balanced ternary, a 3-state classical logic [1]. In the quantum domain, qudits offer a state space with a richer structure than their two-level counterparts, and the merits of this for quantum information have been explored in many contexts [2–10]. Recent experiments have shown even large d quantum systems can be precisely controlled [11, 12]. Here we present a qudit generalisation of a powerful class of quantum error correcting codes, the colour codes [13–17]. Along with surface codes [18, 19], they constitute the most successful topological codes. Using topology, quantum codes have achieved high error thresholds, whilst proving more practical than concatenated codes. Thresholds of qudit surface codes indicate improvements with qudit dimension [7, 9]. Qubit colour codes have several advantages over qubit surface codes, and we show these features can be transferred over into the qudit setting.

* fern.watson10@imperial.ac.uk
There has been some prior investigation into qudit colour codes for prime-power $d$ \cite{20} and colour codes based on more general groups \cite{21}, though these works were restricted to 2D topologies. In this paper, we generalise colour codes to any qudit dimension $d$ and spatial dimensions $\mu$ up to the point where $\mu$ factorial is a multiple of $\mu$. Specifically, given any lattice suitable for constructing qubit colour codes, we show how to use the same lattice to construct a qudit colour code. For qubits, a non-Clifford gate can be implemented in colour codes in 3 and higher spatial dimensions transversally, i.e. by a tensor product of local unitary gates, an inherently fault-tolerant procedure \cite{22, 23}. To avoid confusion between the spatial dimension of the lattice and the Hilbert space dimension of the qudit, we shall always denote the former by the letter $\mu$ and the latter by the letter $d$.

Recently, it has been shown by Bravyi and König \cite{25} that a quantum error correcting code in $\mu$ spatial dimensions can support a gate with constant depth from at most the $\mu$th level of the Clifford hierarchy. The fact that colour codes can be shown to saturate this bound with transversal gates is a very promising feature, and when combined with gauge fixing techniques \cite{15, 26, 28} enables universal quantum computation without the need of magic state distillation \cite{5, 6, 29, 34}. The structure of the qudit Clifford group is very different from its qubit counterpart \cite{4, 35}. Nevertheless, we find that 3D colour codes also provide transversal non-Cliffords in the qudit case.

Recently the colour codes were generalised to gauge colour codes \cite{36}—subsystem codes with many advantageous features—including low weight error detection measurements, universal transversal gates via gauge fixing for $\mu > 2$, fault-tolerant conversion \cite{27} between codes of different spatial dimension \cite{37}, and for the $\mu = 3$ case, single shot error correction \cite{38}—a robustness to measurement errors without the need for repeated measurements. We show that the qudit colour codes introduced here can also be generalised to gauge colour codes.

The main technique which we employ is a bipartition of the vertices in the graph that defines the code into starred and unstarred vertices. We call this the star-bipartition, to distinguish it from the other important colourings which define the colour codes. The commutation properties of the stabilizer and logical operators of the colour codes (and gauge colour codes) in the qubit setting can be reduced to the fact that the pairs of operators $X \otimes X$ and $Z \otimes Z$ commute. The star-bipartition we introduce replaces operators with their complex conjugate on a subset of the vertices, for example, replacing the above operators with $X \otimes X^*$ and $Z \otimes Z^*$, respectively. Crucially, this latter pair of operators does commute for qudits of any dimension, and this becomes the starting point for a generalisation of the colour codes from qubits to qudits of any Hilbert space dimension $d$ and any spatial dimension $\mu$.

Furthermore, the star-bipartition provides a general framework for constructing transversal gates from higher levels of the Clifford hierarchy. While elements of this technique can be seen in earlier work \cite{20, 36}, this is the first time that it has been exploited systematically. The second key technical component of our work is a generalisation of the triorthogonal matrix technique by Bravyi and Haah \cite{31}.

This paper is structured as follows. We start in Sec. II by reviewing the stabilizer formalism for $d$-level systems. In Sec. III we describe how to generalise a qubit colour code in arbitrary spatial dimensions to a qudit colour code by employing the notion of star-conjugation. In Sec. IV we review and generalise the triorthogonal matrix construction. In Sec. V we derive conditions on the lattice that must hold for a transversal non-Clifford gate to be implementable on the code. Sec. VI explains how the Hadamard can be implemented
on the same lattice, although not in the colour code, using the technique of gauge fixing. In Sec. [VII] we show gauge colour codes can naturally be defined for all the codes we have introduced, inheriting the favourable features of the qubit gauge colour codes. We conclude in Sec. [VIII].

II. QU迪T STABILIZER CODES AND THE CLIFFORD HIERARCHY

We will consider $d$-level quantum systems (qudits) as the building blocks for the constructions of quantum codes [2]. Unless stated otherwise, the qudit dimension $d$ is assumed to be any integer greater than two. The conventional basis states for the $d$-level system are taken to be $|j\rangle$, for $j \in \mathbb{Z}_d$. The single qubit Pauli matrices $X$ and $Z$ have natural extensions in higher dimensions [2]. The qudit analogues are

$$X = \sum_{j \in \mathbb{Z}_d} |j + 1\rangle \langle j|, \quad Z = \sum_{j \in \mathbb{Z}_d} \omega^j |j\rangle \langle j|,$$

where $\omega = e^{2\pi i / d}$. With a slight abuse of terminology, we shall say that two operators $A$ and $B$ “$\omega$-commute” if $AB = \omega BA$ and note that $\omega$-commutation holds for $X$ and $Z$. These generalised operators simplify to the familiar Pauli operators for $d = 2$.

The single qudit Pauli group is generated (up to global phases) by $X$ and $Z$ and the $n$-qudit Pauli group $\mathcal{P}^{\otimes n}$ is the $n$-fold tensor product of the single qudit Pauli group. Consider an abelian subgroup $S \subset \mathcal{P}^{\otimes n}$, such that $\omega^j |j\rangle \in S$ for nonzero $j$, then we say that $S$ is a stabilizer group, and we refer to its elements as the stabilizers. The stabilizer group defines an error correcting code with codewords corresponding to states $|\psi\rangle$ that are stabilized by the stabilizers, i.e. $S |\psi\rangle = |\psi\rangle$ for all $S \in S$.

The logical operators correspond to the set of operators that commute with $S$ but are not contained in it. A pair of logical operators $\bar{X}_i$ and $\bar{Z}_i$ $\omega$-commute with each other and hence encode one qudit.

Gottesman and Chuang [39] introduced a classification of quantum gates known as the Clifford hierarchy (CH), which can be defined recursively [40] as

$$\mathcal{P}_l = \{U|P^{\dagger}UPU^{\dagger} \in \mathcal{P}_{l-1} \forall P \in \mathcal{P}_1\},$$

where $\mathcal{P}_1$ is the Pauli group. For example, $\mathcal{P}_2$ is the group of operators that leave the Pauli group invariant under conjugation and is called the Clifford group. We shall describe some important Clifford group gates $H$, $S$ and $\Lambda(X)$ below. In prime qudit dimensions, these are known to generate the whole Clifford group [2, 41]. The gate $H$ is the qudit version of the Hadamard gate (also known as the discrete Fourier transform),

$$H = \frac{1}{\sqrt{d}} \sum_{j,k \in \mathbb{Z}_d} \omega^{jk} |j\rangle \langle k|,$$

the $S$ gate is the generalisation of the qubit $\pi/4$-phase gate,

$$S = \sum_{j \in \mathbb{Z}_d} \omega^{j^2} |j\rangle \langle j|.$$
and $\Lambda(X)$ is the controlled-$X$ gate (also known as the SUM gate),

$$\Lambda(X) = \sum_{j,k \in \mathbb{Z}_d} |j\rangle_c |k \oplus 1\rangle_t \langle j|_c \langle k|_t,$$  \hspace{1cm} (5)

where $c$ and $t$ are the control and target qudits, respectively.

The set of gates $P_l$ in the hierarchy contains all gates from lower levels of the hierarchy. To refer to gates in level $l$ of the hierarchy but not level $l-1$ we shall say that the level $l$ of the hierarchy is the lowest level of the hierarchy for which the gate is a member.

In prime dimensions, it is known that it suffices to supplement the Clifford group with just one non-Clifford gate from the third level of the CH in order to obtain a universal set of gates \cite{6}. Such a gate is not unique, and in the qubit case, the $T$ gate $\text{diag}(1,e^{i\pi/4})$ is usually chosen for this purpose. For the qudit case, we choose the following particularly convenient definition for the $T$ gate, which is valid in all dimensions except when $d = 2, 3, 6$ \cite{4,42},

$$T = \sum_{j \in \mathbb{Z}_d} \omega^{j^3} |j\rangle \langle j|.$$  \hspace{1cm} (6)

It is a consequence of lemmas \cite{1} and \cite{2} (below) that for $d \neq 2, 3, 6$ this gate is non-Clifford and inhabits the third level of the CH. In $d = 3$ and $d = 6$, the gate defined in equation (6) is not non-Clifford, as it reduces to the Pauli $Z$ gate since $j^3 = j \mod 3$ and mod 6. However, the following definition provides a suitable alternative $T$ gate for these dimensions. The gate is non-Clifford and in the third level of the CH:

$$T_{3,6} = \sum_{j \in \mathbb{Z}_d} \gamma^{j^3} |j\rangle \langle j|.$$  \hspace{1cm} (7)

where $\gamma^3 = \omega$ and where the function in the exponent $j^3$ is evaluated in regular arithmetic (or equivalently modulo $3d$). When we refer to “the $T$ gate” in this paper we will always mean a gate of the form of equations (6) or (7), depending on the qudit dimension under consideration. For notational convenience we suppress the dependence of $T$ on $d$, since it will always be clear by the context which $T$ gate we require.

Notice how the $T$ gate has a cubic power in the exponent of $\omega$, in contrast to the quadratic power in the case of the $S$ gate. For prime dimensions, the first investigation characterising all the phase gates from the third level of the CH was performed by Howard and Vala \cite{4}. In general, there is a close correspondence between the order of the polynomial in the exponent of $\omega$ and the lowest level of the CH the phase gate belongs to. Let us define the following family of phase gates in terms of a polynomial function $f_r(j)$ of degree $r$, such that $r \leq d$ with coefficients $a_m \in \mathbb{Z}_d$, so $f_r(j) := \sum_{m=0}^{r} a_m j^m$:

$$R_{f_r} = \sum_{j \in \mathbb{Z}_d} \omega^{f_r(j)} |j\rangle \langle j|,$$  \hspace{1cm} (8)

one can then prove the following useful lemmas.

**Lemma 1.** For all $d$, all $r \leq d$ and all functions $f_r(j)$, the gate $R_{f_r}$ is in the $r$th level of the Clifford hierarchy.
Proof. The proof of this is simple and concise. We begin by calculating
\[ R_{f_{r-1}} = X^\dagger R_{f_{r}} X R_{f_{r}}^\dagger, \]
where \( f_{r-1} \) is a new function \( f_{r-1}(j) = f_r(j+1) - f_r(j) = r a_r j^{r-1} + \cdots \). Expanding out, the degree \( r \) terms cancel, so the leading term is \( r a_r j^{r-1} \) and extra terms are all degree \( r - 2 \) or smaller. We now observe that if \( r = 1 \), \( R_{f_r} \) is a Pauli operator. Using the definition of the Clifford hierarchy in equation (2), the lemma follows by induction.

Lemma 2. For all \( d \), all \( r \leq d \) and all functions \( f_r(j) \) satisfying \( r! a_r \neq 0 \) (mod \( d \)), the gate \( R_{f_r} \) is not in the \( (r-1) \)th level of the Clifford hierarchy.

Proof. If the gate is in the \( (r-1) \)th level of the hierarchy, then applying the inductive transformation in the previous proof \( r-1 \) times must return a Pauli operator, and thus applying it \( r \) times results in an operator proportional to the identity. By chaining together transformations of the form of equation (9) \( n \) times, we obtain \( f_1(j) = r! a_r j + c \). If the corresponding operator is proportional to the identity then this function must be constant, and thus \( r! a_r = 0 \).

These two lemmas provide us with a simple way to generate gates at all levels of the Clifford hierarchy as required. They fail when \( r! a_r = 0 \), which was the case above for \( r = 3 \) and \( d = 2, 3, 6 \). In those exceptional cases, gates can be discovered to moving to higher roots of unity as illustrated by equation (7).

III. QUDIT COLOUR CODES

Colour codes [13] are a class of topological qubit stabilizer codes that be defined on a topological space of any spatial dimension [43] \( \mu \geq 2 \). The \( \mu \)-dimensional manifold is cellaled into a lattice of objects called \( k \)-cells for all spatial dimensions \( 0 \leq k \leq \mu \). For example, vertices are 0-cells, edges are 1-cells and a 2-cell is a cell defined on the faces, or plaquettes, of the lattice. Such a lattice \( \mathcal{L} \) is called a \( \mu \)-colex (colex is short for “colour complex”) whenever

1. it is a cellulation of an orientable manifold without a boundary; and
2. every vertex has \((\mu + 1)\) neighbours ((\(\mu + 1\))-valency); and
3. the \( \mu \)-cells are \((\mu + 1)\)-colourable.

Any \( \mu \)-colex defines a qubit colour code. For example, the smallest 3-dimensional colour code is a 15-qubit code defined on the lattice illustrated in figure (a). The stabilizer group is generated by face operators (with a \( Z \) operators assigned to the vertices around the face or 2-cell) and cell operators (with an \( X \) operators assigned to the vertices around each 3-cell), see figure (b). Our results show a \( \mu \)-colex also defines a qudit colour code

An alternative description of a \( \mu \)-colex is that its dual \( \mathcal{L}^* \) is a simplical lattice. This is lattice where each \( k \)-cell is a simplex, an object with \( k + 1 \) vertices. In the dual, we have conditions: \((2^*)\) every \( \mu \)-cell has \((\mu + 1)\) neighbours; \((3^*)\) the vertices are \((\mu + 1)\) colourable.

Such codes can be constructed [15] by starting with a closed hyperspherical lattice and then removing a vertex to “puncture” the surface. Alternatively, as we show in
Sec. III C, one can construct them directly by defining a suitable boundary on a regular lattice structure. Such a code encodes a single qubit and has the attractive feature of a transversal non-Clifford $T$ gate. The qudit colour codes have many useful properties. For more details, we refer the reader to [13, 15, 43].

The stabilizer generators are guaranteed to commute in this construction. Those of the same type ($X$ or $Z$) trivially commute, while those of different types commute since they always meet in pairs. As already remarked, $X \otimes X$ and $Z \otimes Z$ commute only in $d = 2$. However, by replacing one operator of each pair with its complex conjugate (with respect to the computational basis) we produce a pair of operators that commute for all $d$. These are $X \otimes X^*$ and $Z \otimes Z^*$. Throughout this paper, a starred operator denotes the complex conjugate of the operator with respect to the computational basis. The sharp-eyed reader will note that $X^* = X$. We write the star explicitly on a real matrix here, and throughout, to emphasise symmetry and to simplify notation. Thus we see that as long as a pair of $X$- and $Z$-type stabilizers have in common an equal number of starred qudits, they will commute.

To define colour codes in all qudit dimensions, we need, therefore, to find a construction which allows us to take advantage of the commutation of $X \otimes X^*$ and $Z \otimes Z^*$. This can be achieved by identifying and exploiting a bipartition or bicolouring of the graph defining the lattice. To avoid confusion with other colourings of the lattice important for colour codes we call this the star-bipartition.

We note that a similar construction was defined (for prime power qudit dimension only) in [20]. Here we go further and show that the star-bipartition is the starting point for an identification of a broad family of transversal gates on these codes, including non-Clifford gates saturating Bravyi and König’s bound.

A. Star-bipartition

The star-bipartition is a bipartition of the colour code lattice $\mathcal{L}$. It thus divides the set of lattice vertices into starred and unstarred vertices where neighbouring vertices always belong to different sets. Here we prove the following useful bipartition lemmas

**Lemma 3** (Star-bipartition lemma). Let $\mathcal{L}$ be a $\mu$-colex, then its vertices can be 2-coloured into starred and unstarred sets, $v_*$ and $v_\cdot$, respectively.

Using $|\ldots|$ to denote the number of elements in a set, we also have

**Lemma 4** (Starring of cells lemma). Let $\mathcal{L}$ be a $\mu$-colex, so that its vertices are partitioned into $v_*$ and $v_\cdot$ according to lemma 3. It follows that

1. $|v_*| = |v_\cdot|$; and

2. any $k$-cell $C$ with $0 < k \leq \mu$ contains an equal number of vertices from each partition, so $|C \cap v_*| = |C \cap v_\cdot|$.

Lemmas 3 and 4 or variants thereof, were already proved in prior research [13, 15, 17, 43]. These results are especially important to qudit colour codes and play a fundamental role, and so, for completeness, we present our own proofs. Our presentation of lemma 3 has the merit of being more concise than that of Ref. [17].
Proof. To prove the star-bipartition lemma, we switch to the dual lattice $L^*$. In the dual picture, the vertices are now $(\mu + 1)$-coloured and we need to show that the $\mu$-dimensional cells can be 2-coloured. Consider a single $\mu$-simplex. By smoothly moving or rotating the $\mu$-simplex around the manifold, it defines a notion of orientation at every point on the lattice. There exist two different orientations related by a reflection. By definition, an orientable manifold is one where this orientation is well-defined. That is, on an orientable manifold one cannot generate a reflection by transporting a $\mu$-simplex around some path. Here we have used the textbook concept of orientable manifolds, and from this definition one can show that many familiar manifolds are orientable, including Euclidean, spherical and toroidal manifolds. Consider two $\mu$-simplicies that share a common $(\mu - 1)$ simplex. These $\mu$-simplicies are reflections of each other with respect to the plane of the $(\mu - 1)$ simplex. Therefore, all neighbouring $\mu$-simplicies must have opposite orientations, and these orientations provide partitioning into starred and unstarred simplicies. This proves the lemma.

Proof. Let us now turn to lemma \[4\] and begin with part 1. Consider a bipartite lattice that is $r$-valent. We count the number of edges $N_E$. Every edge is incident to one, and only one, starred vertex. Furthermore, each vertex is contained in $r$ edges, and so the total count is $N_E = |v_\star|r$. The same argument applies to unstarred vertex, so $N_E = |v_\bullet|r$. Equating $|v_\star|r = |v_\bullet|r$, we see that $r > 0$ entails $|v_\star| = |v_\bullet|$. Since a $\mu$-colex is $(\mu + 1)$-valent, we have proven part 1 of lemma \[4\].

Consider any $k$-cell $C$. It defines a sublattice $L_C$ with same vertex set as $C$. If $L_C$ is a $k$-valent with $k > 0$, then the above argument applies and $C$ contains equal number of starred and unstarred vertices. The following reasonings parallels that in Ref. \[15\]. We show explicitly that $L_C$ is a $k$-colex for the $k = \mu - 1$ case, but any $k$ is reached by iteratively applying the argument down to the desired $k$. Again, we must switch to the dual lattice where $L_C^*$ is obtained by removing a single vertex $C^*$ from $L^*$. The sublattice $L_C^*$ retains all simplices containing $C^*$, but with their dimension reduced by the removal of $C^*$. Therefore $L_C^*$ is a simplical lattice of dimension $(\mu - 1)$ and so $L$ is $\mu$-valent. Our proof only requires the correct valency, which we have shown, but note that $L_C$ also has the correct colouring for a $(\mu - 1)$-colex. Specifically, if $C'$ and $C$ intersect on some $(\mu - 1)$-cell of $L_C$, then we assign it the colour of $C'$.

The above results concern a lattice without a boundary. However, if one punctures the code removing a starred vertex then we have

**Corollary 1.** Let $L$ be puncturing of a $\mu$-colex, with the inherited bipartition into $v_\star$ and $v_\bullet$. Then

1. $|v_\star| = |v_\bullet| - 1$; and

2. any $k$-cell $C$ with $0 < k \leq \mu$ contains a equal number of vertices from each partition, so $|C \cap v_\star| = |C \cap v_\bullet|$.

Property 1 is immediately follows from point 1 of lemma \[4\] as a single starred vertex has been removed. Property 2 is identical to its partner in lemma \[4\]. The punctured lattice only keeps cells that did not contain the punctured vertex, and so the property is directly inherited. The dual lattice $L^*$ has been an essential proof tool for these lemmas, but in the rest of the paper we shall only consider the primal lattice.
FIG. 1. (a) The smallest instance of the qudit 3D colour code with the $X$-type stabilizers coloured red, green, blue and yellow. The 1-cells (edges) of the code are also coloured. The vertices are also coloured so the set $v_\bullet$ is represented by black circles and the set $v_\star$ is represented by white stars. (b) A single $Z$ stabilizer of the tetrahedral 3D colour code. The plaquette can take the colour yellow if considered as a face of the green 3-cell, and vice versa. (c) A single $X$ stabilizer of the tetrahedral 3D colour code.

B. Qudit colour codes in arbitrary spatial dimension

Before we write down the stabilizer generators for these codes, we shall use the star-bipartition to define notation for an important family of transversal operators which we call star-conjugate transversal. Throughout this paper, when discussing colour codes we identify qudits with vertices in the lattice defining the code. Let $v$ denote a set of vertices (and thus the corresponding qudits). Let us introduce the notation

$$U[v] = U^{\otimes|v|}$$

(10)

to denote the tensor product of unitary $U$ acting on each qudit identified with the vertices in $v$.

We call an operator star-conjugate transversal when it has the following form

$$\tilde{U} = U[v_\bullet] \otimes U^*[v_\star].$$

(11)

In other words, $\tilde{U}$ consists of $U$ applied to all unstarred vertices and $U^*$ applied to all starred vertices.

To define qudit colour codes we need to introduce two types of stabilizer generators which are defined with respect to cells of different dimensions within the lattice. As in the qubit case, we associate $Z$-type stabilizer generators with the $\mu'$-cells of the lattice and $X$-type stabilizers with the $(\mu - \mu' + 2)$-cells [3]. In this section, for simplicity of presentation, we will take the example of $\mu' = 2$ so the $Z$-type stabilizers are associated with plaquettes (2-cells) and $X$-type stabilizers are associated with $\mu$-cells, although the
construction holds in the more general case, which we shall consider later. For example, in 3D, the $X$ stabilizers act on the vertices contained in a 3-cell of the lattice, and the $Z$ stabilizers act on the vertices contained in a 2-cell (plaquette). This can be seen in figure 1.

Setting $\mu' = 2$ the stabilizer generators therefore take the form

$$S_{X,C} = X[v_\star \cap C] \otimes X^*[v_\star \cap C],$$

$$S_{Z,P} = Z[v_\star \cap P] \otimes Z^*[v_\star \cap P].$$

for all 2-cells $P$, and $\mu$-cells $C$ of the lattice. Recall that we write the conjugate operator $X^*$ explicitly to emphasize the ubiquity of starring, even though $X$ is a real operator in the computational basis and $X = X^*$. It is not just the $S_{Z,P}$ operators where starring is non-trivial, but many logical operators also require starring.

Let us denote the group generated by $S_{X,C}$ operators $S_X$ and $S_{Z,P}$ operators by $S_Z$. All elements of $S_X$ commute as do all elements of $S_Z$. It remains to show that all elements of $S_X$ commute with all elements of $S_Z$. This follows from the fact that $X \otimes X^*$ commutes with $Z \otimes Z^*$. The above construction ensures that whenever cell $C$ and cell $P$ overlap they overlap on an equal number of starred and unstarred vertices, which is a point discussed further in Sec. IV C. Hence these stabilizer generators commute as required. Note that the qubit $d = 2$ case is included in our definition.

We noted above that, in the constructions we consider, the code will contain one more unstarred qudit than there are starred qudits. We can thus define the logical encoded Pauli operators for the code star-conjugate transversally.

$$\tilde{X} = \tilde{X} = X[v_\star] \otimes X^*[v_\star],$$

$$\tilde{Z} = \tilde{Z} = Z[v_\star] \otimes Z^*[v_\star].$$

The fact that $|v_\star| = |v_\bullet| - 1$ ensures that the logical operators satisfy the same commutation properties as $X$ and $Z$. One can verify that these operators commute with the stabilizer operators defined above by recalling that each 2-cell $P$ and and $\mu$-cell $C$ contains an equal number of starred and unstarred vertices.

## C. Constructing codes of any code distance

In topological stabilizer codes we expect to be able to increase the number of physical systems encoding the quantum information in order to protect the information more effectively—a property characterised by the code distance. We have remarked already that suitable codes can be constructed by puncturing a hypersphere. In this section we provide, however, a constructive alternative method to show that in $\mu$ spatial dimensions the colour code distance can be made arbitrarily large.

The code distance is the weight of the smallest logical operator. It was convenient above, in equations (14), to define the logical operator basis star-conjugate transversally. However, multiplying by a subset of the stabilizer group we can localise $\tilde{Z}$ to any edge of the lattice. The code distance is then length of the shortest lattice edge. Thus by showing that the edge length can be made arbitrarily large we demonstrate the ability to create codes of any distance.

We begin with an example in 3D and then generalise the argument to higher spatial dimensions. The particular choice we make for this example is a 4-colourable tiling of
FIG. 2. To increase the distance of the code, we tile 3D space with truncated octahedron cells (with appropriate boundary conditions). Here we show a portion of the lattice. A red cell fits into the hollow formed by the green, blue and yellow cells towards the bottom of the figure, and thus the tiling proceeds, ensuring the 4-colourability of the lattice is preserved.

truncated octahedra, part of which is illustrated in figure 2. This is not a unique choice—for an alternative see [43] where the 3D colour code is defined on a lattice constructed from cubes and truncated octahedrons.

In addition to the two lattices already mentioned, we have identified two further possible choices of regular tilings in 3D Euclidean space on which a colour code may be defined. One is the cantitruncated cubic honeycomb, comprising truncated cuboctahedra (polyhedra with square, hexagonal and octahedral faces), truncated octahedra, and cubes. The second is the omnitruncated cubic honeycomb comprising truncated cuboctahedra and octahedral prisms. This list is not exhaustive, for instance many more tilings may exist in curved space.

The method proceeds by carefully cutting a block of the lattice out, in order to form a tetrahedron, similar to that illustrated in figure 1(a). The main difference is that the new tetrahedron contains more qudits and can have an arbitrarily large (though always odd, in order to ensure the \( \omega \)-commutation of the logical operators) number of qudits contained in its edges. The shape cut from the lattice may not look tetrahedral at first but may be deformed to a regular tetrahedron, with vertices belonging only to a single cell (as opposed to those contained in two or more cells of the lattice) forming the four vertices of the tetrahedron.

In \( \mu \) spatial dimensions a suitable tiling of the space as outlined at the beginning of section III must be found. Besides these requirements placed on the lattice construction, the ‘block’ (polytope) cut out of the lattice to form the higher-dimensional analogue of the tetrahedron must adhere to certain rules.

1. The polytope formed must have \((\mu + 1)\) boundaries, each of a different colour. The colour of the boundary corresponds to the colour of the string-like operators that can end on that boundary.

2. Every edge of the polytope must contain an odd number of qudits. As stated above, this is to ensure the \( \omega \)-commutation of the logical operators is preserved.

3. There should be exactly \( \mu \) vertices that belong only to a single cell of the lattice.
These cells should be different colours and these vertices will form the vertices of the polytope.

D. Error detection

Errors are detected by measuring all the stabilizer generators. We shall not present a detailed analysis here, since errors and syndrome are related for these codes in a similar way to the qubit case. For example, in 3 spatial dimensions a single $Z$-type error on an unstarred qudit will result in a measured eigenvalue of $\omega$ in the $X$ stabilizers corresponding to the four cells containing that vertex. Similarly a single $X$-type error on an unstarred qudit will result in a measured eigenvalue of $\pm \omega$ in the six faces that contain the vertex. A $Z^k$ or $X^k$ error will lead to a measurement outcome of $\omega^\pm k$ respectively. Syndromes arising from sets of multiple errors can be calculated using standard techniques.

A classical algorithm called a decoder may be used to interpret the syndrome and infer the correction operator that must be applied to return the code to its original state. Some examples of decoding the colour code in 2D are given in Refs. [44–49]. The development of efficient decoders for the colour code with $\mu > 2$ is a topic of active research. We shall not present a decoder in this paper, but remark that, unlike in qubit codes where neighbouring errors lead to cancellation of syndrome in between, this does not always occur in larger dimension, and thus in general more information is available to the decoder as the qudit dimension increases [9].

IV. $m^*$-ORTHOGONAL MATRICES AND CODES

Brayvi and Haah introduced a powerful framework for defining codes with a transversal non-Clifford gate; codes defined in terms of triorthogonal matrices. Here we introduce a significant generalisation to Bravyi and Haah’s approach, extending to qudits codes with transversal gates from higher up the Clifford hierarchy.

A. Matrix representation of quantum codes

We begin by reviewing how Bravyi and Haah represent CSS codes with matrices [31]. They use a matrix $G$ over $\mathbb{Z}_2$ that has linearly independent rows under modulo 2 arithmetic. The matrix $G$ is broken up into two blocks $G_1$ (with $k$ rows) and $G_0$ (with $s$ rows). This defines a quantum code with $k$ logical qubits. The elementary logical basis state $|0_L\rangle$ is written

$$|0_L\rangle = \frac{1}{\sqrt{2^s}} \sum_{f \in \text{span}(G_0)} |f\rangle = \frac{1}{\sqrt{2^s}} \sum_{y \in \mathbb{Z}_2^s} |y^T \cdot G_0\rangle.$$  \hspace{1cm} (15)

where ‘$\cdot$’ denotes matrix multiplication modulo $d$, and we take advantage of the fact that the rows of $G_0$ are linearly independent to represent the terms in the superposition by a row vector $y$ transposed and multiplied with $G_0$. For the other logical computational basis
states $|x_L\rangle$ where $x \in \mathbb{Z}_2^k$, we have

$$|x_L\rangle = \frac{1}{\sqrt{2^k}} \sum_{y \in \mathbb{Z}_2^k} |y^T \cdot G_0 \oplus x^T \cdot G_1\rangle,$$

(16)

where addition of row vectors is elementwise modulo 2 as acknowledged by the symbol $\oplus$. Note that in the special case that $k = 1$, $x$ is a scalar (or a $1 \times 1$ row vector) and the transpose operation is trivial. For qudits, quantum codes are defined by a matrix $G$ taking elements from $\mathbb{Z}_d$, so that

$$|x_L\rangle = \frac{1}{\sqrt{d}} \sum_{y \in \mathbb{Z}_d^s} |y^T \cdot G_0 \oplus x^T \cdot G_1\rangle.$$

(17)

The key differences are that we now sum over all $y \in \mathbb{Z}_d^s$ and that arithmetic is modulo $d$.

In the CSS formalism, a quantum code has a stabilizer group which is generated by a product of two distinct generating sets, the $X$-stabilizer generators, which are tensor products of $X$ and $I$ alone and the $Z$-stabilizer generators, which are tensor products of $Z$ and $I$ alone. In addition, one must define logical operators, and in the CSS formalism, logical encoded $X$ operators consist of a tensor product of $X$ and (optionally) $I$ alone, and the logical encoded $Z$ consists of a tensor product of $Z$ and (optionally) $I$ alone. Once the $X$-stabilizer generators and logical $X$ operator (or operators if there are multiple encoded qubits) are defined, the remaining $Z$-stabilizer generators are fixed by conjugation relations which can be elegantly captured via the use of dual codes from classical coding theory. In the qudit setting, $G_0$ will again define the $X$-stabilizer generators for the code, via the mapping $k$ to $X^k$, e.g. the row vector 000123 defines the operator $I \otimes I \otimes I \otimes X \otimes X^2 \otimes X^3$, and $G_1$ will similarly define the logical $X$ operators for the code.

B. Defining $m^\star$-orthogonality

Bravyi and Haah [31] prove that a so-called triorthogonal code supports a transversal non-Clifford gate in the 3rd level of the Clifford hierarchy. We now generalise Bravyi and Haah’s construction. First, we focus on generalising from qubits to qudits. We defined qudit colour codes using a star-bipartition, and it is useful to incorporate that into the definition. We introduce the star sign-flip matrix $F$. Each column of matrix $G$ corresponds to a qudit, and let us order these columns such that the first $p$ columns correspond to unstarred qudits, and the latter $n - p$ columns to starred qudits. We then define the $n \times n$ matrix $F$:

$$F = \text{diag}(1, \ldots, 1, -1, \ldots, -1),$$

(18)

where the first $|v_\star|$ elements on the diagonal are 1 and the remaining elements on the diagonal are $-1$. We note that the ordering of the columns here is arbitrary, so we do not lose any generality by ordering the columns in this manner. The purpose of $F$ will be to flip the sign of the entries of row vectors corresponding to the starred qudits, e.g. if $g = \{|g|_1, |g|_2, \ldots, |g|_n\}$ is a row vector, we define $g \cdot F$ as the row vector where $[gF]_j = [g]_j$ for the first $p$ elements (corresponding to unstarred qudits) and $[gF]_j = -[g]_j$ for the
remaining elements (corresponding to starred vertices). We also define weight \(|g| = \sum_j |g_j|\).

We now will define a significant generalisation of triorthogonal matrices, which we call \(m^*-orthogonality\) (pronounced “\(m^*\) star orthogonality”).

**Definition 1.** An \(n \times n\) matrix \(G\) over \(\mathbb{Z}_d\) with a bipartition of columns into \(\{v_\bullet, v_\ast\}\) is \(m^*\)-orthogonal if both of the following conditions hold.

1. The weight of every elementwise product of any \(m\) rows of \(GF\) (including repeated rows) is equal to 0, except for the following case:

2. For each row of submatrix \(G_1 F\), the weight of the row vector raised to the \(m\)th power elementwise is 1.

where \(F\) is defined as in equation (18) and where the matrices’ columns are ordered to respect to the star-bipartition \(\{v_\bullet, v_\ast\}\).

Notice the above makes no mention of modular arithmetic. If we had instead defined the weights modulo \(d\), we would have a weaker notion of \(m^*\)-orthogonality. Colour codes turn out to satisfy this stronger notion, and so this definition suffices for this paper. We note that the results below rely on the stronger form to deal with the exceptional cases, e.g. \(d = 3, 6\) for the \(T\) gate (see App. A). We also remark on the weaker notion to clarify that for \(d = 2\) it exactly corresponds to the Brayvi-Haah definition of triorthogonality. Other applications of weak \(m^*\)-orthogonality include quantum Reed-Muller codes [6, 34].

A more symbolic statement of \(m^*\)-orthogonality will prove useful in subsequent sections. We make use of \(u \circ v\) symbol to denote element-wise products of vectors \(u\) and \(v\), such that it has elements

\[
[g_a \circ g_b]_j = [g_a]_j [g_b]_j,
\]

which generalises for an arbitrary number of vectors, e.g. for three vectors

\[
[g_a \circ g_b \circ g_c]_j = [g_a]_j [g_b]_j [g_c]_j.
\]

Let us now consider a reformulation of points 1 and 2 of Def. 1. For any set of \(m\) rows, we have a list of \(m\) row indices \(\{a, b, \ldots, y\}\). \(m^*\)-orthogonality demands that for all the

\[
\sum_j [g_a F \circ g_b F \circ \ldots \circ g_y F]_j = 0
\]

unless all rows are identical and come from \(G_1\), in which case the weight is unity.

**C. The \(m^*\)-orthogonality of qudit colour codes**

The concept of \(m^*\)-orthogonality is a powerful tool for studying the properties of qudit colour codes due to the following lemma

**Lemma 5.** A qudit colour code for any \(d \geq 2\) defined on a lattice in spatial dimension \(\mu\) where \(X\)-stabilizer generators are defined by \(\mu'\)-cells with \(\mu' \leq \mu\), and with a star-bipartition \(\{v_\bullet, v_\ast\}\) defined by the bipartition of the lattice, is a \(m^*\)-orthogonal code for all \(m \leq \mu'\).
To prove this, let us first prove a convenient lemma.

**Lemma 6.** Any $m^*$-orthogonal matrix $G$ which includes, as one of its rows, the all-ones vector is also $m'^*$-orthogonal for all $m' < m$.

**Proof.** This follows since an elementwise product of $m$ vectors including the all-ones vector is equal to the elementwise product of $m - 1$ vectors excluding the all-ones vector. \(\square\)

Since the logical $X$ for all colour codes is the transversal $X$ acting on all qudits, the matrix $G_1$ for these codes contains the all-ones row. Hence to prove lemma 5 we now only need to prove that a colour code in $m$ spatial dimensions is $m^*$-orthogonal.

**Proof.** Recall that the $X$-generators of the code are defined by $\mu'$-cells. Taking as an example the 3D lattice in figure $1$, the $X$-stabilizer generators are defined by 3-cells. Consider the following geometric properties of these cells. When $q$ distinct $\mu'$-cells intersect non-trivially, where they meet defines a cell of smaller dimension. For a general lattice, there is no further restriction on the dimension of this cell. However, for a $\mu$-colex it is well known [13, 15, 17, 43] that the intersection of $q$ objects of dimension $\mu'$ yields either an empty set or a cell of dimension $\mu' - q + 1$. For instance, the intersect of two neighbouring $\mu'$-cells defines a $(\mu' - 1)$-cell, where three such $\mu'$-cells meet defines a $(\mu' - 2)$-cell, and so on. Where $\mu'$ $\mu'$-cells meet defines a 1-cell, or lattice edge. For example, in 3D, two adjacent 3-cells meet at a face, and three adjacent 3-cells meet at an edge (four adjacent 3-cells meet at a point).

We use this geometric fact to prove the theorem. Any product of $m$ row vectors in $G_0$ has a geometric representation as the intersection of the vertices of these cells. This corresponds to either an empty cell or cell of dimension no less than $\mu - m + 1$. Each cell in the lattice, of any dimension greater than zero, has an equal number of starred and unstarred vertices (see Cor. 1). When we multiply the product row vector by $F$ we invert the sign of the columns corresponding to the starred vertices, but this corresponds to precisely half of the non-zero elements in the vector. Hence the weight of the resultant vector is zero provided $\mu - m + 1 > 0$. Therefore, the rows of $G_0$ satisfy the conditions for $m^*$-orthogonality whenever $m \leq \mu$.

The other case to consider is the product where all $m$ vectors in the product are the all-ones vector. The element-wise product results trivially in the all-ones vector. The number of unstarred qudits is one more than the starred qudits (see Cor. 1), hence the weight of this vector after multiplying by $F$ is 1. Together with lemma 6, this proves lemma 5. \(\square\)

Note that the bounds in this lemma are tight. A code whose $X$-stabilizer generators are defined by $\mu'$-cells is not $(\mu' + 1)^*$-orthogonal because $(\mu + 1) \mu'$-cells meet at a point, and a single vertex represents a vector which does not have zero weight.

V. TRANSVERSAL OPERATORS

Our attention now turns to transversal gates. The qubit colour codes are the only family of topological codes known to support transversal gates satisfying the Bravyi-König bound. We prove in this paper that (apart from a minority of special cases which need to be treated individually) the qudit codes we have defined also saturate this bound.
**Theorem 1.** The qudit colour codes on a $\mu$-colex, where $X$-stabilizer generators are defined by $\mu' \leq \mu$ cells and where $\mu'! \neq 0 \pmod{d}$, support transversal gates in the $\mu'$th level of the Clifford Hierarchy (CH) (which are not in the $(\mu - 1)$th level of the CH), saturating the Bravyi-König bound.

The qualification $\mu'! \neq 0 \pmod{d}$ is due to lemma 2, namely that when $\mu'! = 0 \pmod{d}$, gates of the form of equation (8) are in the $(\mu' - 1)$th level of the CH (and therefore do not saturate Bravyi-König) as discussed in Sec. II.

For the most important $\mu = 3$ case, however, we additionally prove transversality for the exceptional $d = 3$ and $d = 6$ cases, and can thus state the unqualified result:

**Theorem 2.** The qudit colour codes on a $\mu$-colex, where $X$-stabilizer generators are defined by $\mu' \leq \mu$ cells support transversal non-Clifford gates in the 3rd level of the Clifford Hierarchy (CH) for all $\mu' \geq 3$.

We believe that the qualification in theorem 1 can be removed, following a similar approach to proving theorem 2, but we leave this for future work.

To prove theorem 1, we show that gates of the form of equation (8) are transversal in all $m^*$-orthogonal codes for polynomial degree $r = m$. To prove theorem 2, we also prove that gates of the form in equation (7) are transversal in $3^*$-orthogonal codes. The theorems then follow by virtue of lemmas 1, 2 and 5, together with the known results for qubit colour codes [14]. In particular, the transversal gates we consider are of star-conjugate transversal form, see equation 11.

**Lemma 7.** A $m^*$-orthogonal code has a transversal implementation of the unitary gate $R_{f_m}$ such that:

$$\tilde{R}_{f_m} = R_{f_m}[v_\bullet] \otimes R_{f_m}^*[v_\bullet] \tag{22}$$

implements $R_{f_m}$ on the code space.

We prove this lemma in App. A. As a simplified presentation for the main text of the paper, we prove the special and important case of $r = 3$. The general proof follows the same approach, but is a little notationally unwieldy.

**Proof.**

**Lemma 8.** A $3^*$-orthogonal code has a transversal implementation of the unitary gate $T = R_{j_3}$ such that:

$$\tilde{T} = T[v_\bullet] \otimes T^*[v_\bullet] \tag{23}$$

implements $T$ on the code space.

We now examine the conditions that the star-conjugate transversal $\tilde{T}$ implements a logical $T$. For this to be true, the phases for each computational basis component of each logical codeword in equation (17), must agree with the phases defining the gate in equation (6). Noting that the same phase applies to all terms in the sum in equation (17), we can write

$$\tilde{T} |(x,y)^T \cdot G\rangle = \omega^{x^3} |(x,y)^T \cdot G\rangle, \tag{24}$$
where \( x \in \mathbb{Z}_d \) and \( y \in \mathbb{Z}_d^s \), where \( s \) is the number of stabilizer generators. In other words we fix the number of logical qubits \( k = 1 \).

Applying equations (6) and (23) to the state on the right hand side of equation (24), we recover the following relationship between the phases on both sides of the equation:

\[
((x, y)T \cdot G \cdot F) \circ ((x, y)T \cdot G \cdot F) \circ ((x, y)T \cdot G \cdot F) = x^3 \quad \text{(mod } d)\]

where we have right multiplied the \( F \) matrix to vector \((x, y)^T \cdot G\). It is useful to define \( z = (x, y) \) so

\[
\sum_j ((z^T \cdot G \cdot F) \circ (z^T \cdot G \cdot F) \circ (z^T \cdot G \cdot F))_j = x^3 \quad \text{(mod } d). \tag{26}
\]

Now define \( z = \sum_a z_a e_a \) where \( z_a \) is the value of the \( a \)th position of the vector \( z \), and where \( e_a \) are the basis vectors over \( \mathbb{Z}^{s+1}_2 \). For example \( e_1 = (1, 0, \ldots, 0) \), \( e_2 = (0, 1, \ldots, 0) \) and so on. Re-writing the left hand side of equation (26) we obtain

\[
\sum_j ((z^T \cdot G \cdot F) \circ (z^T \cdot G \cdot F) \circ (z^T \cdot G \cdot F))_j = \sum_j \left( \sum_{a=1}^{s+1} z_a e_a^T \cdot G \cdot F \right) \circ \left( \sum_{b=1}^{s+1} z_b e_b^T \cdot G \cdot F \right) \circ \left( \sum_{c=1}^{s+1} z_c e_c^T \cdot G \cdot F \right) \quad \text{(mod } d). \tag{27}
\]

Note that \( e_a^T \cdot G \) is the \( a \)th row of the matrix \( G \), which we denote \( g_a \). Therefore

\[
\sum_j ((z^T \cdot G \cdot F) \circ (z^T \cdot G \cdot F) \circ (z^T \cdot G \cdot F))_j = \sum_j \left( \sum_{a=1}^{s+1} z_a g_a \cdot F \right) \circ \left( \sum_{b=1}^{s+1} z_b g_b \cdot F \right) \circ \left( \sum_{c=1}^{s+1} z_c g_c \cdot F \right) \quad \text{(mod } d). \tag{28}
\]

we can re-write this as

\[
\sum_j ((z^T \cdot G \cdot F) \circ (z^T \cdot G \cdot F) \circ (z^T \cdot G \cdot F))_j = \sum_{a,b,c=1}^{s+1} (z_a g_a \cdot F) \circ (z_b g_b \cdot F) \circ (z_c g_c \cdot F))_j \quad \text{(mod } d),
\]

\[
= \sum_{a,b,c=1}^{s+1} [(g_a \circ g_b \circ g_c) \cdot F]_j z_{a}z_{b}z_{c} \quad \text{(mod } d). \tag{29}
\]

Summing over \( j \) gives

\[
\sum_j ((z^T \cdot G \cdot F) \circ (z^T \cdot G \cdot F) \circ (z^T \cdot G \cdot F))_j = \sum_{a,b,c=1}^{s+1} |(g_a \circ g_b \circ g_c) \cdot F| z_{a}z_{b}z_{c} \quad \text{(mod } d). \tag{30}
\]

We break the sum into two parts, so that

\[
\sum_j ((z^T \cdot G \cdot F) \circ (z^T \cdot G \cdot F) \circ (z^T \cdot G \cdot F))_j = \sum_{a} |(g_a \circ g_a \circ g_a) \cdot F| z_{a}^3 + \sum_{\{a,b,c|- (a=b=c)\}} |(g_a \circ g_b \circ g_c) \cdot F| z_{a}z_{b}z_{c} \quad \text{(mod } d). \tag{31}
\]
Now we use the definition of \( m^*\)-orthogonal matrices. This ensures that the first term is equal to \( x^3 \) (mod \( d \)) and the second term is equal to zero. Hence equation (25) is satisfied, which completes the proof of lemma 8.

For larger \( m \), the proof follows the same lines but is with more cumbersome algebra. The proof of lemma 7 which we present in App. A completes the proof of theorem 1. To prove theorem 2 we require one further lemma:

**Lemma 9.** A \( 3^*\)-orthogonal qudit code where \( d = 3 \) or \( d = 6 \) has a transversal implementation of the unitary gate \( T_{3,6} \) defined in equation (7) such that:

\[
T_{3,6} = T_{3,6}[v_\bullet] \otimes T_{3,6}^*[v_\bullet]
\]

(32)

The proof is presented in App. B, completing the proof of theorem 2. Finally, we remark, that as CSS codes, all colour codes admit a transversal \( \Lambda(X) \). We conclude this section with a table of star-conjugate transversal gates in qudit colour codes of different spatial dimension in figure 3.

| Logical operator | Defined in Equation | Star-conjugate transversal implementation | Spatial dimension |
|------------------|---------------------|------------------------------------------|------------------|
| \( H \)          | equation (3)        | \( H[v_\bullet] \otimes H^*[v_\bullet] \) | 2                |
| \( \Lambda(X) \) | equation (5)        | Applied blockwise transversally          | \( \geq 2 \)     |
| \( S \)          | equation (4)        | \( S[v_\bullet] \otimes S^*[v_\bullet] \) | \( \geq 2 \)     |
| \( T \)          | equation (6)        | \( T[v_\bullet] \otimes T^*[v_\bullet] \) | \( \geq 3 \)     |
| \( R_{fr} \)     | equation (8)        | \( R_{fr}[v_\bullet] \otimes R_{fr}^*[v_\bullet] \) | \( \geq r \)     |

**FIG. 3.** A table illustrating the star-conjugate transversal implementation of some important logical gates and the spatial dimensions of the codes which support them.

**VI. GAUGE FIXING TO IMPLEMENT THE LOGICAL HADAMARD**

In this section and Sec. VII we show how a universal gate set can be achieved via gauge fixing, or by defining a gauge colour code. Although these results follow directly from the qubit cases presented in [26] and [15] without the need for any further technical innovations, we include a discussion and explanation of these techniques for completeness.

First we consider gauge fixing. We shall define a subsystem code occupying the same lattice as the colour code. This code is capable of realising the logical Hadamard gate on the same encoded qudit as the colour code, while a set of gauge qudits become corrupted. Fortunately there is a simple protocol to fix the corrupted gauge qudits to re-initialise the colour code.

In a \([n,k,d]\) subsystem code [51,53] there are three sets of operators that define the code. First, a set of gauge generators is defined, a set of Pauli operators defined on the lattice that do not necessarily mutually commute. These generate the gauge group \( A \). The center of the gauge group (the elements of the group which commute with all other elements) represents the stabilizer of the code \( S \) (up to a free choice of plus or minus phases which must be chosen to define the zero-error code space). The final set of operators to define is the set of logical Pauli operators on the code space, the Pauli operators that commute with \( A \). Subsystem codes can be understood as stabilizer codes in which some of
the logical qudits have been demoted to gauge qubits. These gauge qubits are not used to carry information and can be corrupted or measured during operations on the code.

The logical operators $\bar{X}$ and $\bar{Z}$ are products of $X$ and $Z$ (and $X^*$ and $Z^*$) operators acting on every qudit. Therefore, a natural candidate for transversal $\bar{H}$ is a global star-conjugate Hadamard. However, for colour codes of $\mu > 2$ it can be seen from the structure of the stabilizer generators that such a unitary will not leave the code space invariant. In particular, the $X$ stabilizers act on the $\mu'$-cells of the lattice whereas the $Z$ stabilizers act on the $(\mu - \mu' + 2)$-cells of the lattice. Therefore unless $\mu' = \mu - \mu' + 2$, the $Z$ stabilizers cannot be transformed into valid $X$ (cell) stabilizers, by a global Hadamard. The solution to this is to switch between the stabilizer colour codes and a different code in which $X$ stabilizer generators and $Z$ stabilizer generators have identical support.

In $\mu$ spatial dimensions the subsystem code stabilizers are defined as the star-conjugate $X$ and $Z$ operators acting on the qudits contained in a $\mu$-cell of the lattice. The logical operators matching the colour code logical operators are the transversal $\bar{X}$ and $\bar{Z}$ defined in equations (14). In addition there are $X$ and $Z$ gauge operators defined on the $(\mu - 1)$-cells, $(\mu - 2)$-cells, and so on to the 2-cells of the lattice, such that they commute with the stabilizer group. They can be identified in $\omega$-commuting pairs, with each pair protecting a gauge qudit.

In higher spatial dimensions there are a larger number of gauge generators and hence more gauge qudits than in lower spatial dimensions. Nevertheless, the basic protocol for realising the logical Hadamard gate in the subsystem code and fixing the gauge to return to the stabilizer code is independent of the spatial dimensions.

The star-conjugated transversal Hadamard gate,

$$\bar{H} = \tilde{H} = H[v] \otimes H^*[v],$$

transforms the transversal logical operators and the stabilizers of the code correctly. However the gauge generators are corrupted under the action of $\bar{H}$, meaning that $\tilde{Z}_j$ (the $Z$ logical operator for qudit $j$) is transformed not to $\bar{X}_j$ as desired but instead to the operator for a different qudit, $\tilde{X}_k$, causing the gauge qudits to become entangled.

These corrupted gauge operators must be fixed, to return the code to the original stabilizer code in which the other transversal operators can be realised. This fixing is enacted by measuring the stabilizer generators. The measured eigenvalues will indicate the Pauli correction operator that should be applied—this is an operator that restores the corrupted gauge qudit to its initial state while commuting with the other logical operators.

Once the corrections are applied, the gauge qudits are re-initialised to their original state so that any entanglement between the gauge qudits is destroyed. This is equivalent to the stabilizer colour code with a logical Hadamard applied to the encoded qudit.

**VII. QUDIT GAUGE COLOUR CODES**

Another approach to the implementation of the transversal Hadamard is to treat the subsystem code as the base code, and to achieve the transversal non-Clifford gates by gauge fixing to the (stabilizer) colour code. This switch in perspective leads to the gauge colour codes, recently introduced by Bombin [36]. Far more than a change in perspective, however, gauge colour codes gain many useful new properties, which the original colour codes do not possess.
The first advantage of the gauge colour code construction is that the outcome of stabilizer measurements can be inferred from measurements of the gauge operators. This offers a practical advantage, since the weight of the gauge generators can be significantly smaller than stabilizer generators. For example, in 3D the $X$ stabilizer generators would include 24-body operators associated with truncated octahedrons, discussed in Sec. III C, requiring coherent 24-body measurement. In the equivalent gauge colour code, however, the $X$ stabilizers can be decomposed into 4- or 6-body $X$ gauge operator measurements, a much more practically feasible proposition.

A second advantage (in the $\mu = 3$ case) is that the outcome of stabilizer measurements is more robustly encoded in the gauge generator measurement. The extra redundancy leads to “single-shot fault-tolerant error correction” [38]—meaning that measurement errors can be accounted for without the need for repeated measurements.

A third advantage is that one can use gauge fixing to fault tolerantly convert [27] between codes of different spatial dimension [37], allowing one to combine the advantageous features of both. For example, one could perform all Clifford computations in a 2-D code while only resorting to a 3-D code to achieve a non-Clifford gate.

We can define qudit gauge colour codes in any spatial dimension analogously to Bombin’s qubit codes [36]. The only difference is that operators are star-conjugated according to the star-bipartition of the lattice.

The most studied gauge colour code exists on a 3D lattice and we shall describe the qudit analogue as an example. Gauge generators consist of star-conjugated face operators of both $X$ and $Z$-type. The stabilizer is generated by 3-cell operators whose values can be obtained from the products of the face operators. The star-conjugate transversality of the Hadamard gate for this code is trivial via its definition. To achieve a transversal $T$ gate one must gauge fix back to the stabilizer colour code by promoting $Z$ face operators to the stabilizer.

Although the basic properties of colour codes transfer immediately to the general qudit case via this definition and by the results presented in previous sections in this paper, the performance of single-shot error correction is less straightforward to analyse, and we leave a detailed study to further work. Single-shot error correction works, in the qubit case, because the relationship between gauge and stabilizer measurement outcomes in the code is such that the values of the stabilizer are mapped to delocalised structures (branching points in string-like extended structures) in the gauge outcomes. This arises because the parity of the outcomes of the face operators around the cell must be equal to the eigenvalue of the stabilizer representing the cell, even parity in for a trivial syndrome, and odd-parity for the non-trivial case.

In qudit codes, the stabilizer measurement outcome may take any of $d - 1$ non-trivial values. The outcomes of gauge measurements must sum to these values. Thus the relationship between gauge measurements and stabilizer measurements is more complicated than the simple “branching point or no branching point” behaviour seen in the qubit code. This extra structure, however, should represent additional information about the location and charge of stabilizer outcomes which a decoder could exploit. Thus, with the development of a suitable decoder, we expect single-shot error correction robust to measurement error to be achieved in qudit gauge colour codes, but we leave its the construction of such a decoder, and analysis of its performance, to future work.
FIG. 4. A cell of the 3D gauge colour code. Its faces are three-colourable and coloured red, green and blue in this example. The stabilizer outcomes for the cell can be constructed by adding together (mod $d$) the outcomes of the gauge measurements for each of the red faces, or each of the blue faces, or each of the green faces.

VIII. DISCUSSION

We have shown how any existing qubit colour code can be generalised to support a qudit colour code in arbitrary spatial dimensions, and with an arbitrarily large distance. To do this we introduced the notion of a star-conjugate lattice and showed how this construction allows the Clifford phase, controlled-$X$ and non-Clifford phase gates to be implemented transversally. The set of Clifford group generators is completed by the Hadamard, which can be implemented in a subsystem code. Techniques for switching between the two codes have been outlined. An additional advantage of considering the colour codes as gauge codes is reduction of the weight of measurements required to obtain the error syndrome.

There are still many open questions remaining—here we shall outline some of the ones we find most interesting. Although a fault-tolerant decoder has been implemented for a 2D code, no decoder has, to our knowledge, been proposed for colour codes in higher spatial dimensions. In Ref. [38] a decoder for a 3D gauge colour code is discussed, unfortunately however, a classical algorithm capable of performing the decoding is not specified. One would expect that in a fully fault-tolerant simulation the gate error threshold for the $\mu$-dimensional colour codes would be prohibitively low, since the stabilizers defined on the $\mu$-cells would require circuits of many time steps to measure. There is hope however that by switching to the corresponding gauge colour code the measurements require fewer gates and therefore the threshold may recover. The development of efficient decoders for both qubit and qudit gauge colour codes is therefore of significant interest.

The construction of the transversal gates in the colour codes and gauge colour codes presented in this work holds for qudits of any dimension, $d$. However, for non-prime $d$ it is not clear that $S$, $H$ and $\Lambda(X)$ are sufficient to generate the Clifford group. We desire a better understanding of universality and the Clifford hierarchy when using systems of non-prime dimension.

We have identified some other lattices that can support a colour code in 3D. It would be interesting to see a comprehensive list of possible lattices. Furthermore, our star-conjugate construction is valid for all of these lattices, but there may be one that offers advantages in terms of the necessary qudit overhead to achieve a desired distance.

Finally, there are many Reed-Muller codes known for qudits for which no equivalent
colour code is known [6, 34]. Is it possible to use one of these to construct a topological code similar to a colour code and study its properties? Here, we found codes where the distance is purely topological in nature, and so depends on the lattice on not whether we use qudits. In contrast, qudit Reed-Muller codes provide an improved distance that is algebraic rather than topological in origin. The potential exists for qudit quantum codes which abstract topological ideas to a more general setting to generate novel codes of arbitrary code distance with a rich family of transversal logical gates.

Acknowledgements

DEB would like to thank Hector Bombin and Ben Brown for helpful and elucidatory discussions on colour codes and gauge colour codes. FHEW and HA acknowledge the financial support of the EPSRC (grant numbers EP/G037043/1 and EP/K022512/1, respectively).

[1] D.E. Knuth, Art of Computer Programming Vol. 2: Seminumerical Algorithms 3rd edition, Addison-Wesley Professional (1997).
[2] D. Gottesman, Chaos, Solitons & Fractals 10, 1749 - 1758 (1999).
[3] D. L. Zhou, B. Zeng, Z. Xu and C. P. Sun, Phys. Rev. A 68, 062303 (2003).
[4] M. Howard and J. Vala, Phys. Rev. A 86, 022316 (2012).
[5] H. Anwar, E.T. Campbell and D.E. Browne, New J. Phys. 14, 063006 (2012).
[6] E.T. Campbell, H. Anwar and D.E. Browne, Phys. Rev. X 2 041021, (2012).
[7] G. Duclos-Cianci and D. Poulin, Phys. Rev. A 87 062338 (2013).
[8] H. Anwar, B.J. Brown, E.T. Campbell and D.E. Browne, New J. Phys. 16, 063038 (2014).
[9] F.H.E. Watson, H. Anwar and D.E. Browne, arXiv:1411.3028
[10] A. Hutter, D. Loss and J.R. Wootton, arXiv:1410.4478
[11] A. Smith, B.E. Anderson, H. Sosa-Martinez, C.A. Riofrío, I.H. Deutsch and P.S. Jessen, Phys. Rev. Lett. 111, 170502 (2013).
[12] B.E. Anderson, H. Sosa-Martinez, C.A. Riofrío, I.H. Deutsch and P.S. Jessen, arXiv:1410.3891
[13] H. Bombin and M.A. Martin-Delgado, Phys. Rev. Lett. 97, 180501 (2006).
[14] H. Bombin and M.A. Martin-Delgado, Phys. Rev. Lett. 98, 160502 (2007)
[15] H. Bombin, arXiv:1311.0879
[16] H. Bombin and R.W. Chihajjany and M. Horodecki and M.A. Martin-Delgado, New J. Phys. 15, 055023 (2013).
[17] A. Kubica and M.E. Beverland, arXiv:1410.0069.
[18] A.Y. Kitaev, Ann. Phys. 303, 2 (2003).
[19] S.B. Bravyi and A.Y. Kitaev, arXiv:quant-ph/9811052
[20] P. Sarvepalli, IEEE ITW 1–5 (2010).
[21] C.G. Brell, arXiv:1408.6238.
[22] E. Knill, R. Laflamme and W.H. Zurek, arXiv:quant-ph/9610011
[23] E. Knill, R. Laflamme and W.H. Zurek, Science 279, 342-345 (1998).
[24] R. Raussendorf, J. Harrington, and K. Goyal, Ann. Phys. 321, 2242 - 2270 (2006).
[25] S. Bravyi and R. König, Phys. Rev. Lett. 110, 170503 (2013).
[26] A. Paetznick and B.W Reichardt, Phys. Rev. Lett. 111, 090505 (2013).
[27] J.T. Anderson, G. Duclos-Cianci and D. Poulin, Phys. Rev. Lett. 113, 080501 (2014).
[28] T. Jochym-O’Connor and R. Laflamme, Phys. Rev. Lett. 112, 010505 (2014).
[29] S. Bravyi and A. Kitaev, Phys. Rev. A 71, 022316 (2005).
[30] B.W. Reichardt, Quant. Inf. and Comp. 9, 1030 (2009).
Appendix A: Proof of lemma 7

Lemma 7. A \( m^{**} \)-orthogonal code has a transversal implementation of the unitary gate \( R_{f_m} \) such that:

\[
\tilde{R}_{f_m} = R_{f_m} \hat{v} \otimes R_{f_m}^* \hat{v}
\]  

(A1)

implements \( R_{f_m} \) on the code space.

Proof. For equation (A1) to be true, for all \( z = (x, y) \) we have

\[
\tilde{R}_{f_m} |z^T \cdot G\rangle = \omega^{5x} |z^T \cdot G\rangle .
\]  

(A2)

Applying equations (8) and (??) to the state on the right hand side of equation (A2), the phases on both sides of the equation are equal if

\[
(z^T \cdot G \cdot F)^{om} = x^m \pmod{d},
\]  

(A3)

where the notation \((\cdot)^{om}\) means a circle-product (see equation (19)) taken between \( m \) identical elements. Proceeding using the same notation as in Sec. V we find

\[
\sum_j (((z^T \cdot G \cdot F)^{om})_j = \sum_j \left( \sum_a z_ag_{\tilde{a}}F^{om} \right)_j \pmod{d},
\]  

(A4)

\[
\sum_j \left( \sum_{\tilde{a}} z_{\tilde{a}}g_{\tilde{a}}F \right)_j \pmod{d},
\]  

(A5)
where \([a, \ldots, b]^c\) is a vector of length \(c\) with elements that take values between \(a\) and \(b\), and where \(g \vec{a}\) is the bitwise product of \(m\) rows of \(G\), so
\[
g \vec{a} = g_{a_1} \circ g_{a_2} \circ \ldots \circ g_{a_m}\). \(\text{(A5)}\)

We can re-write this as
\[
\sum_j ((z^T \cdot G \cdot F)^{om})_j = \sum_j \sum_{\vec{a} \in [1, \ldots, k+s]^m} [g \vec{a} \cdot F]_j z \vec{a} \pmod{d}. \quad \text{(A6)}
\]

Summing over \(j\) gives
\[
\sum_j ((z^T \cdot G \cdot F)^{om})_j = \sum_{\vec{a} \in [1, \ldots, k+s]^m} |g \vec{a} \cdot F| z \vec{a} \pmod{d}. \quad \text{(A7)}
\]

We now use the definition of a \(m^*\)-orthogonal code. This means that all terms satisfying \(\neg(a_1 = \ldots = a_m)\) in the summation on the right hand side vanish. This implies that the only non-zero contribution is represented by \(\vec{a} = (a, a, \ldots, a)\), an \(m\)-element vector where every element is \(a\). Recall also that such terms vanish unless \(a = 1\), in other words it is the \(m\)-fold elementwise product of the vector in \(G_1\). We note that \(z \vec{a} = z_m \vec{a}\), leaving only
\[
\sum_j ((z^T \cdot G \cdot F)^{om})_j = |g \vec{a} \cdot F| z_m \vec{a} \pmod{d}, \quad \text{(A8)}
\]
which proves the desired equation \(\text{(A3)}\) is satisfied. \(\square\)

Appendix B: Proof of lemma 9

**Lemma 9.** A \(3^*\)-orthogonal qudit code where \(d = 3\) or \(d = 6\) has a transversal implementation of the unitary gate \(T_{3,6}\) defined in equation \(7\) such that:
\[
\widetilde{T}_{3,6} = T_{3,6}[v_\bullet] \otimes T^*_{3,6}[v_\ast] \quad \text{(B1)}
\]

where
\[
T_{3,6} = \sum_{j \in \mathbb{Z}_d} \gamma^j \langle j \rangle \quad \text{(B2)}
\]
where \(\gamma^3 = \omega\) and where the function in the exponent \(j^3\) is evaluated in regular arithmetic (or equivalently modulo \(3d\)). We shall prove lemma 9 for \(d = 3\) and \(d = 6\) separately, since each requires calculation in arithmetic with a different modularity.

1. \(d = 3\)

**Proof.** First we consider the case of qutrits, where \(d = 3\). In this case, \(\gamma = e^{i\frac{2\pi}{3}}\). Hence, our calculation will make use of addition modulo 9. Proceeding as before we start with:
\[
\tilde{T}_{3,6}|z^T \cdot G\rangle = \gamma^{x^3} |z^T \cdot G\rangle. \quad \text{(B3)}
\]
The phases on both sides of this equation match if
\[ \sum_j ((z^T \cdot G \cdot F)^{\circ 3})_j = x^3 \pmod{9}. \] (B4)

This expression is a mix of modulo 3 and modulo 9 arithmetic so we must proceed with care. The matrix multiplication is mod 3, whereas all other arithmetic is mod 9. Re-writing the left hand side as
\[ \sum_j ((z^T \cdot G \cdot F)^{\circ 3})_j = \sum_j \sum_{a,b,c=1}^s \{ [g_a \cdot F]_j z_a \pmod{3} \} \circ \{ [g_b \cdot F]_j z_b \pmod{3} \} \circ \{ [g_c \cdot F]_j z_c \pmod{3} \}, \] (B5)

we can use the following identity to express the modulo 3 reduction in terms of standard addition
\[ a \pmod{n} = a - \lfloor a/n \rfloor n. \] (B6)

Setting aside the modulo 9 arithmetic for the moment and evaluating this expression first in standard arithmetic, the left hand side now expands to the following form:
\[ \sum_j ((z^T \cdot G \cdot F)^{\circ 3})_j = \sum_j \sum_{a,b,c=1}^s \left( [g_a \cdot F]_j z_a - 3 \left\{ \frac{[g_a \cdot F]_j z_a}{3} \right\} \right) \circ \left( [g_b \cdot F]_j z_b - 3 \left\{ \frac{[g_b \cdot F]_j z_b}{3} \right\} \right) \circ \left( [g_c \cdot F]_j z_c - 3 \left\{ \frac{[g_c \cdot F]_j z_c}{3} \right\} \right), \]

the last three terms in this summation are integer multiples of 9 and are thus equal 0 \pmod{9}. Thus if we reimpose modulo 9 arithmetic we are left with:
\[ \sum_j ((z^T \cdot G \cdot F)^{\circ 3})_j \pmod{9} = \sum_j \sum_{a,b,c=1}^s \left( (g_a \circ g_b \circ g_c) \cdot F \right)_j z_a z_b z_c \pmod{9} \] (B7)

Finally, invoking \( m^* \)-orthogonality completes the proof. \[ \square \]

Note that we are using the fact that the definition of \( m^* \)-orthogonality is stated in regular arithmetic which implies the modulo 9 orthogonality used here.
2. \( d = 6 \)

Proof. When \( d = 6 \) the phase in \( T_{3,6} \) is \( \gamma = e^{\frac{2\pi}{18}} \) and thus proceed with modulo 18 arithmetic.

\[
\hat{T}_{3,6} |z^T \cdot G\rangle = \gamma^3 |z^T \cdot G\rangle.
\] (B8)

We expand the right hand side and find that

\[
\sum_j ((z^T \cdot G \cdot F)^3)_j = \sum_j \sum_{a,b,c=1}^{s+1} [(g_a \circ g_b \circ g_c) \cdot F]_j z_az_bz_c \quad (\text{mod } 18). \] (B9)

Converting modulo 18 arithmetic to normal arithmetic we find,

\[
\sum_j ((z^T \cdot G \cdot F)^3)_j = \sum_j \sum_{a,b,c=1}^{s+1} [(g_a \circ g_b \circ g_c) \cdot F]_j z_az_bz_c - 18[(g_a \circ g_b) \cdot F]_j z_az_bz_c \left(\frac{[g_c \cdot F]_j z_c}{6}\right)
+ 108[g_a \cdot F]_j z_a \left(\frac{[g_b \cdot F]_j z_b}{6}\right) \left(\frac{[g_c \cdot F]_j z_c}{6}\right)
- 216 \left(\frac{[g_a \cdot F]_j z_a}{6}\right) \left(\frac{[g_b \cdot F]_j z_b}{6}\right) \left(\frac{[g_c \cdot F]_j z_c}{6}\right).
\]

Casting this back into modulo 18 arithmetic we can cancel the terms equal to integer multiples of 18, leaving:

\[
\sum_j ((z^T \cdot G \cdot F)^3)_j \quad (\text{mod } 18) = \sum_j \sum_{a,b,c=1}^{s+1} [(g_a \circ g_b \circ g_c) \cdot F]_j z_az_bz_c \quad (\text{mod } 18) \] (B10)

Finally, via \( m^* \)-orthogonality we recover

\[
\sum_j ((z^T \cdot G \cdot F)^3)_j \quad (\text{mod } 18) = x^3 \quad (\text{mod } 18) \] (B11)

completing the proof. \( \square \)