Quantum Error Correction and Bulk Reconstruction via the HaPPY Code

Michelle Dong, Benjamin Knepper and Anthony Petyuk

University of California, Berkeley, California 94720, USA

PHYS C191: Introduction to Quantum Computing, Final Paper

Dated: May, 2025; Last updated: January, 2026 (Appendix B)

Author Contributions:

Ben wrote part of the abstract, sections 1.1.4, 1.2, and 1.3 of the introduction, and the first two paragraphs of Section 5. Ben wrote all of section 2: HaPPY Code as a QECC, section 3: Causal Wedge Reconstruction, and section 4: Entanglement Wedge Reconstruction. Ben also wrote appendices A-D.

Michelle wrote part of the abstract, the first two introduction paragraphs, sections 1.1.1-1.1.3 and helped write sections 1.1.4 and 1.3 of the introduction. Michelle also contributed to section 5: Conclusion and Future Directions. Michelle also wrote Appendix E and Appendix G, including all figures and simulations therein. Michelle also provided grammatical and formatting edits.

Anthony wrote Appendix F corresponding to section 1.1.3 of the introduction. He also contributed to the final paragraph of the conclusion section.

Abstract: Recent progress in understanding the holographic principle in quantum gravity has been made thorugh the language of quantum error correction. We explore how quantum error-correcting codes model bulk reconstruction in AdS/CFT by analyzing the HaPPY tensor-network code. After reviewing the essentials of the AdS/CFT correspondence, we demonstrate that the HaPPY code is a bonafide quantum error correcting code by showing it is equivalent to a concatenated five-qubit [[5,1,3]] stabilizer code and verifying that it satisfies the Knill–Laflamme conditions. Employing the operator-pushing property of perfect tensors, then we present an explicit example of causal-wedge reconstruction via the "greedy algorithm." Lastly, we propose a novel algorithm to construct the bulk code space for entanglement wedge reconstruction based on a Grover search method.

Contents

Notation: We follow Harlow and abbreviate Hilbert space designations with capital letters. For example, $|\psi\rangle_A \implies |\psi\rangle_A \in \mathcal{H}_A$, *and* $|A| = |\mathcal{H}_A|$.

1 Introduction

Progress in understanding holographic duality has been made in recent years through the language of quantum error correction. In 2007, Guifre Vidal [1] introduced the Multiscale Entanglement Renormalization Ansatz (MERA) and it was interpreted in 2012 by Swingle as a toy model for holography due to its emergent geometry [2]. In 2015, Pastawski, Yoshida, Harlow, and Preskill [3] introduced the HaPPY code named after their initials demonstrating key features such as entanglement wedge reconstruction and bulk operator encoding.

In this project, we explore the HaPPY code, a tensor network that models key features of the holographic principle, specifically the anti-de Sitter/conformal field theory (AdS/CFT) correspondence. The HaPPY code provides a toy model for understanding how bulk spacetime geometry and quantum error correction emerge from boundary degrees of freedom. By encoding logical qubits into entangled physical qubits arranged in a hyperbolic tiling, the code mimics the geometric structure of AdS space and illustrates how local bulk operators can be reconstructed from boundary data.

1.1 Holographic Duality (AdS/CFT Correspondence)

AdS/CFT is an example of holographic duality, or gauge-gravity duality. The main thesis of AdS/CFT is that an asymptotic AdS spacetime in d-dimension bulk is dual to a conformal quantum field theory (CFT) without gravity in $d-1$-dimension boundary.

1.1.1 Anti-de Sitter Space (AdS)

The AdS space is a space with constant negative curvature, i.e. it is a hyperbolic space. Despite this unphysical feature, AdS space provides an ideal playground for exploring the connections between quantum gravity and quantum information. It has a timelike boundary at infinity, allowing for well-defined boundary conditions. It exhibits a high degree of symmetry.

The metric for AdS is given by [4]

$$ds^2 = -(r^2+1)dt^2 + \frac{dr^2}{r^2+1} + r^2 d\Omega_{d-1}^2. \tag{1.1}$$

1.1.2 Conformal Field Theory (CFT)

A conformal field theory is a field theory that is invariant under conformal transformations. Conformal transformations are transformations that locally preserve the angle. Suppose we have a map for conformal transformation φ and metric g, we have $\varphi^* g' = \Lambda g$. For $x' = \varphi(x)$, We can express the conformal transformations as follows: $g'_{\rho\sigma}(x') \frac{\partial x'^{\rho}}{\partial x^{\mu}} \frac{\partial x'^{\sigma}}{\partial x^{\nu}} = \Lambda(x) g_{\mu\nu}(x)$. Together, translations, rotations, dilations, and special conformal transformations generate the full conformal group. The special conformal transformations act on the spacetime coordinate and has the form: $x'^{\mu} = \frac{x^{\mu} - b^{\mu} x^2}{1 - 2b\cdot x + b^2 x^2}$, or $\Lambda(x) = \left(1 - 2b\cdot x + b^2 x^2\right)^{-2}$, where b^{μ} is the constant vector defining the special conformal transformation.

In conformal field theory, primary operators which are local operators that under a conformal transformation transform as $\mathcal{O}(x) \mapsto \mathcal{O}'(x') = \left|\frac{\partial x'}{\partial x}\right|^{-\Delta/d} \mathcal{O}(x)$.

Further details on radial quantization are provided in Appendix E.

1.1.3 Ryu-Takayanagi Formula

In general relativity, black holes are thermodynamic objects and we have the entropy S_{BH} of a black hole given by the Bekenstein-Hawking formula:

$$S_{BH} = \frac{\text{Area}(\Sigma)}{4G_N}$$

where Σ is the event horizon, G_N is the Newton constant.

Ryu and Takaynagi generalized the Bekenstein-Hawking formula in the context of conformal field theory with the use of AdS/CFT correspondence. Let ρ be a state in the Hilbert space of the conformal field theory. Let A be a boundary subregion. We can define a boundary state ρ_A in the subregion A. When ρ has a good semi-classical bulk dual, the von Neumann entropy of ρ_A, defined in Appendix F is given by

$$S(\rho_A) = \text{Tr}(\rho \mathcal{L}_A) + S_{bulk}(\rho_{\mathcal{E}_A})$$

where $\mathcal{L}_A$ is a local operator in the bulk integrated over γ_A. At leading order in Newton's constant G, there is

$$\mathcal{L}_A = \frac{\text{Area}(\gamma_A)}{4G},$$

and $S_{bulk}(\rho_{\mathcal{E}_A})$ is the bulk von Neumann entropy in entanglement wedge $\mathcal{E}_A$.

1.1.4 Bulk Reconstruction

The essence of AdS/CFT can be cast in terms of "bulk reconstruction:" any state and operator of the bulk should have a corresponding state or operator in the CFT. This connection is given explicitly by the "extrapolate dictionary:"

$$\lim_{r\to\infty} r^{\Delta}\phi(r,x) = \mathcal{O}(x) \tag{1.2}$$

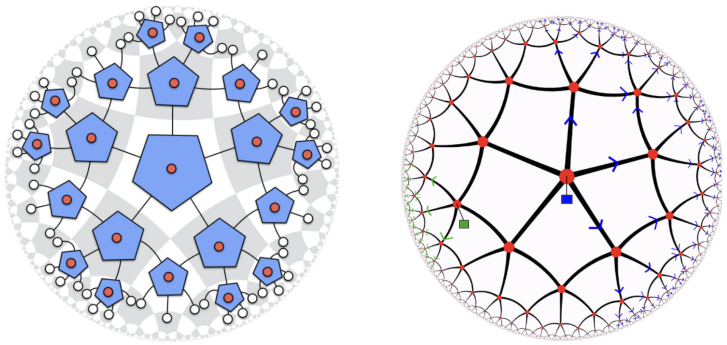
where ϕ is a free scalar field in the bulk, $\mathcal{O}$ is a corresponding operator on the boundary, and $\Delta = \frac{d}{2} + \frac{1}{2}\sqrt{d^2+4m^2}$.

In global AdS-Rindler Coordinates, a generic perscription for bulk reconstruction is of the form

$$\phi(x) = \int_{S^{d-1}\times\mathbb{R}} dY\, K(x,Y)\mathcal{O}(Y) \tag{1.3}$$

where K is a "smearing function" that is expressed in terms of the Fourier modes of the field coefficients f [5]. This is often known as the Hamilton-Kabat-Lifschytz-Lowe (HKLL) reconstruction. While effective, HKLL reconstruction becomes cumbersome as one needs to solve PDE's of the bulk dynamics and leads to the following paradox:

Consider dividing the boundary into two subregions: $R, \bar{R}$, and a bulk operator $\phi(x)$. We can choose R such that its corresponding bulk region contains $\phi(x)$, allowing for reconstruction of $\phi(x)$ on R using 1.3. Then $[\mathcal{O}_R, \mathcal{A}_{\bar{R}}] = 0$ for any operator $\mathcal{A}_{\bar{R}}$ on $\bar{R}$. Since R is arbitrary, there are many other regions of the boundary for which the commutativity relation still holds. However by Schur's lemma, this implies that $\phi(x)$ must be the identity operator, since the only operator that commutes with all other operators on irreducible subspaces is the identity, resulting in a paradox [4, 5].



(a) (b)

Figure 1. **(a)** HaPPY Pentagon Code. Pentagon tiles correspond to bulk logical states, and uncontracted legs at the edge correspond to physical boundary states. **(b)** Operator pushing. Two bulk operators shown in green and blue are reconstructed on the boundary. Figures obtained from [3].

1.2 The HaPPY Code

The key insight by Almheiri et al. [5] that resolves the paradox is the following: operators in the bulk are redundantly encoded in the boundary. Therefore, AdS/CFT can be viewed as an instantiation of quantum error correction, where $\mathcal{H}_{\text{code}}$ corresponds to states in subregions of the bulk and $\mathcal{H}_{phys}$ to regions of the boundary. That is, the operators $\mathcal{O}_R$ for different regions R in equation 1.1.4 are actually physically different operators in the boundary theory, yet they act on the code subspace in the same way. Importantly, this error-correcting code-like structure emerges when we consider operators acting on subspaces of the full bulk space [4, 6].

The HaPPY code provides a concrete model of holographic error correction by tessellating the bulk states with a hyperbolic MERA-like tensor network. Tensor networks provide the DNA for a complex quantum state. Any many-body quantum state of N particles with d spin states (e.g. $d = 2$ for qubits) can be expressed as

$$|\psi\rangle = \sum_{i_1 \ldots i_N}^{d} T_{i_1 \ldots i_N} |i_1\rangle |i_2\rangle \ldots |i_N\rangle \,. \tag{1.4}$$

where the coefficients $T_{i_1 \ldots i_N}$ contain all of the relevant physics. Tensor networks provide a visualization of this state by breaking up the large tensor $T_{i_1 \ldots i_N}$ into connected tensors of lower rank, forming a network of contractions [7, 8].

The HaPPY code implements a tensor network consisting of contracted pentagons, each of which is associated with a six-index tensor containing one logical bulk state (red dots) and five outgoing legs (see Figure 1 (a)). The pentagon contractions provide the encoding map from the bulk logical states to the "boundary" physical states (all uncontracted white dots at the edge), resulting in a state of the form in equation 1.4.

The building blocks for these pentagon tiles are "perfect tensors" T, which have two properties. First, T is an isometry meaning that $\sum_b T^\dagger_{ab} T_{ba'} = \delta_{aa'}$. Second, T becomes unitary when its indices

are evenly partitioned into input and output components [3, 4]. Specifically, one can "shift" input indices of a perfect tensor $T\,|a_2 a_1\rangle \rightarrow \sum_b T_{b a_2 a_1}$ to output indices of a corresponding perfect tensor $\tilde{T}\,|a_1\rangle \rightarrow \sum_{b a_2} |b a_2\rangle\, T_{b a_2 a_1}$ [3]. Since the composition of isometries is still an isometry (see Appendix A), then full contracted HaPPY is an isometric tensor network.

The feature of the HaPPY code perfect tensors that enables bulk reconstruction is called "operator pushing". Consider an operator O acting on an incoming leg of the perfect T. Then because T is an isometry,

$$TO = TOT^\dagger T = O'T \tag{1.5}$$

where $O' = TOT^\dagger$. Therefore, it is physically equivalent to either first act on a state with O and then contract T, or first contract with T and then act with O' [7]. Two examples of HaPPY bulk reconstruction are shown in Figure 1 (b).

1.3 Main Results

Our results consist of the following:

1) We explicitly show that the HaPPY code is a stabilizer quantum error correcting code.

2) We analyze bulk reconstruction in the HaPPY code in two regions: i) we provide a concrete example of bulk reconstruction for a local operator in the "causal wedge," with accompanying Python simulations. ii) We propose a new algorithm to generate the code subspace for "entanglement wedge" reconstruction, based on a Grover search method.

2 HaPPY Code as a QECC

Every quantum error correcting code (QECC) consists of two components: 1) an encoding map, 2) a recovery channel on a set of correctable errors [9, 10]. The utility of holographic codes for bulk reconstruction derives mostly from 1), as they encode logical bulk states into physical boundary states. But if the HaPPY code is indeed a bonafide QECC, then we should be able to identify a set of errors which the HaPPY code can correct by satisfying the Knill-Laflamme conditions.

2.1 Encoding: HaPPY code as a stabilizer code

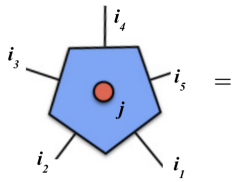
We now show that the HaPPY code is a version of standard five qubit [5,1] stabilizer code.[1] Recall that a stabilizer code on a system of n quantum states is constructed from an abelian subgroup S of the Pauli group $G_n = \pm i\{X, Y, Z, I\}^{\otimes n}$ that is generated by elements M_i which are i) Hermitian, ii) independent, and iii) commuting. Succinctly, $\langle M_i \rangle = S \leq G_n$ [2] [9–11]. The code subspace is then defined as the set of states which are stabilized by the action of the generators of S:

$$|\psi\rangle \in \mathcal{H}_{code} \iff M_i\,|\psi\rangle = |\psi\rangle \quad \forall M_i \in S \tag{2.1}$$

[1] Note that in full generality, the pentagon HaPPY code is actually [6,0] code. This means that it can be represented as a [5,1], [4,2], or [3,3] code in which each tile has 1, 2, or 3 uncontracted indices [3]. However we just consider the most intuitive representation as a [5,1] code here.

[2] We have used the group theory notation $\leq$ to denote subgroup.

As explained in section 1, the perfect six-index tensor associated to each pentagon leaves one index uncontracted to represent a logical qubit $\left|\tilde{j}\right\rangle$ and maps it to five physical qubits (i.e. the edges of a pentagon tile) [4], as shown in Figure 2:



$$= T_{i_1 i_2 i_3 i_4 i_5 j} \equiv 2 \left\langle i_1 i_2 i_3 i_4 i_5 \middle| \tilde{j} \right\rangle$$

Figure 2. The 6-index perfect tensor associated with each pentagon tile maps one qubit $\left|\tilde{j}\right\rangle$ into five physical qubits $|i_1 i_2 i_3 i_4 i_5\rangle$.

The pentagon building blocks of the HaPPY code are therefore equivalent to the well-known 5 qubit [[5,1,3]] stabilizer code [3, 4, 9, 11]. It has $n - k = 5 - 1 = 4$ stabilizers

$$\begin{aligned} M_1 &= X_1 Z_2 Z_3 X_4 I_5 \qquad M_2 = I_1 X_2 Z_3 Z_4 X_5 \\ M_3 &= X_1 I_2 X_3 Z_4 Z_5 \qquad M_4 = Z_1 X_2 I_3 X_4 Z_5, \end{aligned} \tag{2.2}$$

logical X and Z operators

$$\tilde{Z} = Z_1 Z_2 Z_3 Z_4 Z_5 \qquad \tilde{X} = X_1 X_2 X_3 X_4 X_5, \tag{2.3}$$

and encoded states [11, 12]

$$\left|\tilde{0}\right\rangle = \sum_{M \in S} M \left|00000\right\rangle \tag{2.4}$$

$$\left|\tilde{1}\right\rangle = \tilde{X} \left|\tilde{0}\right\rangle . \tag{2.5}$$

The five outgoing legs of each pentagon tile are then contracted with other pentagon tensors, resulting in a concatenated 5 qubit code at the boundary [4].

2.2 Recovery: Satisfying Knill-Laflamme conditions

A quantum error correcting code can achieve sucessful recovery if it satsifes the Knill-Laflamme conditions: for any two orthonormal eigenkets $\left|\tilde{i}\right\rangle, \left|\tilde{j}\right\rangle \in \mathcal{H}_{code}$,

$$\left\langle \tilde{i} \middle| E_{\bar{a}}^{\dagger} E_{\bar{b}} \middle| \tilde{j} \right\rangle = C_{\bar{a}\bar{b}} \delta_{ij} \tag{2.6}$$

where $E_{\bar{a}}, E_{\bar{b}}$ are arbitrary Pauli errors and C_{ab} is an arbitrary matrix that depends only on the indices $\bar{a}, \bar{b}$ [9, 11]. The choice of overbar notation will be made clear in Theorem 1 below.

The type of errors relevant to holographic codes are "erasures." A quantum erasure occurs when the error is unknown, but the location of the error is known [13]. A quantum erasure channel simplifies the Knill-Laflamme conditions because one only needs to account for errors $E_{\bar{a}}$ and $E_{\bar{b}}$

acting on the same location $\bar{A}$. Since a product of single-site errors of the algebra basis is an error still acting on the same site, equation (2.6) becomes

$$\langle \tilde{i} | E_{\bar{A}} | \tilde{j} \rangle = c_{\bar{A}} \delta_{ij} \tag{2.7}$$

where now $c_{\bar{A}}$ is a scalar corresponding to the matrix element $\langle \tilde{i} | E_{\bar{A}} | \tilde{i} \rangle$ [13].

Let us follow [4, 5, 14] and take the three [3,1] qutrit code as a simple illustration. The codewords are $|\tilde{0}\rangle = \frac{1}{\sqrt{3}}(|000\rangle + |111\rangle + |222\rangle)$, $|\tilde{1}\rangle = \frac{1}{\sqrt{3}}(|012\rangle + |120\rangle + |201\rangle)$, and $|\tilde{2}\rangle = \frac{1}{\sqrt{3}}(|021\rangle + |102\rangle + |210\rangle)$. Suppose we want to protect against erasure errors on the third qutrit. Notice that the encoding of the qutrit codewords can be given by a unitary transformation on the three physical qutrits via

$$|\tilde{i}\rangle = U_{12}(|i\rangle_1 \otimes |\chi\rangle_{23}) \tag{2.8}$$

where $|\chi\rangle_{23} = \frac{1}{\sqrt{3}}(|00\rangle + |11\rangle + |22\rangle)$ and U_{12} enacts the transformation given in Appendix B [4, 5].

Since the unitary U_{12} that generates the encoding only has support on the first two qutrits, the QECC is able to protect against erasures errors on the the third qutrit, as desired. This feature generalizes to part of a theorem proposed by Harlow in [4, 5, 14]:

Theorem 1: *Suppose the full physical Hilbert space factorizes into regions A and its compliment $\bar{A}$, $\mathcal{H} = \mathcal{H}_A \otimes \mathcal{H}_{\bar{A}} \supset \mathcal{H}_{code}$. Further consider coupling a reference system R with the same size as the codespace. If $|R| = |code| \leq |A|$, then:*

- *by the division algorithm, $\mathcal{H}_A = (\mathcal{H}_{A_1} \otimes \mathcal{H}_{A_2}) \oplus \mathcal{H}_{A_3}$ with $|A_1| = |R|$, $|A_3| < |R|$*
- *there exists a unitary $U_A : \mathcal{H}_A \to \mathcal{H}_A$ and state $|\chi\rangle_{A_2\bar{A}}$ such that*

$$|\tilde{i}\rangle_{A\bar{A}} = U_A \left(|i\rangle_{A_1} \otimes |\chi\rangle_{A_2\bar{A}}\right) \tag{2.9}$$

where $|\tilde{i}\rangle_{A\bar{A}} \implies |\tilde{i}\rangle \in \mathcal{H}$ under this representation.

Physically, Theorem 1 states that if the boundary is divided into two subregions A and $\bar{A}$, then one can construct a boundary representation of a bulk codeword $|\tilde{i}\rangle$ which ensures it can protect against erasures on $\bar{A}$. A is the region we want to protect, $\bar{A}$ is the region where erasures can occur. It turns out that Theorem 1 implies the Knill-Laflamme conditions in 2.7. We prove this in Appendix A to conserve space. The idea is that if an error $E_{\bar{A}}$ acts only on $\bar{A}$, then it will be proportional to the identity on A given that the encoding map U_A in 2.9 has support only on A.

To be concrete, let us consider how Theorem 1 translates to the HaPPY pentagon 5 qubit code of section 2.1. The code subspace consists of one logical qubit $\tilde{i}$ encoded in five physical qubits i_1, i_2, i_3, i_4, i_5. Since $|A_1| = |code| = 2^1$, without loss of generality say $A_1 \cong \text{span}\{i_1\}$. Since $|A_3| < |R| = |code|$ and since the codespace only has one qubit, then $|A_3| = 0$ so A_3 is empty. The 5 qubit code should protect against erasures on any two of the five qubits [4], say on qubits 4 and 5. Then $\bar{A} \cong \text{span}\{i_4, i_5\}$ and hence $A \cong \text{span}\{i_1, i_2, i_3\}$. This leaves $A_2 = \text{span}\{i_2, i_3\}$ by the division algorithm. Consequently, any encoded pentagon state should be expressible as

$$|\tilde{i}\rangle = U_{123}(|i\rangle_1 \otimes |\chi\rangle_{23,45}). \tag{2.10}$$

By considering $\tilde{i} = \tilde{0}$ and comparing equations 2.10 and 2.4, we can derive a constraint that U_{123} should be given by

$$\left|\tilde{0}\right\rangle = U_{123}(\left|0\right\rangle_1 \left|\chi\right\rangle_{23,45}) = \sum_{M \in S} M \left|00000\right\rangle . \tag{2.11}$$

An explicit U_{123} and $\left|\chi\right\rangle_{23,45}$ are worked out in Appendix B. Therefore, a given pentagon will satisfy the Knill-Laflamme erasure conditions 2.7:

$$\left\langle\tilde{i}\right|E_{45}\left|\tilde{j}\right\rangle = \left\langle i\right|_1 \left\langle\chi\right|_{23} U_{123}^{\dagger} U_{123} \left|j\right\rangle_1 \left|\chi\right\rangle_{23} \left\langle\chi\right|_{45} E_{45} \left|\chi\right\rangle_{45} = \delta_{ij} c_{45}. \tag{2.12}$$

Since each individual [5,1] pentagon code protects against erasure errors, the full concatenated HaPPY code will also protect against erasure errors.

3 Causal Wedge Reconstruction

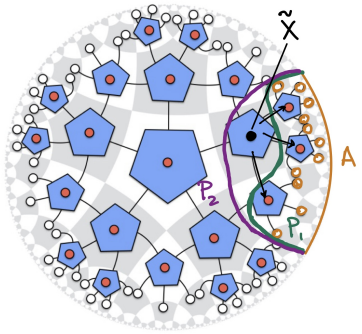


Figure 3. Example of causal wedge reconstruction of $\tilde{X}$. For boundary region A in orange, the causal wedge $C[A]$ is constructed from the Greedy algorithm, which in this case consists of two cuts corresponding to tensors P_1 (green) and P_2 (purple).

We now turn to analyzing how the HaPPY code enables bulk reconstruction. To reconstruct bulk operators that can causally influence bulk points corresponding to a boundary region A, one must perform "causal wedge reconstruction." Formally, the *causal wedge* $C[A]$ of A is defined as the intersection of the bulk future and past of the boundary domain of dependence $D[A]$ (i.e. any point on the boundary for which any causal curve also intersects A)[5]:

$$C[A] = J^{+}[D[A]] \cap J^{-}[D[A]]. \tag{3.1}$$

The left time slice of Figure 4 exhibits two causal wedges corresponding to "top" and " bottom" boundary regions A_t and A_b.

3.1 Example of causal wedge reconstruction using Greedy algorithm

How might one construct a causal wedge $C[A]$ region in the HaPPY code tensor network? The authors of [3] propose a method called the *"greedy algorithm"*. The greedy algorithm consists of making a sequence of cuts $\{c_\alpha\}$ into the bulk and an associated sequence of isometries $\{P_\alpha\}$

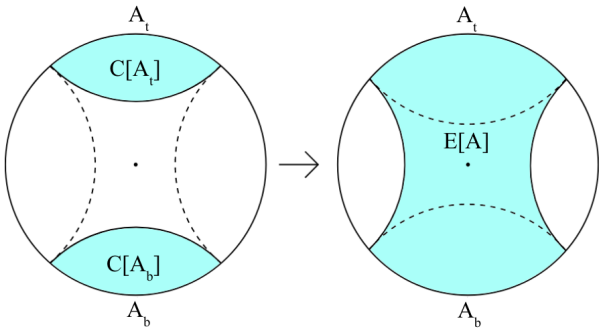


Figure 4. Given a disconnected boundary region $A = A_t \cup A_b$, the left shows the causal wedge $C[A] = C[A_t] \cup C[A_b]$ (the union of two individual causal wedges) and the right shows the entanglement wedge $E[A]$, which extends through the bulk. Figure modified from [3].

generated by the following steps: 1) Perform a trivial cut: start with A itself. 2) Each subsequent cut is made around a perfect tensor $P_{\alpha+1}$ which has at least half of its legs contracted with P_α. This means each pentagon included in $P_{\alpha+1}$ must have at least three legs contracted with a pentagon in P_α. 3) Continue until halt. The causal wedge $C[A]$ will correspond to the set of bulk points reached by the greedy algorithm applied to A, according to [3].

Let us consider reconstructing the operator $\tilde{X}$ as in equation 2.3 acting on a bulk logical state $\left|\tilde{j}\right\rangle \in C[A]$. Suppose that $C[A]$ has been constructed from a greedy algorithm with N cuts, corresponding to a collection of isometries $\{P_i\}_{i=1}^N$. Then, by the tensor pushing formula 1.5, the boundary representation of $\tilde{X}$ will be

$$X' = P_1 \dots P_N \tilde{X} P_N^\dagger \dots P_1^\dagger \tag{3.2}$$

For example, if one chooses the boundary region A as in Figure 3, then the greedy algorithm causal wedge will consist of a sequence of two isometries P_1, P_2. To push the operator $\tilde{X}$ acting on the pentagons in the corresponding $C[A]$, the boundary representation will be $P_1 P_2 \tilde{X} P_2^\dagger P_1^\dagger$. A Python simulation of causal wedge reconstruction for the HaPPY code using the greedy algorithm is discussed in Appendix C.

4 Entanglement Wedge Reconstruction

Another region of interest for bulk reconstruction is the "entanglement wedge." The entanglement wedge is defined as the "set of bulk points in the bulk region bounded by A and γ_A, where γ_A is the minimal bulk geodesic whose boundary matches the boundary of A" [3]. It follows that $C[A] \subset E[A]$, so oftentimes the entanglement wedge extends much further into the bulk than the causal wedge. Entanglement wedge reconstruction naturally arises in settings where the boundary region of interest contains two disconnected subregions such that the entanglement wedge results in a wormhole-like bulk geometry connecting them as shown in Figure 4.

4.1 Difficulty: Entanglement Wedge Reconstruction with Greedy Algorithm

How can one generate the code subspace corresponding to the entanglement wedge in the HaPPY code? The authors of [3] propose the "greedy entanglement wedge" as a possible solution, which is the "set of bulk points reached by applying the greedy algorithm to to all connected components of A simultaneously." However, this greedy entanglement wedge has shortcomings. First, this method requires applying the greedy algorithm to each individual boundary subregion $A_i \in \cup_i A_i = A$, which can present a problem when trying to match cuts coming from different boundary regions deep in the bulk. Second, there might be operators which lie outside of the greedy algorithm region so the greedy entanglement wedge does not always coincide with the true entanglement wedge. As a result, the greedy algorithm does not guarantee that $E[A] \cup E[A^c]$ covers the full bulk, which is an expected property [3].

4.2 Proposal: Grover Search HaPPY Algorithm

As such, we propose a novel algorithm that the HaPPY code could employ to generate the entanglement wedge code subspace. The algorithm is based on a Grover search method inspired by similar techniques used in the Python lunch conjecture [15–17] and in Hayden-Preskill decoding [18]. We start by considering a disconnected boundary region $A = A_t \cup A_b$ as in Figure 4 and suppose that A consists of n qubits so that $|\mathcal{H}_A| = 2^n$. Pastawski et al. [3] state that "according to the entanglement wedge conjecture, we should always be able to reconstruct the center of the bulk from a sufficiently large fraction of the boundary." This necessary condition for entanglement wedge reconstruction of being able to reconstruct on operator on the center pentagon provides the key idea for the Grover algorithm: mark the center pentagon state, $|\tilde{c}\rangle$, and perform a Grover search to amplify it.

First we need to provide a boundary representation of $|\tilde{c}\rangle$ on the physical subregion A. According to Theorem 1 of section 2.2, we can represent the center pentagon encoding as

$$|\tilde{c}\rangle = U_A(|c\rangle_{A_1} |\chi\rangle_{A_2\bar{A}}) \tag{4.1}$$

where the support of U_A is solely on the boundary region $A = A_t \cup A_b$. Now we can formulate a Grover search for $|\tilde{c}\rangle$ in a two-dimensional subspace spanned by $|\tilde{c}\rangle$ and $|\tilde{c}\rangle_\perp$ entirely in the (physical) boundary region A. Each Grover iteration will correspond to a cut further into the bulk from A, similar to the greedy algorithm.

1. Initialization A standard Grover search begins by creating an initial uniform superposition state $|s\rangle$ by applying the n qubit Hadamard gate $|s\rangle = H^{\otimes n}|0\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{N-1} |x\rangle$. Here, we can do even better and use a "generalized Grover search" given that we know some of the structure of $|\tilde{c}\rangle$ [11]. We replace $H^{\otimes n}$ by U_A to create the initial state

$$|a\rangle = U_A |0\rangle_A \tag{4.2}$$

where $|0\rangle_A = |0_1 \dots 0_n\rangle \in \mathcal{H}_A$.

2. Grover rotation. Next we apply the Grover unitary $U_G = U_a U_c$ to $|a\rangle$ where

$$U_c = I_A - 2|\tilde{c}\rangle\langle\tilde{c}| = I_A - 2U_A \left(|c\rangle_{A_1} |\chi\rangle_{A_2\bar{A}}\right) \left(\langle c|_{A_1} \langle\chi|_{A_2\bar{A}}\right) U_A^\dagger \tag{4.3}$$

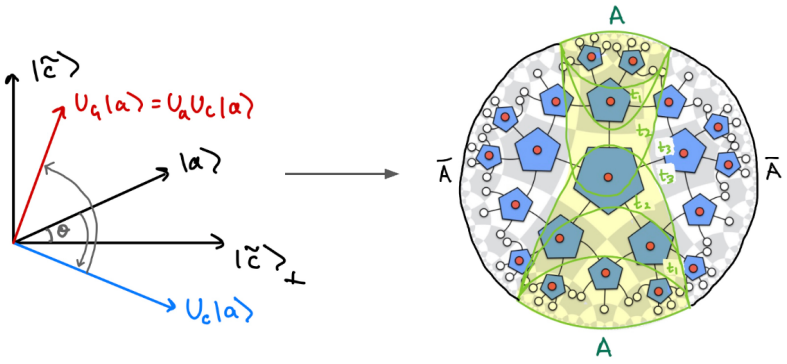


Figure 5. Grover search HaPPY code algorithm for entanglement wedge reconstruction. Schematic on the right shows the Grover rotation (left) iterated three times, t_1, t_2, t_3 in green, resulting in three cuts which define the shaded entanglement wedge.

and

$$U_a = 2\,|a\rangle\langle a| - I_A = U_A(2\,|0\rangle\langle 0|_A - I_A)U_A^\dagger \tag{4.4}$$

so that

$$U_G = U_A\left(2\,|0\rangle\langle 0|_A - I_A\right)\left(I_A - 2\,|c\rangle_{A_1}\,|\chi\rangle_{A_2\bar{A}}\,\langle c|_{A_1}\,\langle\chi|_{A_2\bar{A}}\right)U_A^\dagger. \tag{4.5}$$

We then construct a cut from A into the bulk corresponding to U_G. By leveraging the property that every perfect tensor corresponds to a unitary when its number of input and output indices are balanced, we can make a cut containing perfect tensors P_1 by defining

$$P_1^\dagger = \langle b|U_G|a\rangle \tag{4.6}$$

where $|b\rangle = |i_1 \ldots i_n\rangle$ has the same number of indices as $|a\rangle$ and consists of bulk legs $i_1 \ldots i_n$ one layer in from the boundary region A.[3] However, given the hyperbolic curvature of AdS, it will often be the case that a cut maintaining the same number of bulk indices as uncontracted indices in A cannot be made. To remedy this, we use the other property of perfect tensors that input indices can be pushed to output indices. In this case, the cut corresponding to P_1 is made such that the maximum number of available bulk indices are kept in $|b\rangle$ and the rest are shifted over $|a\rangle$[4]. This procedure is analagous to the "maxi-min" quantum extremal surface protocol discussed in current literature [19].

Moreover, we are guaranteed that we can find a set of tiles in the bulk corresponding to 4.6 because we chose U_G to depend on U_A, the encoding map for $|\tilde{c}\rangle$, and U_A is itself constructed from a composition of encoding maps for individual pentagon tiles. These are precisely the [5,1] encoding maps previously defined in equation 2.11.

[3] Since the Grover method goes from the boundary into the bulk instead of the other direction, P_1 is defined in terms of its Hermitian conjugate.

[4] Specifically, the vector $|a\rangle$ is really $|a\rangle_A\,|0\rangle_{\bar{A}}$ so the remaining bulk indices should be shifted to $|0\rangle_{\bar{A}}$ so as to leave the Grover search on region A unaffected.

3. Iteration. For each iteration t of step 2, the initial vector $|a\rangle \in \mathcal{H}_A$ will further approach $|\tilde{c}\rangle$ by $(U_G)^t |a\rangle$, and we will have made t cuts into the bulk with corresponding perfect tensors

$$P_t^\dagger = \langle b|(U_G)^t|a\rangle \,. \tag{4.7}$$

If the number of contracted bulk indices in layer t is not $|A|$ or if the candidate pentagons in layer t map to any state in $\bar{A}$, then we make P_t an isometry and not a unitary as in step 2 by taking the maximum number of available indices in $|b\rangle$. An interpretation of this process in terms of "post-selection," in line with [15], is given in Appendix D. Figure 5 gives a schematic of the Grover HaPPY algorithm.

4.3 Efficacy of the Grover HaPPY Algorithm

After t iterations, the initial angle θ between $|a\rangle$ and $|\tilde{c}\rangle_\perp$ will have moved to $(2t+1)\theta$ [11]. Therefore, the number of iterations t_f to exactly reach $|\tilde{c}\rangle$ will be given by $(2t_f+1)\theta = \pi/2$, or

$$t_f = \frac{\pi}{4\theta} - \frac{1}{2}. \tag{4.8}$$

Therefore, the HaPPY Grover Algorithm would require pentagon tilings with number of layers $l \geq t_f$. Moroever, it would be optimal on a HaPPY code with an infinite number of tiling layers (see [20]). Even with sufficient layers, we note that the Grover algorithm is not exact, and so there will be some error ϵ in reconstructing the center tile,

$$\left\| |\tilde{c}\rangle - (U_G)^t |a\rangle \right\|^2 = \epsilon. \tag{4.9}$$

These downsides withstanding, we have shown the Grover HaPPY algorithm can define a code subspace which includes the center tile, thus satisfying a necessary condition for EW reconstruction. Does it improve upon the shortcomings of the greedy algorithm?

Indeed, firstly, the Grover HaPPY algorithm runs on the combined space $A = A_t \cup A_b$ and not A_b and A_t individually. Second, we expect it recovers the property that $E[A] \cup E[\bar{A}]$ fills the whole bulk space, for which the greedy algorithm failed. This is a result of how the Grover search process is designed: it projects states onto two subspaces $|\tilde{c}\rangle\langle\tilde{c}|$ and $|\tilde{c}\rangle\langle\tilde{c}|_\perp$. The Grover HaPPY algorithm associates the codespace of $E[A]$ with the $|\tilde{c}\rangle\langle\tilde{c}|$ subspace, given the choice of initialization unitary U_A. Therefore, just as the boundary is divided between A and $\bar{A}$, we expect the bulk to be divided between $|\tilde{c}\rangle\langle\tilde{c}|$ and $|\tilde{c}\rangle\langle\tilde{c}|_\perp$. Applying the Grover HaPPY algorithm to $\bar{A}$ would result in the filling of the complementary space in the bulk (the unshaded region of Figure 5). The error in filling up the whole space will be proportional to ϵ^2.

5 Conclusion and Future Directions

We showed that the HaPPY code is a true QECC, with an encoding given by a concatenated [5,1] 5 qubit code for each contracted pentagon and a recovery channel for erasure errors. Using the tool of operator pushing, we then reconstructed the local bulk operator $\tilde{X}$ acting in the causal

wedge. Lastly, we proposed a novel Grover search HaPPY algorithm that improves on the greedy algorithm for generating the entanglement wedge code subspace.

In the future, we would like to consider applying these techniques to reconstruct a local bulk Hamiltonian and relate dynamics in the bulk to dynamics in the boundary [20, 21]. Moreover, we envision implementing the Grover search algorithm in a Python simulation to see if we can reconstruct an operator in the entanglement wedge as we did for the causal wedge. Doing so will help identify cases in which the Grover search algorithm fails so that it can be improved.

Meanwhile, in the original HaPPY code paper [3], Pastawski et al. also introduce the Qutrit Hexagon Code, which we can further explore to verify that it indeed functions as a QECC and supports both causal and entanglement wedge reconstructions. We can also consider generalizing the HaPPY code framework by exploring n-qudit codes on m-gon tilings. At the same time, since the HaPPY code relies on perfect tensors, it is worth investigating holographic QECCs that do not require perfect tensors. As the perfect-tensor tiling has no preferred past or future direction and clashes with the locality, it is difficult to produce boundary dynamics resembling a CFT. We hope to explore further in understanding time-dependent bulk dynamics with holographic codes.

Acknowledgments

BK, MD, & AP thank Thilo Scharnhorst and Geoff Penington for helpful discussions and feedback on the ideas discussed in this paper in office hours.

A Proofs of some statements throughout

The composition of isometries is an isometry: Let W, V be isometries such that $V^\dagger V = I$ and $W^\dagger W = I$. Then

$$(VW)^\dagger (VW) = W^\dagger V^\dagger V W = W^\dagger I W = I \tag{A.1}$$

so WV is also an isometry. ■

Now we prove the claim in section 2.2 that Theorem 1 implies the Knill-Laflamme conditions. We note that $E_{\bar{A}} = E_{\bar{A}} \otimes I_A$ and substitute equation 2.9 into 2.7,

$$\langle \tilde{i} | E_{\bar{A}} | \tilde{i'} \rangle = \langle i |_{A_1} \langle \chi |_{A_2 \bar{A}} U_A^\dagger (E_{\bar{A}} \otimes I_A) U_A \left| i' \right\rangle_{A_1} | \chi \rangle_{A_2 \bar{A}}$$

Let us write $|\chi\rangle_{A_2\bar{A}} = |\chi\rangle_{A_2} |\chi\rangle_{\bar{A}}$. Noting that $A = A_1 \otimes A_2$,

$$\begin{aligned} \langle \tilde{i} | E_{\bar{A}} | \tilde{i'} \rangle &= \langle i |_{A_1} \langle \chi |_{A_2} U_A^\dagger I_A U_A \left| i' \right\rangle_{A_1} | \chi \rangle_{A_2} \langle \chi |_{\bar{A}} E_{\bar{A}} | \chi \rangle_{\bar{A}} \\ &= \delta_{ii'} c_{\bar{A}} \end{aligned} \tag{A.2}$$

where $c_{\bar{A}} = \langle \chi |_{\bar{A}} E_{\bar{A}} | \chi \rangle_{\bar{A}}$. ■

B Further detail on HaPPY Code as a QECC

The unitary encoding U_{12} for the three qutrit code discussed in section 2.2 transforms according to

$$\begin{array}{lll} |00\rangle \to |00\rangle & |11\rangle \to |20\rangle & |22\rangle \to |10\rangle \\ |01\rangle \to |11\rangle & |12\rangle \to |01\rangle & |20\rangle \to |21\rangle \,. \\ |02\rangle \to |22\rangle & |10\rangle \to |12\rangle & |21\rangle \to |02\rangle \end{array} \tag{B.1}$$

We can similarly model the map U_{123} and state $|\chi\rangle_{23,45}$ for the 5 qubit code.

The full expression for $\left|\tilde{0}\right\rangle$ in equation (2.4) is [12]

$$\begin{aligned}
|\tilde{0}\rangle =&|00000\rangle + M_1|00000\rangle + M_2|00000\rangle + M_3|00000\rangle + M_4|00000\rangle \\
&+ M_1M_2|00000\rangle + M_1M_3|00000\rangle + M_1M_4|00000\rangle \\
&+ M_2M_3|00000\rangle + M_2M_4|00000\rangle + M_3M_4|00000\rangle \\
&+ M_1M_2M_3|00000\rangle + M_1M_2M_4|00000\rangle + M_1M_3M_4|00000\rangle \\
&+ M_2M_3M_4|00000\rangle + M_1M_2M_3M_4|00000\rangle \\
=&|00000\rangle + |10010\rangle + |01001\rangle + |10100\rangle \\
&+ |01010\rangle - |11011\rangle - |00110\rangle - |11000\rangle \\
&- |11101\rangle - |00011\rangle - |11110\rangle - |01111\rangle \\
&- |10001\rangle - |01100\rangle - |10111\rangle + |00101\rangle
\end{aligned} \tag{B.2}$$

and $\left|\tilde{1}\right\rangle$ is

$$\begin{aligned}
|\tilde{1}\rangle =&\bar{X}|\tilde{0}\rangle \\
=&|11111\rangle + |01101\rangle + |10110\rangle + |01011\rangle \\
&+ |10101\rangle - |00100\rangle - |11001\rangle - |00111\rangle \\
&- |00010\rangle - |11100\rangle - |00001\rangle - |10000\rangle \\
&- |01110\rangle - |10011\rangle - |01000\rangle + |11010\rangle.
\end{aligned} \tag{B.3}$$

Notice that in the subspace of the first three qubits, there are $2^3 = 8$ unique possible states. Each of these eight is repeated once to make up the sixteen states of B.2. Keeping with the Theorem 1 representation (2.10)

$$\left|\tilde{i}\right\rangle = U_{123}(|i\rangle_1 \otimes |\chi\rangle_{23,45}), \tag{B.4}$$

we provide one possible explicit form for U_{123} and $|\chi\rangle$ below. Suppose

$$\begin{aligned}
|\chi\rangle_{23,45} =&|0000\rangle + |0010\rangle + |1001\rangle + |0100\rangle \\
&+ |1010\rangle - |1011\rangle - |0110\rangle - |1000\rangle \\
&- |1101\rangle - |0011\rangle - |1110\rangle - |1111\rangle \\
&- |0001\rangle - |1100\rangle - |0111\rangle + |0101\rangle.
\end{aligned} \tag{B.5}$$

Then U_{123}, with action $U_{123}\,|i_1i_2i_3\rangle \otimes I_4I_5\,|i_4i_5\rangle$ on each term $|i_1i_2i_3i_4i_5\rangle \in |i_1\rangle\,|\chi\rangle_{23,45}$, would be given by the following:

If $i_1 = 0$:

$$U_{123} = \begin{cases} I_1I_2I_3 & \text{if } \sum_{k=1}^{5} i_k \bmod 2 = 0 \\ X_1I_2I_3 & \text{else} \end{cases} \tag{B.6}$$

Else if $i_1 = 1$:

U_{123}	I_1	X_1	Z_1	X_1Z_1
$i_2i_3i_4i_5$	0011	0001	0000	0010
	0101	0111	0110	0100
	1010	1000	1001	1011
	1100	1110	1111	1101

where each case of U_{123} implicitly also acts with I_2I_3.

It can be checked that these definitions for $|\chi\rangle$ and U_{123} recover $|\tilde{0}\rangle$ and $|\tilde{1}\rangle$ from (B.4).

C Causal Wedge Reconstruction Simulations

We have written Python code generating a random tensor network model of the HaPPY code with two layers (see Figure 6). We use random tensors, since it was stated in [4] that such tensors could resolve problems that stabilizer perfect tensors have.

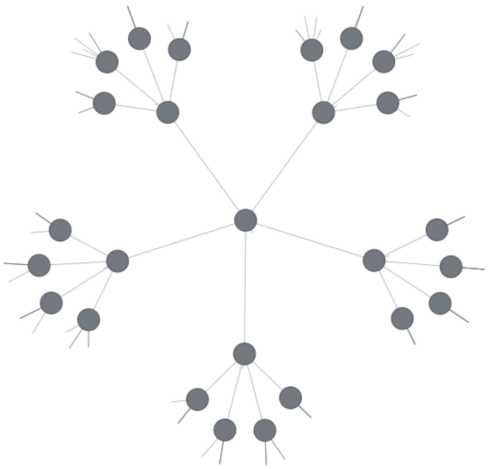


Figure 6. Python tensor network representation of HaPPY code with random tensors

One can construct a causal wedge in this code by implementing the greedy algorithm directly and performing operator pushing. A more streamlined approach is to use the LEGO HQEC software developed in [7], which has preset HaPPY operator pushing encodings. An operator pushing simulation which represents all stabilizers on the boundary was performed. For example, on a bulk tensor labeled 4, the operator $IZXIZX$ has the boundary representation as shown in Figure 7.

```
Starting from tensor 4, aiming to push operator: ['I', 'Z', 'X', 'I', 'X', 'Z']
Internal legs: IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII
reading_boundary_complete: True
Result: IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIXXIZXXXXXIIIIIXXXXXZXIXIIIIIIIIIIIIIIII
Logical: I
```

Figure 7. Python output of LEGO HQEC simulation

D Further Details on Grover HaPPY Algorithm

An alternative way of viewing the process of shifting indices in P_t is post-selection, as discussed in [15–17]. In this case we can view $P_t^\dagger$ as a unitary if we include ancilla qubits $|0\rangle_R$ from the reference system $\mathcal{H}_R$ such that $P_t^\dagger : U_A |a\rangle_A |b_r\rangle_{\bar{A}} \mapsto |b_{n-r}\rangle |0^{\otimes r}\rangle_R$ where the subscript r denotes the "remaining" indices discussed above (cf. Figure 8 of [15]). Therefore, $|\tilde{c}\rangle$ actually can be viewed as the post-selected state

$$V|a\rangle = \sqrt{N}\left\langle 0^{\otimes n}\right| (U_G)^{t_f} \left|0^{\otimes k}\right\rangle |a\rangle \tag{D.1}$$

where $k \geq n$ (cf. equation 2.0 of [16]).

Another necessary condition for entanglement wedge reconstruction that could be checked is the "JLMS" condition [6, 22]. Dividing the boundary into $\mathcal{H}_{phys} = \mathcal{H}_A \otimes \mathcal{H}_{\bar{A}}$ and bulk into $\mathcal{H}_{code} = \mathcal{H}_a \otimes \mathcal{H}_{\bar{a}}$, where $\mathcal{H}_{\bar{a}}$ conssits of states in the entanglement wedge $E[A]$, we should verify that the Grover algorithm preserves

$$S(\rho_{\bar{A}}|\sigma_{\bar{A}}) = S(\rho_{\bar{a}}|\sigma_{\bar{a}}) \tag{D.2}$$

where S is the relative entropy between states ρ and σ in the specified subspaces.

E Radial Quantization and Operator-State Correspondence

Consider conformal field theories on 2-dimensional Euclidean space, with Euclidean time direction x^0 and space direction x^1. We can compactify Euclidean space direction x^1 on a circle of radius R and choose $R = 1$. The conformal field theory can now be defined on a cylinder of infinite length for which we have the complex coordinate

$$z = x^0 + ix^1, \quad z \sim z + 2\pi i.$$

To radial quantize the above theory, we map the cylinder to the complex plane with a change of variables

$$w = e^z = e^{x^0} \cdot e^{ix^1}.$$

In radial quantization, inserting an operator $\mathcal{O}(0)$ at the origin creates a state on a small circle around $w = 0$. In particular, we define a state

$$|\mathcal{O}\rangle := \lim_{w \to 0} \mathcal{O}(w)|0\rangle,$$

where $|0\rangle$ is the vacuum. That is, inserting the operator $\mathcal{O}$ at the origin is equivalent to preparing a state $|\mathcal{O}\rangle$ on a small circle around $w = 0$. Conversely, any state prepared on a circle of radius r (say $r \to 0$) can be associated with some local operator insertion at $w = 0$. This is the state-operator correspondence: States on a circle correspond to local operators at the origin.

F Entanglement Entropy

Entanglement entropy is a measure of entanglement. Von Neumann entanglement entropy is defined as:

$$S(\rho) = -\operatorname{Tr}(\rho \ln(\rho))$$

which has the operational definition of being the asymptotic number of Bell states per copy required to distill a large number of copies of ρ. When a pure state is written in the Schmidt basis, the Von Neumann entropy is exactly analogous to Gibbs' thermodynamic entropy. Renyi entanglement entropy is defined as:

$$S_\alpha(\rho) = \frac{1}{1-\alpha} \operatorname{Tr}(\rho^\alpha)$$

which is useful because it can be calculated as the expectation value of a permutation operator $S_n(\rho) = \operatorname{Tr}(\sigma \rho^{\otimes n})$ and Von Neumann entropy can be recovered as a limit:

$$S(\rho) = \lim_{\alpha \to 1} S_\alpha(\rho)$$

G Proof of Explicit Operator Maps from the Bulk to the Boundary (Gesteau & Kang, 2020)

From Gesteau and Kang [20], we have an example for constructing the stabilizer generators for each level of the infinite-dimensional generalization of the HaPPY code. In this section, we summarize the example.

We have shown in the above sections that the original perfect tensor is constructed with the 5 qubit $[[5, 1, 3]]$ stabilizer code, with stabilizer generated by the cyclic permutations of $S = XZZXI$. We can multiply by $ZZX1X$ to get the stabilizer $S = YXXY1$.

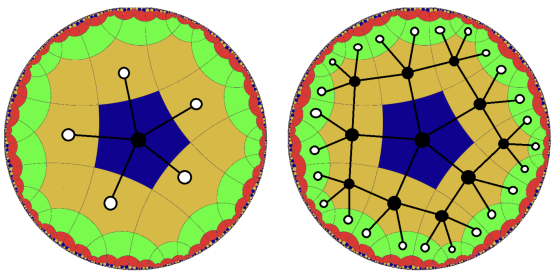


Figure 8. The left diagram shows the tensor network produced at level 1, which has 1 bulk qubit and 5 boundary qubits. The right diagram shows the tensor network produced at level 2, which has 11 bulk qubits and 25 boundary qubits. [20].

From Figure 8, for the pentagon tiling, we can see that, while in the first level, the central pentagon contributes 5 boundary nodes, most of the outside level of each bulk node provide 2 or 3 boundary modes, named as 2-clusters and 3-clusters, with the number of 2-clusters as a_n and the number of 3-clusters as b_n, for $n \geq 2, n \in \mathbb{N}$. We have the resursion relations of a_n and b_n as follows:

$$\begin{cases} a_n = 2a_{n-1} + 3b_{n-1}, & a_2 = 5 \\ b_n = a_{n-1} + 2b_{n-1}, & b_2 = 5 \end{cases}$$

n	2	3	4	5	6	7	8	9	10
a_n	5	25	95	355	1325	4945	18455	68825	256775
b_n	5	15	55	205	765	2855	10655	39705	147305

Table 1. Values of a_n and b_n from level $n = 2$ to $n = 10$.

From Table 1, we can see the number of boundary clusters and nodes grows exponentially with each level.

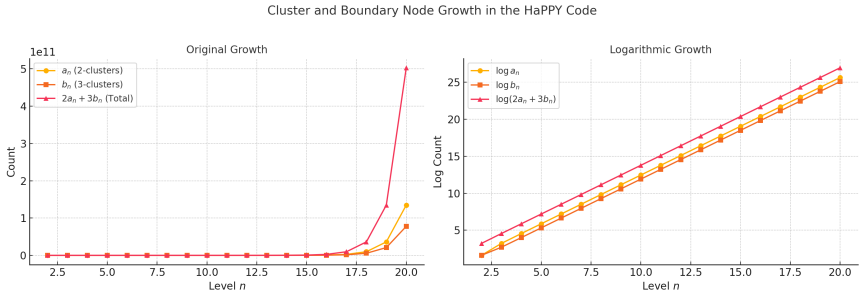


Figure 9. Cluster and Boundary Node Growth

Now we want to put each level of HaPPY code as a stabilizer code, which we can write the number of physical qubits p_n at level n as follows:

$$p_n = \widetilde{N}_n = 2a_n + 3b_n = a_{n+1};$$

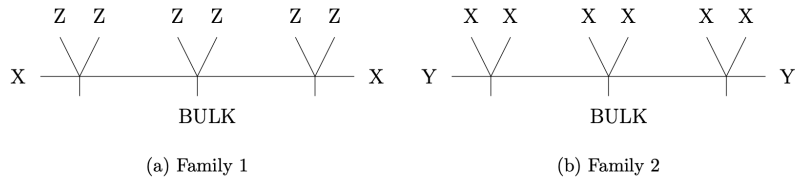
and number of logical qubits l_n as

$$\ell_{n+1} - \ell_n = N_{n+1} - N_n = a_{n+1} + b_{n+1}.$$

Since we can have stabilizers generated by cyclic permutations of either $XZZXI$ and $YXXYI$, we have two families of stabilizers:

n	2	3	4	5	6	7	8	9	10
p_n	25	95	355	1325	4945	18455	68815	256385	955165
ℓ_n	11	51	201	761	2851	10651	39701	148681	557041
ℓ_n/p_n	0.44	0.537	0.566	0.574	0.577	0.577	0.577	0.580	0.583

Table 2. Values of p_n, ℓ_n, and the ratio ℓ_n/p_n from level $n = 2$ to $n = 10$.



(a) Family 1 (b) Family 2

Figure 10. Two families of stabilizer [20].

We can push stabilizers one layer at a time using the rule in Figure 10: a weight-two operator YY on an internal edge is replaced by the weight-four string $YXXY$ on the next layer's legs, so that each node has the five legs attached it being appropriate cyclic permutation of the stabilizer $YXXYI$. In the case of the central tensor, as shown in Figure 12, we start with $S = YXXYI$ and push to $S' = YXXYIIYYIIYXXYIIIIIIII$, and the clusters such as YIY are traced out so they are not included in this example stabilizer.

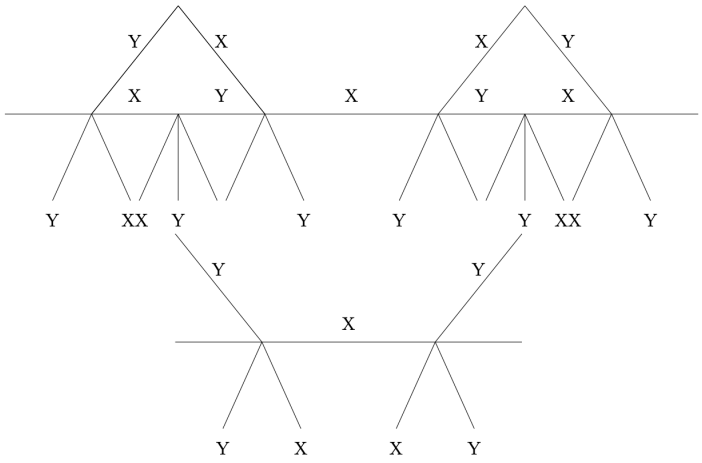


Figure 11. Pushing YY to $YXXY$ and Pushing $YXXY$ to $YXXYIIYYIIYXXY$[20].

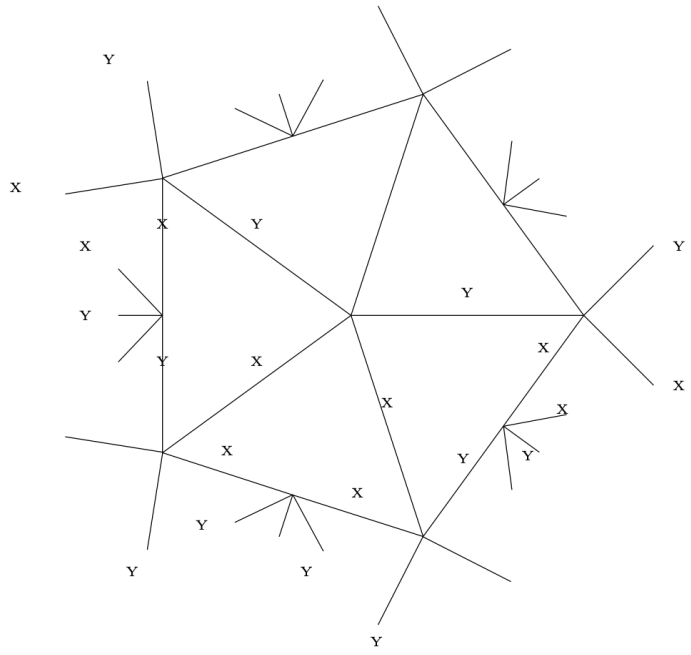


Figure 12. Pushing Stabilizer in the case of the central tensor[20].

In the recursive process of pushing bulk operators to the boundary, the patterns of non-trivial Pauli operators become increasingly sparse, with identity operators interspersed throughout. This spreading implies that even small, disconnected regions of the boundary can collectively contain enough information to reconstruct certain bulk operators, which is closely related to the concept "uberholography," introduced by Pastawski and Preskill, suggesting the bulk information can be encoded in boundary regions with fractal structures. This process is aligned with the principle of entanglement wedge reconstruction.

Meanwhile, in the following sections in the paper [20], Gesteau and Kang define a nontrivial bulk Hamiltonian exhibiting a phase transition, using trapeze operators. However, due to this "uberholography," when this Hamiltonian is pushed to the boundary, it fails to reproduce expected CFT behavior such that the boundary state is too disentangled, missing the long-range correlations that define a CFT.

References

[1] G. Vidal, *Class of quantum many-body states that can be efficiently simulated*, *Physical Review Letters* **101** (2008) 110501 [quant-ph/0610099].

[2] B. Swingle, *Entanglement renormalization and holography*, *Phys. Rev. D* **86** (2012) 065007.

[3] F. Pastawski, B. Yoshida, D. Harlow and J. Preskill, *Holographic quantum error-correcting codes: toy models for the bulk/boundary correspondence*, *J. High Energ. Phys.* **2015** (2015) 149.

[4] D. Harlow, *TASI Lectures on the Emergence of the Bulk in AdS/CFT*, July, 2018. 10.48550/arXiv.1802.01040.

[5] A. Almheiri, X. Dong and D. Harlow, *Bulk locality and quantum error correction in AdS/CFT*, *J. High Energ. Phys.* **2015** (2015) 163.

[6] J.-W. Kim, *Explicit reconstruction of the entanglement wedge*, *Journal of High Energy Physics* **2017** (2017) 131.

[7] J. Fan, M. Steinberg, A. Jahn, C. Cao, A. Sarkar and S. Feld, *Lego hqec: A software tool for analyzing holographic quantum codes*, 2024.

[8] A. Jahn and J. Eisert, *Holographic tensor network models and quantum error correction: a topical review*, *Quantum Science and Technology* **6** (2021) 033002.

[9] G. Penington, A. Sipahigil and B. Whaley, *Lecture Notes for Physics C191: Introduction to Quantum Computing*, University of California, Berkeley (2025).

[10] M.A. Nielsen and I.L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition*, Cambridge University Press (2010).

[11] J. Preskill, *Lecture Notes for Physics 229: Quantum Information and Computation.*, California Institute of Technology (1998).

[12] D. Gottesman, *Stabilizer codes and quantum error correction*, 1997.

[13] M. Grassl, T. Beth and T. Pellizzari, *Codes for the quantum erasure channel*, *Physical Review A* **56** (1997) 33–38.

[14] D. Harlow, *The ryu–takayanagi formula from quantum error correction*, *Communications in Mathematical Physics* **354** (2017) 865–912.

[15] A.R. Brown, H. Gharibyan, G. Penington and L. Susskind, *The python's lunch: geometric obstructions to decoding hawking radiation*, 2019.

[16] N. Engelhardt, G. Penington and A. Shahbazi-Moghaddam, *Finding Pythons in Unexpected Places*, *Class. Quantum Grav.* **39** (2022) 094002.

[17] N. Engelhardt, G. Penington and A. Shahbazi-Moghaddam, *Twice upon a time: Timelike-separated quantum extremal surfaces*, 2023.

[18] B. Yoshida and A. Kitaev, *Efficient decoding for the hayden-preskill protocol*, 2017.

[19] C. Akers and G. Penington, *Quantum minimal surfaces from quantum error correction*, *SciPost Physics* **12** (2022) .

[20] E. Gesteau and M.J. Kang, *The infinite-dimensional HaPPY code: entanglement wedge reconstruction and dynamics*, May, 2020. 10.48550/arXiv.2005.05971.

[21] H. Apel, T. Kohler and T. Cubitt, *Holographic duality between local hamiltonians from random tensor networks*, *Journal of High Energy Physics* **2022** (2022) .

[22] X. Dong, D. Harlow and A.C. Wall, *Reconstruction of bulk operators within the entanglement wedge in gauge-gravity duality*, *Physical Review Letters* **117** (2016) .