Calderbank-Shor-Steane holographic quantum error-correcting codes

Robert J. Harris,[1,*] Nathan A. McMahon,[2,1] Gavin K. Brennen,[2] and Thomas M. Stace[1]

[1]*ARC Centre for Engineered Quantum Systems, School of Mathematics and Physics, The University of Queensland,
St Lucia, QLD, 4072, Australia*

[2]*Center for Engineered Quantum Systems, Department of Physics and Astronomy, Macquarie University, 2109 New South Wales, Australia*

We expand the class of holographic quantum error-correcting codes by developing the notion of block perfect tensors, a wider class that includes previously defined perfect tensors. The relaxation of this constraint opens up a range of other holographic codes. We demonstrate this by introducing the self-dual Calderbank-Shor-Steane (CSS) heptagon holographic code, based on the 7-qubit Steane code. Finally, we show promising thresholds for the erasure channel by applying a straightforward, optimal erasure decoder to the heptagon code and benchmark it against existing holographic codes.

The correspondence between anti-de Sitter (AdS) space and conformal field theories (CFT) [1] is an example of the holographic principle between a bulk $d + 1$-dimensional AdS and a boundary d-dimensional CFT [2]. AdS space is a maximally symmetric solution to the vacuum Einstein equations, in particular, it is a solution with negatively curved space-time. Boundary CFTs are quantum field theories invariant under conformal transformations. This is currently the most precise realization of the holographic principle and has spurred much work in this field [3]. It has been conjectured that any CFT can interpreted as a theory of quantum gravity which is an asymptotically AdS space [3], with an appropriate choice of metric on both sides.

A feature of the correspondence is the Ryu-Takayanagi (RT) formula, which relates the von Neumann entropy of a d-dimensional CFT boundary region to the minimal surface area of the $d + 1$-dimensional AdS bulk that subtends the boundary region [4,5]. This correspondence suggests that the boundary degrees of freedom possess substantial redundancy, making it a candidate for robustly encoding quantum information. Holographic codes were first proposed as a way to connect quantum information with the bulk-boundary correspondence [6,7]. Here we focus on the construction introduced in Ref. [8], which combines three desirable features: they are stabilizer codes and thus exactly solvable, they are quantum error correction codes (QECC), and their encoding is described by a tensor network which is a uniform tiling of hyperbolic space. Pastawski *et al.* demonstrated that a family of holographic codes based on the 5-qubit QECC satisfies the RT formula (although since the two-point correlators are not scale invariant the stabilizer codes do not correspond to a CFT on the boundary). A key requirement of Pastawski *et al.* is that the network be comprised exclusively of *perfect tensors*, described below, which strongly restricts the encoding circuit.

Holographic codes have tuneable rates,[1] from zero, in which a single logical qubit is encoded in a large network of physical qubits, up to a maximum rate set by the choice of seed code. Finite-rate codes whose code distance increases with the code size are desirable, since they enjoy a fixed overhead in practical implementations [9–11]. Further, low-density stabilizers are important for controlling the gate complexity of the error syndrome measurements. Finite-rate holographic codes approximately satisfy these desirable criteria, since most of the stabilizers are relatively localized, with very few that are highly delocalized. As such, they have potential as useful practical codes.

A useful preliminary metric against which an error-correcting code family can be judged is its resistance to erasure errors [12], in part because it is possible to map Pauli and erasure errors onto a modified erasure channel [13]. The erasure thresholds shown for these codes are comparable to the performance of certain tree networks [14] and the surface code [15–17]. A favorable comparison between the surface code and the pentagon holographic code suggest this is a promising avenue for practical codes. However, there are a number of scenarios where it is beneficial to use Calderbank-Shor-Steane (CSS) codes, for example, in building large-scale cluster states for measurement-based computation [18–20], or for building foliated codes [21] to use as long-range quantum repeaters.

In this paper, we show that perfect tensors are not required for constructing holographic codes and describe a CSS heptagon code based on the 7-qubit Steane code. We also implement an exact erasure decoder for holographic codes and demonstrate its performance on the heptagon code and the original pentagon code. The optimal decoder outperforms the greedy algorithm from [8] and gives a threshold of ~1/3 for the heptagon code.

Pastawski *et al.* [8] constructed holographic quantum error-correcting codes based on the $[[n, k, d]] = [[5, 1, 3]]$ 5-qubit

---

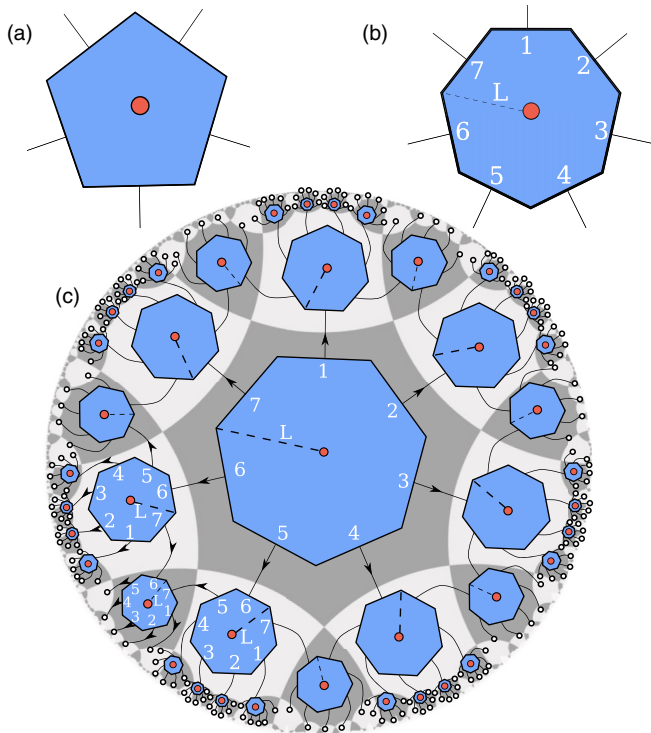[1]The ratio of the number of logical to physical qubits.

FIG. 1. (a) Graphical representation of the seed tensor for the 5-qubit pentagon code. The central red dots represent a logical input qubit. (b) Graphical representation of the seed tensor for the Steane 7-qubit heptagon code. The cyclic ordering of indices ensures a block perfect Steane code presentation as in Eq. (3). (c) Tessellation of the heptagonal seed in a larger tensor network representation of radius $R = 3$ heptagon code. The small circles around the boundary represent physical qubits. The arrows show the conventional direction from the center to the boundary, which we adopt for constructive purposes. Numbered indices indicate a consistent ordering required to produce an isometry from the bulk logical qubits to boundary physical qubits.

code [22]. The 5-qubit code generates a rank $n + 1 = 6$ *seed tensor*, T, represented graphically as a pentagon in Fig. 1(a). The central vertex represents a logical input qubit, and the n planar legs represent output qubits. The tensor specifies an isometry from logical input operators to output operators.

A larger tensor network is built from the seed tensor by tessellating it in the $\{4, n\}$ Schläfli geometry (i.e., with four polygons meeting at each vertex), forming a surface with negative curvature illustrated in Fig. 1(c). Neighboring indices on adjacent tensors are contracted, which we represent graphically by connecting the corresponding planar legs [23]. The tessellation terminates at a certain radius R, which is given by the minimal number of edges from the boundary to the central *bulk* logical qubit. Input vertices on each seed tensor in the bulk represent logical bulk qubits; the uncontracted legs at the boundary terminate at physical qubits, denoted by hollow circles.

The seed tensors in the pentagon holographic code are *perfect tensors*, which we briefly review. For a rank-$2m$ tensor, T, we may partition its indices into an ordered set A and its

ordered complement $\bar{A}$ such that $|A| \leqslant |\bar{A}| = 2m - |A|$. We interpret T as a linear map from the logical Hilbert space on the input indices in set A to the image Hilbert space on the output indices in set $\bar{A}$, i.e., $T^{\bar{A} \leftarrow A} : \mathcal{H}_A \mapsto \mathcal{H}_{\bar{A}}$.

Pastawski *et al.* [8] define T to be a perfect tensor if it is an isometry for *all* bipartitions of the indices $\{A|\bar{A}\}$ with $|A| \leqslant |\bar{A}|$, i.e., T satisfies

$$(T^{\bar{A} \leftarrow A})^{\dagger} T^{\bar{A} \leftarrow A} = \mathbb{I}_{\mathcal{H}_A}. \tag{1}$$

This is a very restrictive constraint on T.

For later discussion, we note that we can express a given a bipartition of indices $\{A|\bar{A}\}$ as a permutation Π with respect to some reference index ordering $\underline{J} = \{j_1, j_2, \ldots, j_{2m}\}$, i.e., $\{A|\bar{A}\} = \Pi[\underline{J}]$. A perfect tensor is therefore an isometry for *all* permutations Π. This formulation will help when we define the less restricted class of *block perfect tensors*.

Operators acting on the physical qubits on the boundary of the space are defined via *operator pushing* [8] from logical bulk qubits, through the tensor network, to the physical boundary qubits. For constructive purposes we assign a direction to each leg in the network, shown by arrows in Fig. 1(c), indicating that an "output" index from one tensor contracts with an "input" index to an adjacent tensor. Each tensor in the network translates operators acting on input indices to operators acting on output indices according to

$$O^A = T^{A \leftarrow \bar{A}} O^{\bar{A}} (T^{A \leftarrow \bar{A}})^{\dagger}. \tag{2}$$

For example, an identity operator on the logical input leg of a stabilizer code translates to any of the stabilizer group on the output legs.

We note that the bulk logical qubits near the boundary map, via operator pushing, to boundary operators that are localized on a small wedge of the physical boundary qubits, while logical operators for qubits deep within the bulk are highly delocalized over the boundary. A constructive method to generate a complete set of stabilizers is shown in the Appendix.

The perfect tensor property guarantees that tensor legs can be arbitrarily partitioned into inputs and outputs. While this was inspired by diffeomorphism invariance in the underlying AdS space [8], it is very restrictive. They show that the rank-6 seed tensor for the 5-qubit code is indeed a perfect tensor. The resulting holographic code has finite rate $r = k/n = 1/\sqrt{5}$ in the asymptotic limit.

We now show that the perfect tensor constraint can be relaxed to a less restrictive class that still generates a holographic code. This is motivated by the observation that in the hyperbolic tessellation in [8], input legs to a given seed tensor can be grouped into a contiguous block. This new class includes the 7-qubit Steane code [24] tensor, which we use in a CSS holographic code based on a tiling of heptagons.

We define *block perfect tensors* to be those that are isometries for all *cyclic* permutations $\Pi = \sigma^p$ of $\underline{J}$, i.e., those for which $\{A|\bar{A}\} = \sigma^p[\underline{J}]$, where $\sigma^p : j_i \mapsto j_{i+p}$ is a cyclic shift. This coincides with the description of perfect tangles developed independently [25]. We believe this constraint on the seed tensor more closely corresponds to a discretization of diffeomorphism invariance, though we do not comment on this further here.

Steane tensor. We exemplify this relaxation of perfection by showing that the rank-8 Steane tensor, $T_{\text{Steane}}^{j_1,\ldots,j_L,j_7}$, which is generated from the Steane code, is block perfect, but not perfect. This tensor is defined through the unique simultaneous $+1$ eigenstate, $|T_{\text{Steane}}\rangle$, of the eight stabilizers

Index label:	1	2	3	4	5	6	L	7	
	X	X	$\mathbb{I}$	$\mathbb{I}$	$\mathbb{I}$	X	$\mathbb{I}$	X	$\equiv S_1$
	$\mathbb{I}$	X	X	X	$\mathbb{I}$	$\mathbb{I}$	$\mathbb{I}$	X	$\equiv S_2$
	$\mathbb{I}$	$\mathbb{I}$	$\mathbb{I}$	X	X	X	$\mathbb{I}$	X	$\equiv S_3$
	Z	Z	$\mathbb{I}$	$\mathbb{I}$	$\mathbb{I}$	Z	$\mathbb{I}$	Z	$\equiv S_4$,
	$\mathbb{I}$	Z	Z	Z	$\mathbb{I}$	$\mathbb{I}$	$\mathbb{I}$	Z	$\equiv S_5$
	$\mathbb{I}$	$\mathbb{I}$	$\mathbb{I}$	Z	Z	Z	$\mathbb{I}$	Z	$\equiv S_6$
	X	X	X	X	X	X	X	X	$\equiv S_{\bar{X}}$
	Z	Z	Z	Z	Z	Z	Z	Z	$\equiv S_{\bar{Z}}$

$$(3)$$

via the Choi-Jamiołkowski isomorphism [26]. That is, $T_{\text{Steane}}^{j_1,\ldots,j_L,j_7} = \langle j_1, \ldots, j_L, j_7 | T_{\text{Steane}} \rangle$, where $|T_{\text{Steane}}\rangle$ satisfies $S_\alpha | T_{\text{Steane}}\rangle = |T_{\text{Steane}}\rangle$ for all α. The index labels in Eq. (3) are consistent with the ordering shown in Fig. 1(b). With respect to this index label ordering, we have exhaustively checked that T_{Steane} is block perfect.

It is straightforward to see that T_{Steane} is *not* a perfect tensor by considering the noncontiguous partition of indices $A = \{3, 4, 5, L\}$ and $\bar{A} = \{1, 2, 6, 7\}$. If T_{Steane} were perfect, then $T^{A\leftarrow\bar{A}}$ would be unitary. Then according to Eq. (2), $T^{A\leftarrow\bar{A}} X^{\otimes A}(T^{A\leftarrow\bar{A}})^\dagger \neq \mathbb{I}^{\otimes\bar{A}}$. However, according to S_1 and the Choi-Jamiołkowski isomorphism, $T^{A\leftarrow\bar{A}} X^{\otimes A}(T^{A\leftarrow\bar{A}})^\dagger = \mathbb{I}^{\otimes\bar{A}}$, implying that for this partition $T^{A\leftarrow\bar{A}}$ is not a unitary map. Hence T_{Steane} is not perfect.

Heptagon holographic code. As with the pentagon code, the heptagon code is built on a two-dimensional (2D) tiling with negative curvature. This is a tessellation of heptagons, with four heptagons meeting at each vertex (the {4, 7} Schläfli geometry), as shown in Fig. 1(c).

The hyperbolic tiling of the heptagon code requires a consistent assignment of index contractions between adjacent tensors. Figure 1(c) shows one such assignment for a subset of the tiles. This ensures that every seed tensor indeed acts as an isometry from inputs to outputs, so that the entire network is an isometry from bulk inputs to boundary outputs.

Because the seed code is a self-dual CSS code, it is clear that pushing X-like tensors will lead to X-like holographic stabilizers, and similarly for the Z-like stabilizers. This means the heptagon holographic code is a self-dual CSS code. Similarly to the pentagon code, the heptagon code is a finite-rate code, with asymptotic rate $r = 1/\sqrt{21}$.

Block-perfect tensors in a hyperbolic tiling generate a holographic code according to the definition in Pastawski *et al.*, based on the existence of a greedy algorithm. The greedy algorithm constructs a recoverability region $\mathcal{R}$ of bulk logical data by recursively adding bulk tensors T to $\mathcal{R}$ according to the local update rules: (1) boundary qubits are recoverable if they are not erased, and (2) given some set $\bar{A}$ of tensor indices for T that are recoverable, then if there is an isometry from A to $\bar{A}$ (with $|A| \leqslant |\bar{A}|$), then we add T to $\mathcal{R}$. The fixed point of these rules defines $\mathcal{R}$. For holographic codes built from perfect tensors, the tensor indices in $\bar{A}$ can be arbitrary, whereas for block perfect tensors they must be in

contiguous order. Starting from a contiguous region B on the boundary, the region $\mathcal{R}$ produced by this algorithm has an inner boundary that approximates, to within a small constant, the discrete bulk geodesic γ_B connecting the end points of B. As shown in [8], this implies an RT formula of $S_B \propto |\gamma_B|$.

Erasure decoders. Having defined the heptagon holographic code, we are interested in the resilience of the code to errors. Delfosse *et al.* [12] propose using the robustness of a code to erasure errors as a proxy for performance of the code under more general error channels. As such, we now describe an erasure decoder for this code with which to quantify the code performance.

Loss errors are heralded so that we know where they have occurred. This enables us to use the error pattern as part of the error decoding algorithm, making an exact decoder computationally feasible.

Recovery algorithm. The algorithm we detail here is optimal for any stabilizer code, including the holographic codes. Computationally, it relies on matrix row reduction, which for an $a \times b$-dimensional matrix has run time $\sim O(a^2 b)$. (There are more sophisticated algorithms with lower complexity [27].) In the optimal decoder a is the number of erasure errors and $b = n - k$ is the number of stabilizers.

For simplicity we describe the algorithm for CSS codes; however, it is straightforwardly adapted to any stabilizer code. The stabilizers for an $[[n, k, d]]$ CSS code are specified by a set of binary support vectors $\underline{s}_j$ such that the X-like stabilizers are given by $S_j = \hat{X}^{\otimes \underline{s}_j}$. Likewise, a logical support vector $\underline{\ell}$ defines an X-like logical operator $\bar{X} = X^{\otimes \underline{\ell}}$ [28]. Z-like stabilizers and logical operators are defined similarly.

Logical operators are equivalent up to multiplication by stabilizers, so that $\bar{X}' = X^{\otimes \underline{\ell}'} \sim \bar{X} = X^{\otimes \underline{\ell}}$ if and only if

$$\underline{\ell}' = \underline{\ell} + \sum_j \lambda_j \underline{s}_j \quad \text{mod } 2 \tag{4}$$

for some $\lambda_j \in \mathbb{Z}_2$.

Suppose a subset of physical qubits is erased. This error is defined by a binary support vector $\underline{\varepsilon}$, in which an entry 1 in position i indicates that the ith qubit is lost. Providing we can construct a logical operator $X^{\otimes \underline{\ell}'}$ which has no support on the lost qubits, i.e., $\underline{\ell}' \cdot \underline{\varepsilon} = 0$, then the corresponding logical information is recoverable. (Note that the dot product here is not modular.)

It is clear that to satisfy this condition it is necessary and sufficient to find $\underline{\ell}'$ which has zeros at positions where $\underline{\varepsilon}$ is 1, i.e., $\ell_i' = 0$ if $\varepsilon_i = 1$. We define a *filtered* support vector $\underline{a}^{(\varepsilon)}$, which is the restriction of the support vector $\underline{a}$ to the positions at which $\underline{\varepsilon} = 1$. Then $\underline{\ell}' \cdot \underline{\varepsilon} = 0$ if and only if we can find λ_j such that

$$\underline{\ell}^{(\varepsilon)} + \sum_j \lambda_j \underline{s}_j^{(\varepsilon)} = 0 \quad \text{mod } 2. \tag{5}$$

The existence (and solution where required) of satisfying λ_j's can be determined efficiently with row reduction of the matrix of filtered stabilizer support vectors augmented with the filtered logical support vector.

Monte Carlo simulations. To evaluate the performance of this decoder, and the performance of the heptagon code under erasure, we simulate the recovery of the central logical qubit after loss using Monte Carlo simulations. We generate
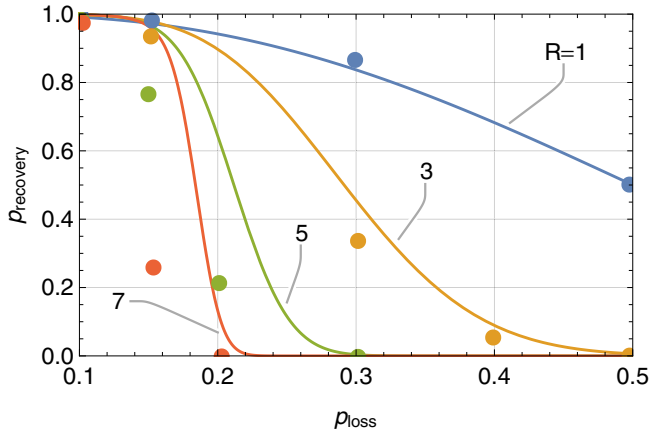
FIG. 2. Recovery probability for the pentagon code, based on recovery of the central logical qubit using a greedy algorithm (points) [8], and the optimal row reduction algorithm calculated according to Eq. (6) (solid lines) for codes of radius $R = 1, 3, 5, 7$. Note that the $R = 1$ code is the 5-qubit code. As R grows, the optimal decoder performs increasingly better then the greedy decoder.

FIG. 3. Recovery probability for the heptagon code, $p_{rec}(p_{loss})$ as in Eq. (6), for recoverability of the central logical qubit for codes of radius $R = 1, 2, 3, 4, 5$. Note that the $R = 1$ code is the Steane code. There is a threshold at $p_{loss} < p_{hept}^*$.

independent and identically distributed patterns of physical qubit erasure for a fixed number of errors $a = \text{wt}(\underline{\varepsilon})$, and then use the algorithm detailed above to determine whether each pattern is recoverable. We iterate over all $a \in \{0, 1, \ldots, n\}$ to estimate the recovery probability, $P_{rec}(a, n)$, and use the binomial formula

$$p_{rec}(p, n) = \sum_a \binom{n}{a} p^a (1-p)^{n-a} P_{rec}(a, n) \qquad (6)$$

to calculate the recovery rate for different loss rates p.

We use the pentagon code to benchmark the row reduction algorithm against the greedy algorithm in [8]. This code does not have an erasure threshold, as the distance of the central qubit of code does not increase with the radius. The results are shown in Fig. 2, where lines are the results of numerical simulations with the optimal decoder and Eq. (6), and points are from the heuristic greedy algorithm in [8]. As the radius of the network increases, a growing discrepancy between the optimal row reduction and the heuristic greedy algorithm is evident, albeit with no threshold appearing.

We now examine the performance of the heptagon holographic code against erasure, as measured by the recovery probability for the central logical qubit. The performance curves are shown in Fig. 3 up to a radius $R = 5$ code. In contrast to the pentagon code, we do find a threshold, $p_{hept}^* \approx 1/3$, for erasure in the heptagon code. That is, for an erasure probability $p_{loss} < p_{hept}^*$, the code performance improves with increasing radius.

We compare the performance of the heptagon code to the mixed pentagon/hexagon code of Pastawski *et al.*, where the asymptotic rate is reduced by a factor of around $1/2$, to find a threshold of around $p_{pent/hex}^* \approx 1/3$. We calculate the asymptotic rate of their pentagon/hexagon code to be $r_{pent/hex} = (13\sqrt{6} - 12)/90 \approx 1/\sqrt{20.5}$, which is very similar to the rate for the heptagon code $r_{hept} = 1/\sqrt{21}$ proposed here.

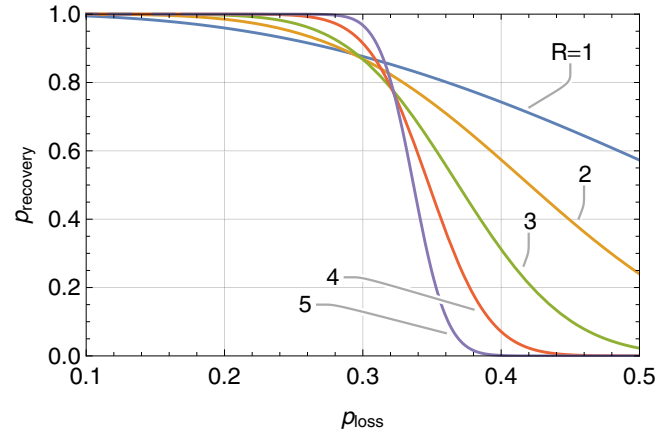The high erasure threshold and code rate suggest that the heptagon code might be an attractive candidate for a practical error-correcting code in networks that have weak geometric constraints, such as optical architectures [20]. Its performance against other logical error channels is the subject of ongoing research. Further, since it is of CSS form, the heptagon code can be constructed from measurements on a suitably prepared cluster state [21]. However, because some stabilizers have relatively large weight, the cluster states resulting from that construction would usually be high valence, which tends to amplify imperfections during cluster creation. We note here that the heptagon holographic code can be implemented with a low-valence cluster state; details will be reported in a forthcoming publication.

To conclude, we have developed the notion of block perfect tensors, a less restricted class than the perfect tensors introduced in earlier work. This makes a range of other codes available for tessellation in a holographic tensor network, including the self-dual CSS Steane code, with which we have exemplified the general construction. The stabilizers have a spectrum of weights, with most stabilizers being local and low weight, and proportionally fewer having larger weights. As such, they may be suitable for cluster-state-based measurement protocols [21]. Finally, we have applied a straightforward, optimal erasure decoder based on matrix row reduction on filtered support vectors to characterize the performance of holographic codes, yielding promising thresholds.

APPENDIX: OPERATOR CONSTRUCTION

Given a particular code associated to each of the seed tensors, there is a unique choice of stabilizers where each stabilizer is centered on a particular seed tensor. Each stabilizer can be considered to stabilize a particular logical input.

In Fig. 1(c) there is a direction associated to each physical leg on the seed tensors. For the central seed tensor all physical

legs are outputs; for all other seed tensors there are either one or two input physical legs.

Starting with seed tensors at the boundary, we produce a subset of stabilizers for the holographic code associated to these boundary seed tensors. This is done by selecting the stabilizers which have identity on all input legs. For example, using the Steane code as the seed tensor, the stabilizers associated to each boundary seed tensor with identity on index 6 and logical are:

Index label: 1	2	3	4	5	6	L	7	
$\mathbb{I}$	X	X	X	$\mathbb{I}$	$\mathbb{I}$	$\mathbb{I}$	X	$\equiv S_2$
X	X	$\mathbb{I}$	X	X	$\mathbb{I}$	$\mathbb{I}$	$\mathbb{I}$	$\equiv S_1 S_3$
$\mathbb{I}$	Z	Z	Z	$\mathbb{I}$	$\mathbb{I}$	$\mathbb{I}$	Z	$\equiv S_5$
Z	Z	$\mathbb{I}$	Z	Z	$\mathbb{I}$	$\mathbb{I}$	$\mathbb{I}$	$\equiv S_4 S_6$

(A1)

A stabilizer for the holographic code, associated with the boundary seed tensors, is the operators above on the output physical qubits from the particular seed tensor along with identity on all other physical qubits. As alternate choices from Eq. (A1) come from the Steane code generating set, they create independent stabilizers for the holographic code. So all options from Eq. (A1), with identities on all other physical qubits, can be added to the generating set for the heptagon code.

Since different boundary seed tensors do not share physical qubits, the procedure above will generate independent holographic stabilizers for each boundary seed tensor. For the same reason they are all guaranteed to commute. This procedure generates a subset of the stabilizer generators for the holographic code.

Adding the additional restriction of identities on two planar legs (indices 6 and 7), we have the following stabilizers:

Index label: 1	2	3	4	5	6	L	7	
X	X	$\mathbb{I}$	X	X	$\mathbb{I}$	$\mathbb{I}$	$\mathbb{I}$	$\equiv S_1 S_3$
Z	Z	$\mathbb{I}$	Z	Z	$\mathbb{I}$	$\mathbb{I}$	$\mathbb{I}$	$\equiv S_4 S_6$

(A2)

As for the stabilizers from Eq. (A1), these can produce further stabilizers for the generating set, which again are independent and commuting for the same reasons as above.

For nonboundary seed tensors, the approach above constitutes the initial stage of generating the stabilizers. From this, the operators are pushed to the boundary to form the holographic stabilizer.

Operator pushing is essentially taking a seed tensor and inputting either an X or Z operator for each incoming leg. We can use then use stabilizer $S_1 S_2$ or $S_4 S_5$ to pull these onto all other legs:

Index label: 1	2	3	4	5	6	L	7	
X	$\mathbb{I}$	X	X	$\mathbb{I}$	X	$\mathbb{I}$	$\mathbb{I}$	$\equiv S_1 S_2$
Z	$\mathbb{I}$	Z	Z	$\mathbb{I}$	Z	$\mathbb{I}$	$\mathbb{I}$	$\equiv S_4 S_5$

(A3)

This push is unique up to the product of stabilizers (with identity on legs L and 6) of the seed tensor we are pushing this operator through.

In the normal operator pushing language we have the seed tensor T, which is proportional to an isometry from input physical legs (and logical leg) to outgoing physical legs and can define operator O' from O by

$$T O \propto T O(T^\dagger T) = (T O T^\dagger)T = O'T. \qquad (A4)$$

Note operators are only defined up to multiplication by stabilizers.

When there are multiple incoming legs, then we can do the same using the appropriate choice from seed tensors:

Index label: 1	2	3	4	5	6	L	7	
X	$\mathbb{I}$	X	X	$\mathbb{I}$	X	$\mathbb{I}$	$\mathbb{I}$	$\equiv S_1 S_2$
$\mathbb{I}$	X	X	X	$\mathbb{I}$	$\mathbb{I}$	$\mathbb{I}$	X	$\equiv S_2$
X	X	$\mathbb{I}$	$\mathbb{I}$	$\mathbb{I}$	X	$\mathbb{I}$	X	$\equiv S_1$
Z	$\mathbb{I}$	Z	Z	$\mathbb{I}$	Z	$\mathbb{I}$	$\mathbb{I}$	$\equiv S_4 S_5$
$\mathbb{I}$	Z	Z	Z	$\mathbb{I}$	$\mathbb{I}$	$\mathbb{I}$	Z	$\equiv S_5$
Z	Z	$\mathbb{I}$	$\mathbb{I}$	$\mathbb{I}$	Z	$\mathbb{I}$	Z	$\equiv S_4$

(A5)

Note all identity inputs push through to all identity outputs.

Repeated applications of operator pushing is sufficient to push the operators from the nonboundary seed tensor to the boundary physical qubits. This generates a further subset of nonboundary stabilizers that completes the stabilizer generating set for the holographic code.

This procedure generates six stabilizers for the central tensor, four stabilizers for each tensor with one input physical leg and two stabilizers for two input physical legs. This can be shown to sum to the number of required stabilizers. As they are all commuting and independent, this is a stabilizer generating set.

[1] J. M. Maldacena, Int. J. Theor. Phys. **38**, 1113 (1999).

[2] J. M. Maldacena, J. High Energy Phys. 04 (2003) 021.

[3] D. Harlow, Rev. Mod. Phys. **88**, 15002 (2016).

[4] S. Ryu and T. Takayanagi, Phys. Rev. Lett. **96**, 181602 (2006).

[5] V. E. Hubeny, M. Rangamani, and T. Takayanagi, J. High Energy Phys. 07 (2007) 062.

[6] J. I. Latorre and G. Sierra, arXiv:1502.06618.

[7] B. Yoshida, Ann. Phys. **338**, 134 (2013).

[8] F. Pastawski, B. Yoshida, D. Harlow, and J. Preskill, J. High Energy Phys. 06 (2015) 149.

[9] D. Poulin, J. Tillich, and H. Ollivier, IEEE Trans. Inf. Theory **55**, 2776 (2009).

[10] J.-P. Tillich and G. Zemor, IEEE Trans. Inf. Theory **60**, 1193 (2014).

[11] N. Delfosse, *Proceedings of the 2013 IEEE International Symposium on Information Theory, Istanbul* (IEEE, Piscataway, NJ, 2013), pp. 917–921.

[12] N. Delfosse, P. Iyer, and D. Poulin, arXiv:1611.04256.

[13] N. Delfosse and N. H. Nickerson, arXiv:1709.06218.

[14] M. Varnava, D. E. Browne, and T. Rudolph, Phys. Rev. Lett. **97**, 120501 (2006).

[15] T. M. Stace, S. D. Barrett, and A. C. Doherty, Phys. Rev. Lett. **102**, 200501 (2009).

[16] T. M. Stace and S. D. Barrett, Phys. Rev. A **81**, 022317 (2010).

[17] S. D. Barrett and T. M. Stace, Phys. Rev. Lett. **105**, 200502 (2010).

[18] R. Raussendorf, J. Harrington, and K. Goyal, Ann. Phys. **321**, 2242 (2006).

[19] H. J. Briegel, D. E. Browne, W. Dür, R. Raussendorf, and M. Van den Nest, Nat. Phys. **5**, 19 (2009).

[20] T. Rudolph, APL Photon. **2**, 030901 (2017).

[21] A. Bolt, G. Duclos-Cianci, D. Poulin, and T. M. Stace, Phys. Rev. Lett. **117**, 070501 (2016).

[22] R. Laflamme, C. Miquel, J. P. Paz, and W. H. Zurek, Phys. Rev. Lett. **77**, 198 (1996).

[23] J. C. Bridgeman and C. T. Chubb, J. Phys. Math. Theor. **50**, 223001 (2017).

[24] A. Steane, Proc. R. Soc. Math. Phys. Eng. Sci. **452**, 2551 (1996).

[25] J. Berger and T. J. Osborne, arXiv:1804.03199.

[26] M. Jiang, S. Luo, and S. Fu, Phys. Rev. A **87**, 022310 (2013).

[27] J. R. Bunch and J. E. Hopcroft, Math. Comput. **28**, 231 (1974).

[28] D. Gottesman, Ph.D. Thesis, California Institute of Technology, 1997.