

Single-Shot Universality in Quantum LDPC Codes via Code-Switching

Shi Jie Samuel Tan^{1,4,*†}, Yifan Hong^{1,2,*‡}, Ting-Chun Lin^{3,4,*§},
 Michael J. Gullans^{1,¶}, and Min-Hsiu Hsieh^{4,||}

¹Joint Center for Quantum Information and Computer Science, University of Maryland and NIST, College Park, MD, USA

²Joint Quantum Institute, University of Maryland and NIST, College Park, MD, USA

³University of California San Diego, La Jolla, CA, USA

⁴Hon Hai Research Institute, Taipei, Taiwan

October 10, 2025

Abstract

Code-switching is a powerful technique in quantum error correction that allows one to leverage the complementary strengths of different codes to achieve fault-tolerant universal quantum computation. However, existing code-switching protocols that encapsulate recent generalized lattice surgery approaches often either require many rounds of measurements to ensure fault-tolerance or suffer from low code rates. We present a single-shot, universal protocol that uses code-switching between high-rate quantum codes to perform fault-tolerant quantum computation. To our best knowledge, our work contains the first universal fault-tolerant quantum computation protocol that achieves what we term *single-shot universality* on high-rate codes that is characterized by (i) single-shot error correction, (ii) single-shot state preparation, as well as (iii) universal logical gates and logical measurements with constant depth circuits. We achieve this feat with single-shot code-switching between constant-rate 2D hypergraph product (HGP) codes and high-rate 3D HGP codes that can be viewed as a generalization of Bombín’s dimensional jump for color codes and Hillmann et al.’s single-shot lattice surgery for higher-dimensional topological codes. In addition, we prove the fault-tolerance of our code-switching protocol under both the adversarial and local-stochastic noise models. We introduce a vastly simpler recipe to construct high-rate 3D HGP codes with transversal CCZ gates that grants immense flexibility in the choice of expander graphs and local codes, allowing us to expand the search space for codes with good parameters and interesting logical gates. Our work opens an alternative path towards universal fault-tolerant quantum computation with low space-time overhead by circumventing the need for magic state distillation.

*These authors contributed equally to this work.

†stan97@umd.edu

‡yhong137@umd.edu

§til022@ucsd.edu

¶mgullans@umd.edu

||min-hsiu.hsieh@foxconn.com

Contents

1	Introduction	3
1.1	Related Works	5
1.2	Our Results	6
1.2.1	Dimensional Mechanics through Single-Shot code-switching	6
1.2.2	Single-Shot Universality	8
1.3	Discussion and Outlook	10
1.4	Outline	11
1.5	Acknowledgements	12
2	Preliminaries	12
2.1	Notation	12
2.2	Graphs and Expansion Parameters	12
2.3	Classical Linear Codes	14
2.4	Classical Tanner Codes on Spectral Expanders	14
2.5	Quantum Codes	15
2.6	Chain Complexes	15
2.7	Homological Product	16
2.8	Hypergraph Product Codes and Higher-Dimensional Variants	17
2.9	Higher-Dimensional Hypergraph Product Codes	18
2.10	Confinement and Soundness	20
2.11	Coboundary expansion	22
2.12	Coboundary Invariance and CCZ gates	22
3	Static Fault-Tolerant Single-Shot Universal Computation	24
4	Homomorphic CNOT between a 2D HGP code and a 3D HGP code	28
4.1	Homomorphic CNOT between \mathcal{Q} and \mathcal{Q}_G	29
5	Single-Shot Code Switching	33
5.1	Dimensional Expansion	33
5.2	Dimensional Contraction	35
6	Dynamical Fault-Tolerant Universal Computation	36
6.1	Single-Shot State Preparation	37
6.2	Single-Shot Error Correction	37
6.3	Single-Shot Logical Measurement	38
6.4	Logical Clifford Gates	38
6.5	Logical CCZ Gates	39
7	Linear Soundness in the 3D HGP Code	42
8	Fault Tolerance for Adversarial Noise	44
8.1	Fault Tolerance of Gadgets	44
8.2	Fault Tolerance of Single-Shot State Preparation	45
8.3	Fault Tolerance of Single-Shot Error Correction	47
8.4	Fault Tolerance of Single-Shot Code-Switching	48
8.5	Fault-Tolerance of Logical Gate Gadgets	48

8.6	Fault-Tolerance of the Single-Shot, Universal Protocol	48
9	Fault Tolerance for Local-Stochastic Noise	49
9.1	State preparation of the 3D HGP code	50
9.2	Fault tolerance of the entire protocol	54

1 Introduction

Quantum error correction—and, more broadly, fault-tolerant quantum computation—seeks to deliver reliable large-scale quantum algorithms by expending extra resources. For a long time, the surface code [Kit03] and other topological codes [BMD06] have been the leading candidates for fault-tolerant quantum computation due to their high thresholds for different noise models and compatibility with 2D qubit architectures [DKLP02, FMMC12, Lit19, TPMP24]. However, these codes suffer from high space-time overheads because they encode only a constant number of logical qubits per code block [BPT10]. Following the introduction of topological codes, there has been sustained effort to shrink these costs, culminating in the emergence of high-rate quantum low-density parity-check (QLDPC) codes [TZ14, BE21, PK22, LZ22, DHLV23, BCG⁺24]. In recent years, asymptotically good QLDPC codes that have constant rate and linear distance have been proven to exist [PK22, LZ22, DHLV23]. These codes forgo geometric locality to encode logical qubits more efficiently than topological codes. Quantum Tanner codes, one of these asymptotically good QLDPC codes, have also been shown to be amenable to single-shot error correction, making them extremely efficient both in terms of space and time overheads [GTC⁺24, DHLV23, LZ23, LZ25]. Gottesman showed that constant-rate QLDPC families can support fault-tolerant computation with asymptotically constant spatial overhead, though with linear time overhead relative to the unencoded circuit [Got13]. Since then, there has been a flurry of work to reduce this time overhead [FGL20, YK24, TKY24, NP25]. Most if not all of these schemes rely on magic state distillation (MSD) [BH12] to introduce non-Clifford gates and promote the native Clifford gates of the codes to a universal gate set to achieve universal fault-tolerant quantum computation. Because our ultimate goal is to achieve fast and efficient universal fault-tolerant quantum computation, it is natural to ask whether we can bypass expensive MSDs altogether by directly implementing a universal set of logical gates transversally. Nonetheless, the Eastin-Knill theorem [EK09] forbids the existence of a quantum code that can implement a universal set of logical gates transversally.

Code-switching which refers to the process of switching between different quantum error-correcting codes, has been proposed as an alternative method to sidestep the Eastin-Knill theorem [EK09] and expand the set of transversal logical gates that can be implemented fault-tolerantly. It was first proposed in the form of code deformation for topological codes where a given surface code is iteratively deformed to obtain CNOT gates [BMD09]. The idea was later generalized to topological subsystem color codes which can now implement Clifford logical gates via code deformation [Bom11]. Other works have investigated code-switching between small quantum stabilizer codes to toggle between memory and computation codes [HFWH11]. Hill, et al. looked at using a five-qubit code for quantum memory and switching to a seven-qubit Steane code for computation. This allows for some physical overhead savings as the five-qubit code is more space-efficient than the Steane code. Paetznick and Reichardt [PR13] proposed a scheme to achieve universal fault-tolerant quantum computation by using the triorthogonal codes introduced by Bravyi and Haah [BH12]. Their scheme uses transversal CCZ and H gates to achieve universality. However, the triorthogonal codes do not have good distance scaling and hence are not suitable for large-scale quantum computation. Beyond code-switching, concatenation has also been explored as a method

for achieving universal computation at the expense of the distance of the code [JOL14]. Yoder et al. uses concatenation and pieceable fault-tolerance to achieve universal fault-tolerant quantum computation. However, the scalability and efficiency of their scheme is unclear and their work primarily focused on small codes with small distances [YTC16].

The first scalable protocol proposed for code-switching with the aim of achieving universal fault-tolerant quantum computation was by Bombín [Bom15a, Bom16, KB15]. His proposal was to toggle between 2D and 3D gauge color codes via the method of gauge-fixing, exploiting the fact that a quantum code can access transversal gates from different levels of the Clifford hierarchy if they are defined in different spatial dimensions [BK13]. To be explicit, the 2D color code admits transversal Cliffords gates like the Hadamard gate and the CNOT gate while the 3D color code supports a transversal non-Clifford T gate. Bombín’s proposal uses *dimensional jumping* to promise constant-time logical gates using only local operations and leverages single-shot error correction in subsystem color codes. Anderson et al. [ADCP14] also stated a scheme involving switching between the Steane and Quantum Reed-Muller code shares a lot of similarities with Bombín’s proposal. Beyond the color code, Brown utilized the just-in-time gauge fixing technique introduced by Bombín [Bom18] to propose a code deformation scheme for the 2D surface code to implement the non-Clifford CCZ gate [Bro20]. This is interesting because the surface code is known to have better error correcting performance than the color code due to its lower stabilizer weight and better decoding algorithms [DKLP02, Del14, DN21]. Kubica and Vasmer were also able to adapt Bombín’s dimensional jump scheme to the case of the 3D subsystem toric code, which is often viewed as the canonical instance of 3D topological codes, to achieve single-shot error correction gadgets and logical gates [KV22]. Coupled with the single-shot state preparation ideas in Ref. [BGKT20], it shows that it is possible to achieve universal fault-tolerant quantum computation with single-shot error correction and state preparation using only local operations.

While these protocols are very promising and exciting due to the existence of a growing family of codes that work well in practice, they rely on topological codes that are known to have poor asymptotic spatial overhead scaling due to the fact that they only encode a constant number of logical qubits [BPT10]. Recently, Hong proposed a dimensional collapse protocol [Hon24] that is a generalization of a part of Bombín’s dimensional jump to the hypergraph product (HGP) codes that are known to have a constant rate [TZ14]. The protocol allows a *one-way* switch from a specific 3D HGP code to a 2D HGP code. This is a particularly exciting result because it opens up the possibility for code-switching between high-rate HGP codes. However, a recent result by Fu, et al. [FZLL25] shows that there are fundamental limitations to the set of transversal logical gates that can be implemented on HGP codes. To be specific, they show that 3D HGP codes cannot afford transversal single-qubit non-Clifford gates. Despite these limitations, several recent constructions of high-performance quantum codes with transversal CCZ gates have been proposed [ZSP⁺23, WHY24, Ngu24, GG24, GL25, Lin24, BDET24, Zhu25a, Zhu25b]. Some of these constructions are actually 3D HGP codes that are carefully constructed using classical Sipser-Spielman codes [SS96]. One can think of these as the high-rate generalizations of past results that showed how one can climb the Clifford hierarchy by increasing the spatial dimension of topological codes [KYP15, Mou16, VB19, Bro20].

While these code constructions are very promising for shaving space overhead, we also require every operation in the protocol to only require a constant-depth circuit implementation for fast and efficient fault-tolerant quantum computation. We introduce the term *single-shot universality* to refer to a universal fault-tolerant quantum computation protocol that satisfies the following properties: (i) single-shot error correction, (ii) single-shot state preparation, as well as (iii) universal logical gates and logical measurements with constant-depth circuit implementations. Single-shot universality is a desirable property for efficient quantum computation because it allows us to

perform every necessary operation for quantum computation in a constant-depth circuit, thereby reducing the time overhead and the accumulation of errors. This is especially of relevance for quantum hardware devices with fast qubit decoherence times as well as slow measurement times. For the case of the recently constructed asymptotically good qLDPC codes, Given the recent code construction developments discussed above and the need for single-shot universality for efficient quantum computation, it is natural to ask:

Can we generalize Bombín’s dimensional jump scheme for the color codes to a single-shot code-switching scheme for high-rate codes with transversal CCZ gates to achieve low spatial overhead universal fault tolerant quantum computation with single-shot universality?

1.1 Related Works

As discussed above, Hong proposed a dimensional collapse protocol [Hon24] that allows a *one-way* switch from a 3D HGP code to a 2D HGP code. This is particularly useful for single-shot state preparation of logical $|0\rangle$ and $|+\rangle$ states in the 2D HGP code since it harnesses the soundness property in the higher-dimensional HGP code. However, it requires certain 1D-like structure in the third dimension, and it is not immediate how to apply the protocol to our setting.

Lattice surgery, a method that can also be understood as a form of code-switching, has also been explored as a method to expand the set of logical gates that can be implemented fault-tolerantly. It was first proposed for the surface code to implement CNOT gates [HFDVM12]. Recently, generalized lattice surgery has been proposed for arbitrary high-rate QLDPC codes as a means to perform Pauli-based computation [BSS16]. For these high-rate codes with many logical qubits, it is often challenging to address individual logical qubits through targeted Pauli measurements to perform logical gates. These generalized lattice surgery protocols merge the code block together with an ancilla code block to form a larger code block, allowing one to perform joint logical measurements fault-tolerantly on multiple logical qubits before unmerging the code blocks [CB24, Cow24, CHRY24, SJOY24, IGND25, CHWY25, HCWY25]. While the state-of-the-art generalized lattice surgery protocols are extremely powerful and general, they ultimately still suffer from a time overhead that scales with the distance of the code due to the need for multiple rounds of measurements to ensure fault-tolerance. A recent work by Hillmann, et al. [HDTV24] showed that it is possible to perform single-shot lattice surgery that only requires a single round of measurement to reliably merge and split two higher-dimensional topological code blocks. While they were able to reduce the time overhead of the surgery procedure to a constant, their protocol still relies on topological codes that can only encode a constant number of logical qubits.

We note that Xu et al. addressed the problem of single-shot targeted joint logical Pauli measurement for constant rate HGP codes in Ref. [XZZ⁺25] where they constructed special homomorphic CNOTs and used a Steane measurement scheme to perform the joint logical measurements. Their scheme allows for efficient logical teleportation between HGP codes that differ by punctures and augmentations. However, their scheme does not address the problem of switching between HGP codes of different dimensions to access different transversal logical gates.

Heußen and Hilder recently proposed a code-switching protocol that leverages one-way transversal CNOT gates to enable efficient fault-tolerant code-switching between codes [HH25] via logical teleportation. Their protocol is interesting because it uses teleportation to switch between different codes, thereby avoiding the need for direct code deformation. However, the one-way CNOT gates they use are designed for color codes and it is unclear how to generalize their protocol to high-rate codes.

1.2 Our Results

In this work, we answer the question posed in the introduction in the affirmative by providing a framework for single-shot code-switching for high-rate HGP codes that are known to support single-shot error correction [FGL18, Cam19, QVRC21], efficient syndrome extraction circuits [MC25, TS24, BTHG25], fast Clifford gates [XZZ⁺25], and transversal CCZ gates [GL25, Lin24, Zhu25a, Zhu25b]. By carefully designing new gadgets and incorporating existing state-of-the-art gadgets, our proposed scheme achieves *single-shot universality*. In other words, our protocol possesses single-shot code-switching, single-shot error correction, single-shot state preparation, as well as universal logical gates and measurements that can be done in a constant-depth circuit (without d rounds of measurements) fault-tolerantly. To our best knowledge, this is the first universal FTQC protocol on high-rate codes that possesses this property. We summarize our main result in the informal theorem statement below:

Theorem 1.1 (Informal Statement of Theorem 6.1). *There exists a fault-tolerant quantum computation protocol that implements a universal gate set via single-shot code-switching between high-rate HGP codes using only gadgets that can be performed in a single-shot manner with a constant-depth circuit. The protocol can tolerate a number of adversarial errors that scales linearly with the distance of the HGP codes. In addition, the protocol is fault-tolerant under the local-stochastic noise model.*

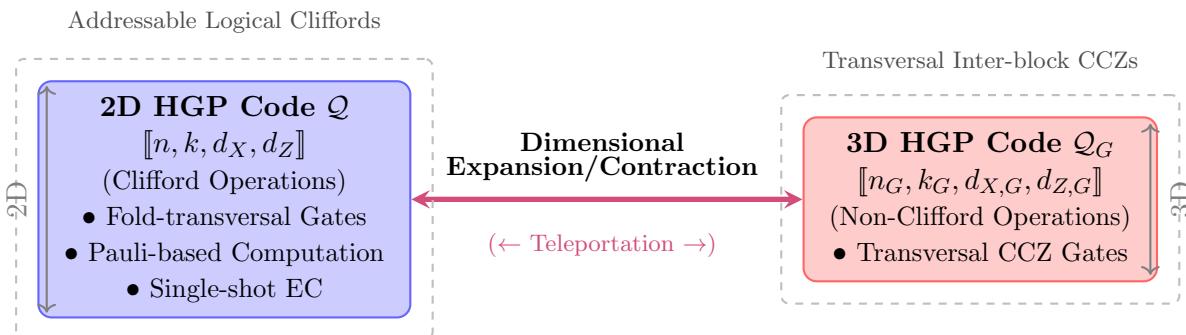
From a high level, our scheme takes advantage of the Clifford gates and single-shot error correction in constant-rate 2D HGP codes as well as the transversal CCZ gate and single-shot state preparation in certain 3D HGP codes to achieve single-shot universality. Compared to the usual efficient HGP constructions that utilizes lossless expanders, we construct HGP codes using classical Tanner codes. The slight difference in our construction means that we have to prove certain error correcting and single-shot properties for this different class of HGP codes. While it is possible to construct constant-rate 3D HGP codes, we make the design choice to trade the rate of the 3D HGP code in the new third dimension for greater flexibility in the first two dimensions, allowing us to use any constant-rate 2D HGP code for code-switching into high-rate 3D HGP codes with transversal CCZ gates. Even though these high-rate 3D HGP codes essentially encode the same number of logical qubits as the original constant-rate 2D HGP codes, this is sufficient for all practical purposes because we always have to code-switch back to the 2D HGP code for error correction and Clifford gates so there is typically no need to have the capability of encoding more logical qubits in the 3D case. This is discussed in greater detail in Section 1.2.2. The key ingredient that enables our scheme is a new single-shot code-switching protocol that allows us to fault-tolerantly switch between 2D and 3D HGP codes using only a constant-depth circuit. By switching back and forth between the two codes, we can implement the Hadamard gate in the 2D HGP code and the CCZ gate in the 3D HGP code to achieve universal quantum computation. We provide a schematic overview of our scheme in Figure 1. We now proceed to provide a technical overview of the results and techniques used in this work to prove Theorem 1.1.

1.2.1 Dimensional Mechanics through Single-Shot code-switching

Single-shot code-switching between HGP codes can be challenging because of circuit depth concerns. From a high level, a 2D HGP code is amenable to (fold)-transversal Hadamard gates because a constant-depth circuit is sufficient to swap the X and Z logical operators that are supported on 1D objects. On the other hand, a 3D HGP code is amenable to transversal inter-block CCZ gates because the 2D X logical operators always intersect at an odd number of points which grants the

Single-Shot code-switching Scheme

for Fault-Tolerant Universal Quantum Computation



Key Features:

- Constant spatial overhead (except during non-Clifford and state preparation)
- Single-shot Universality

Code Properties:

- Sipser-Spielman HGP
- Single-shot EC

Computational Universality:

- Clifford: 2D HGP code
- Non-Clifford: 3D HGP code

Figure 1: Overview of the single-shot universal protocol for fault-tolerant quantum computation using 2D and 3D HGP codes. The scheme leverages single-shot error correction, state preparation, and code-switching to achieve addressable Clifford and non-Clifford operations with low spatial overhead and time complexity.

necessary X stabilizer invariance that the CCZ logical action requires. When we only have 1D X logical operators, there is no guarantee on the parity of the intersection points when we shift the logical operators with stabilizers. Therefore, the goal for our single-shot code-switching protocol is to expand a 2D HGP code into a 3D HGP code in a way that grows the X logical operators from 1D to 2D objects while ensuring that the circuit is fault-tolerant and constant-depth. However, when the second dimension of the X logical operators scales with the distance of the code, one might expect that the circuit depth must also scale with the distance of the code. However, we show that it is possible to achieve this goal.

We introduce a new single-shot code-switching protocol that allows us to fault-tolerantly switch between a D -dimensional HGP code and a $(D+1)$ -dimensional HGP code built from the former and a generalized repetition code constructed using the Sipser-Spielman classical expander codes [SS96]. This can be interpreted as a single-shot lattice surgery protocol that is performed via logical teleportation instead of code deformation. The informal theorem statement below summarizes our second protocol:

Theorem 1.2 (Informal Statement of Theorem 5.1). *Let \mathcal{Q} be an D -dimensional HGP code constructed from classical expander codes for $D \geq 2$. Let \mathcal{Q}_G be an $(D+1)$ -dimensional HGP codes constructed from \mathcal{Q} and a classical generalized repetition code. There exists a SSCS protocol that fault-tolerantly switches between \mathcal{Q} and \mathcal{Q}_G using only a constant-depth logical teleportation circuit that can tolerate a number of adversarial errors that scales linearly with the distance of the codes. In addition, the protocol exhibits a threshold against local-stochastic noise.*

The high-level idea of our protocol is to utilize the homomorphic CNOT framework developed by Huang, et al. [HJOY23] to develop a *one-way* logical CNOT gate between HGP codes. This addresses the open question raised in Ref. [HH25] of how to generalize the one-way CNOT gate for color codes to high-rate codes for the purpose of code-switching via logical teleportation. We prove that there exists a sparse chain map between the chain complexes of \mathcal{Q} and \mathcal{Q}_G that allows us to implement the homomorphic CNOT between them with a depth-1 circuit. We utilize the grid Pauli product measurement (GPPM) framework developed by Xu, et al. [XZZ⁺25] to perform the necessary single-shot joint logical measurements along with the homomorphic CNOTs for the entanglement to complete the logical teleportation between \mathcal{Q} and \mathcal{Q}_G for the code-switching.

1.2.2 Single-Shot Universality

In this section, we discuss how every single gadget can be made single-shot or constant-depth to achieve single-shot universality. We discuss each gadget in a separate paragraph below.

Single-Shot State Preparation To prepare logical $|\bar{0}\rangle$ and $|\bar{+}\rangle$ states in the 2D HGP code \mathcal{Q} in a single shot, we use the protocol of Bergamaschi and Liu [BL24], which requires a spatial overhead of $\Theta(\sqrt{n})$, i.e. $\Theta(n^{3/2})$ additional ancilla qubits. Since 2D HGP codes generically lack any metacheck structure, they do not typically possess single-shot state preparation without additional overhead.

Single-Shot Error Correction. It is known that 2D HGP codes built from classical expander codes can perform single-shot error correction [FGL18, Cam19, QVRC21]. To be specific, 2D HGP codes are known to exhibit single-shot error correction under both the adversarial and local-stochastic noise models [FGL18, Cam19]. Less is known about single-shot error correction in higher-dimensional HGP codes. Campbell and Quintavalle, et al. [QVRC21] showed that 3D HGP

codes built from classical expander codes can perform single-shot error correction under the adversarial noise model in one Pauli basis. Our new result discussed above in the single-shot state preparation section implies that 3D HGP codes can perform single-shot error correction under the local-stochastic noise model in one Pauli basis. Even though we can only perform single-shot error correction in a single Pauli basis under both noise models, we can always delay the error correction in the other Pauli basis until we switch back to a 2D HGP code where we can perform single-shot error correction in both Pauli bases. This is sufficient for the purpose of our fault-tolerance analysis because all of our gadgets have a constant-depth implementation and hence we can bound the accumulation of errors. Most importantly, for most applications, our code only stays in 3D momentarily to perform the transversal CCZ gate before switching back to 2D. Therefore, we do not expect a significant accumulation of errors in the other Pauli basis during the short time spent in 3D.

Transversal Logical Clifford and Non-Clifford Gates There are many options for implementing logical Clifford gates in 2D HGP codes. We can use the fold-transversal framework developed by Breuckmann and Burton [BB24] to implement Clifford gates in a constant-depth circuit [QWV23]. Automorphisms are also a powerful tool for implementing logical Clifford gates in HGP codes [BGH⁺25]. Pauli-based computation [BSS16] is another powerful framework for implementing logical Clifford gates in HGP codes. We can use the GPPM framework developed by Xu, et al. [XZZ⁺25] to perform targeted logical Pauli measurements by entangling the HGP code block with a punctured or/and augmented HGP code block using homomorphic CNOTs. This allows us to perform addressable logical Clifford gates in a constant-depth circuit when coupled with fold-transversal gates and logical automorphisms.

Before we discuss how the transversal CCZ gates can be implemented on our HGP codes, we first address a possible question that readers might have. One might ask why do we need to code switch to \mathcal{Q}_G when we already have an adapter 3D HGP code block \mathcal{Q}_S that has the ability to perform dimensional jump when coupled with our single-shot lattice surgery protocol. The reason is that the 3D HGP code \mathcal{Q}_S may not have a transversal CCZ gate because the constraints for allowing CCZ gates are typically extremely restrictive and difficult to satisfy. Using results regarding sheaf codes from Ref. [Lin24] and the HGP constructions from Ref. [Zhu25a, Zhu25b], we prove that \mathcal{Q}_G is amenable to a transversal CCZ gate even when \mathcal{Q} is freely constructed from classical Sipser-Spielman codes built from arbitrary expanders and local codes. The only constraint we have to impose is that the three blocks of 3D HGP code must have matching expander graphs in the same physical dimension and the corresponding local codes in the same physical dimension must satisfy the *multiplication property*. While choosing an asymptotically good classical code for the third dimension can help with achieving constant rate for our 3D HGP codes, we intentionally choose a generalized repetition code for the third dimension to allow us to satisfy the multiplication property in a simple, systematic way even though it may not be the most space-efficient choice. Because we are using a repetition code as one of the local codes, this multiplication property is automatically satisfied when the other two local codes in the same physical dimension of the two other 3D HGP codes are chosen to be dual to each other. This vastly simplifies the construction of high-rate 3D HGP codes with transversal CCZ gates and allows us to search and consider a much larger family of classical codes as constituents for our HGP codes than before, thus expanding the design space for finite-size codes with good parameters, logical gates, and code automorphisms [BGH⁺25]. This is because we can effectively pick any two classical codes for the construction of the 2D HGP code \mathcal{Q} that is used to construct our first 3D HGP code block and then use it to constrain the construction of the other two 3D HGP code blocks. We summarize our result regarding transversal CCZ gates

in the informal theorem statement below:

Theorem 1.3 (Informal Statement of Theorem 6.4). *Let $\mathcal{Q}_1, \mathcal{Q}_2, \mathcal{Q}_3$ be three 3D HGP code constructed with a generalized repetition code in one of its dimensions and two classical codes in the other two dimensions. Let the following three conditions hold:*

- *The three 3D HGP codes have the same expander graphs in the same physical dimensions.*
- *The generalized repetition codes are all oriented in different physical dimensions.*
- *For each physical dimension, the two classical codes used in the other two 3D HGP codes are dual to each other in a local way.*

Then the transversal CCZ gate on the three code blocks implements a logical CCZ gate.

Transversal Measurement Transversal measurement of logical Pauli operators in HGP codes can be done in a single-shot manner without the need for multiple rounds of measurements. By using the GPPM framework developed by Xu, et al. [XZZ⁺25], we can perform targeted logical Pauli measurements by entangling the HGP code block with a punctured or/and augmented HGP code block using homomorphic CNOTs before performing transversal measurements on the ancilla code block using the Steane measurement framework. The measurement outcomes can then be processed classically to infer the desired logical measurement outcome fault-tolerantly. If we are interested in measuring all the logical qubits in some Pauli basis, we can simply perform transversal measurements on all the physical qubits in the code block without the need for any ancilla code blocks just like the case for any CSS code.

1.3 Discussion and Outlook

In this work, we provided a framework for single-shot code-switching between high-rate HGP codes that allows us to achieve single-shot universality for fault-tolerant quantum computation. Our protocol leverages existing and new gadgets that can be performed in a single-shot manner with constant-depth circuits to achieve single-shot universality. To our best knowledge, this is the first universal fault-tolerant quantum computation protocol for high-rate codes that possesses this property.

Remark 1.4. *We recently became aware of independent and concurrent work by Li, et al. [LPX25] that also considers universal fault-tolerant quantum computation that uses ideas related to single-shot code-switching. Our code-switching approach is different from theirs and we believe that both approaches are interesting and complementary to each other. In addition, we provide an explicit construction of high-rate 3D HGP codes with transversal CCZ gates that can be used for universal fault-tolerant quantum computation.*

Independently, Golowich, et al. [GCZ25] and Qian, et al. [XZB⁺25] also have upcoming works that consider code-switching between high-rate codes for addressable gates or/and high-rate logical operations. Their work is complementary to ours because they focus on code-switching between high-rate codes for logical Clifford gates while we focus on code-switching between 2D and 3D HGP codes to access transversal CCZ gates.

We believe that our work opens up many exciting future directions. We discuss some of the avenues below.

Open Problem 1. *Can we achieve constant spatial overhead for universal fault-tolerant quantum computation with single-shot universality for HGP codes under the local-stochastic noise model?*

Achieving constant rate in 3D is not typically required for code-switching since we ultimately have to discard the newly obtained logical qubits when we revert back to the 2D code. Nonetheless, having constant rate in 3D could still be useful for logical compilation purposes and the additional logical qubits can be useful as additional logical workspace in certain quantum algorithms. In addition, fault-tolerance under the adversarial noise is a strong assumption that is not typically satisfied in practice. Therefore, it would be interesting to see if we can go beyond individual fault-tolerance for the gadgets with respect to local-stochastic noise but also show that an arbitrary composition of these gadgets can also tolerate local-stochastic noise. A recent result by He et al. [HNP25] provides some insights into this direction.

Open Problem 2. *Can we generalize our single-shot code-switching framework to other families of quantum error-correcting codes beyond HGP codes?*

Another promising direction is to explore the application of our single-shot code-switching framework to other exotic families of quantum error-correcting codes. This could potentially lead to new insights and techniques for achieving fault tolerance in a wider range of quantum computing architectures. Bivariate bicycle codes and trivariate tricycle codes are promising candidates because of their extremely high encoding rates and translational symmetries [BCG⁺24, JMB25, MBAM⁺25]. Other exotic product codes may also give us better distance scalings and qubit overheads [PK21, BE21].

Open Problem 3. *Does our single-shot code-switching framework possess an advantage over the state-of-the-art magic state distillation (MSD) techniques?*

We note that recent advances in MSD and lattice surgery techniques have made them extremely efficient both in the practical and asymptotic regime [YSR⁺25, NP25]. Comparisons between code-switching and magic state distillation in Ref. [BKS21] for topological codes have showed that code-switching does not offer substantial savings over state distillation both in terms of space and space-time overhead. Therefore, most of the community has focused on improving MSD and lattice surgery techniques instead of code-switching. However, these comparisons were done for topological codes. It remains an open question whether code-switching can outperform MSD for high-performing QLDPC codes. Our work takes a step towards answering this question by providing a framework for code-switching for high-rate QLDPC codes. A more detailed comparison is left for future work.

1.4 Outline

The rest of the paper is organized as follows. In Section 2, we provide the necessary preliminaries regarding our notation as well as graphs, classical and quantum codes, chain complexes and homology, HGP codes, soundness and confinement, and the coboundary invariance property required for CCZ gates. In Section 3, we present a simple static scheme for universal fault-tolerant quantum computation using 3D HGP codes with transversal CCZ gates just to illustrate the power of having CCZ gates. In Section 4, we present a primitive that allows us to perform a one-way homomorphic CNOT between two different 3D HGP codes that would be useful for our single-shot code-switching protocol. In Section 5, we detail our single-shot code-switching protocol between 2D and 3D HGP codes using the homomorphic CNOT developed in Section 4 and demonstrate how it can be used to implement a universal set of logical gates with single-shot universality. In Section 7, we prove that 3D HGP codes can exhibit linear confinement which is a new result that is useful for our single-shot state preparation and error correction. In Section 8, we analyze the fault tolerance of our single-shot universal protocol as well as its constituent gadgets under the adversarial noise

model. In Section 9, we analyze the fault tolerance of the gadgets we utilize in our single-shot universal protocol under the local-stochastic noise model.

1.5 Acknowledgements

We thank Noah Berthelsen, Xiaozhen Fu, Daniel Gottesman, Yaoyun Shi, and Qian Xu for helpful discussions. We also thank Aleksander Kubica for helpful discussions on single-shot error correction. This material is based upon work supported in part by the Defense Advanced Research Projects Agency (DARPA) under Agreement HR00112490357, the NSF QLCI award OMA2120757, the NSF-funded NQVL:QSTD: Pilot: DLPQC and the DoE ASCR Quantum Testbed Pathfinder program (awards No. DE-SC0019040 and No. DE-SC0024220). This work was performed in part at the Kavli Institute for Theoretical Physics (KITP), which is supported by grant NSF PHY-2309135. SJST acknowledges funding and support from Joint Center for Quantum Information and Computer Science (QuICS) Lanczos Graduate Fellowship and the National University of Singapore (NUS) Development Grant.

AI Usage: the authors acknowledges the use of Claude in generating some of the figures in this paper and ChatGPT in proofreading for grammar and spelling.

2 Preliminaries

This section provides the necessary notation and preliminaries for the paper.

2.1 Notation

For a positive integer n , let $[n] = \{1, 2, \dots, n\}$. For a prime power q , let \mathbb{F}_q denote the finite field of order q . Given a vector $\mathbf{v} \in \mathbb{F}_q^n$, let $\|\mathbf{v}\| = |\{i \in [n] : \mathbf{v}_i \neq 0\}|$ denote the Hamming weight of \mathbf{v} , i.e., the number of nonzero coordinates in the standard basis. We use boldface uppercase letters (e.g., \mathbf{H}) to denote matrices and boldface lowercase letters (e.g., \mathbf{v}) to denote vectors. We denote the standard basis vector as $\mathbf{e}_i \in \mathbb{F}_q^n$, which has a 1 in the i -th position and 0s elsewhere. We also assume that all \mathbb{F}_q -vector spaces in our note are finite-dimensional which implies that all Hamming weights are finite. We provide a summary for the notation and conventions that we use in our paper in Table 1.

2.2 Graphs and Expansion Parameters

In this section, we introduce the necessary graph-theoretic concepts and expansion parameters used in our paper. We will be interested in regular graphs with uniform degree Δ . We begin by first reciting the definition of a spectral expander.

Definition 2.1 (Spectral expander). *Given a simple graph $\mathcal{G} = (V, E)$ with adjacency matrix $A(\mathcal{G}) \in \mathbb{F}_2^{|G| \times |G|}$. We say that \mathcal{G} is a λ -spectral expander when λ is the second-largest eigenvalue of $A(\mathcal{G})$.*

For a Δ -regular, λ -spectral expander graph $\mathcal{G} = (V, E)$, the Cheeger inequalities tell us that the edge expansion constant

$$h(\mathcal{G}) = \min_{S \subset V: |S| \leq |V|/2} \frac{|\partial S|}{|S|}, \quad (1)$$

Symbol	Meaning	Type / domain	Notes
q	prime power (field size)	$\mathbb{Z}_{>0}$	Field \mathbb{F}_q
\mathbb{F}_q^n	length- n vector space	vector space	Standard basis unless stated
\mathbf{e}_i	standard basis vector	vector	i -th entry is 1, others 0
$\ \mathbf{v}\ $	Hamming weight	\mathbb{N}	Number of nonzeros in \mathbf{v}
$\mathcal{C} \subseteq \mathbb{F}_q^n$	classical linear code	code	Dual \mathcal{C}^\perp , $\rho = k/n$
	$[n, k, d]_q$		
\mathbf{G}, \mathbf{H}	generator / parity-check of \mathcal{C}	matrices	$\mathbf{H}\mathbf{G}^\top = 0$
\mathcal{Q}	quantum CSS code	quantum code	$\mathbf{H}_X\mathbf{H}_Z^\top = 0$
	$\llbracket n, k, d \rrbracket$		
d_X, d_Z, d	X/Z distances; $d = \min(d_X, d_Z)$	\mathbb{N}	—
w	LDPC locality (max row/col weight)	\mathbb{N}	—
$\overline{X}_i, \overline{Z}_i$	logical Paulis	operators	i -indexed logical qubits
\mathcal{A}	chain complex	complex	$\cdots \rightarrow A_{i+1} \xrightarrow{\partial_{i+1}} A_i \xrightarrow{\partial_i} A_{i-1} \rightarrow \cdots$
∂_i, δ_i	boundary and coboundary maps	linear map	$\delta_i = \partial_{i+1}^\top$
$Z_i(\mathcal{A}), B_i(\mathcal{A}), H_i(\mathcal{A})$	cycles, boundaries, homology	subspaces	—
$d_i(\mathcal{A}), d^i(\mathcal{A})$	systolic and cosystolic distance	\mathbb{N}	Min. weight Z and X logical operators
$\text{HGP}(\mathbf{H}_1, \mathbf{H}_2)$	hypergraph product (HGP) code	construction	—
$\mathcal{G} = (L, R, E)$	biregular bipartite graph	graph	Degrees (ℓ, r) , neighbors $N(\cdot)$
(γ, α)	expansion parameters	reals	Vertex/edge expansion
$T(\mathcal{G}, \mathcal{C})$	Tanner code with local code \mathcal{C}	classical code	—
\mathbf{M}_Z	Z-metacheck matrix	matrix	Syndrome repair
$C^{(\ell)}Z$	multi-controlled-Z gates	gates	Level $\ell+1$ of Clifford hierarchy
$\mathcal{Q}', \tilde{\mathcal{Q}}$	thickened / expanded code	quantum codes	—

Table 1: Symbols and conventions used throughout the paper.

where $\partial S \subset E$, obeys $\frac{1}{2}(\Delta - \lambda) \leq h(\mathcal{G}) \leq \sqrt{2\Delta(\Delta - \lambda)}$ [HLW06]. Thus, if we desire large edge-expansion, then we want λ to be as small as possible. Explicit constructions of regular graphs with such optimal spectral expansion are known [Lub14].

The spectral expander graph is a useful tool for constructing Tanner codes with good distance properties.

2.3 Classical Linear Codes

This section states the basic definition for classical linear codes that anchor classical coding theory. We also briefly define the dual code of a classical linear code.

Definition 2.2 (Classical Linear Code). *For a finite field \mathbb{F}_q , a classical linear code of length n and dimension k over \mathbb{F}_q is a k -dimensional linear subspace $\mathcal{C} \subseteq \mathbb{F}_q^n$. The rate of \mathcal{C} is $\rho := k/n$. The distance d of \mathcal{C} is the minimum Hamming weight of a nonzero element of \mathcal{C} , that is $d = \min_{\mathbf{c} \in \mathcal{C} \setminus \{0\}} \|\mathbf{c}\|$. In general, we refer to \mathcal{C} as an $[n, k, d]_q$ code. The dual code $\mathcal{C}^\perp \subseteq \mathbb{F}_q^n$ of \mathcal{C} is defined by $\mathcal{C}^\perp = \{\mathbf{x} \in \mathbb{F}_q^n : \mathbf{x} \cdot \mathbf{y} = 0 \forall \mathbf{y} \in \mathcal{C}\}$, where $\mathbf{x} \cdot \mathbf{y} = \sum_{i \in [n]} \mathbf{x}_i \mathbf{y}_i$ denotes the standard bilinear form.*

2.4 Classical Tanner Codes on Spectral Expanders

In this section, we introduce the classical Tanner codes defined on spectral expanders that we introduced in Definition 2.1. These classical Tanner codes have good error-correcting capabilities that are inherited from the spectral expanders. These classical codes are then used as ingredients in our quantum code constructions.

Definition 2.3. *Given a (Δ_ℓ, Δ_r) -biregular bipartite graph $\mathcal{G} = (L, R, E)$ where $|L| = n$ and $|R| = m$, and a code $\mathcal{C}_0 \subseteq \mathbb{F}_2^{\Delta_r}$, we define the Tanner code of \mathcal{G} and \mathcal{C}_0 as*

$$T(\mathcal{G}, \mathcal{C}_0) = \{ \mathbf{c} \in \mathbb{F}_2^n \mid \forall u \in R, \mathbf{c}|_{N(u)} \in \mathcal{C}_0 \}$$

where $\mathbf{c}|_{N(u)} \in \mathbb{F}_2^{\Delta_r}$ denotes the subsequence of \mathbf{c} formed by the bits corresponding to the neighbors of $u \in L$ in the graph \mathcal{G} .

When \mathcal{C}_0 is the regular parity check code, then $T(\mathcal{G}, \mathcal{C}_0)$ is simply a classical code whose parity check matrix H is given by the graph adjacency matrix of the graph \mathcal{G} . When \mathcal{C}_0 is linear, the Tanner code $T(\mathcal{G}, \mathcal{C}_0)$ is also a classical linear code. This means that the Tanner code $T(\mathcal{G}, \mathcal{C}_0)$ can be described by a parity check matrix H that is constructed from the graph adjacency matrix of the graph \mathcal{G} and the parity check matrix of the code \mathcal{C}_0 . For the sake of concreteness, let us suppose that \mathcal{C}_0 is a $[n_0 = \Delta_r, k_0, d_0]$ code with m_0 checks. The parity check matrix $H_T(\mathcal{G}, \mathcal{C}_0)$ which we abbreviate as H_T of the Tanner code $T(\mathcal{G}, \mathcal{C}_0)$ can be constructed as follows:

1. Initialize an empty matrix with a number of rows equal to $m \cdot m_0$ and n columns (one for each node in L).
2. For each check node $w_i \in R$ for $i \in [m]$:
 - (a) Take the matrix $H_0 \in \mathbb{F}_2^{m_0 \times n_0}$ of the local code \mathcal{C}_0 which acts only on $v \in L$ such that $v \in N(w_i)$.
 - (b) Embed each row of H_0 into the global matrix H_T by placing the coefficients at the columns corresponding to $N(w_i)$ and zeros elsewhere.

For our work, we are interested in Tanner codes defined on Δ -regular spectral expanders and a linear local code of length Δ . In particular, we can position the bits of the Tanner code on the edges of a Δ -regular graph whereas the checks are the vertices of the graph. Each check imposes the local code's constraints on the Δ edges incident to it. This class of codes are referred to as Sipser-Spielman codes or classical expander codes. We shall refer to these classical codes as classical expander codes, Tanner codes, and Sipser-Spielman codes interchangeably in this paper. Building the codes with spectral expanders and short linear codes gives us classical expander codes with asymptotically good parameters and efficient, linear-time decoding algorithms [SS96].

2.5 Quantum Codes

This section states the basic definitions in quantum coding theory. In particular, we pay close attention to quantum CSS codes.

Definition 2.4 (Quantum CSS Codes). *For a finite field \mathbb{F}_q , a quantum CSS code of length n over \mathbb{F}_q is specified by a pair of classical codes $\mathcal{C}_X, \mathcal{C}_Z \subseteq \mathbb{F}_q^n$ such that $\mathcal{C}_X^\perp \subseteq \mathcal{C}_Z$. Let \mathcal{Q} denote the resulting CSS code. Its dimension is $k := \dim(\mathcal{C}_Z) - \dim(\mathcal{C}_X^\perp)$, and its rate is $\rho := k/n$. The distance of \mathcal{Q} is*

$$d := \min \left\{ \min_{\mathbf{c} \in \mathcal{C}_X \setminus \mathcal{C}_Z^\perp} \|\mathbf{c}\|, \min_{\mathbf{c} \in \mathcal{C}_Z \setminus \mathcal{C}_X^\perp} \|\mathbf{c}\| \right\}.$$

Sometimes, we differentiate between the X and Z distances of the code and define them as follows:

$$d_X := \min_{\mathbf{c} \in \mathcal{C}_X \setminus \mathcal{C}_Z^\perp} \|\mathbf{c}\|, \quad d_Z := \min_{\mathbf{c} \in \mathcal{C}_Z \setminus \mathcal{C}_X^\perp} \|\mathbf{c}\|.$$

The locality w of \mathcal{Q} is the maximum number of nonzero entries in any row or column of the parity-check matrices \mathbf{H}_X and \mathbf{H}_Z of \mathcal{C}_X and \mathcal{C}_Z .

For qLDPC codes, the locality w is a constant that is independent of the code length n . This implies that each stabilizer generator checks at most a constant number of qubits and each qubit is checked by at most a constant number of stabilizer generators.

2.6 Chain Complexes

This section states the basic definitions in homological algebra.

Definition 2.5 (Chain Complexes). *A chain complex \mathcal{A}_* over a field \mathbb{F}_q consists of a sequence of \mathbb{F}_q -vector spaces $(A_i)_{i \in \mathbb{Z}}$ and linear boundary maps $(\partial_i^{\mathcal{A}} : A_i \rightarrow A_{i-1})_{i \in \mathbb{Z}}$ satisfying $\partial_{i-1}^{\mathcal{A}} \circ \partial_i^{\mathcal{A}} = 0$ for all $i \in \mathbb{Z}$. When clear from context, we omit the superscript and subscript and write $\partial = \partial_i = \partial^{\mathcal{A}} = \partial_i^{\mathcal{A}}$. Assuming that each A_i has a fixed basis, then the locality $w^{\mathcal{A}}$ of \mathcal{A}_* is the maximum number of nonzero entries in any row or column of any matrix ∂_i in this fixed basis. If there exist bounds $\ell < m \in \mathbb{Z}$ such that for all $i < \ell$ and $i > m$ we have $A_i = 0$, then we may truncate the sequence and say that \mathcal{A}_* is the $(m - \ell + 1)$ -term chain complex*

$$\mathcal{A}_* = \left(A_m \xrightarrow{\partial_m} A_{m-1} \xrightarrow{\partial_{m-1}} \dots \xrightarrow{\partial_{\ell+1}} A_\ell \right).$$

We furthermore define the following (standard) vector spaces for $i \in \mathbb{Z}$:

$$\text{the space of } i\text{-cycles: } Z_i(\mathcal{A}) := \ker(\partial_i) \subseteq A_i, \tag{2}$$

$$\text{the space of } i\text{-boundaries: } B_i(\mathcal{A}) := \text{im}(\partial_{i+1}) \subseteq A_i, \tag{3}$$

$$\text{the space of } i\text{-homology: } H_i(\mathcal{A}) := Z_i(\mathcal{A}) / B_i(\mathcal{A}). \tag{4}$$

The cochain complex \mathcal{A}^* associated to \mathcal{A}_* has vector spaces $(A^i := A_i)_{i \in \mathbb{Z}}$ and boundary maps given by the coboundary maps $(\delta_i = \partial_{i+1}^\top : A^i \rightarrow A^{i+1})_{i \in \mathbb{Z}}$ obtained by transposing all the boundary maps of \mathcal{A}_* . Thus, the cochain complex is defined as such:

$$\mathcal{A}^* = \left(A^m \xleftarrow{\delta_{m-1}} A^{m-1} \xleftarrow{\delta_{m-2}} \dots \xleftarrow{\delta_\ell} A^\ell \right).$$

We can analogously define the spaces of cohomology $H^i(\mathcal{A}) = Z^i(\mathcal{A})/B^i(\mathcal{A})$, cocycles $Z^i(\mathcal{A}) = \ker(\delta_i)$, and coboundaries $B^i(\mathcal{A}) = \text{im}(\delta_{i-1})$. We typically refer to the chain complex instead of the cochain complex. We therefore often denote the chain complex by \mathcal{A} and omit the $*$ for notational convenience.

Definition 2.6. For a chain complex \mathcal{A} , the i -systolic distance $d_i(\mathcal{A})$ and the i -cosystolic distance $d^i(\mathcal{A})$ are defined as

$$d_i(\mathcal{A}) = \min_{\mathbf{c} \in Z_i(\mathcal{A}) \setminus B_i(\mathcal{A})} \|\mathbf{c}\|, \quad d^i(\mathcal{A}) = \min_{\mathbf{c} \in Z^i(\mathcal{A}) \setminus B^i(\mathcal{A})} \|\mathbf{c}\|.$$

It is well-known that classical linear codes can be described by 2-term chain complexes where the two vector spaces are the spaces of bits and checks respectively. These two vector spaces are connected by a linear boundary map that can be written as the check matrix \mathbf{H} . Quantum CSS codes can be described with a 3-term chain complex by associating the X stabilizers, qubits, and Z stabilizers with the three vector spaces in a 3-term chain complex. The condition “the boundary of a boundary is trivial” is compatible with the CSS orthogonality condition i.e., $\partial_{i-1} \circ \partial_i = 0$ and $\mathbf{H}_X \mathbf{H}_Z^\top = 0$. To see how the Pauli logical operators fit in the chain complex picture, we associate the X stabilizers, qubits, and Z stabilizers to the \mathbb{F}_2 -vector spaces A_0, A_1 , and A_2 , then the X and Z logical operator representatives are given by the basis elements of the 1-cohomology space and 1-homology space respectively. The X and Z distances of the quantum code are then $d^1(\mathcal{A})$ and $d_1(\mathcal{A})$.

2.7 Homological Product

This section states the basic notions of the homological product.

Definition 2.7 (Homological Product). For chain complexes \mathcal{A} and \mathcal{B} , the homological product $\mathcal{D} = \mathcal{A} \otimes \mathcal{B}$ is the chain complex given by the vector spaces

$$D_i := \bigoplus_{j \in \mathbb{Z}} A_j \otimes B_{i-j}$$

and the boundary maps

$$\partial_i^{\mathcal{D}} := \bigoplus_{j \in \mathbb{Z}} (\partial_j^{\mathcal{A}} \otimes I + (-1)^j I \otimes \partial_{i-j}^{\mathcal{B}}). \quad (5)$$

Because we typically work with \mathbb{F}_2 , the $(-1)^j$ signs can be ignored.

We now state a well-known result about the homological product.

Proposition 2.8 (Künneth Formula). Let \mathcal{A} and \mathcal{B} be chain complexes over a field \mathbb{F}_q , each with a finite number of nonzero terms. Then for every $i \in \mathbb{Z}$,

$$H_i(\mathcal{A} \otimes \mathcal{B}) \cong \bigoplus_{j \in \mathbb{Z}} H_j(\mathcal{A}) \otimes H_{i-j}(\mathcal{B}). \quad (6)$$

Furthermore, for $\mathbf{a} \in Z_j(\mathcal{A})$ and $\mathbf{b} \in Z_{i-j}(\mathcal{B})$, the isomorphism above maps

$$\mathbf{a} \otimes \mathbf{b} + B_i(\mathcal{A} \otimes \mathcal{B}) \mapsto (\mathbf{a} + B_j(\mathcal{A})) \otimes (\mathbf{b} + B_{i-j}(\mathcal{B})).$$

2.8 Hypergraph Product Codes and Higher-Dimensional Variants

The hypergraph product (HGP) code is one of the first constructions of quantum CSS codes that has constant rate and large distance [TZ14]. The HGP construction is essentially the same as taking the homological product of two 2-term chain complexes that correspond to two classical linear codes. Let parity-check matrices $\mathbf{H}_1 \in \mathbb{F}_2^{m_1 \times n_1}$ and $\mathbf{H}_2 \in \mathbb{F}_2^{m_2 \times n_2}$ correspond to two classical linear codes with parameters $[n_1, k_1, d_1]$ and $[n_2, k_2, d_2]$. By taking a homological product of their chain complexes, we obtain the product complex

The hypergraph product (HGP) construction is among the earliest families of CSS quantum codes achieving constant rate together with large distance [TZ14]. Conceptually, HGP is realized by taking the homological product of two 2-term chain complexes associated with a pair of classical linear codes. Let $\mathbf{H}_1 \in \mathbb{F}_2^{m_1 \times n_1}$ and $\mathbf{H}_2 \in \mathbb{F}_2^{m_2 \times n_2}$ correspond to two classical linear codes with parameters $[n_1, k_1, d_1]$ and $[n_2, k_2, d_2]$. Forming the homological product of their chain complexes yields the product complex

$$\begin{array}{ccccc}
& & S_1 \otimes S_2 & & \\
& \nearrow \mathbf{H}_1 \otimes \mathbb{1} & & \swarrow \mathbb{1} \otimes \mathbf{H}_2 & \\
B_1 \otimes S_2 & & & & S_1 \otimes B_2 \\
\downarrow \mathbb{1} \otimes \mathbf{H}_2 & & & \uparrow \mathbf{H}_1 \otimes \mathbb{1} & \\
& & B_1 \otimes B_2 & & \\
& & & & S_X \\
& & & \uparrow \mathbf{H}_X^\top & \\
& & & Q & \\
& & & \uparrow \mathbf{H}_Z & \\
& & S_Z & &
\end{array} \tag{7}$$

where S_X, Q, S_Z denote, respectively, the binary spaces of X -syndromes, qubit errors, and Z -syndromes. If we split the columns so that the left block corresponds to $B_1 \otimes S_2$ and the right block to $S_1 \otimes B_2$, then from (7) the CSS parity-check matrices of the HGP code are

$$\mathbf{H}_X = (\mathbb{1}_{n_1} \otimes \mathbf{H}_2^\top \mid \mathbf{H}_1^\top \otimes \mathbb{1}_{n_2}) \tag{8a}$$

$$\mathbf{H}_Z = (\mathbf{H}_1 \otimes \mathbb{1}_{m_2} \mid \mathbb{1}_{m_1} \otimes \mathbf{H}_2). \tag{8b}$$

Commutation of X and Z stabilizers is immediate, since $\mathbf{H}_X \mathbf{H}_Z^\top = 2\mathbf{H}_1^\top \otimes \mathbf{H}_2^\top = 0$. The quantum code HGP($\mathbf{H}_1, \mathbf{H}_2$) thus obtained has parameters

$$n = n_1 m_2 + m_1 n_2 \tag{9a}$$

$$k = k_1 k_2^\top + k_1^\top k_2 \tag{9b}$$

$$d_Z = \min(d_1^\top, d_2^\top) \tag{9c}$$

$$d_X = \min(d_1, d_2), \tag{9d}$$

with d_i^\top the distance of the dual code defined by \mathbf{H}_i for $i \in \{1, 2\}$. As usual, we have the quantum code distance $d = \min(d_X, d_Z)$.

Given $\mathbf{H}_1, \mathbf{H}_2$, choose generator matrices $\mathbf{G}_1, \mathbf{G}_2$ (so that $\mathbf{H}\mathbf{G}^\top = 0$) whose rows span the respective classical code spaces. In the HGP setting—and mirroring the data-qubit decomposition—the logical qubits split into $k_1 k_2^\top$ “left” and $k_1^\top k_2$ “right” logicals. A convenient canonical basis for the left logical \overline{X} and \overline{Z} operators is [QC22, QWV23]

$$\mathbf{G}_{Z,L} = (\{\mathbf{e}_i\} \otimes \mathbf{G}_2^\top \mid \mathbf{0}) \tag{10a}$$

$$\mathbf{G}_{X,L} = (\mathbf{G}_1 \otimes \{\mathbf{e}_j\} \mid \mathbf{0}), \tag{10b}$$

where $\mathbf{e}_i \notin \text{im}(\mathbf{H}_1^\top)$ and $\mathbf{e}_j \notin \text{im}(\mathbf{H}_2)$ are unit vectors. This guarantees that distinct logical operators are not stabilizer-equivalent. By placing \mathbf{G} in standard form, $\mathbf{G} = (\mathbb{1}_k \mid A)$, one can arrange that each pair of logical $\overline{X}/\overline{Z}$ either has disjoint support or intersects in exactly one qubit indexed by (i, j) in (10); this single-site intersection can be used to label the left logicals. This refinement is optional: for any row of (10b), there exists a linear combination of rows of (10a) that anticommutes with it and commutes with the rest. In this canonical choice, left logical \overline{Z} operators occupy single columns of the left block and left logical \overline{X} operators occupy single rows of the left block; the same statements hold for the right sector on the right block. Moreover, left logicals have minimum weight $\min(d_1^\top, d_2)$, while right logicals have minimum weight $\min(d_1, d_2^\top)$. If the right logicals are treated as gauge and ignored, the distance reduces to $d = \min(d_1^\top, d_2)$. A precise justification follows from examining the stabilizer structure of the left/right blocks.

2.9 Higher-Dimensional Hypergraph Product Codes

In this section, we discuss the construction of higher-dimensional hypergraph product codes as well as review some of their well-known facts. Let D denote the dimension of the HGP code we wish to construct. In particular, we consider the case where $D \geq 3$ and let $\{\mathbf{H}^{(a)} \in \mathbb{F}_2^{m_a \times n_a}\}_{a \in [D]}$ be parity-check matrices of D classical linear codes. For each $a \in [D]$, let $\mathcal{A}^{(a)}$ be the 2-term chain complex associated with $\mathbf{H}^{(a)}$ and vector spaces $A_0^{(a)} = B_a$ and $A_1^{(a)} = S_a$ correspond to the bit and check vector spaces of the classical code respectively. Then, consider the D -fold tensor product complex:

$$\mathcal{A} = \mathcal{A}^{(1)} \otimes \mathcal{A}^{(2)} \otimes \cdots \otimes \mathcal{A}^{(D)}$$

with degree- s cell spaces given by

$$A_s = \bigoplus_{\substack{J \subseteq [D] \\ |J|=s}} \left(\bigotimes_{i \in J} S_i \otimes \bigotimes_{j \in [D] \setminus J} B_j \right)$$

for $s \in \{0, 1, \dots, D\}$. This is the chain complex that corresponds to the D -dimensional hypergraph product code constructed from the D classical codes. To obtain a valid CSS code, we can choose any interior degree $s \in \{1, 2, \dots, D-1\}$ and associate the vector space A_s with the qubits, A_{s-1} with the X checks, and A_{s+1} with the Z checks. The resulting CSS parity-check matrices are given by

$$\mathbf{H}_X = \partial_s : A_S \rightarrow A_{s-1} \tag{11a}$$

$$\mathbf{H}_Z = \partial_{s+1} : A_{s+1} \rightarrow A_s. \tag{11b}$$

where the boundary operators can be derived from Equation (5). With a consistent choice of degree labeling, we see that $s = 1$ reproduces the usual $D = 2$ formulas above for the HGP code. We will adopt this convention throughout so that the 2D case is recovered by substitution and the 3D case is given by the $D = 3$ instance of the same construction.

The logical operators of the D -dimensional HGP code can be derived from the Künneth formula in Proposition 2.8. In particular, the X and Z logical operators correspond to the basis elements of the s -cohomology space $H^s(\mathcal{A})$ and s -homology space $H_s(\mathcal{A})$ respectively which can be derived

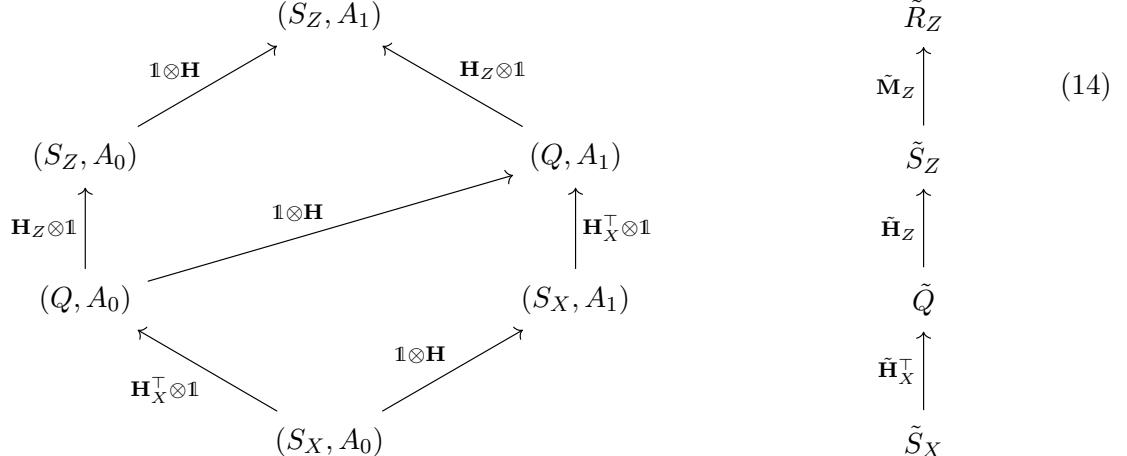
from Equation (6).

$$H_s(\mathcal{A}) \cong \bigoplus_{t_1+\dots+t_D=s} \bigotimes_{a \in [D]} H_{t_a}(\mathcal{A}^{(a)}) \quad (12)$$

$$H^s(\mathcal{A}) \cong \bigoplus_{t_1+\dots+t_D=s} \bigotimes_{a \in [D]} H^{t_a}(\mathcal{A}^{(a)}) \quad (13)$$

The X and Z distances of the code are then given by $d^s(\mathcal{A})$ and $d_s(\mathcal{A})$, i.e., the s -cosystolic and s -systolic distances of the chain complex \mathcal{A} respectively.

Because we mainly work with 2D and 3D HGP codes in our paper, we provide some concrete details regarding the construction of a 3D HGP code. For an input CSS code $[\![n_Q, k_Q, (d_X, d_Z)]\!]$ with parity-checks $\mathbf{H}_X, \mathbf{H}_Z \in \mathbb{F}_2^{m_Q \times n_Q}$ and a classical code $[n_c, k_c, d_c]$ with parity-check $\mathbf{H} \in \mathbb{F}_2^{m_c \times n_c}$, we can form the Kronecker (tensor) product of their underlying chain complexes and associate the result with a homological product code [FH13, BH14, ZP19, Cam19]. This construction generalizes the usual hypergraph product to higher “dimensions”/folds in the algebraic—not strictly topological—sense (though several higher-dimensional topological codes fit this framework). As an illustration, the 3D surface code can be viewed either as the homological product of the 2D surface code with a 1D repetition code, or equivalently as a 3-fold hypergraph product of classical repetition codes, i.e., a 3D HGP code. Mirroring the 3D surface code’s freedom over which Pauli type becomes a membrane, the homological product admits two choices. Without loss of generality, we take the version that extends the quantum code’s logical Z -type; the X case follows by reversing the product-complex orientation. The tensor-product complex is



where A_0 and A_1 are the bit and check vector spaces of the classical code respectively. The new CSS parity-check matrices are given by

$$\tilde{\mathbf{H}}_X = (\mathbf{H}_X \otimes \mathbf{1} \mid \mathbf{1} \otimes \mathbf{H}^\top) \quad (15a)$$

$$\tilde{\mathbf{H}}_Z = \left(\begin{array}{c|c} \mathbf{H}_Z \otimes \mathbf{1} & \mathbf{0} \\ \mathbf{1} \otimes \mathbf{H} & \mathbf{H}_X^\top \otimes \mathbf{1} \end{array} \right). \quad (15b)$$

Geometrically, the Tanner graph looks like the Euclidean graph product of the inputs, with nodes relabeled as in (14). When the input parity-checks are full rank, homology yields the parameters

[ZP19]:

$$\tilde{n} = n_Q n_c + m_Z m_c \quad (16a)$$

$$\tilde{k} \geq k_Q k_c \quad (16b)$$

$$\tilde{d}_X = d_X d_c \quad (16c)$$

$$\tilde{d}_Z = d_Z. \quad (16d)$$

In (16b), we used the Künneth formula (Proposition 2.8) to compute \tilde{k} . Let us write the (co)homologies of the quantum and classical codes as $H[\mathcal{Q}]$ and $H[\mathcal{C}]$ respectively. Adopting the convention that logical \overline{X} operators of the CSS code correspond to $H^1[\mathcal{Q}]$, while the classical code's codespace is $H^0[\mathcal{C}]$ (bits/checks placed on the 0/1-cells of its 2-term complex), Künneth gives

$$\tilde{H}^1 \cong H^1[\mathcal{Q}] \otimes H^0[\mathcal{C}].$$

Since the rank of \tilde{H}^1 counts the logical qubits of the product code, summing the products of ranks of the constituent groups yields (16b).

From (15) we also see, as in HGP, a left/right decomposition of qubits. With full-rank inputs, only the left sector carries logical qubits. For these, a canonical basis of logical \overline{X} and \overline{Z} operators is

$$\tilde{\mathbf{G}}_{Z,L} = (\mathbf{G}_Z \otimes \{\mathbf{e}_i\} \mid \mathbf{0}) \quad (17a)$$

$$\tilde{\mathbf{G}}_{X,L} = (\mathbf{G}_X \otimes \mathbf{G} \mid \mathbf{0}), \quad (17b)$$

with \mathbf{G}_Z and \mathbf{G}_X a symplectic logical basis for the input quantum code, $\{\mathbf{e}_i\} \notin \text{rs}(h)$, and \mathbf{G} is the generator matrix for the classical code \mathcal{C} . If the input CSS code is itself HGP, we may take \mathbf{G}_Z and \mathbf{G}_X from the canonical basis (10) and index logicals by triples (i, j, k) .

2.10 Confinement and Soundness

Confinement and soundness are two important properties of quantum codes that are used to analyze the performance of quantum error-correcting codes. In this section, we state the definitions of confinement and soundness for quantum codes.

Definition 2.9 (Confinement [QVRC21]). *Let $t > 0$ be an integer and $f : \mathbb{Z} \rightarrow \mathbb{R}$ be an increasing function. For a parity-check matrix $\mathbf{H} \in \mathbb{F}_2^{m \times n}$, we say it is (t, f) -confining if for any Pauli errors \mathbf{x} with reduced weight $\|\mathbf{x}\| \leq t$, its syndrome $\sigma(\mathbf{x}) = \mathbf{Hx}$ obeys*

$$f(\|\sigma(\mathbf{x})\|) \geq \|\mathbf{x}\|.$$

Confinement can be understood as a condition that lower bounds the weight of the error syndrome given the weight of the error. In other words, if the weight of the error is large, then the weight of the syndrome cannot be too small. Given Definition 2.9, we can also define what good linear confinement entails.

Definition 2.10 (Good Linear Confinement [QVRC21]). *Consider an infinite check family \mathcal{M}_n . We say the family has good linear confinement if each \mathcal{M}_n is (t, f) -confined where:*

1. *t grows with n such that $t \geq an^b$ for some positive constants a, b . That is $t \in \Omega(n^b)$ with $b > 0$;*

2. and $f(x)$ is some linear function that is monotonically increasing with x and independent of n .

In Ref. [QVRC21], the authors show that good linear confinement is sufficient to ensure that the quantum code is single-shot against both adversarial and stochastic noise. Prior to their work, Leverrier, Tillich and Zemor showed that hypergraph products of classical expander codes built from lossless expanders have good linear confinement [LTZ15].

Lemma 2.11 (Linear Confinement of Expander HGP Codes [LTZ15]). *There exist families of hypergraph product codes, called quantum expander codes, such that an arbitrary error \mathbf{x} with reduced weight $\|\mathbf{x}\| < t = \Theta(\sqrt{n})$ has a syndrome with weight bounded from below as $\|\sigma(\mathbf{x})\| \geq \|\mathbf{x}\|/3$.*

In other words, expander HGP codes have good linear confinement and are therefore single-shot against both adversarial and stochastic noise. Thus, it is possible to just perform a single round of (noisy) syndrome measurement for error correction. This is in contrast to the case of surface codes where we need to perform $O(d)$ rounds of syndrome measurements to perform error correction. In addition, the authors of Ref. [QVRC21] also prove the following for all 3D homological product codes. Letting X -confinement be the confinement with respect to Pauli Z errors, we have the following theorem.

Theorem 2.12 (Good X Confinement of 3D Homological Product Codes [QVRC21, Theorem 3]). *All 3D homological product codes have (t, f) X -confinement, where $t = d_Z$ and $f(x) = x^3/4$ or better.*

Now, we can define soundness for quantum codes.

Definition 2.13 (Soundness [Cam19]). *Let t be an integer and $f : \mathbb{Z} \rightarrow \mathbb{R}$ be some function called the soundness function with $f(0) = 0$. Given some set of Pauli checks M , we say it is (t, f) -sound if for all Pauli errors \mathbf{x} with $|\sigma(\mathbf{x})| = x < t$, it follows that there exists an \mathbf{e}' with $\sigma(\mathbf{e}') = \sigma(\mathbf{x})$ such that $|\mathbf{e}'| \leq f(x)$.*

Soundness can be understood as a condition that upper bounds the weight of the error given the weight of the syndrome. In other words, if the weight of the syndrome is small, then there exists an error that matches the syndrome and has small weight. We can also define what good soundness entails.

Definition 2.14 (Good Soundness [Cam19]). *Consider an infinite check family \mathcal{M}_n . We say the family has good soundness if each \mathcal{M}_n is (t, f) -sound where:*

1. t grows with n such that $t \geq an^b$ for some positive constants a, b . That is $t \in \Omega(n^b)$ with $b > 0$;
2. and $f(x)$ is some polynomial function that is monotonically increasing with x and independent of n .

While redundant checks are not strictly necessary for good soundness of the quantum code, it should not be too hard to see that increasing the number of redundant checks will typically improve the soundness of the quantum code as it raises t and makes the function f “smaller”. We can also use Definitions 2.9 and 2.13 to conclude that good soundness implies good confinement for qLDPC codes.

Lemma 2.15 (Good Soundness Implies Good Confinement [QVRC21, Lemma 2]). *Consider a qLDPC code that is (t, f) -sound with increasing f . If its qubit degree is at most ω , then it has $(t/\omega, f)$ -confinement.*

Next, we state the following lemma from Ref. [Cam19]:

Lemma 2.16 (Soundness of 4D Homological Product Codes [Cam19, Restatement of Lemma 6]).
Let

$$\mathcal{A} = \left(A_0 \xrightarrow{\delta_0} A_1 \xrightarrow{\delta_1} A_2 \right)$$

be a chain complex such that δ_1^\top is (t, f) -sound and δ_0 is (t, f) -sound with $f(x) = x^2/4$. Applying the homological product with two 1-term chain complexes of two classical codes, we obtain a new length-5 chain complex

$$\check{\mathcal{A}} = \left(\check{A}_{-1} \xrightarrow{\check{\delta}_{-1}} \check{A}_0 \xrightarrow{\check{\delta}_0} \check{A}_1 \xrightarrow{\check{\delta}_1} \check{A}_2 \xrightarrow{\check{\delta}_2} \check{A}_3 \right)$$

where the map $\check{\delta}_1$ is (t, g) -sound and $\check{\delta}_0^\top$ is (t, g) -sound with $g(x) = x^3/4$.

2.11 Coboundary expansion

In this section, we review the notions of (co)boundary expansion for (co)chain complexes and their relations to confinement and soundness in the previous subsection.

Definition 2.17 (Small-set coboundary expansion [HL22]). Let $A_0 \xrightarrow{\delta_0} A_1 \xrightarrow{\delta_1} A_2$ be a cochain complex. We say that δ_1 is (t, α) -small-set coboundary expanding if for all $a_1 \in A_1$ with $|a_1| \leq t$, we have

$$|\delta_1 a_1| \geq \alpha \cdot \min_{a_0 \in A_0} |a_1 + \delta_0 a_0|. \quad (18)$$

Intuitively, coboundary expansion asserts that if a_1 is far from any coboundary with respect to δ_0 , then its coboundary with respect δ_1 must be large. The small-set label just means that this property only holds below a certain radius t .

There is also a closely related notion of local (co)minimality and locally (co)minimal distance.

Definition 2.18 (Locally co-minimal [DHLV23]). Let $A_0 \xrightarrow{\delta_0} A_1 \xrightarrow{\delta_1} A_2$ be a cochain complex. We say that $a_1 \in A_1$ is locally co-minimal if $\forall a_0 \in A_0$ with $|a_0| = 1$, we have

$$|a_1| \leq |a_1 + \delta_0 a_0|. \quad (19)$$

Observe that if a_1 is not locally co-minimal, then we locally reduce its weight by applying a single coboundary.

Definition 2.19 (Locally co-minimal distance [DLV24]). Let $A_0 \xrightarrow{\delta_0} A_1 \xrightarrow{\delta_1} A_2$ be a cochain complex. The locally co-minimal distance (at level 1) is given by

$$d_{\text{coLM}} := \min\{|a_1| : a_1 \in \ker \delta_1, a_1 \text{ is locally co-minimal}\}. \quad (20)$$

2.12 Coboundary Invariance and CCZ gates

In this section, we discuss the coboundary invariance property of certain quantum codes and its implications for the implementation of CCZ gates. Broadly speaking, coboundary invariance guarantees that a set of physical CCZ gates map codewords to codewords since the logical action remains the same even as the code state differs by a coboundary i.e. X stabilizers. We mostly restate the definitions and ideas from Ref. [Lin24] and refer the reader to the original paper for more details.

Consider the case where we have three quantum codes $\mathcal{Q}_1, \mathcal{Q}_2, \mathcal{Q}_3$ and we want to implement a CCZ gate between the logical qubits of these codes. Suppose we have a trilinear map $f : \mathbb{F}_2^{n_1 \times n_2 \times n_3} \rightarrow \mathbb{F}_2$ where n_i is the length of the code \mathcal{Q}_i . Recall that a function is trilinear if the function is linear when we fix any of the other two inputs. The trilinear map f is constructed such that it captures the desired action of the following unitary operator CCZ^f :

$$CCZ^f |\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\rangle = (-1)^{f(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3)} |\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\rangle$$

where $\mathbf{x}_i \in \mathbb{F}_2^{n_i}$ is any computational basis state of the code \mathcal{Q}_i and $|\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\rangle$ is the state on $n_1 + n_2 + n_3$ qubits. Another important way to look at these states is to think of them as the qubits which we apply Pauli X operators on.

In order for CCZ^f to be a valid logical operator, it has to map codewords to codewords. Before we discuss the actual condition in greater detail, we note that any codeword in $\mathcal{Q}_1 \otimes \mathcal{Q}_2 \otimes \mathcal{Q}_3$ can be described as follows:

$$|[\zeta_1], [\zeta_2], [\zeta_3]\rangle := \sum_{\beta_1 \in B^1(\mathcal{A}_1)} \sum_{\beta_2 \in B^1(\mathcal{A}_2)} \sum_{\beta_3 \in B^1(\mathcal{A}_3)} |\zeta_1 + \beta_1, \zeta_2 + \beta_2, \zeta_3 + \beta_3\rangle$$

for $\zeta_i \in Z^1(\mathcal{A}_i)$ and $B^1(\mathcal{A}_i)$ being the space of X stabilizers of the code \mathcal{Q}_i for $i \in \{1, 2, 3\}$. Note that $[\zeta_i]$ is the logical equivalence class of the element ζ_i .

Because the action of CCZ^f is essentially a change of phase when applied to any computational basis state, we can guarantee that CCZ^f maps codewords to codewords if and only if

$$CCZ^f |[\zeta_1], [\zeta_2], [\zeta_3]\rangle \propto |[\zeta_1], [\zeta_2], [\zeta_3]\rangle.$$

The aforementioned condition is the same as the following:

$$f(\zeta_1, \zeta_2, \zeta_3) = f(\zeta_1 + \beta_1, \zeta_2 + \beta_2, \zeta_3 + \beta_3)$$

for all $\zeta_i \in Z^1(\mathcal{A}_i)$ and $\beta_i \in B^1(\mathcal{A}_i)$. This intuitively means that the action of CCZ^f is uniquely defined on the logical equivalence classes of the codewords.

We now define what it means for a triple of classical codes to satisfy what we term the *multiplication property*:

Definition 2.20 (Multiplication Property). *Let $\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3 \subseteq \mathbb{F}_2^\Delta$ be classical linear codes of length Δ . Let us denote the classical code $\mathcal{C}_1 \odot \mathcal{C}_2 \odot \mathcal{C}_3$ as the code generated by the following:*

$$\mathcal{C}_1 \odot \mathcal{C}_2 \odot \mathcal{C}_3 = \text{span} \{ \mathbf{c}_1 * \mathbf{c}_2 * \mathbf{c}_3 \mid \mathbf{c}_1 \in \mathcal{C}_1, \mathbf{c}_2 \in \mathcal{C}_2, \mathbf{c}_3 \in \mathcal{C}_3 \}$$

where $*$ denotes the component-wise multiplication of vectors.

We say that the triple $(\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3)$ satisfies the multiplication property if $\mathcal{C}_1 \odot \mathcal{C}_2 \odot \mathcal{C}_3 \subseteq \mathcal{C}' \subseteq \mathbb{F}_2^\Delta$ for the classical parity check code \mathcal{C}' , which is the dual of the repetition code. In other words, we have

$$\sum_{i=1}^{\Delta} (\mathbf{c}_1)_i (\mathbf{c}_2)_i (\mathbf{c}_3)_i = 0.$$

In the recent work done by Lin, an explicit connection was made between the multiplication property and a quantum code's amenability to transversal logical CCZ gates [Lin24]. While it was originally stated for sheaf codes, this connection can be made for higher-dimensional HGP codes as well since they are captured by the same sheaf formalism. Before we state the theorem from Ref. [Lin24], we first define the cup product operation on cochains:

Definition 2.21 (Cup Product on Cochains). *Let A^k be the space of k -cochains on a cell complex. The cup product is a bilinear map*

$$\smile : A^k \times A^\ell \rightarrow A^{k+\ell}$$

defined on the level of cochains by

$$(\mathbf{c}_1 \smile \mathbf{c}_2)(\tau) = \mathbf{c}_1(\sigma_1)\mathbf{c}_2(\sigma_2)$$

where σ_1 and σ_2 are the faces of τ such that $\sigma_1 \smile \sigma_2 = \tau$, and \mathbf{c}_1 and \mathbf{c}_2 are cochains in A^k and A^ℓ , respectively. The cup product is associative and satisfies the Leibniz rule with respect to the coboundary operator δ , i.e.,

$$\delta(\mathbf{c}_1 \smile \mathbf{c}_2) = (\delta\mathbf{c}_1) \smile \mathbf{c}_2 + (-1)^k \mathbf{c}_1 \smile (\delta\mathbf{c}_2),$$

where $\mathbf{c}_1 \in A^k$ and $\mathbf{c}_2 \in A^\ell$.

Now, we are ready to state the theorem from Ref. [Lin24] that connects the multiplication property to the implementation of transversal CCZ gates on 3D HGP codes.

Theorem 2.22 (Local Multiplication and CCZ Property [Lin24, Restatement of Theorem 6.8]). *Let $\mathcal{Q}_1(\{G_i\}_{i=1}^3, \{\mathcal{C}_{1,j}\}_{j=1}^3)$, $\mathcal{Q}_2(\{G_i\}_{i=1}^3, \{\mathcal{C}_{2,j}\}_{j=1}^3)$, and $\mathcal{Q}_3(\{G_i\}_{i=1}^3, \{\mathcal{C}_{3,j}\}_{j=1}^3)$ be three 3D HGP codes where we have G_i for $i = 1, 2, 3$ as the respective base spectral expander graphs and $\mathcal{C}_{1,j}, \mathcal{C}_{2,j}, \mathcal{C}_{3,j}$ are the linear local codes for each of the spectral expanders used to construct the Sipser-Spielman codes used in the HGP construction for each of the three 3D HGP codes. Let us define a trilinear function*

$$\begin{aligned} f : A^1 \times A^1 \times A^1 &\rightarrow \mathbb{F}_2 \\ f(\mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3) &\mapsto \sum_{\tau \in A_3} ((\mathbf{q}_1 \smile \mathbf{q}_2) \smile \mathbf{q}_3)(\tau) \end{aligned}$$

where the three A^1 's are the 1-cochains of the cochain complexes associated with the three 3D HGP codes and $\tau \in A_3$ denotes the basis elements of the 3-chains of the chain complex constructed from the tensor product between the three spectral expanders $\{G_i\}_{i=1}^3$. Define the unitary operator CCZ^f as the following:

$$CCZ^f = \prod_{j_1, j_2, j_3 \in [n]} CCZ_{j_1, j_2, j_3}^{f(\mathbf{e}_{j_1}, \mathbf{e}_{j_2}, \mathbf{e}_{j_3})}$$

where $CCZ_{j_1, j_2, j_3}^{f(\mathbf{e}_{j_1}, \mathbf{e}_{j_2}, \mathbf{e}_{j_3})}$ acts on the j_1, j_2, j_3 -th physical qubits of $\mathcal{Q}_1, \mathcal{Q}_2, \mathcal{Q}_3$ respectively if the trilinear map f evaluates nontrivially on the corresponding basis elements $\mathbf{e}_{j_1}, \mathbf{e}_{j_2}, \mathbf{e}_{j_3}$ of the 1-cochains. If $(\mathcal{C}_{1,j} \odot \mathcal{C}_{2,j} \odot \mathcal{C}_{3,j})$ satisfies the multiplication property for every $j = 1, 2, 3$, then the CCZ^f implements a logical CCZ gate across the logical qubits of the three 3D HGP codes.

3 Static Fault-Tolerant Single-Shot Universal Computation

In this section, we present a *static* fault-tolerant single-shot universal computation scheme based on some of the recent constructions of (near-)asymptotically good QLDPC codes with transversal CCZ gates [GL25, Lin24, BDET24, Zhu25a, Zhu25b]. We term it *static* because it does not involve dimension jumping or code switching. The codes constructed in the references listed above are all of the homological product variety. To be precise, all of them are 3D HGP codes that are obtained from the homological product of three classical Sipser-Spielman codes [SS96]. To describe how we might be able to achieve fault-tolerant universal computation, we state the two important facts. The first fact pertains to the computational universality of a small gate set [Shi02, Aha03, Chi17].

Fact 3.1 ([Shi02, Aha03, Chi17, Computationally Universal Gate Set]). *The gate set that contains the Hadamard gate H and the controlled-controlled-Z gate CCZ is computationally universal.*

Fact 3.1 implies that we can approximate any orthogonal matrix to arbitrary precision using the gate set $\{H, CCZ\}$ as long as we have access to ancilla qubits. However, because both H and CCZ have real matrix representations, they are clearly unable to approximate complex unitary matrices. Nonetheless, it is possible for us to simulate the effect of arbitrary complex unitary operations using H and CCZ gates by simulating the real and imaginary part of the operations separately.

Let us define the $C^{(\ell)}Z$ gate as the controlled-Z gate that is controlled on ℓ qubits and applies Z to the target qubit if and only if all the control qubits are in the state $|1\rangle$. It is known that the $C^{(\ell)}Z$ gate is in the $(l+1)$ -th level of the Clifford hierarchy. In other words, the CCZ gate is in the third level of the Clifford hierarchy and lies outside of the Clifford group that contains the CZ gate. We now state another well-known fact:

Fact 3.2 (Descending the Diagonal Clifford Hierarchy). *Let $C^{(\ell)}Z$ be a gate that is controlled on the first ℓ qubits and applies Z to the $(\ell+1)$ -th qubit if and only if all the control qubits are in the state $|1\rangle$. For any $i \in [\ell]$, we have*

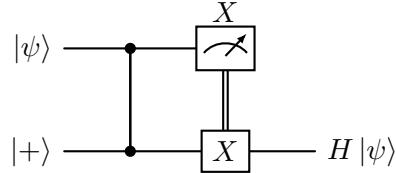
$$X_i C^{(\ell)}Z_{1,\dots,\ell+1} X_i C^{(\ell)}Z_{1,\dots,\ell+1} = C^{(\ell-1)}Z_{1,\dots,i-1,i+1,\dots,\ell+1}.$$

Note that the $C^{(\ell-1)}Z$ gate is controlled on the same ℓ qubits except for the i -th qubit.

In other words, assuming we have a transversal implementation of $C^{(\ell)}Z$, we can descend the Clifford hierarchy by conjugating the gate with X_i for any $i \in [\ell]$ to obtain a transversal implementation of $C^{(\ell-1)}Z$ as long as the target qubit remains the same. This fact was exploited in Ref. [HVWZ25] to build quantum codes with addressable CCZ gates. Using Fact 3.2, we obtain the following fact:

Fact 3.3 (Transversal CCZ to CZ). *All the CSS codes that have transversal CCZ gates constructed in Refs. [ZSP⁺23, WHY24, GL25, Lin24, Ngu24, BDET24, SPW24, Zhu25a, HVWZ25] possess a transversal implementation of the CZ gate.*

Now, we proceed to describe a quantum teleportation scheme that allows us to implement the Hadamard gate using the CZ gate, Pauli-X measurements, and an ancilla $|+\rangle$ state.



In the case where $|\psi\rangle$ and $|+\rangle$ are logical states of a 3D HGP code $\tilde{\mathcal{Q}}$ that encodes many logical qubits, performing the logical Hadamard on any subset of logical qubits might appear to be challenging especially when we are constrained by the addressability of the transversal logical CZ gate and the transversal logical X measurement gadget that might collapse all logical qubits in the code block. Fortunately, we can employ the Grid Pauli Product Measurement (GPPM) gadget introduced by Xu *et al.* [XZZ⁺25] to perform the necessary logical teleportation and measurements to ensure that the circuit shown above only acts on the logical qubits that we want to perform a logical Hadamard on.

By combining Fact 3.3 and the quantum teleportation scheme in the above figure, we obtain the ability to perform a computationally universal gate set that contains the Hadamard gate and

the CCZ gate. In addition, the transversal implementation of both components imply that we have a fault-tolerant implementation of these gates that take $O(1)$ logical cycles. The only caveat that remains is to show how we can fault-tolerantly prepare the ancilla logical $|\mp\rangle$ state for the gate teleportation scheme.

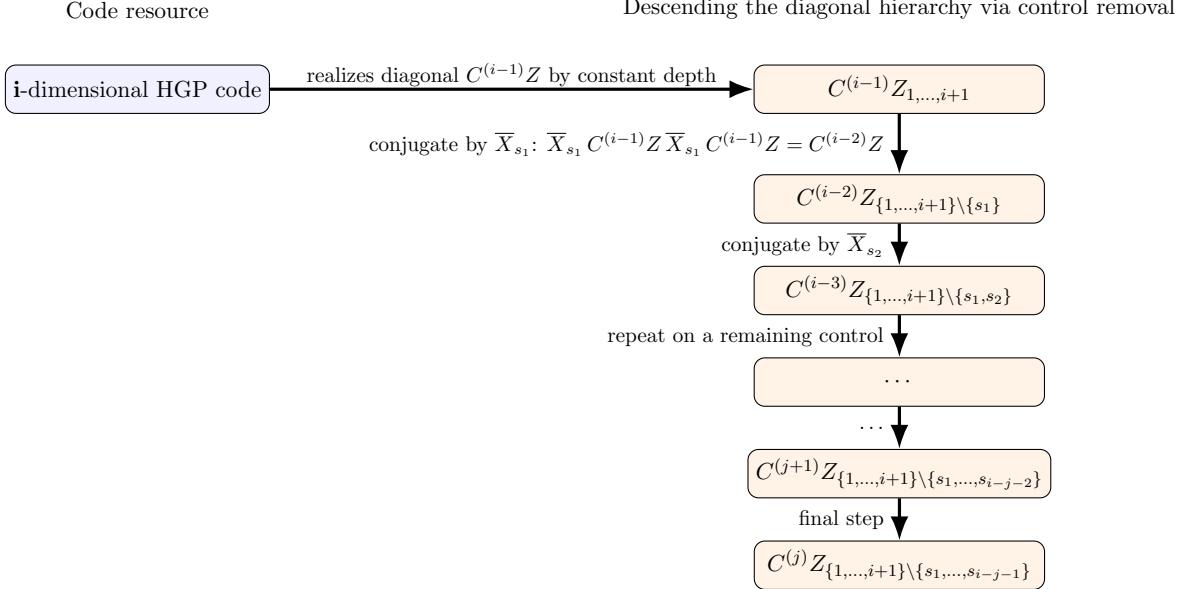
Because our 3-dimensional homological product is not a 5-term chain complex but a 4-term chain complex, only one of our chain maps can be (t, g) -sound. We now reuse and adapt the notation from Lemma 2.16 that discusses the soundness of a 4D HGP code from the chain complex picture. Let us assign $\tilde{A}_0, \tilde{A}_1, \tilde{A}_2$ to be the vector spaces that correspond to the X stabilizer generators, qubits, and Z stabilizer generators of the 3D HGP code respectively. In addition, let the code parameters of the 3D HGP code $\tilde{\mathcal{Q}}$ be $[\tilde{n}, \tilde{k}, \tilde{d}_X, \tilde{d}_Z]$. Suppose we let \tilde{A}_3 be a non-trivial vector space in our 4-term chain complex that is spanned by the Z metachecks that are associated to the Z stabilizer generators by the map $\tilde{\delta}_1$. The result from Lemma 2.16 still holds for the Z checks of our 3D HGP code. Now, we can use these Z metachecks to prepare the ancilla $|+\rangle$ state in our 3D HGP code $\tilde{\mathcal{Q}}$ in a single-shot fashion as described in the following protocol:

1. Initialize all physical qubits in the $|+\rangle^{\otimes \tilde{n}}$ state.
2. Measure all Z stabilizer generators to obtain a preliminary Z syndrome \vec{s}_0 .
3. Digitally compute the Z metacheck syndrome based on \vec{s}_0 to obtain a Z metasyndrome \vec{m} .
4. Use the metasyndrome \vec{m} to obtain a repaired syndrome \vec{s} .
5. Decode using the repaired syndrome \vec{s} and $\tilde{\delta}_1$ and apply the correction to obtain $|\mp\rangle^{\otimes \tilde{k}}$.

With the above gadgets, we are now able to perform computationally universal fault-tolerant quantum computation on these 3D HGP codes with transversal CCZ gates. As long as we have access to ancilla blocks that are crucial for gate teleportation for the Hadamard gates, we can achieve the desired computation. Suppose we have 3 blocks of such 3D HGP codes that are each equipped with an ancilla 3D HGP code, we can now perform (in parallel) logical CCZ on logical qubits across the 3 blocks as well as the Hadamard gate on the logical qubits within each of the 3 blocks using their individual ancilla code blocks. While our 3D HGP code does not directly allow for a single-shot state preparation of the $|\bar{0}\rangle$ state, we can use the Z metachecks to prepare the $|\mp\rangle$ state in a single-shot fashion and then apply a transversal H gate using the gate teleportation gadget to obtain the $|\bar{0}\rangle$ state. In our scheme, we do not focus too much on the routing problem of moving logical qubits between different code blocks for the implementation of logical gates. As mentioned before, the Pauli-measurement-based logical gates defined in Ref. [XZZ⁺25] may be integrated to the scheme to facilitate logical qubit teleportation between 3D HGP code blocks as well as standard Clifford computation using Pauli-based computation techniques. In addition, we might also be able to take advantage of the automorphisms inherited from the underlying classical codes of the 3D HGP code and utilize the automorphism gadgets for homological product codes to permute logical qubits fault-tolerantly [BGH⁺25]. Coupled with the inter-block CCZ gates and Hadamard gates, we can perform universal computation in a single-shot and fault-tolerant fashion. We defer a more in-depth discussion of the single-shotness to the later sections since they overlap heavily with the single-shotness of the dynamical scheme that we will introduce in Section 6.

While the static scheme we have constructed above uses 3D homological product codes, we point out that the scheme can be generalized to any i^{th} -dimensional homological product code. By using i blocks of i -dimensional homological product codes, we can descend the Clifford hierarchy down to any $C^{(j)}Z$ gate for $j \leq i$ and use the gadgets described above to perform Clifford gates fault-tolerantly. Thus, we can achieve fault-tolerant universal computation using i -dimensional

homological product codes as long as we have access to ancilla blocks that are crucial for gate teleportation for the Hadamard gates.



We note that an alternative approach is to consider six-dimensional hypergraph product (6D HGP) codes. The 3D HGP codes we have considered so far only have one chain map that is (t, g) -sound, which allows us to perform single-shot state preparation and measurement in only one basis (the X basis in our case). This is because the 4-term chain complex that corresponds to a 3D HGP code only has one non-trivial boundary map that can be made (t, g) -sound. To achieve single-shot state preparation in both bases, we need soundness in both X and Z bases without giving up non-Clifford transversality. A 6D HGP code yields a longer chain complex that can be arranged so that two distinct boundary maps are (t, g) -sound, providing independent metachecks for repairing both Z and X syndromes and hence single-shot state preparation in both bases. At the same time, by associating the appropriate vector spaces in the chain complex with qubits and checks such that the X logical operators are 4-dimensional membranes and the Z logical operators are 2-dimensional membranes, the HGP code now supports a transversal CCZ gate. At the same time, the code has soundness in both basis which allows for single-shot error correction and state preparation in both basis. One can think of this as the high-rate version of the 6D color code that Bombín explored in Ref. [BCHMD09]. Combined with Hadamard via teleportation, this preserves computational universality while achieving dual-basis soundness. However, it is known that the rate of the code worsens as we increase the dimensionality. There are practical reasons to stay in 3D instead of 6D. Suppose the classical codes used to construct the HGP codes have rate approximately 1/2. Then, a back of the envelope calculation shows that the 3D HGP code will have rate 1/8 and the 6D HGP code will have rate 1/64.

From the above discussion regarding the rate of higher-dimensional HGP codes, it should be apparent that the static scheme we have constructed above is not necessarily the most efficient way to achieve universal computation. While we are able to enjoy potential time savings from not having to switch between codes, a 3D homological product code suffers from a lower rate compared to a 2D code. In addition, for certain hardware architectures, it may be more efficient or even necessary to keep the code in 2D for the most parts unless we need to perform logical non-Clifford gates which we then switch to 3D codes for the duration of the non-Clifford computation. In some sense, this shares parallels with the motivations that underlie the work done by Hill *et al.* [HFWH11].

Thus, it may be overall more efficient in terms of space-time costs to use a dynamical scheme that switches between 2D and 3D codes to achieve universal computation. An exact characterization of the conditions under which such a scheme would be beneficial is an important open question for future work.

In the subsequent sections, we first describe a homomorphic CNOT primitive before showing how it can be utilized for a dynamical fault-tolerant single-shot universal computation scheme. This primitive is designed to help facilitate the switching between different code dimensions and ensure the overall efficiency of the computation.

4 Homomorphic CNOT between a 2D HGP code and a 3D HGP code

In this section, we describe the 2D and 3D HGP code constructions we utilize for the dimensional expansion and contraction steps. We then describe a chain map between these two HGP codes. Finally, we describe the homomorphic CNOT that we can perform between these codes.

Recall from Definition 2.3 that we denote the n -bit Sipser-Spielman code constructed from a Δ -regular spectral expander and a code \mathcal{C}_0 as $T(\mathcal{G}, \mathcal{C}_0)$. When \mathcal{C}_0 is the regular parity check code, then $T(\mathcal{G}, \mathcal{C}_0)$ is simply a classical code whose parity check matrix \mathbf{H} is given by the graph adjacency matrix of the graph \mathcal{G} . In our case, we are interested in the case where \mathcal{C}_0 is a length- Δ repetition code with parameters $[\Delta, 1, \Delta]$ that has a parity-check matrix \mathbf{H}_0 .

Note that we have chosen \mathbf{H}_0 so that every bit is involved in two checks by introducing a redundant check. The typical repetition code check matrix typically only has $\Delta - 1$ linearly independent checks but we introduce an extra check so that the graph representation now forms a loop. This is useful for the expander graph construction that we will use in the subsequent sections. In this form, each of the n classical bits of $T(\mathcal{G}, \mathcal{C}_0)$ is involved in an even number of checks.

We now state the following simple lemma that provides a useful property of the Sipser-Spielman code $T(\mathcal{G}, \mathcal{C}_0)$.

Lemma 4.1 (Codewords of $T(\mathcal{G}, \mathcal{C}_0)$). *Let $\mathcal{G} = (L, R, E)$ be a Δ -regular spectral expander and let $\mathcal{C}_0 \subseteq \mathbb{F}_2^\Delta$ be a $[\Delta, 1, \Delta]$ repetition code. If \mathcal{G} is a connected graph, then*

$$T(\mathcal{G}, \mathcal{C}_0) = \{0^n, 1^n\}$$

and $\dim(T(\mathcal{G}, \mathcal{C}_0)) = 1$.

Proof. We first note that the codewords of the repetition code \mathcal{C}_0 are simply the all-zero and all-one vectors in \mathbb{F}_2^Δ . Suppose the first bit $v_1 \in L$ in the codeword of $T(\mathcal{G}, \mathcal{C}_0)$ is 0. From the fact that \mathcal{G} is bipartite and connected, for all $v' \in R$ such that $v_1 \in N(v')$, all bits corresponding to the neighbors of v' in L must also be 0 because of the constraints from \mathbf{H}_0 . We can then extend this argument to all bits in L iteratively because G is connected. Since the only allowed vectors in \mathbb{F}_2^Δ are 0^Δ and 1^Δ , we conclude that

$$T(\mathcal{G}, \mathcal{C}_0) = \{0^n, 1^n\}.$$

□

Now, we discuss the choice of the connected bipartite graph \mathcal{G} that we will use to construct the Sipser-Spielman code $T(\mathcal{G}, \mathcal{C}_0)$. In particular, we can use the spectral expanders to construct Sipser-Spielman codes with good distance properties that are equipped with extremely efficient decoding algorithms. By choosing to plant \mathcal{C}_0 as the local codes in the spectral expander graph, we can

ensure that the resulting Sipser-Spielman code inherits the good properties of both the expander graph and the repetition code. We refer to this code $T(\mathcal{G}, \mathcal{C}_0)$ as the *generalized repetition code* \mathcal{C}_G due to the fact that it is a repetition code defined on an expanding bipartite graph with local codes \mathcal{C}_0 . We refer to the classical repetition code that has the standard parity check matrix \mathbf{H}_S with linearly independent checks as the *standard repetition code* and denote it by \mathcal{C}_S . We provide a pictorial representation of these different repetition codes in Figure 2.

Now that we are done with describing the generalized repetition code, we proceed to describe an important 3D HGP quantum code \mathcal{Q}_G . Consider an arbitrary 2D HGP code \mathcal{Q} constructed from two classical Sipser-Spielman expander codes. We can construct \mathcal{Q}_G by taking a tensor product of \mathcal{Q} with the generalized repetition code \mathcal{C}_G . More formally, we have the following definition.

Definition 4.2 (Quantum Code \mathcal{Q}_G). *Suppose we have a 2D HGP code \mathcal{Q} constructed from two classical Sipser-Spielman expander codes \mathcal{C}_1 and \mathcal{C}_2 that have associated 2-term chain complexes $\mathcal{A}_{\mathcal{C}_1}$ and $\mathcal{A}_{\mathcal{C}_2}$ respectively. Then, the 2D HGP code \mathcal{Q} has an associated 3-term chain complex $\mathcal{A}_{\mathcal{Q}}$ given by $\mathcal{A}_{\mathcal{Q}} = \mathcal{A}_{\mathcal{C}_1} \otimes \mathcal{A}_{\mathcal{C}_2}$. Define $\mathcal{A}_{\mathcal{Q}_G} = \mathcal{A}_{\mathcal{Q}} \otimes \mathcal{A}_{\mathcal{C}_G}$ i.e., the homological product between the chain complex of the 2D HGP code \mathcal{Q} and the chain complex \mathcal{C}_G . Then, let the 3D HGP code \mathcal{Q}_G be the code associated with the chain complex $\mathcal{A}_{\mathcal{Q}_G}$.*

4.1 Homomorphic CNOT between \mathcal{Q} and \mathcal{Q}_G

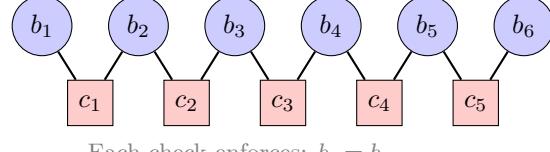
In this section, we describe the homomorphic CNOT that we can perform between two HGP quantum codes \mathcal{Q}_G and \mathcal{Q} which we will define later. To be concrete, the homomorphic CNOT is controlled on the logical qubits of the 3D HGP code \mathcal{Q}_G and targeted on the logical qubits of the 2D HGP code \mathcal{Q} . We note that our homomorphic CNOT construction is a *one-way CNOT* where the control and target codes are fixed. In other words, we cannot swap the control and target codes. This is a generalization of the homomorphic CNOT construction in Ref. [HH25] and addresses the open question of constructing such a homomorphic CNOT between codes with a growing number of logical qubits. Intuitively, reversing the control and target codes would require a chain map in the opposite direction which may not be sparse considering we would need to map a string-like X logical operator in \mathcal{Q} to a surface-like X logical operator in \mathcal{Q}_G . Our homomorphic CNOT is a crucial component of our dimension expansion and contraction steps.

To construct the homomorphic CNOT, we first define what we mean by a chain map between two k -term chain complexes for arbitrary integer k . A chain map is a linear map that commutes with the boundary operators of the two chain complexes.

Definition 4.3 (Chain Map between k -term Chain Complexes). *Suppose we are given two k -term \mathbb{F}_2 -chain complexes for $k \in \mathbb{Z}^+$*

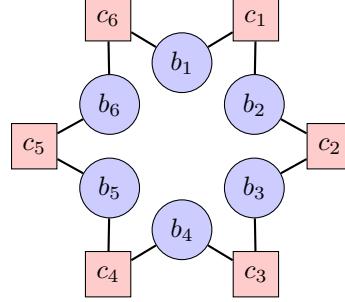
$$\begin{aligned}\mathcal{A} &= \left(A_k \xrightarrow{\partial_k^A} A_{k-1} \xrightarrow{\partial_{k-1}^A} \cdots \xrightarrow{\partial_1^A} A_0 \right), \text{ and} \\ \mathcal{B} &= \left(B_k \xrightarrow{\partial_k^B} B_{k-1} \xrightarrow{\partial_{k-1}^B} \cdots \xrightarrow{\partial_1^B} B_0 \right).\end{aligned}$$

A chain map $\varphi : \mathcal{A} \rightarrow \mathcal{B}$ is a collection of linear maps $\{\varphi_i : A_i \rightarrow B_i\}_{i=0}^k$ such that for all $1 \leq i \leq k$,



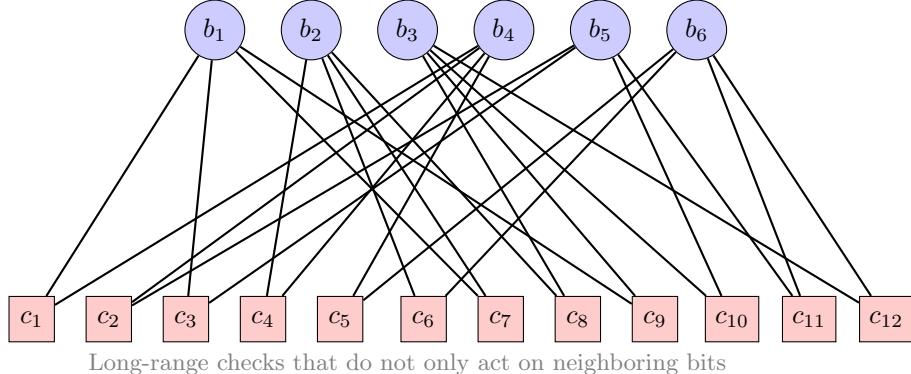
Each check enforces: $b_i = b_{i+1}$

(a) Standard repetition code \mathcal{C}_S with n bits and $n - 1$ checks for $n = 6$.



Ring topology with cyclic constraints

(b) Circular repetition local code \mathcal{C}_0 with n bits and n checks for $n = 6$.



Long-range checks that do not only act on neighboring bits

(c) Generalized repetition code \mathcal{C}_G with n bits and $> n$ checks for $n = 6$. \mathcal{C}_G is constructed from an $(n = 6, m = 4, 2, \Delta = 3)$ left-right expander $\mathcal{G} = (L, R, E)$ and a $[3, 1, 3]$ repetition code \mathcal{C}_0 as the local code. For example, c_1, c_2, c_3 are the parity checks decomposed from a node $u_1 \in V$ that was adjacent to nodes $b_1, b_4, b_6 \in E$. Similarly, c_4, c_5, c_6 are the parity checks decomposed from a node $u_2 \in V$ that was adjacent to nodes $b_2, b_4, b_6 \in E$.

Figure 2: Comparison of three repetition code structures: (a) standard repetition code \mathcal{C}_S , (b) circular repetition local code \mathcal{C}_0 , and (c) generalized repetition code \mathcal{C}_G .

the following diagram commutes:

$$\begin{array}{ccc} A_i & \xrightarrow{\partial_i^A} & A_{i-1} \\ \varphi_i \downarrow & & \downarrow \varphi_{i-1} \\ B_i & \xrightarrow{\partial_i^B} & B_{i-1} \end{array}$$

That is, $\varphi_{i-1} \circ \partial_i^A = \partial_i^B \circ \varphi_i$ for all $1 \leq i \leq k$.

Given k -term chain complexes, we are interested in the existence of a chain map between the two chain complexes where each of the linear maps φ_i is w -limited for some integer $w \in \mathbb{Z}^+$. A w -limited linear map is a sparse linear map where its matrix representation has at most w non-zero entries in each column and row. Next, we state the homomorphic CNOT framework defined in Refs. [HJOY23, XZZ⁺25].

Definition 4.4 (Homomorphic CNOT [XZZ⁺25, Restatement of Definition 3]). *Let \mathcal{Q} and \mathcal{Q}' be two quantum CSS codes associated with two 3-term chain complexes $\mathcal{A} = \{\{A_i\}_{i=0}^2, \{\partial_i\}_{i=1}^2\}$, and $\mathcal{A}' = \{\{A'_i\}_{i=0}^2, \{\partial'_i\}_{i=1}^2\}$, respectively. Let $\varphi = \{\varphi_i : A'_i \rightarrow A_i\}_{i=0}^2$ be a homomorphism between the two chain complexes, i.e., the following diagram is commutative:*

$$\begin{array}{ccccc} A_2 & \xrightarrow{\partial_2} & A_1 & \xrightarrow{\partial_1} & A_0 \\ \varphi_2 \uparrow & & \varphi_1 \uparrow & & \varphi_0 \uparrow \\ A'_2 & \xrightarrow{\partial'_2} & A'_1 & \xrightarrow{\partial'_1} & A'_0 \end{array} \tag{21}$$

Then physical \mathcal{Q} -controlled CNOTs specified by φ_1 , i.e. a physical CNOT controlled by the i -th qubit of \mathcal{Q} and targeted the j -th qubit of \mathcal{Q}' is applied if and only if $\varphi_1[i, j] = 1$, give some \mathcal{Q} -controlled logical CNOT gates between \mathcal{Q} and \mathcal{Q}' . We refer to such a logical gadget as a homomorphic CNOT associated with the homomorphism φ .

With Definition 4.2, we can now describe the homomorphic CNOT from the quantum code \mathcal{Q}_G to \mathcal{Q} by applying the framework stated in Definition 4.4 to the chain complexes $\mathcal{A}_{\mathcal{Q}_G}$ and $\mathcal{A}_{\mathcal{Q}}$ defined in Definition 4.2 with a 1-limited chain map φ that we define in the following proposition.

Proposition 4.5 (Homomorphic CNOT from \mathcal{Q}_G to \mathcal{Q}_S). *Let \mathcal{Q} and \mathcal{Q}_G be the quantum codes associated with the chain complexes $\mathcal{A}_{\mathcal{Q}}$ and $\mathcal{A}_{\mathcal{Q}_G}$ defined in Definition 4.2 respectively. Then, there exists a homomorphic CNOT from \mathcal{Q}_G to \mathcal{Q} specified by some 1-limited chain map φ . In particular, the homomorphic CNOT uses a transversal implementation of inter-block CNOTs and induces a logical CNOT from each logical qubit of \mathcal{Q}_G to the corresponding logical qubit of \mathcal{Q} .*

Proof. To prove the above proposition, we need to construct a chain map from the chain complex $\mathcal{A}_{\mathcal{Q}}$ to the chain complex $\mathcal{A}_{\mathcal{Q}_G}$ that is 1-limited. We can then use Definition 4.4 to conclude that the physical CNOTs specified by the chain map give a homomorphic CNOT from \mathcal{Q}_G to \mathcal{Q} .

We can decompose the 4-term chain complex $\mathcal{A}_{\mathcal{Q}_G}$ into the following form:

$$\begin{aligned} A_{\mathcal{Q}_G,0} &= A_{\mathcal{C}_1,0} \otimes A_{\mathcal{C}_2,0} \otimes G_0, \\ A_{\mathcal{Q}_G,1} &= \bigoplus_{i,j,k \in \{0,1\} : i+j+k=1} A_{\mathcal{C}_1,i} \otimes A_{\mathcal{C}_2,j} \otimes G_k, \\ A_{\mathcal{Q}_G,2} &= \bigoplus_{i,j,k \in \{0,1\} : i+j+k=2} A_{\mathcal{C}_1,i} \otimes A_{\mathcal{C}_2,j} \otimes G_k \\ A_{\mathcal{Q}_G,3} &= A_{\mathcal{C}_1,1} \otimes A_{\mathcal{C}_2,1} \otimes G_1, \end{aligned}$$

where G_k for $k \in \{0, 1\}$ are the vector spaces in the chain complex $\mathcal{A}_{\mathcal{C}_G}$ that is associated with the generalized repetition code $T(\mathcal{G}, \mathcal{C}_0)$.

Now, we can construct the chain map $\varphi : \mathcal{A}_{\mathcal{Q}} \rightarrow \mathcal{A}_{\mathcal{Q}_G}$. Let g_0 be an arbitrary standard basis vector in G_0 . Then, we can define the chain map as follows:

$$\begin{aligned} \varphi_0 : A_{\mathcal{Q},0} &\rightarrow A_{\mathcal{Q}_G,0}, \\ a_0 &\mapsto a_0 \otimes g_0, \\ \varphi_1 : A_{\mathcal{Q},1} &\rightarrow A_{\mathcal{Q}_G,1}, \\ a_1 &\mapsto a_1 \otimes g_0, \\ \varphi_2 : A_{\mathcal{Q},2} &\rightarrow A_{\mathcal{Q}_G,2}, \\ a_2 &\mapsto a_2 \otimes g_0, \end{aligned}$$

where a_i for $i \in \{0, 1, 2\}$ are arbitrary vectors in the vector spaces $A_{\mathcal{Q},i}$ that belong to the chain complex $\mathcal{A}_{\mathcal{Q}}$. It is clear from the construction of the chain map φ that each of the linear maps φ_i for $i \in \{0, 1, 2\}$ is 1-limited since each basis vector in $A_{\mathcal{Q},i}$ is mapped to a unique basis vector in $A_{\mathcal{Q}_G,i}$. This property directly implies that the physical CNOTs specified by $\varphi_{\mathcal{Q},1}^{\mathcal{O}(1)}$ act transversally: each physical qubit in \mathcal{Q} is paired with exactly one unique physical qubit in \mathcal{Q}_G for the CNOT operation, with no overlap or entanglement between different pairs. It is straightforward to verify that the above chain map φ satisfies the commutative diagram in Equation 21 in Definition 4.4.

Similarly, it is easy to check that the application of the physical CNOTs leads to logical CNOTs between the corresponding logical qubits in \mathcal{Q}_G and \mathcal{Q} . The Z logical operators are strings in \mathcal{Q} that are mapped to strings in \mathcal{Q}_G in the layer associated with the basis vector g_0 ; the X logical operators are membranes in \mathcal{Q}_G whose string-like boundaries that reside on the layer associated with the basis vector g_0 are mapped to strings in \mathcal{Q} under the chain map φ . Thus, we conclude that the homomorphic CNOT specified by φ is transversal at the physical level and induces logical CNOTs between individual pairs of logical qubits in \mathcal{Q}_G and \mathcal{Q} , as claimed. \square

Remark 4.6. We remark that our homomorphic CNOT gadget is compatible with the Grid Pauli Product Measurement framework formulated in Ref. [XZZ⁺25]. This stems from the well-known fact that chain maps compose to form another chain map. In particular, if we have two chain maps $\varphi : \mathcal{A} \rightarrow \mathcal{B}$ and $\psi : \mathcal{B} \rightarrow \mathcal{C}$ between three k -term chain complexes \mathcal{A}, \mathcal{B} , and \mathcal{C} , then the composition of the two chain maps $\psi \circ \varphi : \mathcal{A} \rightarrow \mathcal{C}$ is also a chain map. This property directly implies that if we have a homomorphic CNOT from a quantum code \mathcal{Q}_1 to another quantum code \mathcal{Q}_2 specified by a chain map φ , and another homomorphic CNOT from \mathcal{Q}_2 to a third quantum code \mathcal{Q}_3 specified by a chain map ψ , then we can construct a homomorphic CNOT from \mathcal{Q}_1 to \mathcal{Q}_3 specified by the composition of the two chain maps $\psi \circ \varphi$. In other words, we can puncture or/and augment the classical codes that constitute the HGP codes and only perform a logical CNOT on select logical

qubits between the 2D and 3D HGP codes. This flexibility can potentially be useful in practice for addressable code-switching applications.

Remark 4.7. We also remark that the homomorphic CNOT that we have developed can be generalized to the case where the third dimension of the code block \mathcal{Q}_G is constructed from a different classical code than the generalized repetition code \mathcal{C}_G . Instead of performing a single transversal homomorphic CNOT between \mathcal{Q}_G and \mathcal{Q} , we now transversal homomorphic CNOTs from \mathcal{Q}_G to multiple copies of \mathcal{Q} where the number of copies scales with the dimension of the classical code used to construct the third dimension of \mathcal{Q}_G . By expressing the parity-check matrix of the classical code in some canonical basis where the information bits are separated from the parity bits, the logical strings in \mathcal{Q}_G can be mapped to disjoint layers in \mathcal{Q}_G . Likewise the boundaries of the logical membranes in \mathcal{Q}_G can also be mapped to disjoint layers in \mathcal{Q}_G . We can then perform a transversal homomorphic CNOT from each of these layers to a different copy of \mathcal{Q} in parallel without blowing up the depth of the circuit. This generalization can be useful in practice if we want to use a classical code with a better rate than the generalized repetition code to construct the third dimension of \mathcal{Q}_G .

5 Single-Shot Code Switching

In this section, we describe the code-switching protocol that allows us to switch between different high-rate QLDPC codes in a single-shot manner. Our code-switching involves two kinds of operations: dimensional expansion and dimensional contraction that take place via logical teleportation. We now state our single-shot code-switching theorem.

Theorem 5.1 (Single-Shot Code-switching). *Given $D \geq 2$, suppose we are given a D -dimensional HGP code \mathcal{Q} with parameters $\llbracket n, \Theta(n), \Omega(n^{1/D}) \rrbracket$. Let \mathcal{Q}_G be an $\llbracket \Theta(n^{(D+1)/D}), \Theta(n), \Omega(n^{1/D}) \rrbracket$ $(D+1)$ -dimensional HGP code that has a chain complex $\mathcal{A}_{\mathcal{Q}_G}$ that is obtained from the tensor product between the chain complex $\mathcal{A}_{\mathcal{Q}}$ that corresponds to the HGP code \mathcal{Q} and a 2-term chain complex $\mathcal{A}_{\mathcal{C}_G}$ that corresponds to a classical Tanner code $\mathcal{C}_G = T(\mathcal{G}, \mathcal{C}_0)$ where \mathcal{G} is a Δ -regular spectral expander with $\Omega(n^{1/D})$ edges and \mathcal{C}_0 is a $[r, 1, r]$ classical repetition local code for some constants $\Delta > 0$. Then, there exists a single-shot code-switching protocol that fault-tolerantly switches between \mathcal{Q} and \mathcal{Q}_G using a constant-depth circuit that can tolerate $\Omega(n^{1/D})$ adversarial errors. In addition, the single-shot code-switching protocol exhibits a threshold against a constant physical error rate under the local stochastic noise model.*

Proof. We prove Theorem 5.1 by constructing an explicit single-shot code-switching protocol and provide an intuitive explanation for its fault-tolerance. We defer a more rigorous proof of fault-tolerance under both the adversarial and local stochastic noise models to Sections 8 and 9 respectively. We now state the dimensional expansion scheme and break it down into several constant-depth circuit components that utilizes known gadgets.

5.1 Dimensional Expansion

In this section, we describe the dimensional expansion protocol for the code \mathcal{Q} to the code \mathcal{Q}_G . We utilize the homomorphic CNOT primitive developed in Section 4.1 as well as a single-shot state preparation scheme that is discussed in greater detail in Section 6.1 to perform the dimensional expansion. Our dimensional expansion scheme is effectively a logical teleportation scheme that allows us to transfer the logical state of the code \mathcal{Q} into the code \mathcal{Q}_G . This addresses the open question raised in Ref. [HH25] of how to generalize the one-way CNOT gate for color codes to

high-rate codes for the purpose of code-switching via logical teleportation. The effective logical circuit for the dimensional expansion is shown in Figure 3.

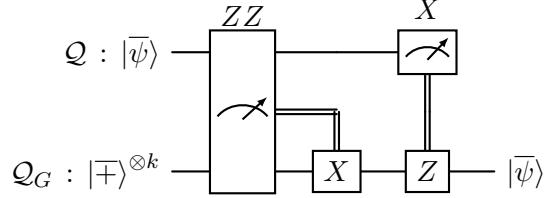


Figure 3: Logical quantum circuit for performing logical teleportation between two code blocks \mathcal{Q} and \mathcal{Q}_G using Pauli-based measurements and gates.

To perform the X measurement shown in Figure 3 in a single-shot and fault-tolerant manner, we can simply employ the Steane measurement technique and measure all of the physical qubits in \mathcal{Q} in the X basis. As for the ZZ measurement, we can use the Grid Pauli Product Measurement (GPPM) gadget developed in Ref. [XZZ⁺25] which gives us the following circuit shown in Figure 4.

This teleportation scheme is very similar to the standard EPR teleportation scheme. From the onset, there are a few technical challenges involved in this dimensional expansion gadget that we have devised here. Typically, we teleport logical qubits between codes of the same “shape” or “size” which allows us to use standard transversal entangling gadgets. However, when we are teleporting logical qubits from a 2D HGP code to a 3D HGP code, we cannot directly use the standard transversal CNOT gate because the two codes have different “shapes”. While it is possible to employ the homomorphic CNOT gadget, a naive implementation of the homomorphic CNOT gadget would likely require a physical CNOT circuit with CNOT depth that grows with the length of the third dimension of the 3D HGP code. The intuition behind this is that the logical X operators of the 3D HGP code are membrane-like while the logical X operators of the 2D HGP code are string-like. In order for the physical CNOT gadget to perform the intended logical CNOT action, we have to connect the physical qubits that lie in the support of these string-like operators in the 2D code to the physical qubits that lie in the support of these membrane-like operators in the 3D code.

To sidestep this technical issue, we utilize the transversal homomorphic CNOT gadget that we designed in Section 4.1. In other words, we first entangle the logical qubits in the 3D HGP code \mathcal{Q}_G with the logical qubits in some 2D HGP ancilla code block \mathcal{Q} that is initialized in the $|\bar{0}\rangle^{\otimes k}$ state using the transversal homomorphic CNOT gadget before entangling 2D ancilla code block with the original 2D HGP data code block \mathcal{Q} using the standard CSS transversal CNOT gate gadget. This avoids the need to perform a high-depth CNOT circuit, allowing us to preserve the transversal property and constant-time overhead for the procedure.

The only remaining thing to address is the state preparation of the ancilla code block \mathcal{Q} and the 3D HGP code block \mathcal{Q}_G . To prepare the 3D HGP code block \mathcal{Q}_G in the $|\pm\rangle^{\otimes k}$ state, we can simply perform the standard CSS state preparation protocol and then utilize the redundant Z checks to ensure that the state preparation is single-shot and fault-tolerant. This is essentially the same as the state preparation protocol discussed in Ref. [Hon24] except we do not collapse the third dimension of the code block. As for the state preparation of the 2D HGP ancilla code block \mathcal{Q} in the $|\bar{0}\rangle^{\otimes k}$ state, we can utilize the single-shot state preparation protocol described in Bergamaschi and Liu’s recent work [BL24].

The rest of the gadgets shown in Figure 4 are just the standard transversal logical CNOT gadget and Pauli logical operations that can be implemented in the standard manner. The X and

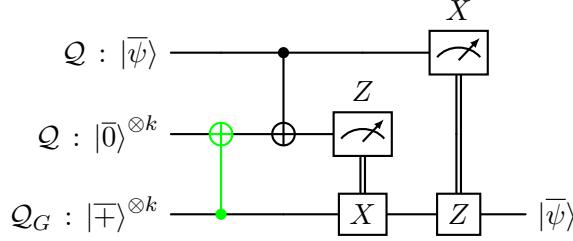


Figure 4: Logical quantum circuit for performing logical teleportation from a code block \mathcal{Q} to another code block \mathcal{Q}_G using an adapter ancilla code block \mathcal{Q} . The sequence of gates including the two CNOTs as well as the Z measurement is effectively the ZZ measurement. The first CNOT (colored in green) is the homomorphic CNOT described in Section 4.1. The last CNOT is the standard transversal logical CNOT between two 2D HGP codes.

Z measurements can be done using the Steane measurement technique by measuring all physical qubits in the X and Z basis respectively. The classical post-processing of the measurement outcomes can be done using standard classical computation.

5.2 Dimensional Contraction

In this section, we discuss the process of collapsing the 3D HGP code \mathcal{Q}_G back to a 2D HGP code \mathcal{Q} . Because our dimensional expansion scheme is essentially a logical teleportation scheme, we can leverage the exact same techniques used in the expansion process to achieve contraction. For the sake of being concrete, we provide a quantum circuit for the dimensional contraction scheme in the following figure shown in Figure 5.

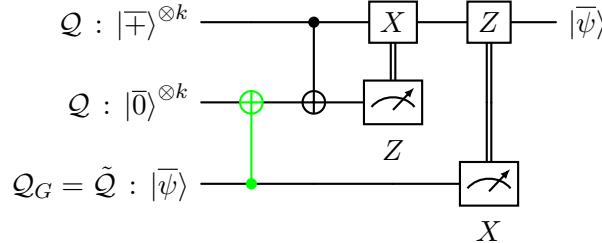


Figure 5: Logical quantum circuit for performing logical teleportation from a code block \mathcal{Q}_G to another code block \mathcal{Q} using an adapter ancilla code block \mathcal{Q} . The sequence of gates including the two CNOTs as well as the Z measurement is effectively the ZZ measurement. The first CNOT (colored in green) is the homomorphic CNOT described in Section 4.1. The last CNOT is the standard transversal logical CNOT between two 2D HGP codes.

For the fault-tolerance of the code-switching protocol, there are a few different possible error locations. The state preparation step is fault-tolerant by the analysis done in Refs. [Hon24] and [BL24]. The transversal CNOT gates including the homomorphic CNOT gate are clearly fault-tolerant. The transversal X and Z measurements that may be done using the Steane measurement technique are also fault-tolerant. Thus, the entire code-switching protocol is fault-tolerant. We defer a more rigorous proof of fault-tolerance under both the adversarial and local stochastic noise models to Sections 8 and 9 respectively. \square

Remark 5.2. *The single-shot code-switching protocol described here can be understood as a generalization of the single-shot lattice surgery protocol to high-rate codes. Suppose we are interested in performing single-shot lattice surgery to merge a 2D HGP code \mathcal{Q} with another 3D HGP code \mathcal{Q}_G . That can be achieved via teleporting the logical qubits in \mathcal{Q} to some merged 3D HGP code $\tilde{\mathcal{Q}}_G$ that is obtained from merging \mathcal{Q}_G with another copy of \mathcal{Q} . The logical teleportation can be done using the exact same circuit as shown in Figure 4. The splitting step can also be understood as a logical teleportation from the merged 3D HGP code $\tilde{\mathcal{Q}}_G$ to the original 2D HGP code \mathcal{Q} . This single-shot lattice surgery can be done between higher-dimensional HGP codes as long as the two codes have soundness properties in the appropriate dimensions and basis.*

6 Dynamical Fault-Tolerant Universal Computation

In this section, we present a dynamical fault-tolerant universal computation scheme based on recent HGP constructions of QLDPC codes with transversal CCZ gates [GL25, Lin24, BDET24, Zhu25a, Zhu25b]. Our dynamical scheme involves a code-switching protocol that involves dimensional jumps between two different high-rate QLDPC codes to sidestep the Eastin-Knill theorem [EK09] and achieve fault-tolerant universal computation. We describe our scheme in detail for the case of hypergraph product codes, but it can very likely be generalized to other product codes that include lifted product codes and balanced product codes. We break down our scheme into two components: the first component is a dimensional expansion of an D -dimensional HGP code to a $(D + 1)$ -dimensional HGP code, and the second component is a dimensional contraction of an $(D + 1)$ -dimensional HGP code to a D -dimensional HGP code. While our scheme holds for any integer $D \geq 2$, we will focus on the case of $D = 2$ for simplicity as well as to highlight the change between the Clifford and non-Clifford logical gates in the two codes.

Before we describe the explicit procedures for the dimensional expansion and contraction, we first describe the general framework of our scheme. For the purpose of all Clifford gates and error correction, our code will be a 2D HGP code \mathcal{Q} with parameters $\llbracket n, k, d_X, d_Z \rrbracket$ constructed using classical codes with the Sipser-Spielman construction. Our construction will be very similar to the one stated by Golowich and Lin in Ref. [GL25] and Zhu in Refs. [Zhu25a, Zhu25b]. By Lemma 2.11, \mathcal{Q} is amenable to single-shot error correction. With a combination of the fold-transversal logical gates [BB24, QWV23] and the parallel Pauli-measurement-based logical gates from Ref. [XZZ⁺25], we can perform addressable Clifford gates in a fault-tolerant manner. When we want to perform logical non-Clifford gates, we perform a dimensional expansion of the code \mathcal{Q} by teleporting the logical qubits of \mathcal{Q} into a 3D HGP code \mathcal{Q}_G with parameters $\llbracket n_G, k_G, d_{X,G} \cdot n_C, d_{Z,G} \rrbracket$ for integer $n_C = \min(d_X, d_Z)$. The code \mathcal{Q}_G has the same 2D form as \mathcal{Q} but its third dimension is given by the classical code \mathcal{C}_G i.e. the generalized repetition code. Note that \mathcal{Q}_G is effectively an instance of the 3D HGP code described in Section 4.1. We show that a careful choice for the classical codes that are used to construct \mathcal{Q}_G can make it amenable to inter-block transversal logical CCZ gates. We note that the dimensional expansion step is a single-shot procedure that does not require any additional rounds of syndrome measurements. After we finish performing the logical non-Clifford gates, we perform a dimensional contraction of the code \mathcal{Q}_G back to \mathcal{Q} to keep the “non-constant” spatial overhead limited to the time steps where we perform the logical non-Clifford gates. The dimensional contraction step is also a single-shot procedure that does not require any additional rounds of syndrome measurements. Lastly, when we want to perform state preparation, we either use the single-shot state preparation scheme developed by Bergamaschi and Liu in Ref. [BL24] to prepare logical states directly in the 2D HGP code \mathcal{Q} , or the scheme developed by Hong in Ref. [Hon24] to prepare logical states in the 3D HGP code \mathcal{Q}_G depending on the dimensionality

of the code block in which we hope to prepare the logical state. With this framework, we can perform a fault-tolerant universal computation scheme with single-shot error correction and state preparation with constant spatial overhead except for the time steps where we perform the logical non-Clifford gates and the state preparation. Most importantly, every gadget in our scheme is single-shot and takes constant time to implement. We present a schematic overview of our scheme in Figure 1. We now state our main theorem for this section.

Theorem 6.1 (Single-Shot, Universal Protocol via Code-switching). *Let \mathcal{Q} be an $\llbracket n, \Theta(n), \Omega(\sqrt{n}) \rrbracket$ 2D HGP code constructed from classical Sipser-Spielman codes. Let \mathcal{Q}_G be a 3D HGP code with parameters $\llbracket \Theta(n^{3/2}), \Theta(n), \Omega(\sqrt{n}) \rrbracket$ that has a chain complex $\mathcal{A}_{\mathcal{Q}_G}$ that is obtained from the tensor product between the chain complex $\mathcal{A}_{\mathcal{Q}}$ that corresponds to the HGP code \mathcal{Q} and a 2-term chain complex $\mathcal{A}_{\mathcal{C}_G}$ that corresponds to a classical Tanner code $\mathcal{C}_G = T(\mathcal{G}, \mathcal{C}_0)$ where \mathcal{G} is a Δ -regular spectral expander and \mathcal{C}_0 is a $[\Delta, 1, \Delta]$ classical repetition local code for some constants $\Delta, \gamma, \alpha > 0$. Then, there exists a universal fault-tolerant quantum computation protocol implements a universal gate set using only gadgets that satisfy single-shot universality. The protocol can tolerate $\Omega(\sqrt{n})$ adversarial faults and also exhibits a threshold under local-stochastic noise.*

Proof. We break down the proof into several subsections that addresses each individual gadget. Similar to the proof of Theorem 5.1, we provide some intuition for the fault-tolerance of the protocol and defer a more rigorous analysis of the fault-tolerance to Section 8.

6.1 Single-Shot State Preparation

To initialize logical states $|\bar{0}\rangle$ and $|\bar{+}\rangle$ in the 2D HGP code \mathcal{Q} , we simply employ the single-shot state preparation scheme developed by Bergamaschi and Liu in Ref. [BL24]. If we are interested in preparing logical states in the 3D HGP code \mathcal{Q}_G , we can either use the same scheme by Bergamaschi and Liu or the state preparation scheme developed by Hong in Ref. [Hon24] depending on the basis in which the 3D HGP code \mathcal{Q}_G is sound. If the 3D HGP code \mathcal{Q}_G is sound in the Z (X) basis, we can use Hong’s scheme to prepare $|\bar{+}\rangle$ ($|\bar{0}\rangle$) directly in \mathcal{Q}_G . Otherwise, we can utilize the other single-shot state preparation scheme by Bergamaschi and Liu [BL24] to prepare the desired logical state in \mathcal{Q}_G .

6.2 Single-Shot Error Correction

HGP codes built from classical expander codes are known to support single-shot error correction. Specifically, 2D HGP codes built from random lossless expanders admit single-shot correction under both adversarial and local-stochastic noise models [FGL18, Cam19, QVRC21, HGL25] with high probability. We now state a theorem regarding the single-shot correction capability for the 2D HGP codes that we use for our protocol.

Theorem 6.2 (Linear confinement of 2D HGP codes from Sipser-Spielman codes [DHLV23, Restatement of result]). *Let \mathcal{Q} be a distance d 2D HGP code constructed from two classical Sipser-Spielman codes built from spectral expanders and random local codes. Then, with high probability, \mathcal{Q} has $(O(d), f)$ -confinement such that $f(x) = \Theta(x)$ in both Pauli bases.*

Proof. The proof follows directly from the results in Ref. [DHLV23]. The main idea is that the product of two random local codes will satisfy the product expansion property with high probability. Satisfying the product expansion property on good spectral expanders gives good boundary expansion, which in turn implies small-set coboundary expansion and linear confinement in both Pauli bases for the 2D HGP code. \square

This shows that the 2D HGP codes built from the Sipser-Spielman codes discussed above exhibit linear confinement with high probability, which is a sufficient condition for single-shot correction under adversarial noise and local-stochastic noise models. As for higher dimensional HGP codes constructed with a base 2D HGP code built from such Sipser-Spielman codes, it is still very much an open question. Campbell, Quintavalle, and co-authors [QVRC21] showed that higher-dimensional HGP codes from random lossless expanders achieve single-shot correction under the adversarial model in at least one Pauli basis.

For our 3D HGP codes, although single-shot correction may be available only in one Pauli basis under adversarial noise, we can postpone correction in the other basis until we return to a 2D HGP where both bases support single-shot procedures. This is adequate for our fault-tolerance analysis because all our gadgets run in constant depth, so error growth can be bounded. Moreover, we occupy the 3D code only briefly—to execute the transversal CCZ—before switching back to 2D, so additional errors in the non-corrected basis are not expected to accumulate significantly.

6.3 Single-Shot Logical Measurement

Logical Pauli measurements on HGP codes can be executed in one shot, i.e., with a single round of physical readout rather than repeated syndrome cycles. Using the grid Pauli product measurement (GPPM) framework of Xu et al. [XZZ⁺25], we first target the desired logical by coupling the 2D data HGP code block to an ancilla 2D HGP code block that is punctured and/or augmented to expose the right operators. The ancilla code block can be prepared with the Bergamaschi and Liu scheme. This coupling is implemented via homomorphic CNOTs, which map the logical Pauli on the data into a Pauli on the ancilla. We then perform transversal single-qubit measurements of the ancilla in a Steane-style scheme. Finally, a lightweight classical postprocessing step—accounting for ancilla stabilizers and known byproducts—combines the raw outcomes to infer the logical eigenvalue fault-tolerantly. The procedure is constant-depth, avoids multiple measurement rounds, and isolates the targeted logical without disturbing others, making it compatible with standard HGP decoding pipelines. If we are planning to measure all logical qubits, we can simply perform transversal single-qubit measurements of the data HGP code block in the desired basis and perform classical postprocessing to infer the logical eigenvalues fault-tolerantly.

6.4 Logical Clifford Gates

2D HGP codes admit several complementary strategies for realizing logical Clifford gates, many of which achieve constant depth. One route is the fold-transversal construction of Breuckmann and Burton [BB24], which—together with recent partitioning techniques [QWV23]—yields depth-bounded Clifford implementations. A second option is code automorphisms: permutations of qubits and checks that act as logical Cliffords [BGH⁺25]. A third, measurement-driven route is Pauli-based computation [BSS16]. Here we employ the GPPM framework of Xu et al. [XZZ⁺25]: the data HGP block is entangled with a punctured and/or augmented HGP ancilla via homomorphic CNOTs, enabling targeted logical Pauli measurements through transversal readout of the ancilla. This allows us to perform addressable logical Clifford gates in a constant-depth circuit when coupled with fold-transversal gates and logical automorphisms. In the case when certain Clifford gates are not achievable in the 2D HGP code \mathcal{Q} , we can temporarily switch to the 3D HGP code \mathcal{Q}_G to use the logical CCZ gate to descend to the CZ gate and then obtain the logical Hadamard which is sufficient to achieve universal quantum computation along with the logical CCZ gate as shown in Section 3. Alternatively, we can use the homomorphic CNOTs discussed in Remark 4.7 to teleport the logical qubits from our 2D HGP code \mathcal{Q} to another 2D HGP code that has the necessary symmetries to

implement the desired logical Clifford gates. For example, we can use the transversal homomorphic CNOT gadget to teleport the logical qubits from our 2D HGP code $\mathcal{Q} = \mathcal{C}_1 \times \mathcal{C}_2$ to a 3D HGP code $\mathcal{C}_1 \times \mathcal{C}_2 \times \mathcal{C}_2$ before teleporting the logical qubits again with another round of homomorphic CNOT to a 2D HGP code $\mathcal{C}_2 \times \mathcal{C}_2$ that has the symmetries to implement the desired logical Hadamard gate. Subsequently, we can teleport the logical qubits back to our original 2D HGP code \mathcal{Q} with two more rounds of homomorphic CNOTs. This addressable Clifford approach is explored in greater detail in Refs. [XZB⁺25] and [GCZ25].

6.5 Logical CCZ Gates

In this subsection, we describe how to choose the right spectral expanders and local codes to construct 3D HGP codes that allow us to perform logical CCZ gates across them. The construction relies on Theorem 2.22, which imposes the local multiplication property on the corresponding local codes of three blocks of 3D HGP codes. We shall use the Tanner code notation of $T(\mathcal{G}, \mathcal{C})$ to denote the Tanner code constructed from a spectral expander \mathcal{G} and a classical local code \mathcal{C} . Recall that we denote the classical repetition code with a redundant check by \mathcal{C}_0 .

Definition 6.3. Let \mathcal{Q}_1^{3D} , \mathcal{Q}_2^{3D} , and \mathcal{Q}_3^{3D} be three 3D HGP codes constructed as follows:

$$\begin{aligned}\mathcal{Q}_1^{3D} &= T(\mathcal{G}_1, \mathcal{C}_1) \otimes T(\mathcal{G}_2, \mathcal{C}_2^\perp) \otimes T(\mathcal{G}_3, \mathcal{C}_0), \\ \mathcal{Q}_2^{3D} &= T(\mathcal{G}_1, \mathcal{C}_0) \otimes T(\mathcal{G}_2, \mathcal{C}_2) \otimes T(\mathcal{G}_3, \mathcal{C}_3^\perp), \\ \mathcal{Q}_3^{3D} &= T(\mathcal{G}_1, \mathcal{C}_1^\perp) \otimes T(\mathcal{G}_2, \mathcal{C}_0) \otimes T(\mathcal{G}_3, \mathcal{C}_3),\end{aligned}$$

where \mathcal{C}_i^\perp denotes the dual code of \mathcal{C}_i for $i = 1, 2, 3$.

Note that the graphs \mathcal{G}_1 , \mathcal{G}_2 , and \mathcal{G}_3 and the classical codes \mathcal{C}_1 , \mathcal{C}_2 , and \mathcal{C}_3 need not be the same. In addition, each $T(\mathcal{G}_i, \cdot)$ for $i = 1, 2, 3$ can be understood as the skeleton for each of the three geometric dimensions. In other words, we can map $i = 1, 2, 3$ to the X, Y, Z directions, respectively. Then, \mathcal{Q}_1^{3D} has $T(\mathcal{G}_1, \mathcal{C}_1)$, $T(\mathcal{G}_2, \mathcal{C}_2^\perp)$, and $T(\mathcal{G}_3, \mathcal{C}_0)$ oriented in the X, Y , and Z directions respectively. All three codes \mathcal{Q}_1^{3D} , \mathcal{Q}_2^{3D} , and \mathcal{Q}_3^{3D} are different instances of the 3D HGP code \mathcal{Q}_G . Since dimensional expansion grows a 2D HGP code into a 3D HGP code with \mathcal{C}_0 in the new dimension, we can imagine the string-like X logical operator of the 2D HGP code in the $X - Y$ plane growing into a membrane-like X logical operator in the Z direction. The same interpretation can be applied to \mathcal{Q}_2^{3D} and \mathcal{Q}_3^{3D} where their X logical operators grow from the $Y - Z$ and $X - Z$ planes in the X and Y directions respectively. One can intuitively see how the three membrane-like X logical operators of \mathcal{Q}_1^{3D} , \mathcal{Q}_2^{3D} , and \mathcal{Q}_3^{3D} can intersect at a single point, which is the key requirement for implementing a transversal CCZ gate. We provide a diagram to illustrate this idea in Figure 6.

Now, we state the following lemma that shows how the above simple construction satisfies the local multiplication property.

Theorem 6.4 (Satisfaction of the Local Multiplication Property). *Given arbitrary spectral expanders \mathcal{G}_1 , \mathcal{G}_2 , and \mathcal{G}_3 as well as arbitrary classical local codes \mathcal{C}_1 , \mathcal{C}_2 , and \mathcal{C}_3 , the 3D HGP codes \mathcal{Q}_1 , \mathcal{Q}_2 , and \mathcal{Q}_3 constructed above in Definition 6.3 satisfy the local multiplication property required for Theorem 2.22.*

Proof. We need to show that the local codes of \mathcal{Q}_1 , \mathcal{Q}_2 , and \mathcal{Q}_3 satisfy the local multiplication property. This follows from the structure of the local codes and their relationships. Specifically, we need to verify that for any three copies of Tanner codes $T(\mathcal{G}_i, \cdot)$ that corresponds to any

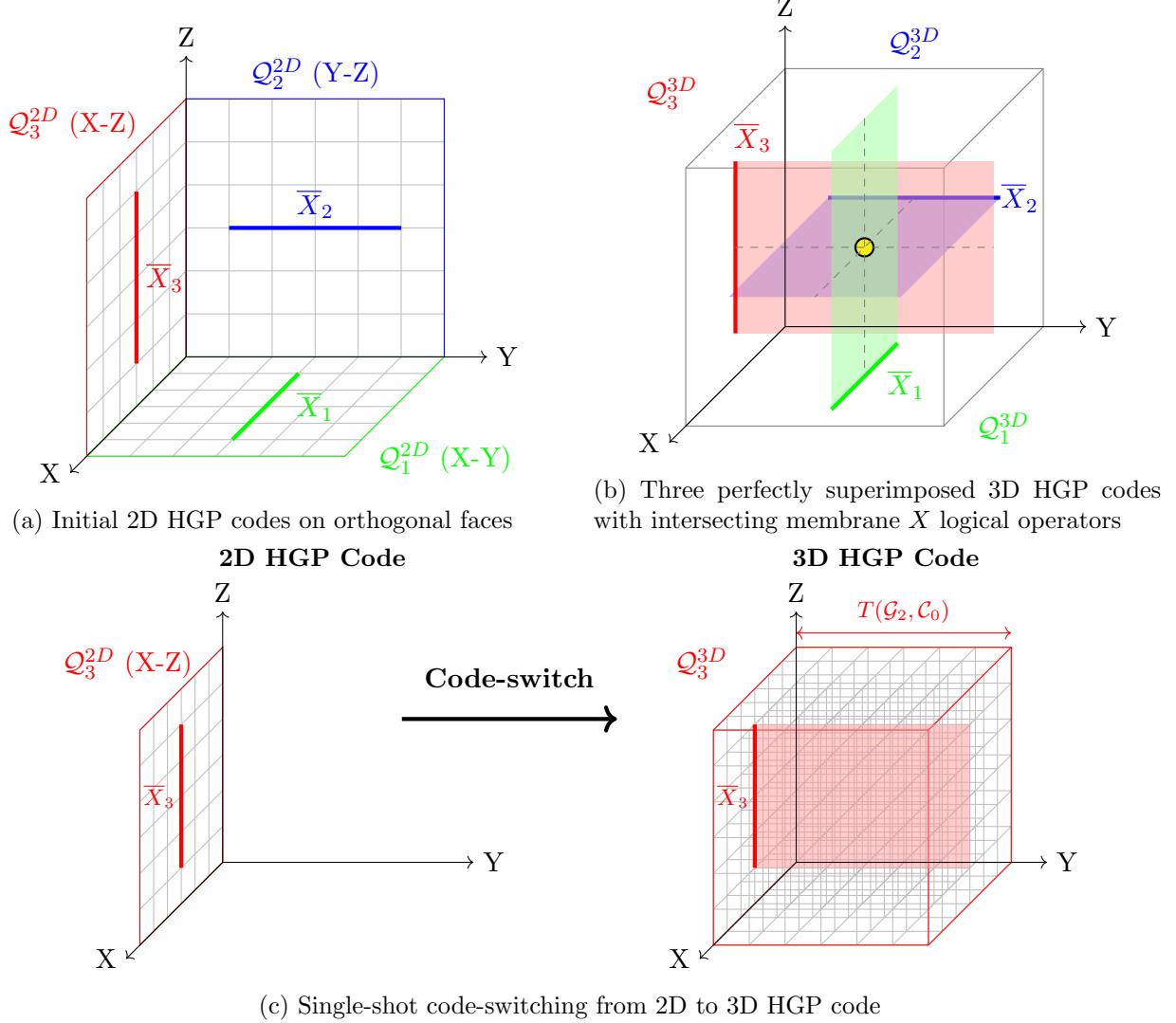


Figure 6: Construction of 3D hypergraph product codes for transversal logical CCZ gates. (a) Three 2D HGP codes \mathcal{Q}_1^{2D} (X-Y), \mathcal{Q}_2^{2D} (Y-Z), and \mathcal{Q}_3^{2D} (X-Z) positioned on orthogonal faces of a cube. Note that \mathcal{Q}_1^{2D} (X-Y) is simply \mathcal{Q}_1^{3D} but without the generalized repetition code component in the Z direction. The other 2D HGP codes can be defined in the same way. (b) After code-switching, each 2D HGP code grows into a 3D HGP code. The three 3D HGP codes can be perfectly superimposed on each other. The now membrane-like logical X operators \bar{X}_1 , \bar{X}_2 , and \bar{X}_3 intersect at a single point (yellow), enabling transversal CCZ implementation. (c) Schematic of the single-shot code-switching from 2D to 3D HGP code for \mathcal{Q}_3 . The generalized repetition code component $T(\mathcal{G}_2, \mathcal{C}_0)$ in the Y direction is indicated. The X logical operator grows from a string to a membrane. The same procedure applies to \mathcal{Q}_1 and \mathcal{Q}_2 .

$i \in \{1, 2, 3\}$, the local codes $\mathcal{C}_{1,\text{loc}}, \mathcal{C}_{2,\text{loc}}, \mathcal{C}_{3,\text{loc}} \subseteq \mathbb{F}_2^\Delta$ corresponding to $\mathcal{Q}_1, \mathcal{Q}_2$, and \mathcal{Q}_3 satisfy the following condition:

$$\sum_i^{\Delta} (\mathbf{c}_1)_i (\mathbf{c}_2)_i (\mathbf{c}_3)_i = 0 \pmod{2}, \quad \forall \mathbf{c}_1 \in \mathcal{C}_{1,\text{loc}}, \mathbf{c}_2 \in \mathcal{C}_{2,\text{loc}}, \mathbf{c}_3 \in \mathcal{C}_{3,\text{loc}}.$$

This can be shown by analyzing the structure of the local codes and their interactions. Let us do that for \mathcal{G}_1 and the same argument applies to \mathcal{G}_2 and \mathcal{G}_3 . In this case, the three local codes that we have to check are given by $\mathcal{C}_1, \mathcal{C}_0, \mathcal{C}_1^\perp$ respectively. By the definition of the dual code, we have

$$\sum_i^{\Delta} (\mathbf{c}_1)_i (\mathbf{c}_3)_i = 0 \pmod{2}, \quad \forall \mathbf{c}_1 \in \mathcal{C}_1, \mathbf{c}_3 \in \mathcal{C}_1^\perp.$$

Because the codewords of \mathcal{C}_0 are either 0^Δ or 1^Δ , we have

$$\sum_i^{\Delta} (\mathbf{c}_1)_i (\mathbf{c}_2)_i (\mathbf{c}_3)_i = 0 \pmod{2}, \quad \forall \mathbf{c}_1 \in \mathcal{C}_1, \mathbf{c}_2 \in \mathcal{C}_0, \mathbf{c}_3 \in \mathcal{C}_1^\perp.$$

This completes the proof for \mathcal{G}_1 . The same argument applies to \mathcal{G}_2 and \mathcal{G}_3 . \square

The lemma above provides the foundation for implementing transversal CCZ gates in the described construction. The local multiplication property ensures compatibility between the local codes, which is critical for fault-tolerant CCZ computation. This construction is inspired by the ideas in Zhu's paper [Zhu25a] and can be viewed as a vastly simpler and cleaner version of the construction used in Ref. [GL25], where much more advanced techniques were used to achieve better code parameters with punctured Reed-Solomon local codes on a special expander graph at the expense of the LDPC property. Similar to the construction in Ref. [Zhu25a], our construction offers great flexibility in the choice of the expander graphs and local codes, allowing for a wide range of code parameters and properties. This flexibility is particularly useful for finding interesting finite size examples that may have useful logical gates and code automorphisms [BGH⁺25]. We note that Theorem 2 in Ref. [Zhu25a] provides a concrete way to identify the necessary cohomology classes that give rise to non-trivial logical CCZ gates. By using Zhu's result, it is possible to find the number of logical CCZ gates that can be implemented transversally for a given choice of the expander graphs and local codes.

We now proceed to show how the construction defined in Definition 6.3 can satisfy the rate scaling described in Theorem 6.1.

Lemma 6.5. *The following 2D HGP codes constructed using the Sipser-Spielman codes built from spectral expander graphs \mathcal{G}_i and classical local codes \mathcal{C}_i for $i = 1, 2, 3$ have constant rate:*

- $\mathcal{Q}_1^{2D} = T(\mathcal{G}_1, \mathcal{C}_1) \otimes T(\mathcal{G}_2, \mathcal{C}_2^\perp)$
- $\mathcal{Q}_2^{2D} = T(\mathcal{G}_2, \mathcal{C}_2) \otimes T(\mathcal{G}_3, \mathcal{C}_3^\perp)$
- $\mathcal{Q}_3^{2D} = T(\mathcal{G}_1, \mathcal{C}_1^\perp) \otimes T(\mathcal{G}_3, \mathcal{C}_3)$

if the classical local codes $\mathcal{C}_1, \mathcal{C}_2$, and \mathcal{C}_3 have rates r_1, r_2 , and r_3 , respectively with $r_1, r_2, r_3 > 1/2$.

Proof. It is known that the rate of a Sipser-Spielman code is at least $2r - 1$ where r is the rate of the classical local code [SS96]. Thus, when we define \mathcal{C}_i to have rate greater than $1/2$, the Sipser-Spielman code $T(\mathcal{G}_i, \mathcal{C}_i)$ has positive rate, implying that the first homology group associated

with its chain complex has dimension that scales linearly with the number of physical bits. On the other hand, $T(\mathcal{G}_i, \mathcal{C}_i^\perp)$ would have a vanishing rate but its transpose would have positive rate, implying that the zeroth homology group associated with its chain complex has dimension that scales linearly with the number of physical bits. By the Künneth formula stated in Proposition 2.8, the 2D HGP codes \mathcal{Q}_1^{2D} , \mathcal{Q}_2^{2D} , and \mathcal{Q}_3^{2D} would achieve constant rate. \square

The rate of the 3D HGP codes \mathcal{Q}_1^{3D} , \mathcal{Q}_2^{3D} , and \mathcal{Q}_3^{3D} constructed in Definition 6.3 can be derived from the rate of the corresponding 2D HGP codes \mathcal{Q}_1^{2D} , \mathcal{Q}_2^{2D} , and \mathcal{Q}_3^{2D} and the rate of the generalized repetition code $T(\mathcal{G}_i, \mathcal{C}_0)$ using the Künneth formula stated in Proposition 2.8. Thus, we have proven the theorem statement. \square

7 Linear Soundness in the 3D HGP Code

In this section, we will show that our 3D HGP code \mathcal{Q}_G exhibits small-set linear soundness in the membrane basis. Suppose our 3D HGP code of length $\Delta = \Theta(n^{3/2})$ is extended in the X basis (X logicals are membrane-like). Then we have the following 4-term cochain complex:

$$\tilde{S}_X \xrightarrow{\tilde{H}_X^\top} \tilde{Q} \xrightarrow{\tilde{H}_Z} \tilde{S}_Z \xrightarrow{\tilde{M}_Z} \tilde{R}_Z, \quad (22)$$

where S_X is the space of X syndromes, Q the space of physical qubits, S_Z the space of Z syndromes, and R_Z the space of Z -check relations. M_Z is the Z -metacheck matrix, and H_X and H_Z are the usual CSS parity-check matrices. We want to show that H_Z is (t, f) -sound with $t = \Theta(n^{1/3})$ and $f(x) = \alpha x$ for some $\alpha = O(1)$. To get this result, we will show small-set coboundary expansion for the metachecks M_Z . It is known that small-set coboundary expansion can be reduced to the property of spectral expansion of the graphs and the product expansion of the underlying classical codes [DLV24]. By construction the graphs we use are spectral expanders, and so it suffices to check the product expansion of the underlying classical codes. It is known that a pair of random codes are product-expanding [DHLV23, KP22]. As such, the local codes we will use are two random codes and a repetition code. We show below that adding a repetition code to a product-expanding pair of codes preserves the product expansion property.

Given two linear codes $C_1, C_2 \subseteq \mathbb{F}_2^\Delta$, we define $C_1 \boxplus C_2 = C_1 \otimes \mathbb{F}_2^\Delta + \mathbb{F}_2^\Delta \otimes C_2 \subseteq \mathbb{F}_2^{\Delta \times \Delta}$. More generally, given a collection of linear codes $C = (C_1, C_2, \dots, C_D)$ of linear codes over \mathbb{F}_2 , we define the codes

$$C^{(i)} := \mathbb{F}_2^\Delta \otimes \cdots \otimes C_i \otimes \cdots \otimes \mathbb{F}_2^\Delta \quad (23)$$

which is formed by tensors $x \in \mathbb{F}_2^{\Delta^D}$ whose restriction along the i -th axis are codewords in C_i . It is clear that

$$C_1 \boxplus \cdots \boxplus C_D = C^{(1)} + \cdots + C^{(D)}. \quad (24)$$

For each $i \in [D]$, let L_i be the set of lines parallel to the i -th axis in the D -dimensional grid $[\Delta]^D$. For each tensor $x \in \mathbb{F}_2^{\Delta^D}$, we define $|x|$ as the Hamming weight of x , and define $|x|_i$ as the number of lines $\ell \in L_i$ such that x is nonzero on that line, $|x|_\ell \neq 0$.

Definition 7.1 (Product expansion [KP25]). *Given a collection of linear codes $C_i \subset \mathbb{F}_2^\Delta$, we say that (C_1, \dots, C_D) is ρ -product-expanding if every codeword $c \in C_1 \boxplus \cdots \boxplus C_D$ can be represented as a sum $c = \sum_{i=1}^D a_i$, where $a_i \in C^{(i)}$ for all $i \in [D]$, and the following inequality holds:*

$$\rho \Delta \sum_{i \in [D]} |a_i|_i \leq |c|. \quad (25)$$

Lemma 7.2 (Product expansion with repetition). *If (C_1, C_2) is ρ -product-expanding, then the $(C_1, C_2, C_{\text{rep}})$ is $\rho/3$ -product-expanding.*

Proof. Let $c \in C_1 \boxplus C_2 \boxplus C_{\text{rep}}$ be a codeword. Consider the 2D slices of c parallel to the first two axes, which we write $c = \sum_{i=1}^{\Delta} c_i \otimes e_i$ where $c_i \in \mathbb{F}_2^{\Delta \times \Delta}$ is the i -th slice and $e_i \in \mathbb{F}_2^{\Delta}$ is the i -th standard basis vector. Because C_{rep} is the repetition code, any tensor in $C^{(3)}$ is constant on each line parallel to the third axis. Therefore, $c_i - c_j \in C_1 \boxplus C_2$ for all $i, j \in [\Delta]$. Let $\varepsilon \in \mathbb{F}_2^{\Delta \times \Delta}$ be the difference between c_1 and its closest codeword in $C_1 \boxplus C_2$ in the Hamming distance. Then $c_i - \varepsilon \in C_1 \boxplus C_2$ for all $i \in [\Delta]$. It is $|\varepsilon| \leq |c_i|$ for all $i, j \in [\Delta]$. Let $c' = \sum_{i=1}^{\Delta} (c_i - \varepsilon) \otimes e_i$ and let $c'' = \varepsilon \otimes (1, 1, \dots, 1)$. Then $c = c' + c''$ where $c' \in C^{(1)} + C^{(2)}$ and $c'' \in C^{(3)}$. Since (C_1, C_2) is ρ -product-expanding, we can write $c_i - \varepsilon = a_{i,1} + a_{i,2}$, where $a_{i,1} \in C_1 \otimes \mathbb{F}_2^{\Delta}$ and $a_{i,2} \in \mathbb{F}_2^{\Delta} \otimes C_2$ such that $\rho\Delta(|a_{i,1}|_1 + |a_{i,2}|_2) \leq |c_i - \varepsilon|$.

Using $a_{i,1}$, $a_{i,2}$, and ε , we are ready to construct a decomposition of c into $c = a_1 + a_2 + a_3$. Let a_1 be the tensor whose i -th slice is $a_{i,1}$, and let a_2 be the tensor whose i -th slice is $a_{i,2}$. Let a_3 be the tensor $\varepsilon \otimes (1, 1, \dots, 1)$ which is constant on each line parallel to the third axis. It is clear that $a_1 \in C^{(1)}$, $a_2 \in C^{(2)}$, and $a_3 \in C^{(3)}$.

We now verify the inequality (25). For each 2D slice, we have

$$\rho\Delta(|a_{i,1}|_1 + |a_{i,2}|_2) \leq |c_i - \varepsilon|, \quad (26)$$

which implies

$$|a_{i,1}|_1 + |a_{i,2}|_2 + \frac{|\varepsilon|}{\Delta} \leq \frac{|c_i - \varepsilon|}{\rho\Delta} + \frac{|\varepsilon|}{\Delta} \leq \frac{|c_i| + |\varepsilon| + |\varepsilon|}{\rho\Delta}, \leq \frac{3|c_i|}{\rho\Delta}, \quad (27)$$

where we used the fact that $\rho \leq 1$ and $|\varepsilon| \leq |c_i|$. Summing over all $i \in [\Delta]$,

$$|a_1|_1 + |a_2|_2 + |a_3|_3 = \sum_{i=1}^{\Delta} \left(|a_{i,1}|_1 + |a_{i,2}|_2 + \frac{|\varepsilon|}{\Delta} \right) \leq \sum_{i=1}^{\Delta} \frac{3|c_i|}{\rho\Delta} = \frac{3|c|}{\rho\Delta}. \quad (28)$$

It follows that $(C_1, C_2, C_{\text{rep}})$ is $\rho/3$ -product-expanding. \square

Using the product-expansion promised by Lemma 7.2, we can now show small-set coboundary expansion of \tilde{M}_Z in (22).

Theorem 7.3 (3D Metacheck expansion). *The 3D HGP code \mathcal{Q}_G exhibits small-set coboundary expansion in \tilde{M}_Z . In particular, there exists a $t = \Theta(\tilde{n}^{1/3})$ such that for any $s \in S_Z$ with $|s| \leq t$,*

$$|\tilde{M}_Z s| \geq \kappa \min_{e \in Q} |s + \tilde{H}_Z e|, \quad (29)$$

where $\kappa = \Theta_{\rho, \lambda, \Delta}(1)$. Moreover, we also have $|e| \leq \alpha|s|$.

Proof. Since $\mathcal{G}_1, \mathcal{G}_2, \mathcal{G}_3$ are λ -spectral expanders, the subgraphs of $\mathcal{G}_1 \times \mathcal{G}_2 \times \mathcal{G}_3$ corresponding to the i -th direction are also λ -spectral expanders up to size $\Theta(|\mathcal{G}_i|) = \Theta(\tilde{n}^{1/3})$. Combined with the fact that $(C_1, C_2, C_{\text{rep}})$ is ρ -product-expanding, it follows from [DLV24, Proposition 7.1] that there exists $t = \Theta(\tilde{n}^{1/3}) < d_{\text{coLM}}$, where d_{coLM} is the locally co-minimal distance (Definition 2.19). Small-set coboundary expansion (29) similarly follows from Theorem 3.6 in [DLV24].

We now prove the statement $|e| \leq \alpha|s|$. If s is not locally co-minimal (Definition 2.18), then there exists a local flip $s_1 = \tilde{H}_Z e_1$ such that $|s + s_1| < |s|$. Since $s + s_1$ has a smaller weight than $|s| < d_{\text{coLM}}$, we can repeat the process to obtain a sequence of local flips that eventually reduces s to its locally co-minimal representative. Let s_1, s_2, \dots, s_T be these local flips, where each $s_i = \tilde{H}_Z e_i$. This implies $s = \sum_{i=1}^T s_i$ and we define $e = \sum_{i=1}^T e_i$. It is clear that $s = \tilde{H}_Z e$.

To bound the weight of e , note that each step decreases the weight of s by at least 1, which implies $T \leq |s|$. Since the chain complex is LDPC, each local flip e_i has bounded weight $|e_i| = \Theta(1)$. Therefore, we have

$$|e| \leq \sum_{i=1}^T |e_i| = O(T) = O(|s|), \quad (30)$$

which implies $|e| \leq \alpha|s|$ for some constant α . \square

An immediate corollary of Theorem 7.3, for the case when $s \in \ker \tilde{M}_Z$, is (small-set) linear soundness in our 3D HGP code.

Corollary 7.4. *The 3D HGP code \mathcal{Q}_G is (t, f) -sound with $t = \Theta(\tilde{n}^{1/3})$ and $f(x) = \alpha x$ for $\alpha = O(1)$. In particular, for any $s \in \ker \tilde{M}_Z$ with $|s| \leq t$, there exists $e \in Q$ such that $s = \tilde{H}_Z e$ and $|e| \leq \alpha|s|$.*

8 Fault Tolerance for Adversarial Noise

In this section, we provide a rigorous analysis of the fault tolerance of the gadgets in our universal protocol against adversarial noise. To be explicit, we consider an adversarial noise model where an adversary can choose any set of s fault locations in the circuit and apply arbitrary Pauli errors at those locations. This includes errors on the data and ancilla qubit errors, measurement errors, preparation errors, and gate errors. We first review the necessary background on the fault tolerance properties of gadgets. Subsequently, we show that our gadgets satisfy these properties, thereby establishing the fault tolerance of our universal protocol by a simple composition argument.

8.1 Fault Tolerance of Gadgets

In this section, we mostly restate the definitions and results from Ref. [Got24]. We start off by reviewing what it means for a gate gadget to be fault-tolerant under the single-shot universality framework. Note that a gate gadget essentially captures the idea of a logical gate implemented on an encoded state, and so it is a more general notion than a transversal gate. In fact, it also encapsulates lattice surgery and code switching protocols.

Definition 8.1 (Fault-tolerant gate gadget [Got24, Restatement of Definitions 10.10 and 10.11]). *Suppose we have a gate gadget that is associated with a QLDPC code with distance $2t + 1$ that has u input and output bits for some constant u . Let r_i for $i = 1, \dots, u$ be the incoming faults for each of the u input qubits, and let s be the total number of faults that occur during the gadget. A gate gadget is said to be fault-tolerant if it satisfies the following properties when $\sum_{i=1}^u r_i + f(s) \leq t$ for some increasing function f :*

- (Gate Error Propagation Property) *The output u -qubit state has at most $\sum_{i=1}^u r_i + f(s)$ faults on each of the u output qubits.*
- (Gate Correctness Property) *The output u -qubit state when passed through an ideal decoder implements the intended logical operation on the input state. In other words, the output state from the noisy gate gadget when passed through an ideal decoder is the same as if we had passed the input state through an ideal decoder followed by the ideal logical gate operation.*

It should not be too hard to see that transversal, fold-transversal, and swap-transversal gates are all examples of fault-tolerant gate gadgets with a constant factor difference in the function f .

Next, we state the definition of fault-tolerance for a state-preparation gadget, which is a procedure to prepare an encoded logical state.

Definition 8.2 (Fault-tolerant state-preparation gadget [Got24, Restatement of Definitions 10.13 and 10.14]). *Suppose we have a state-preparation gadget that is associated with a QLDPC code with distance $2t + 1$. Let s be the total number of faults that occur during the gadget. A state-preparation gadget is said to be fault-tolerant if it satisfies the following properties when $f(s) \leq t$ for some increasing function f :*

- (*State-Preparation Error Propagation Property*) *The output state has at most $f(s)$ faults.*
- (*State-Preparation Correctness Property*) *The output state when passed through an ideal decoder is the intended logical state.*

Subsequently, we state the definition of fault-tolerance for a logical measurement gadget, which is a procedure to measure a logical operator on an encoded state.

Definition 8.3 (Fault-tolerant measurement gadget [Got24, Restatement of Definition 10.15]). *Suppose we have a measurement gadget that is associated with a QLDPC code with distance $2t + 1$. Let r be the number of incoming faults, and let s be the total number of faults that occur during the gadget. A logical measurement gadget is said to be fault-tolerant if it satisfies the following properties when $r + f(s) \leq t$ for some increasing function f :*

- (*Measurement Correctness Property*) *The measurement outcome is the same as if we had passed the input state through an ideal decoder followed by an ideal logical measurement.*

Finally, we state the definition of fault-tolerance for a single-shot error-correction gadget.

Definition 8.4 (Fault-tolerant single-shot error-correction gadget [Got24, Restatement of Definitions 10.17 and 10.18]). *Suppose we have a single-shot error-correction gadget that is associated with a QLDPC code with distance $2t + 1$. Let s be the total number of faults that occur during the gadget. A single-shot error-correction gadget is said to be fault-tolerant if it satisfies the following properties:*

- (*Error-Correction Error Recovery Property*) *When $f(s) \leq t$ for some increasing function f , the output state has at most $f(s)$ faults.*
- (*Error-Correction Correctness Property*) *Let r be the number of incoming faults into the gadget. When $r + f(s) \leq t$ for some increasing function f , the output state when passed through an ideal decoder is the same as if we had passed the input state through an ideal decoder.*

One can interpret the error recovery property as a guarantee that our fault-tolerant gadget will always return us to some codeword with some bounded number of errors no matter how many faults there are to start with. On the other hand, the correctness property guarantees that when the total number of faults is small enough, the encoded logical state does not deviate from the ideal logical state during the gadget.

8.2 Fault Tolerance of Single-Shot State Preparation

We now analyze the fault tolerance of our single-shot state-preparation gadgets against adversarial errors. With slight abuse of notation, we will reuse the same letter for both a vector and its support, and it should be clear by context which one we are referring to.

For the 3D HGP code, we have the following chain complex for the membrane sector:

$$Q \xrightarrow{\mathbf{H}} S \xrightarrow{\mathbf{M}} R \quad (31)$$

where Q, S, R are the vector spaces of qubits, syndromes and relations respectively, and \mathbf{H} and \mathbf{M} are the (low-density) parity-check and metacheck matrices respectively. We call $\text{im } \mathbf{H}$ the space of *physical* syndromes since these are the ones that can be produced by errors on the data qubits in Q . The systolic distance, or smallest nontrivial weight in $\ker \mathbf{M} \setminus \text{im } \mathbf{H}$, is called the single-shot or metacode distance and is labeled as d_{ss} .

For the setting of adversarial noise, the 3D HGP code is amenable to single-shot error correction in at least one of the Pauli basis using the metacheck matrix, which has been described by Campbell [Cam19]. For completeness, we recite the relevant results below. We will assume a theoretical minimum-weight decoder for both the metacode and the physical code; note that some constants (like $1/2$) can be modified and provide the same qualitative behavior if one wishes to relax the minimum-weight assumption on the decoder.

We first introduce some relevant notation. Let \mathbf{s}_{true} be the true syndrome and \mathbf{s}_{err} the syndrome error such that $\mathbf{s}_{\text{obs}} = \mathbf{s}_{\text{true}} + \mathbf{s}_{\text{err}}$ is the observed, corrupted syndrome. Our decoder for M , which we will call a metadecoder, will output a syndrome correction \mathbf{s}_{corr} such that $\mathbf{s}_{\text{rep}} = \mathbf{s}_{\text{obs}} + \mathbf{s}_{\text{corr}}$ is the repaired or inferred syndrome. The metadecoder succeeds if $\mathbf{s}_{\text{rep}} \in \text{im } \mathbf{H}$; since $\mathbf{s}_{\text{true}} \in \text{im } \mathbf{H}$ by default, the success requirement is equivalent to $\mathbf{s}_{\text{err}} + \mathbf{s}_{\text{corr}} \equiv \mathbf{s}_{\text{res}} \in \text{im } \mathbf{H}$. We call \mathbf{s}_{res} the *residual* syndrome, since it will end up as the syndrome for the eventual residual error.

At a high level, the distance d_{ss} of the metacode will tell us how many syndrome errors we can tolerate before $\mathbf{s}_{\text{res}} = \mathbf{s}_{\text{err}} + \mathbf{s}_{\text{corr}}$ lies outside of $\text{im } \mathbf{H}$ and hence cannot be decoded to a physical correction. Then the soundness property will tell us how large the corresponding residual error \mathbf{e}_{res} can be, given the residual syndrome $\mathbf{s}_{\text{res}} \in \text{im } \mathbf{H}$. If the initial physical and syndrome errors are small enough, then the combination of the residual error with a new round of physical and syndrome errors will still be correctable, and hence sustainable single-shot QEC is achieved over many (noisy) syndrome extraction rounds.

Lemma 8.5 (Metacode success; Lemma 1 in [Cam19]). *If $|\mathbf{s}_{\text{err}}| < d_{ss}/2$, and we have access to a minimum-weight metadecoder, then the residual syndrome $\mathbf{s}_{\text{res}} \in \text{im } \mathbf{H}$ is physical.*

Proof. Since \mathbf{s}_{err} itself is also a valid syndrome correction, by minimality of our metadecoder, we have $|\mathbf{s}_{\text{corr}}| \leq |\mathbf{s}_{\text{err}}|$. Hence $|\mathbf{s}_{\text{res}}| = |\mathbf{s}_{\text{err}} + \mathbf{s}_{\text{corr}}| \leq |\mathbf{s}_{\text{err}}| + |\mathbf{s}_{\text{corr}}| \leq 2|\mathbf{s}_{\text{err}}| < d_{ss}$, which implies that \mathbf{s}_{res} belongs to the trivial class in $\ker \mathbf{M} / \text{im } \mathbf{H}$, i.e. $\mathbf{s}_{\text{res}} \in \text{im } \mathbf{H}$. \square

Note that the definition of soundness (Definition 2.13) requires $t < d_{ss}$, since it asserts that any syndrome with weight less than or equal to t is physical and so belongs to $\text{im } \mathbf{H}$.

Lemma 8.6 (Residual error bound; Lemma 2 in [Cam19]). *Assume our code is (t, f) -sound with f an increasing function, and let \mathbf{s}_{err} the syndrome error. If $|\mathbf{s}_{\text{err}}| < t/2$, then the residual error satisfies*

$$|\mathbf{e}_{\text{res}}| \leq f(2|\mathbf{s}_{\text{err}}|) < f(t). \quad (32)$$

Proof. By minimality of our metadecoder, $|\mathbf{s}_{\text{res}}| \leq 2|\mathbf{s}_{\text{err}}| < t$, and so the residual syndrome lies within the soundness radius of the code. Therefore, by Definition 2.13 and the fact that f is increasing, we have $|\mathbf{e}_{\text{res}}| < f(t)$. \square

For our 3D HGP \mathcal{Q}_G , Lemma 6 in [Cam19] tells us that our soundness radius $t = \Theta(d_c)$, assuming that the repetition code in the third dimension has distance $\Theta(d_c)$, where d_c is the distance of the two classical codes in the 2D HGP. All that remains is to determine the soundness function f . By Corollary 7.4, we have $f(x) = \Theta(x)$, i.e., linear soundness.

Now, we are ready to prove the fault tolerance of our single-shot state-preparation gadgets. Note that the following statement holds for both the 2D HGP codes and the 3D HGP codes \mathcal{Q}_S and \mathcal{Q}_G . In other words, we can prepare the logical $|\bar{0}\rangle$ and $|\bar{+}\rangle$ states of both the 2D and 3D HGP codes in a fault-tolerant manner. For the 2D HGP code case, there is just the additional dimensional reduction step whose fault tolerance in the adversarial noise model has already been established in Ref. [Hon24].

Lemma 8.7 (Fault-tolerant single-shot state preparation). *Our single-shot state-preparation gadgets for the logical computational basis states of the 3D HGP code \mathcal{Q}_G (in the basis with the redundant checks) are fault-tolerant according to Definition 8.2 with $f(s) = \Theta(s)$.*

Proof. Suppose our 3D HGP code has X membrane logical operators and Z string logical operators without loss of generality. In addition, consider the case where we have Pauli X errors \mathbf{e}_X , Pauli Z errors \mathbf{e}_Z , and syndrome measurement errors \mathbf{s}_{err} during the state-preparation gadget. The total number of faults is $s = |\mathbf{e}_X| + |\mathbf{e}_Z| + |\mathbf{s}_{\text{err}}| \leq t$ where $t = (d - 1)/2$ and d is the distance of the 3D HGP code. Then, a minimum-weight metadecoder and the decoders stated in Ref. [Hon24] ensure that \mathbf{e}_X and \mathbf{s}_{err} will be converted into an X residual error with weight that is bounded by some linear function of $|\mathbf{s}_{\text{err}}|$ by Lemma 8.6 and the linear soundness of our 3D HGP codes given by Corollary 7.4. Combining this residual error with the original Z error \mathbf{e}_Z will give us a total residual error with weight that is upper-bounded by some linear function of s . This establishes the error propagation property. By choosing to define the linear function f with the appropriate constants, we can also ensure that $f(s) \leq t$, which establishes the correctness property when given an ideal decoder. \square

For the case of preparing the logical computational basis state in the 2D HGP code \mathcal{Q} , we committed to using the Bergamaschi and Liu protocol [BL24]. While the fault-tolerance of their state preparation gadget has only been proven for the local stochastic noise model, their protocol can be easily adapted to be fault-tolerant under the adversarial noise model. As long as we set the “thickness” of the gadget which they parameterized as T to be the same as the distance of the code d , then one can show that their protocol can fail when there are at least $\Omega(d)$ faults in the circuit. Therefore, by a similar argument as in Lemma 8.7, their state-preparation gadget is also fault-tolerant under the adversarial noise model with $f(s) = \Theta(s)$.

8.3 Fault Tolerance of Single-Shot Error Correction

The fault tolerance of our single-shot error-correction gadgets against adversarial errors. Recall that we only perform QEC in the 2D HGP code. From Theorem 6.2, we know that our 2D HGP codes have good linear confinement. Thus, they satisfy the residual error bound stated in Lemma 1 of Ref. [QVRC21]. In particular, the residual error has weight that scales linearly with the number of syndrome measurement errors. Therefore, by the same argument as in Lemma 8.7, our single-shot error-correction gadgets satisfy all the conditions for fault tolerance for single-shot error correction under adversarial noise as stated in Definition 8.4.

8.4 Fault Tolerance of Single-Shot Code-Switching

The fault tolerance of our single-shot code-switching gadgets against adversarial errors is effectively a composition of the fault-tolerance of the single-shot state preparation gadget, transversal and homomorphic CNOT gate gadgets, and transversal measurement of code blocks. Because the transversal and homomorphic CNOT gates as well as the transversal measurements are trivially fault-tolerant according to Definitions 8.1 and 8.3 under the adversarial noise model, all of the gadgets that we use in our code-switching protocol are fault-tolerant according to the appropriate definitions. Because all the gadgets are fault-tolerant with growing function f that are at most linear in the number of faults, the entire code-switching protocol is fault-tolerant according to Definition 8.1 with a growing function f that is also linear in the number of faults.

8.5 Fault-Tolerance of Logical Gate Gadgets

We begin by analyzing the fault-tolerance of our logical Clifford gate gadgets against adversarial errors. The (fold)-transversal gates are trivially fault-tolerant according to Definition 8.1 under the adversarial noise model, since a transversal gate will only propagate errors on each qubit to at most one other qubit. Thus, it is only essential to analyze the fault-tolerance of the GPPM gadget introduced in Ref. [XZZ⁺25]. The GPPM gadget is effectively a constant number of compositions of single-shot state preparation, homomorphic CNOT gates, transversal single-qubit gates, and transversal measurements. In fact, our general Clifford gate gadget discussed in Section 6.4 is just a composition of a constant number of the gadgets mentioned above. Because all of these gadgets are fault-tolerant with at worst linear growing functions according to the appropriate definitions under the adversarial noise model, the entire GPPM gadget is fault-tolerant according to Definition 8.1 with a growing function f that is linear in the number of faults.

For the logical CCZ gate gadget, it is simply a single round of code-switching from 2D to 3D HGP code \mathcal{Q}_G , followed by a constant depth circuit of CCZ gates, and then code-switching back to the 2D HGP code. Again, because all the gadgets that we use in our CCZ gadget are fault-tolerant with at worst linear growing functions according to the appropriate definitions under the adversarial noise model, the entire CCZ gadget is fault-tolerant according to Definition 8.1 with a growing function f that is linear in the number of faults.

8.6 Fault-Tolerance of the Single-Shot, Universal Protocol

In order for our universal fault-tolerant scheme to be robust over many logical operations, we need to interleave single-shot error correction between all of our constant-depth gadgets. The reasoning is as follows. Suppose we have some gadget that is not a single-shot error correction gadget and it is fault-tolerant according to the definitions stated in Section 8.1 with a growing function f that is polynomial in the number of faults s that occur during the gadget. In addition, suppose that the code has r physical faults going into the gadget. Then, the output state will have at most $r + f(s)$ faults. If we do not perform error correction after this gadget, then the next gadget will have $r + f(s)$ incoming faults instead of just r incoming faults. If we keep stacking gadgets without performing error correction, then the number of incoming faults will keep increasing and eventually exceed the error-correcting radius of the code, and so the entire protocol will not be fault-tolerant. On the other hand, if we perform single-shot error correction after the gadget, then we have $r + f(s)$ faults going into the single-shot error-correction gadget. Suppose s' faults occur during the single-shot error-correction gadget. By the error-recovery and correctness properties of the single-shot error-correction gadget stated in Definition 8.4, the output state will have at most $f'(s')$ faults for some growing function f' that is linear in s' provided that $r + f(s) + f'(s')$ is less than or equal to

the number of errors that the code can correct. The subsequent gadget will then have at most $f'(s')$ incoming faults instead of $r + f(s)$ incoming faults, and so the number of incoming faults will not keep increasing over many gadgets. Therefore, by interleaving single-shot error correction between all of our constant-depth gadgets, we can ensure that the entire protocol is fault-tolerant according to the appropriate definitions under the adversarial noise model. In particular, the combination of the residual error from the previous round of single-shot error correction with the new errors in the current round remains correctable over many error-correction cycles. In other words, the residual error remains controlled over many noisy QEC cycles.

It is perhaps not too difficult to see that the correctness of the entire protocol follows from the correctness properties of the individual gadgets by a simple composition argument. Nonetheless, this guarantee is broken when too many faults occur in any one of the gadgets. To be more concrete, when $r + f(s)$ exceeds the number of errors that the code can correct for any one of the gadgets, then the correctness of the entire protocol is not guaranteed. Thus, the number of adversarial faults that our entire protocol can tolerate is upper-bounded by the smallest number of adversarial faults that any one of the gadgets can tolerate. In particular, this is characterized by the largest f of the gadgets. Notably, when all the gadgets have f that is linear in the number of faults, then the entire protocol can tolerate a number of adversarial faults that is $O(d)$ where d is the distance of the code. This is because each gadget now outputs a state with a number of residual faults that is linear in the number of physical faults that occur during the gadget. As long as the constants in the linear number of faults are managed appropriately, one can easily show that the single-shot error-correction gadget can always correct the combination of the residual error from the previous gadget with any new errors that occur in the error-correction gadget, provided that the total number of faults is still less than or equal to half the distance of the code. Thus, the entire protocol can tolerate a number of adversarial faults that scales $O(d) = O(\sqrt{n})$.

9 Fault Tolerance for Local-Stochastic Noise

In this section, we show that all 3D gadgets in our protocol exhibit thresholds under local-stochastic noise. Results guaranteeing fault tolerance against local-stochastic noise are typically more relevant than adversarial noise since local-stochastic noise can encompass more realistic error models for LDPC codes such as circuit-level noise. Specifically, we will show that any LDPC code equipped with an LDPC syndrome metacode and exhibiting small-set linear soundness admits a single-shot threshold for transversal logical state preparation. While our analytical results will require certain expansion properties of the underlying code or chain complex, we note that numerical evidence already suggests that generic (LDPC) 3D HGP codes, whose underlying chain complexes are not necessarily expanding, exhibit sustainable single-shot thresholds with respect to two-stage [QVRC21] and single-stage decoders [HB23].

We first review the definition of local-stochastic noise, which encompasses both the independent error model as well as short-range correlations from constant-depth circuit-level noise.

Definition 9.1 (Local-stochastic error). *An error pattern e on a vertex set V is local-stochastic with parameter $0 \leq p \leq 1$ if, for all subsets $v \subseteq V$,*

$$\mathbb{P}[v \subseteq e] \leq p^{|v|}. \quad (33)$$

Note that the independent single-qubit error model with probability p corresponds to an equality in (33). For a local-stochastic error model on n qubits, there will typically be $\Theta(n)$ errors at a given instance, which can be much larger than the minimum distance of a code; e.g. all of our codes have

$d = o(n)$. One may then worry that such a large error will be uncorrectable. However, one can show that these random errors rarely conspire to form a logical operator of the code [KP13, Got13]. The main idea is that since our code is LDPC, there are large regions of physical qubits that do not talk to each other through the parity checks. As a consequence, a typical error configuration at low error rates consists of small, disconnected clusters that are each handled independently by a decoder. To quantify how large a typical error cluster can be, also known as the cluster size distribution, we will utilize some known combinatorial results on graphs with bounded degrees.

Lemma 9.2 (Cluster bound; Lemma 2 in [Got13], Lemma 5 in [AGP08]). *Consider a vertex set V of size t in a simple graph with maximum degree z . Let $N_s(V)$ be the number of vertex sets of size $s \geq t$ whose connected components each contain a vertex in V . Then*

$$N_s(V) \leq e^{t-1}(ze)^{s-t}, \quad (34)$$

where e is the usual base of the natural logarithm.

9.1 State preparation of the 3D HGP code

In this subsection, we will prove that the logical $|\bar{0}\rangle / |\bar{+}\rangle$ state preparation of 3D HGP code \mathcal{Q}_G is fault-tolerant against sufficiently small local-stochastic noise. In particular, assuming that the initial syndrome error is local-stochastic with a sufficiently small but constant parameter $q < 1$, we will show that the residual error after two-stage decoding is also local-stochastic with a small (but possibly worse) parameter $\gamma(q) < 1$ that vanishes with q . As a reminder, to (transversally) prepare the logical $|\bar{0}\rangle$ state, we initialize all physical qubits in $|0\rangle$ and subsequently measure all X -checks.

It will be useful to define the syndrome adjacency graph \mathcal{G}_S as the simple graph where parity checks are vertices, and two vertices share an edge if and only if the corresponding parity checks share a metacheck in \mathbf{M} . If \mathbf{M} is (v_c, v_r) -LDPC, any check participates in at most v_c metachecks, and each of these neighboring metachecks involves at most $v_r - 1$ other checks. Hence, \mathcal{G}_S has maximum degree $v_c(v_r - 1)$. Likewise, define the qubit adjacency graph \mathcal{G}_Q as the simple graph where qubits are vertices, and two vertices share an edge if and only if the corresponding qubits share a parity check in \mathbf{H} . If \mathbf{H} is (w_c, w_r) -LDPC, then \mathcal{G}_Q has maximum degree $w_c(w_r - 1)$.

Let \mathbf{s}_{true} be the true syndrome and \mathbf{s}_{err} the syndrome error such that $\mathbf{s}_{\text{obs}} = \mathbf{s}_{\text{true}} + \mathbf{s}_{\text{err}}$ is the observed, corrupted syndrome. Our decoder for \mathbf{M} , which we will call a metadecoder, will output a syndrome correction \mathbf{s}_{corr} such that $\mathbf{s}_{\text{rep}} = \mathbf{s}_{\text{obs}} + \mathbf{s}_{\text{corr}}$ is the repaired or inferred syndrome. The metadecoder succeeds if $\mathbf{s}_{\text{rep}} \in \text{im } \mathbf{H}$; since $\mathbf{s}_{\text{true}} \in \text{im } \mathbf{H}$ by default, the success requirement is equivalent to $\mathbf{s}_{\text{err}} + \mathbf{s}_{\text{corr}} \equiv \mathbf{s}_{\text{res}} \in \text{im } \mathbf{H}$. We call \mathbf{s}_{res} the *residual* syndrome, since it will end up as the syndrome for the eventual residual error.

First, assuming a certain minimality condition on our metadecoder, we will adapt an argument by Bombín [Bom15b] to show that the residual syndrome inherits the local-stochastic property.

Lemma 9.3 (Residual syndrome is local-stochastic [Bom15b]). *Suppose $\mathbf{M} \in \mathbb{F}_2^{r \times m}$ is (v_c, v_r) -LDPC with $v_c, v_r = O(1)$, and we have access to a minimum-weight decoder for \mathbf{M} . Suppose the original syndrome error was local-stochastic with parameter $q < 1/(2\nu)^2$ with $\nu = ev_c(v_r - 1)$. Then for any $s \in S$,*

$$\mathbb{P}[s \subseteq s_{\text{res}}] \leq \beta^{|s|} \quad , \quad \beta = \frac{2\nu\sqrt{q}}{1 - 2\nu\sqrt{q}}. \quad (35)$$

In other words, the residual syndrome is local-stochastic with parameter β .

Proof. The metadecoder takes in a syndrome error s_{err} and outputs a residual syndrome $s_{\text{res}} \in \ker \mathbf{M}$. Since codewords of \mathbf{M} form connected clusters on the syndrome adjacency graph \mathcal{G}_S , the residual syndrome s_{res} will partition into connected components on \mathcal{G}_S . Since, within each component $c_{\text{err}} \subseteq s_{\text{err}}$, our metadecoder chooses a c_{corr} of minimal weight, we have $|c_{\text{corr}}| \leq |c_{\text{err}}|$ by noting that c_{err} is itself a possible solution for c_{corr} . As a consequence, within the residual syndrome s_{res} , at least half of its support must contain the entire syndrome error. For any connected $c \in \ker \mathbf{M}$, the probability that c is a connected component of the residual syndrome can then be bounded by

$$\mathbb{P}[c \subseteq s_{\text{res}}] \leq \sum_{j=\lceil |c|/2 \rceil}^{|c|} \binom{|c|}{j} q^{|j|} \leq 2^{|c|} q^{|c|/2} = (2\sqrt{q})^{|c|}. \quad (36)$$

Now, the probability that $s \in S$ is a subset of the residual syndrome can be bounded by counting the number of clusters with connected components that contain s in the syndrome adjacency graph \mathcal{G}_S and multiplying this number by (36). Let $\nu = ev_c(v_r - 1)$, i.e. the maximum degree of \mathcal{G}_S times e . The number of such clusters $N_u(s)$ of size $u \geq |s|$ that contain s is upper-bounded by $N_u(s) \leq e^{s-1} \nu^{u-s} \leq \nu^u$ from Lemma 9.2. Hence, we have

$$\mathbb{P}[s \subseteq s_{\text{res}}] \leq \sum_{\substack{c \supseteq s \\ c \in \ker \mathbf{M}}} \mathbb{P}[c \subseteq s_{\text{res}}] \leq \sum_{j=|s|}^m \nu^j (2\sqrt{q})^j = \sum_{j=|s|}^m (2\nu\sqrt{q})^j. \quad (37)$$

When $2\nu\sqrt{q} < 1$, the geometric series in the last expression converges, and we arrive at

$$\mathbb{P}[s \subseteq s_{\text{res}}] \leq \frac{(2\nu\sqrt{q})^{|s|}}{1 - 2\nu\sqrt{q}} \leq \left(\frac{2\nu\sqrt{q}}{1 - 2\nu\sqrt{q}} \right)^{|s|} \equiv \beta^{|s|}, \quad (38)$$

where $\beta \rightarrow 0$ when $q \rightarrow 0$. \square

We will now show that, with high probability, the residual syndrome outputted by the metadecoder is physical, i.e. $s_{\text{res}} \in \text{im } \mathbf{H}$.

Lemma 9.4 (Syndrome repair success). *Suppose $\mathbf{M} \in \mathbb{F}_2^{r \times m}$ is (v_c, v_r) -LDPC with $v_c, v_r = O(1)$, and we have access to a minimum-weight decoder for \mathbf{M} . If the syndrome noise is local-stochastic with parameter $q < 1/(4\nu)^2$ with $\nu = ev_c(v_r - 1)$ and $d_{\text{ss}} = \Omega(m^b)$ for some $b > 0$, then*

$$\mathbb{P}[s_{\text{res}} \in \text{im } \mathbf{H}] = 1 - e^{-\Omega(d_{\text{ss}})}. \quad (39)$$

Proof. A metacheck failure occurs when the metadecoder outputs a repaired syndrome belonging to a nontrivial homology class in $\ker \mathbf{M} / \text{im } \mathbf{H}$. A nontrivial operator in $\ker \mathbf{M} \setminus \text{im } \mathbf{H}$ must contain a connected cluster of size at least d_{ss} on \mathcal{G}_S ; otherwise, each of its connected components is individually in $\ker \mathbf{M}$ and has size less than d_{ss} . By the Union lemma (Lemma 2 in [BPT10]), the entire operator then belongs to $\text{im } \mathbf{H}$ by definition of d_{ss} .

Our minimum-weight metadecoder can only fail when there exists a connected component of the residual syndrome on \mathcal{G}_S of size $u \geq d_{\text{ss}}$. Let $\nu = ev_c(v_r - 1)$. The number of connected clusters N_u of size u can be crudely upper-bounded by $N_u \leq m\nu^u$, since there are m initial vertices to seed the cluster and at most ν^u clusters of size u originating from the seed vertex from Lemma 9.2. The probability of a metacheck failure can then be bounded by

$$\mathbb{P}[s_{\text{res}} \notin \text{im } \mathbf{H}] \leq \sum_{\substack{\text{connected } c \\ |c| \geq d_{\text{ss}}}} \mathbb{P}[c \subseteq s_{\text{res}}] \leq \sum_{u=d_{\text{ss}}}^m m\nu^u (2\sqrt{q})^u = m \sum_{u=d_{\text{ss}}}^m (2\nu\sqrt{q})^u, \quad (40)$$

where in the second inequality we used (36). When $2\nu\sqrt{q} < 1$, the geometric series in the last expression converges, and we arrive at

$$\mathbb{P}[s_{\text{res}} \notin \text{im } \mathbf{H}] \leq m \cdot \frac{(2\nu\sqrt{q})^{d_{\text{ss}}}}{1 - 2\nu\sqrt{q}} \leq m \cdot \left(\frac{2\nu\sqrt{q}}{1 - 2\nu\sqrt{q}} \right)^{d_{\text{ss}}} = e^{-\Omega(d_{\text{ss}})} \quad (41)$$

which is exponentially decaying with d_{ss} when $2\nu\sqrt{q} < 1/2$ and $d_{\text{ss}} = \Omega(m^b)$ for any $b > 0$. \square

At this point, we have shown that the entire residual syndrome is physical with high probability by showing it for each connected component. It remains to show that the residual error corresponding to this residual syndrome is also local-stochastic. To obtain the residual error from the residual syndrome, we will again employ a minimum-weight decoder.

Lemma 9.5 (Residual error is local-stochastic). *Suppose that $H \in \mathbb{F}_2^{m \times n}$ is (w_c, w_r) -LDPC and (t, f) -sound with $t = \Omega(n^b) < d_{\text{ss}}$ and $f(x) = \alpha x$ for $b, \alpha > 0$. Let $\omega = ew_c(w_r - 1)$. Then for $\omega\beta < 1$, with probability $1 - e^{-\Omega(t)}$, for any error $e \in Q$,*

$$\mathbb{P}[e \subseteq e_{\text{res}}] \leq \gamma^{\|e\|}, \quad \gamma = \left(\frac{\omega\beta}{1 - \omega\beta} \right)^{1/\alpha}, \quad (42)$$

where $\gamma \rightarrow 0$ when $\beta \rightarrow 0$.

Proof. Lemma 9.4 tells us that when $\nu\beta < 1$, then with probability $1 - e^{-\Omega(d_{\text{ss}})}$, the residual syndrome $s_{\text{res}} \in \text{im } H$ is the union of connected syndrome clusters on \mathcal{G}_S each of size less than d_{ss} . Note that $t < d_{\text{ss}}$ by the definitions of soundness and single-shot distance. Lemma 2.15 tells us that our linear soundness implies linear confinement up to radius $t_c = t/w_c$. For $t = \Omega(n^b)$ with $b > 0$, the same argument tells us that with probability $1 - e^{-\Omega(t_c)}$, each residual syndrome component is associated with a residual error cluster with size less than the linear confinement radius t_c . Since each residual syndrome component is disjoint by definition, the entire residual error obeys linear confinement. The probability of an error $e \in Q$ being the residual error is bounded by

$$\mathbb{P}[e = e_{\text{res}}] = \mathbb{P}[s(e) = s_{\text{res}}] \leq (2\sqrt{q})^{\|e\|/\alpha}. \quad (43)$$

Now consider the subset event $e \subseteq e_{\text{res}}$ for some $e \in Q$. Linear confinement implies that the residual syndrome must then have weight at least $\|e\|/\alpha$. Observe that if the other errors in $e_{\text{res}} - e$ are disconnected from e in \mathcal{G}_Q , then they would not affect the fact that $s(e) \subseteq s_{\text{res}}$, and we would have $\mathbb{P}[e \subseteq e_{\text{res}}] = \mathbb{P}[s(e) \subseteq s_{\text{res}}]$. However, there could be errors connected to e in \mathcal{G}_Q that could cause some unsatisfied checks in $s(e)$ to become satisfied and thus nullify the condition $s(e) \subseteq s_{\text{res}}$. Nonetheless, due to linear confinement, doing so will spawn new unsatisfied checks. We can upper bound $\mathbb{P}[e \subseteq e_{\text{res}}]$ by counting the number of vertex sets in \mathcal{G}_Q (with maximum degree $w_c(w_r - 1)$) whose connected components contain a vertex in e and multiplying each vertex set by the probability of encountering its syndrome in the residual syndrome. Let $\omega = ew_c(w_r - 1)$. The number of choices of such $c \in Q$ of size ℓ that can cover e in \mathcal{G}_Q is at most $N_\ell(e) \leq e^{\|e\|-1}\omega^{\ell-\|e\|} \leq \omega^\ell$ by Lemma 9.2. The probability that e belongs to the residual error is then bounded by

$$\mathbb{P}[e \subseteq e_{\text{res}}] \leq \sum_{c \supseteq e} \mathbb{P}[s(c) \subseteq s_{\text{res}}] \leq \sum_{\ell \geq \|e\|/\alpha} N_\ell(e) \beta^\ell \leq \sum_{\ell \geq \|e\|/\alpha} (\omega\beta)^\ell. \quad (44)$$

When $\omega\beta < 1$, the geometric series in the last expression converges, and we arrive at

$$\mathbb{P}[e \subseteq e_{\text{res}}] \leq \frac{(\omega\beta)^{\|e\|/\alpha}}{1 - \omega\beta} \leq \left(\frac{\omega\beta}{1 - \omega\beta} \right)^{\|e\|/\alpha} = \gamma^{\|e\|}, \quad (45)$$

where $\gamma \rightarrow 0$ when $\beta \rightarrow 0$. \square

We note that linear soundness is a key ingredient of this lemma, in order to translate from a local-stochastic syndrome to a local-stochastic error. Specifically, we used the exponential suppression from linear confinement to beat the combinatorial growth when counting covering sets. To this end, Corollary 7.4 tells us that our 3D HGP code \mathcal{Q}_G exhibits the required linear soundness for the residual error to be local-stochastic. The final ingredient is to show that this local-stochastic residual error is correctable by our minimum-weight decoder with high probability. For LDPC codes, it has been shown that $d = \Omega(\log n)$ is sufficient for a threshold against sufficiently small local-stochastic noise, using percolation arguments [KP13, Got13]. We will use essentially the same arguments, but adapted to our setting.

Theorem 9.6 (Residual error is correctable). *Suppose $M \in \mathbb{F}_2^{r \times m}$ is (v_c, v_r) -LDPC with $v_c, v_r = O(1)$ and $d_{ss} = m^a$ with $a > 0$. In addition, suppose that $H \in \mathbb{F}_2^{m \times n}$ is (w_c, w_r) -LDPC with $w_c, w_r = O(1)$ and also (t, f) -sound with $t = \Omega(n^b) < d_{ss}$ and $f(x) = \alpha x$ for $b, \alpha > 0$. Assume we also have access to minimum-weight decoders for both M and H . If the syndrome error is local-stochastic with parameter $q < [2\nu(1 + \omega + 16^\alpha \omega^{1+2\alpha})]^{-2}$ for $\nu = ev_c(v_r - 1)$ and $\omega = ew_c(w_r - 1)$, then with probability $1 - e^{-\Omega(t)}$, the residual error e_{res} is correctable with minimum-weight decoding.*

Proof. Lemma 9.4 tells us that our residual syndrome $s_{\text{res}} \in \text{im } H$ with probability $1 - e^{-\Omega(t)}$ when $q < 1/(4\nu)^2$. Lemma 9.5 further tells us that, with probability $1 - e^{-\Omega(t)}$, the residual error is local-stochastic with parameter $\gamma(q)$. Since our decoder chooses a correction \hat{e}_{res} of minimal weight, we have $\|\hat{e}_{\text{res}}\| \leq \|e_{\text{res}}\|$ by minimality. By an analogous calculation to (36), the combined residual error and its correction $\tilde{e}_{\text{res}} = e_{\text{res}} + \hat{e}_{\text{res}}$ is then local-stochastic with parameter $2\sqrt{\gamma}$. By definition of correction, we must have $\tilde{e}_{\text{res}} \in \ker H$, which means that it is either trivial ($\|\tilde{e}_{\text{res}}\| = 0$) or a nontrivial logical operator ($\|\tilde{e}_{\text{res}}\| \geq d$).

Recall that the qubit adjacency graph \mathcal{G}_Q has maximum degree $w_c(w_r - 1)$. By the Union lemma [BPT10], a nontrivial logical operator in $\ker H$ must correspond to a connected cluster of size at least d on \mathcal{G}_Q ; otherwise, one of its connected components will be simultaneously nontrivial and have size less than d , which would contradict the definition of minimum distance. Since we have $d > t_c = \Omega(n^b)$ by the definition of confinement, a connected cluster of size d would also imply a connected cluster of size t_c . Hence the decoding failure probability can be bounded by the probability of encountering a cluster of size t_c or greater. Let $\omega = ew_c(w_r - 1)$. The total number of clusters of size u can be bounded by $N_u \leq n\omega^u$ by Lemma 9.2. At the same time, Lemma 9.5 tells us that the probability of encountering any particular cluster decays exponentially in its size. Hence, we have

$$\mathbb{P}[e_{\text{res}} \text{ uncorrectable}] \leq \sum_{j=t_c}^n n\omega^j (2\sqrt{\gamma})^j \leq n \sum_{j \geq t_c} (2\omega\sqrt{\gamma})^j. \quad (46)$$

When $2\omega\sqrt{\gamma} < 1$, the geometric series in the last expression converges, and we arrive at

$$\mathbb{P}[e_{\text{res}} \text{ uncorrectable}] \leq n \cdot \frac{(2\omega\sqrt{\gamma})^{t_c}}{1 - 2\omega\sqrt{\gamma}} \leq n \cdot \left(\frac{2\omega\sqrt{\gamma}}{1 - 2\omega\sqrt{\gamma}} \right)^{t_c} = e^{-\Omega(t_c)} \quad (47)$$

when $2\omega\sqrt{\gamma} < 1/2$ and $t_c = \Omega(n^b)$ for any $b > 0$. Plugging in the definitions of γ (Lemma 9.5) and β (Lemma 9.3), the previous condition becomes $q < [2\nu(1 + \omega + 16^\alpha \omega^{1+2\alpha})]^{-2}$. \square

As a final remark, we note that the above proof can straightforwardly be adapted to the case where one wishes to add additional (independent) local-stochastic physical noise on top of the residual error: since both distributions are local-stochastic, one can substitute in the combined distribution, which would also be local-stochastic (with a slightly worse parameter).

9.2 Fault tolerance of the entire protocol

We note that the composableity of our gadgets with local-stochastic noise follows a similar analysis to that with adversarial noise; we briefly mention the modifications here. In particular, the residual errors for the state preparation gadgets and single-shot decoder is local-stochastic, which means that each transversal gadget in Figure 4 will only add additional local-stochastic noise to the already existing local-stochastic noise for the next gadget. Assuming that the local-stochastic noise in the separate gadgets are independent, then their composition is also local-stochastic. As long as the composed local-stochastic noise is small enough, then it will be corrected by the single-shot decoder.

References

- [ADCP14] Jonas T Anderson, Guillaume Duclos-Cianci, and David Poulin. Fault-tolerant conversion between the steane and reed-muller quantum codes. *Physical review letters*, 113(8):080501, 2014. [1](#)
- [AGP08] Panos Aliferis, Daniel Gottesman, and John Preskill. Accuracy threshold for post-selected quantum computation. *Quantum Info. Comput.*, 8(3):181–244, March 2008. [9.2](#)
- [Aha03] Dorit Aharonov. A simple proof that toffoli and hadamard are quantum universal. *arXiv preprint quant-ph/0301040*, 2003. [3](#), [3.1](#)
- [BB24] Nikolas P Breuckmann and Simon Burton. Fold-transversal clifford gates for quantum codes. *Quantum*, 8:1372, 2024. [1.2.2](#), [6](#), [6.4](#)
- [BCG⁺24] Sergey Bravyi, Andrew W Cross, Jay M Gambetta, Dmitri Maslov, Patrick Rall, and Theodore J Yoder. High-threshold and low-overhead fault-tolerant quantum memory. *Nature*, 627(8005):778–782, 2024. [1](#), [1.3](#)
- [BCHMD09] Hector Bombin, Ravindra W Chhajlany, M Horodecki, and Miguel-Angel Martin-Delgado. Self-correcting quantum computers. *arXiv preprint arXiv:0907.5228*, 2009. [3](#)
- [BDET24] Nikolas P Breuckmann, Margarita Davydova, Jens N Eberhardt, and Nathanan Tantivasadakarn. Cups and gates i: Cohomology invariants and logical quantum operations. *arXiv preprint arXiv:2410.16250*, 2024. [1](#), [3](#), [3.3](#), [6](#)
- [BE21] Nikolas P Breuckmann and Jens N Eberhardt. Balanced product quantum codes. *IEEE Transactions on Information Theory*, 67(10):6653–6674, 2021. [1](#), [1.3](#)
- [BGH⁺25] Noah Berthelsen, Michael J Gullans, Yifan Hong, Maryam Mudassar, and Shi Jie Samuel Tan. Automorphism gadgets in homological product codes. *arXiv preprint arXiv:2508.04794*, 2025. [1.2.2](#), [3](#), [6.4](#), [6.5](#)
- [BGKT20] Sergey Bravyi, David Gosset, Robert König, and Marco Tomamichel. Quantum advantage with noisy shallow circuits. *Nature Physics*, 16(10):1040–1045, 2020. [1](#)
- [BH12] Sergey Bravyi and Jeongwan Haah. Magic-state distillation with low overhead. *Physical Review A—Atomic, Molecular, and Optical Physics*, 86(5):052329, 2012. [1](#)

- [BH14] Sergey Bravyi and Matthew B Hastings. Homological product codes. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, pages 273–282, 2014. [2.9](#)
- [BK13] Sergey Bravyi and Robert König. Classification of topologically protected gates for local stabilizer codes. *Physical review letters*, 110(17):170503, 2013. [1](#)
- [BKS21] Michael E Beverland, Aleksander Kubica, and Krysta M Svore. Cost of universality: A comparative study of the overhead of state distillation and code switching with color codes. *PRX Quantum*, 2(2):020341, 2021. [1.3](#)
- [BL24] Thiago Bergamaschi and Yunchao Liu. On fault tolerant single-shot logical state preparation and robust long-range entanglement. *arXiv preprint arXiv:2411.04405*, 2024. [1.2.2](#), [5.1](#), [5.2](#), [6](#), [6.1](#), [8.2](#)
- [BMD06] Hector Bombin and Miguel Angel Martin-Delgado. Topological quantum distillation. *Physical review letters*, 97(18):180501, 2006. [1](#)
- [BMD09] Héctor Bombín and Miguel Angel Martin-Delgado. Quantum measurements and gates by code deformation. *Journal of Physics A: Mathematical and Theoretical*, 42(9):095302, 2009. [1](#)
- [Bom11] Hector Bombin. Clifford gates by code deformation. *New Journal of Physics*, 13(4):043005, 2011. [1](#)
- [Bom15a] Héctor Bombín. Gauge color codes: optimal transversal gates and gauge fixing in topological stabilizer codes. *New Journal of Physics*, 17(8):083002, 2015. [1](#)
- [Bom15b] Héctor Bombín. Single-shot fault-tolerant quantum error correction. *Phys. Rev. X*, 5:031043, Sep 2015. [9.1](#), [9.3](#)
- [Bom16] Héctor Bombín. Dimensional jump in quantum error correction. *New Journal of Physics*, 18(4):043038, 2016. [1](#)
- [Bom18] Hector Bombin. Transversal gates and error propagation in 3d topological codes. *arXiv preprint arXiv:1810.09575*, 2018. [1](#)
- [BPT10] Sergey Bravyi, David Poulin, and Barbara Terhal. Tradeoffs for reliable quantum information storage in 2d systems. *Phys. Rev. Lett.*, 104:050503, Feb 2010. [1](#), [9.1](#), [9.1](#)
- [Bro20] Benjamin J Brown. A fault-tolerant non-clifford gate for the surface code in two dimensions. *Science advances*, 6(21):eaay4929, 2020. [1](#)
- [BSS16] Sergey Bravyi, Graeme Smith, and John A Smolin. Trading classical and quantum computational resources. *Physical Review X*, 6(2):021043, 2016. [1.1](#), [1.2.2](#), [6.4](#)
- [BTHG25] Noah Berthelsen, Shi Jie Samuel Tan, Eric Huang, and Daniel Gottesman. Adaptive syndrome extraction. *PRX Quantum*, 6(3):030307, 2025. [1.2](#)
- [Cam19] Earl T Campbell. A theory of single-shot error correction for adversarial noise. *Quantum Science and Technology*, 4(2):025006, 2019. [1.2](#), [1.2.2](#), [2.9](#), [2.13](#), [2.14](#), [2.10](#), [2.16](#), [6.2](#), [8.2](#), [8.5](#), [8.6](#), [8.2](#)

- [CB24] Alexander Cowtan and Simon Burton. Css code surgery as a universal construction. *Quantum*, 8:1344, 2024. [1.1](#)
- [Chi17] Andrew M Childs. Lecture notes on quantum algorithms. *Lecture notes at University of Maryland*, 5, 2017. [3](#), [3.1](#)
- [CHRY24] Andrew Cross, Zhiyang He, Patrick Rall, and Theodore Yoder. Improved qldpc surgery: Logical measurements and bridging codes. *arXiv preprint arXiv:2407.18393*, 2024. [1.1](#)
- [CHWY25] Alexander Cowtan, Zhiyang He, Dominic J Williamson, and Theodore J Yoder. Parallel logical measurements via quantum code surgery. *arXiv preprint arXiv:2503.05003*, 2025. [1.1](#)
- [Cow24] Alexander Cowtan. Ssip: automated surgery with quantum ldpc codes. *arXiv preprint arXiv:2407.09423*, 2024. [1.1](#)
- [Del14] Nicolas Delfosse. Decoding color codes by projection onto surface codes. *Physical Review A—Atomic, Molecular, and Optical Physics*, 89(1):012317, 2014. [1](#)
- [DHLV23] Irit Dinur, Min-Hsiu Hsieh, Ting-Chun Lin, and Thomas Vidick. Good quantum ldpc codes with linear time decoders. In *Proceedings of the 55th annual ACM symposium on theory of computing*, pages 905–918, 2023. [1](#), [2.18](#), [6.2](#), [7](#)
- [DKLP02] Eric Dennis, Alexei Kitaev, Andrew Landahl, and John Preskill. Topological quantum memory. *Journal of Mathematical Physics*, 43(9):4452–4505, 2002. [1](#)
- [DLV24] Irit Dinur, Ting-Chun Lin, and Thomas Vidick. Expansion of high-dimensional cubical complexes: with application to quantum locally testable codes. In *2024 IEEE 65th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 379–385. IEEE, 2024. [2.19](#), [7](#), [7](#)
- [DN21] Nicolas Delfosse and Naomi H Nickerson. Almost-linear time decoding algorithm for topological codes. *Quantum*, 5:595, 2021. [1](#)
- [EK09] Bryan Eastin and Emanuel Knill. Restrictions on transversal encoded quantum gate sets. *Physical review letters*, 102(11):110502, 2009. [1](#), [6](#)
- [FGL18] Omar Fawzi, Antoine Gospellier, and Anthony Leverrier. Efficient decoding of random errors for quantum expander codes. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 521–534, 2018. [1.2](#), [1.2.2](#), [6.2](#)
- [FGL20] Omar Fawzi, Antoine Gospellier, and Anthony Leverrier. Constant overhead quantum fault tolerance with quantum expander codes. *Communications of the ACM*, 64(1):106–114, 2020. [1](#)
- [FH13] Michael H Freedman and Matthew B Hastings. Quantum systems on non- k -hyperfinite complexes: A generalization of classical statistical mechanics on expander graphs. *arXiv preprint arXiv:1301.1363*, 2013. [2.9](#)
- [FMMC12] Austin G Fowler, Matteo Mariantoni, John M Martinis, and Andrew N Cleland. Surface codes: Towards practical large-scale quantum computation. *Physical Review A—Atomic, Molecular, and Optical Physics*, 86(3):032324, 2012. [1](#)

- [FZLL25] Xiaozhen Fu, Han Zheng, Zimu Li, and Zi-Wen Liu. No-go theorems for logical gates on product quantum codes. *arXiv preprint arXiv:2507.16797*, 2025. [1](#)
- [GCZ25] Louis Golowich, Kathleen (Katie) Chang, and Guanyu Zhu. Constant-overhead addressable gates via single-shot code switching. 2025. [1.4](#), [6.4](#)
- [GG24] Louis Golowich and Venkatesan Guruswami. Asymptotically good quantum codes with transversal non-clifford gates. *arXiv preprint arXiv:2408.09254*, 2024. [1](#)
- [GL25] Louis Golowich and Ting-Chun Lin. Quantum ldpc codes with transversal non-clifford gates via products of algebraic codes. In *Proceedings of the 57th Annual ACM Symposium on Theory of Computing*, pages 689–696, 2025. [1](#), [1.2](#), [3](#), [3.3](#), [6](#), [6.5](#)
- [Got13] Daniel Gottesman. Fault-tolerant quantum computation with constant overhead. *arXiv preprint arXiv:1310.2984*, 2013. [1](#), [9](#), [9.2](#), [9.1](#)
- [Got24] Daniel Gottesman. Surviving as a quantum computer in a classical world. *Textbook manuscript preprint*, 2024. [8.1](#), [8.1](#), [8.2](#), [8.3](#), [8.4](#)
- [GTC⁺24] Shouzhen Gu, Eugene Tang, Libor Caha, Shin Ho Choe, Zhiyang He, and Aleksander Kubica. Single-shot decoding of good quantum ldpc codes. *Communications in Mathematical Physics*, 405(3):85, 2024. [1](#)
- [HB23] Oscar Higgott and Nikolas P. Breuckmann. Improved single-shot decoding of higher-dimensional hypergraph-product codes. *PRX Quantum*, 4(2), May 2023. [9](#)
- [HCWY25] Zhiyang He, Alexander Cowtan, Dominic J Williamson, and Theodore J Yoder. Extractors: Qldpc architectures for efficient pauli-based computation. *arXiv preprint arXiv:2503.10390*, 2025. [1.1](#)
- [HDTV24] Timo Hillmann, Guillaume Dauphinais, Ilan Tzitrin, and Michael Vasmer. Single-shot and measurement-based quantum error correction via fault complexes. *arXiv preprint arXiv:2410.12963*, 2024. [1.1](#)
- [HFDVM12] Dominic Horsman, Austin G Fowler, Simon Devitt, and Rodney Van Meter. Surface code quantum computing by lattice surgery. *New Journal of Physics*, 14(12):123011, 2012. [1.1](#)
- [HFWH11] Charles D Hill, Austin G Fowler, David S Wang, and Lloyd CL Hollenberg. Fault-tolerant quantum error correction code conversion. *arXiv preprint arXiv:1112.2417*, 2011. [1](#), [3](#)
- [HGL25] Yifan Hong, Jinkang Guo, and Andrew Lucas. Quantum memory at nonzero temperature in a thermodynamically trivial system. *Nature Communications*, 16(1), January 2025. [6.2](#)
- [HH25] Sascha Heußen and Janine Hilder. Efficient fault-tolerant code switching via one-way transversal cnot gates. *Quantum*, 9:1846, 2025. [1.1](#), [1.2.1](#), [4.1](#), [5.1](#)
- [HJOY23] Shilin Huang, Tomas Jochym-O'Connor, and Theodore J Yoder. Homomorphic logical measurements. *PRX Quantum*, 4(3):030301, 2023. [1.2.1](#), [4.1](#)
- [HL22] Max Hopkins and Ting-Chun Lin. Explicit lower bounds against $\omega(n)$ -rounds of sum-of-squares, 2022. [2.17](#)

- [HLW06] Shlomo Hoory, Nathan Linial, and Avi Wigderson. Expander graphs and their applications. *Bull. Amer. Math. Soc.*, 43(04):439–562, August 2006. [2.2](#)
- [HNP25] Zhiyang He, Quynh T Nguyen, and Christopher A Pattison. Composable quantum fault-tolerance. *arXiv preprint arXiv:2508.08246*, 2025. [1.3](#)
- [Hon24] Yifan Hong. Single-shot preparation of hypergraph product codes via dimension jump. *arXiv preprint arXiv:2410.05171*, 2024. [1](#), [1.1](#), [5.1](#), [5.2](#), [6](#), [6.1](#), [8.2](#), [8.2](#)
- [HVWZ25] Zhiyang He, Vinod Vaikuntanathan, Adam Wills, and Rachel Yun Zhang. Quantum codes with addressable and transversal non-clifford gates. *arXiv preprint arXiv:2502.01864*, 2025. [3](#), [3.3](#)
- [IGND25] Benjamin Ide, Manoj G Gowda, Priya J Nadkarni, and Guillaume Dauphinais. Fault-tolerant logical measurements via homological measurement. *Physical Review X*, 15(2):021088, 2025. [1.1](#)
- [JMB25] Abraham Jacob, Campbell McLaughlan, and Dan E Browne. Single-shot decoding and fault-tolerant gates with trivariate tricycle codes. *arXiv preprint arXiv:2508.08191*, 2025. [1.3](#)
- [JOL14] Tomas Jochym-O’Connor and Raymond Laflamme. Using concatenated quantum codes for universal fault-tolerant quantum gates. *Physical review letters*, 112(1):010505, 2014. [1](#)
- [KB15] Aleksander Kubica and Michael E Beverland. Universal transversal gates with color codes: A simplified approach. *Physical Review A*, 91(3):032330, 2015. [1](#)
- [Kit03] A.Yu. Kitaev. Fault-tolerant quantum computation by anyons. *Annals of Physics*, 303(1):2–30, January 2003. [1](#)
- [KP13] Alexey A. Kovalev and Leonid P. Pryadko. Fault tolerance of quantum low-density parity check codes with sublinear distance scaling. *Phys. Rev. A*, 87:020304, Feb 2013. [9](#), [9.1](#)
- [KP22] Gleb Kalachev and Pavel Panтелейев. Two-sided robustly testable codes. *arXiv preprint arXiv:2206.09973*, 2022. [7](#)
- [KP25] Gleb Kalachev and Pavel Panтелейев. Maximally extendable product codes are good coboundary expanders. *arXiv preprint arXiv:2501.01411*, 2025. [7.1](#)
- [KV22] Aleksander Kubica and Michael Vasmer. Single-shot quantum error correction with the three-dimensional subsystem toric code. *Nature communications*, 13(1):6272, 2022. [1](#)
- [KYP15] Aleksander Kubica, Beni Yoshida, and Fernando Pastawski. Unfolding the color code. *New Journal of Physics*, 17(8):083026, 2015. [1](#)
- [Lin24] Ting-Chun Lin. Transversal non-clifford gates for quantum ldpc codes on sheaves. *arXiv preprint arXiv:2410.14631*, 2024. [1](#), [1.2](#), [1.2.2](#), [2.12](#), [2.12](#), [2.12](#), [2.22](#), [3](#), [3.3](#), [6](#)
- [Lit19] Daniel Litinski. A game of surface codes: Large-scale quantum computing with lattice surgery. *Quantum*, 3:128, 2019. [1](#)

- [LPX25] Christine Li, John Preskill, and Qian Xu. Transversal dimension jump for product qldpc codes. Manuscript in preparation, 2025. [1.4](#)
- [LTZ15] Anthony Leverrier, Jean-Pierre Tillich, and Gilles Zémor. Quantum expander codes. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*, pages 810–824. IEEE, 2015. [2.10](#), [2.11](#)
- [Lub14] Alexander Lubotzky. Ramanujan complexes and high dimensional expanders. *Japanese Journal of Mathematics*, 9(2):137–169, 2014. [2.2](#)
- [LZ22] Anthony Leverrier and Gilles Zémor. Quantum tanner codes. In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 872–883. IEEE, 2022. [1](#)
- [LZ23] Anthony Leverrier and Gilles Zémor. Decoding quantum tanner codes. *IEEE Transactions on Information Theory*, 69(8):5100–5115, 2023. [1](#)
- [LZ25] Anthony Leverrier and Gilles Zémor. Efficient decoding up to a constant fraction of the code length for asymptotically good quantum codes. *ACM Transactions on Algorithms*, 21(4):1–34, 2025. [1](#)
- [MBAM⁺25] Varun Menon, J Pablo Bonilla-Ataides, Rohan Mehta, Daniel Bochen Tan, and Mikhail D Lukin. Magic tricycles: efficient magic state generation with finite block-length quantum ldpc codes. *arXiv preprint arXiv:2508.10714*, 2025. [1.3](#)
- [MC25] Argyris Giannisis Manes and Jahan Claes. Distance-preserving stabilizer measurements in hypergraph product codes. *Quantum*, 9:1618, 2025. [1.2](#)
- [Mou16] Jonathan E Moussa. Transversal clifford gates on folded surface codes. *Physical Review A*, 94(4):042316, 2016. [1](#)
- [Ngu24] Quynh T Nguyen. Good binary quantum codes with transversal ccz gate. *arXiv preprint arXiv:2408.10140*, 2024. [1](#), [3.3](#)
- [NP25] Quynh T Nguyen and Christopher A Pattison. Quantum fault tolerance with constant-space and logarithmic-time overheads. In *Proceedings of the 57th Annual ACM Symposium on Theory of Computing*, pages 730–737, 2025. [1](#), [1.3](#)
- [PK21] Pavel Panteleev and Gleb Kalachev. Quantum ldpc codes with almost linear minimum distance. *IEEE Transactions on Information Theory*, 68(1):213–229, 2021. [1.3](#)
- [PK22] Pavel Panteleev and Gleb Kalachev. Asymptotically good quantum and locally testable classical ldpc codes. In *Proceedings of the 54th annual ACM SIGACT symposium on theory of computing*, pages 375–388, 2022. [1](#)
- [PR13] Adam Paetznick and Ben W Reichardt. Universal fault-tolerant quantum computation with only transversal gates? format?; and error correction. *Physical review letters*, 111(9):090505, 2013. [1](#)
- [QC22] Armando O. Quintavalle and Earl T. Campbell. Reshape: A decoder for hypergraph product codes. *IEEE Transactions on Information Theory*, 68(10):6569–6584, 2022. [2.8](#)

- [QVRC21] Armanda O Quintavalle, Michael Vasmer, Joschka Roffe, and Earl T Campbell. Single-shot error correction of three-dimensional homological product codes. *PRX Quantum*, 2(2):020340, 2021. [1.2](#), [1.2.2](#), [2.9](#), [2.10](#), [2.10](#), [2.10](#), [2.12](#), [2.15](#), [6.2](#), [6.2](#), [8.3](#), [9](#)
- [QWV23] Armanda O. Quintavalle, Paul Webster, and Michael Vasmer. Partitioning qubits in hypergraph product codes to implement logical gates. *Quantum*, 7:1153, 2023. [1.2.2](#), [2.8](#), [6](#), [6.4](#)
- [Shi02] Yaoyun Shi. Both toffoli and controlled-not need little help to do universal quantum computation. *arXiv preprint quant-ph/0205115*, 2002. [3](#), [3.1](#)
- [SJOY24] Esha Swaroop, Tomas Jochym-O'Connor, and Theodore J Yoder. Universal adapters between quantum ldpc codes. *arXiv preprint arXiv:2410.03628*, 2024. [1.1](#)
- [SPW24] Thomas R Scruby, Arthur Pesah, and Mark Webster. Quantum rainbow codes. *arXiv preprint arXiv:2408.13130*, 2024. [3.3](#)
- [SS96] M. Sipser and D.A. Spielman. Expander codes. *IEEE Transactions on Information Theory*, 42(6):1710–1722, 1996. [1](#), [1.2.1](#), [2.4](#), [3](#), [6.5](#)
- [TKY24] Shiro Tamiya, Masato Koashi, and Hayata Yamasaki. Polylog-time-and constant-space-overhead fault-tolerant quantum computation with quantum low-density parity-check codes. *arXiv preprint arXiv:2411.03683*, 2024. [1](#)
- [TPMP24] Shi Jie Samuel Tan, Christopher A Pattison, Matt McEwen, and John Preskill. Resilience of the surface code to error bursts. *arXiv preprint arXiv:2406.18897*, 2024. [1](#)
- [TS24] Shi Jie Samuel Tan and Lev Stambler. Effective distance of higher dimensional hgps and weight-reduced quantum ldpc codes. *arXiv preprint arXiv:2409.02193*, 2024. [1.2](#)
- [TZ14] Jean-Pierre Tillich and Gilles Zemor. Quantum LDPC codes with positive rate and minimum distance proportional to the square root of the blocklength. *IEEE Transactions on Information Theory*, 60(2):1193–1202, 2014. [1](#), [2.8](#)
- [VB19] Michael Vasmer and Dan E Browne. Three-dimensional surface codes: Transversal gates and fault-tolerant architectures. *Physical Review A*, 100(1):012312, 2019. [1](#)
- [WHY24] Adam Wills, Min-Hsiu Hsieh, and Hayata Yamasaki. Constant-overhead magic state distillation. *arXiv preprint arXiv:2408.07764*, 2024. [1](#), [3.3](#)
- [XZB⁺25] Qian Xu, Hengyun Zhou, Dolev Bluvstein, Madelyn Cain, Marcin Kalinowski, John Preskill, Mikhail D. Lukin, and Nishad Maskara. Batched high-rate logical operations for quantum ldpc codes. Manuscript in preparation, 2025. [1.4](#), [6.4](#)
- [XZZ⁺25] Qian Xu, Hengyun Zhou, Guo Zheng, Dolev Bluvstein, J Pablo Bonilla Ataides, Mikhail D Lukin, and Liang Jiang. Fast and parallelizable logical computation with homological product codes. *Physical Review X*, 15(2):021065, 2025. [1.1](#), [1.2](#), [1.2.1](#), [1.2.2](#), [1.2.2](#), [3](#), [3](#), [4.1](#), [4.4](#), [4.6](#), [5.1](#), [6](#), [6.3](#), [6.4](#), [8.5](#)
- [YK24] Hayata Yamasaki and Masato Koashi. Time-efficient constant-space-overhead fault-tolerant quantum computation. *Nature Physics*, 20(2):247–253, 2024. [1](#)

- [YSR⁺25] Theodore J Yoder, Eddie Schoute, Patrick Rall, Emily Pritchett, Jay M Gambetta, Andrew W Cross, Malcolm Carroll, and Michael E Beverland. Tour de gross: A modular quantum computer based on bivariate bicycle codes. *arXiv preprint arXiv:2506.03094*, 2025. [1](#), [3](#)
- [YTC16] Theodore J Yoder, Ryuji Takagi, and Isaac L Chuang. Universal fault-tolerant gates on concatenated stabilizer codes. *Physical Review X*, 6(3):031039, 2016. [1](#)
- [Zhu25a] Guanyu Zhu. A topological theory for qldpc: non-clifford gates and magic state fountain on homological product codes with constant rate and beyond the $N^{1/3}$ distance barrier. *arXiv preprint arXiv:2501.19375*, 2025. [1](#), [1.2](#), [1.2.2](#), [3](#), [3.3](#), [6](#), [6.5](#)
- [Zhu25b] Guanyu Zhu. Transversal non-clifford gates on qldpc codes breaking the \sqrt{N} distance barrier and quantum-inspired geometry with \mathbb{Z}_2 systolic freedom. *arXiv preprint arXiv:2507.15056*, 2025. [1](#), [1.2](#), [1.2.2](#), [3](#), [6](#)
- [ZP19] Weilei Zeng and Leonid P Pryadko. Higher-dimensional quantum hypergraph-product codes with finite rates. *Physical review letters*, 122(23):230501, 2019. [2.9](#), [2.9](#)
- [ZSP⁺23] Guanyu Zhu, Shehryar Sikander, Elia Portnoy, Andrew W Cross, and Benjamin J Brown. Non-clifford and parallelizable fault-tolerant logical gates on constant and almost-constant rate homological quantum ldpc codes via higher symmetries. *arXiv preprint arXiv:2310.16982*, 2023. [1](#), [3.3](#)