Lowering Connectivity Requirements for Bivariate Bicycle Codes Using Morphing Circuits

Shaw, Mackenzie H.; Terhal, Barbara M.

DOI
10.1103/PhysRevLett.134.090602

Publication date
2025

Document Version
Final published version

Published in
Physical review letters

Lowering Connectivity Requirements for Bivariate Bicycle Codes Using Morphing Circuits

Mackenzie H. Shaw and Barbara M. Terhal

QuTech, Delft University of Technology, Lorentzweg 1, 2628 CJ Delft, The Netherlands
and Delft Institute of Applied Mathematics, Delft University of Technology,
Mekelweg 4, 2628 CD Delft, The Netherlands

(Received 21 August 2024; accepted 5 February 2025; published 4 March 2025)

In reference Bravyi *et al.*, [Nature (London) **627**, 778 (2024)], Bravyi *et al.* found examples of bivariate bicycle (BB) codes with similar logical performance to the surface code but with an improved encoding rate. In this work, we generalize a novel parity-check circuit design principle called *morphing circuits* and apply it to BB codes. We define a new family of BB codes whose parity check circuits require a qubit connectivity of degree five instead of six while maintaining their numerical performance. Logical input or output to an ancillary surface code is also possible in a biplanar layout. Finally, we develop a general framework for designing morphing circuits and present a sufficient condition for its applicability to two-block group algebra codes.

DOI: 10.1103/PhysRevLett.134.090602

Introduction—Quantum error correction (QEC) is crucial for achieving fault-tolerant universal quantum computation. One of the most widely studied QEC codes is the surface code [1–3], whose strengths include its planar qubit connectivity and good performance at experimentally achievable error rates. However, since only one logical qubit is encoded in each surface code patch, the qubit overhead becomes extremely high at the low error rates required for practical quantum algorithms.

One alternative approach is to use low-density parity check (LDPC) codes which encode more than one logical qubit per code block, at the expense of no longer having a purely planar connectivity. Recently, Bravyi *et al.* [4] introduced a set of LDPC codes called bivariate bicycle (BB) codes—a subset of the more general two-block group algebra (2BGA) codes [5,6]—that, for the first time, match the logical performance of surface codes even at relatively high physical error rates. As was noted in Ref. [4], physically implementing these BB codes using, for example, superconducting qubits, presents an additional experimental challenge: each qubit performs a CNOT gate with six other qubits in a biplanar layout.

In this work, we simplify the experimental requirements by constructing a set of closely related BB codes whose physical implementation only requires each qubit to interact with five other qubits in a biplanar layout. The new codes are designed using a recently proposed parity check circuit design philosophy that we refer to as *morphing*. Originally called "middle-out" circuits, these morphing circuits have been applied to both surface codes and color codes to reduce the connectivity requirements in those codes [7,8]. We choose the name morphing circuits as the procedure is also related to the concept of *morphing* quantum codes from Ref. [9].

Our contribution is to generalize the idea of morphing circuits to generate parity check circuits for general codes and apply this methodology to the BB codes in Ref. [4]. The procedure takes as input a known code C and outputs a pair of new "end-cycle" codes $\tilde{C}_1$ and $\tilde{C}_2$ along with a pair of parity check circuits. Each parity check circuit measures all of the stabilizer generators of one end-cycle code $\tilde{C}_i$ while simultaneously transforming into the other code $\tilde{C}_{i'}$, see Fig. 1. Moreover, midway through the parity check circuit the joint state of the data and ancilla qubits is encoded in the original, known code C. Despite this, the end-cycle codes may bear little resemblance to the codes from which they are derived—indeed, in our case, the weight of the stabilizers of the end-cycle codes is nine instead of six.

To demonstrate the practicality of our new codes, we investigate their performance against uniform circuit-level depolarizing noise using the BP-OSD decoder [10,11]. We find that the new codes perform at least as well as those in Ref. [4], and therefore provide the same overhead savings versus the surface code as those in Ref. [4]. Moreover, we

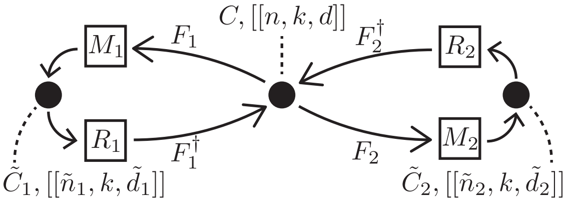
$C, [[n,k,d]]$
M_1
F_1
$F_2^\dagger$
R_2
R_1
$F_1^\dagger$
F_2
M_2
$\tilde{C}_1, [[\tilde{n}_1,k,\tilde{d}_1]]$
$\tilde{C}_2, [[\tilde{n}_2,k,\tilde{d}_2]]$

FIG. 1. Graphical summary of the operation of the morphing protocol, including the known mid-cycle code C, the end-cycle codes $\tilde{C}_i$, the contraction circuits F_i, and the measurement and reset rounds M_i, R_i.

demonstrate that the input and output (I/O) of arbitrary logical qubits from the new codes to the surface code is possible using morphing circuits with a biplanar graph layout. To the best of our knowledge, our I/O construction is the first time morphing circuits have been used to perform a lattice surgery operation.

Mid- and end-cycle codes—We begin by introducing some terminology (following Ref. [7]) for an arbitrary parity check circuit that measures the stabilizer generators of the code. Each round of parity checks begins and ends with a measurement and reset of a set of ancilla qubits. During this time, we say that the remaining data qubits are encoded in the *end-cycle* code. Next, a circuit of Clifford gates is performed. At each step during this circuit, the data and ancilla qubits are also encoded in some QEC code. In particular, we define the *mid-cycle* code as the code that arises precisely midway through the circuit [12]. The stabilizer generators of this mid-cycle code can be determined using the Gottesman-Knill theorem [13] and originate from two sources: the stabilizers of the end-cycle code, and the reset of the ancilla qubits at the start of the QEC cycle. This latter set of stabilizers ensures that the mid-cycle code has the same number of logical qubits as the end-cycle code despite being encoded across both the data and ancilla qubits.

QEC through morphing circuits—The standard approach to designing a parity check circuit is to assume that the end-cycle code corresponds to some specified, "known" code. In contrast, to design a morphing parity check circuit we instead assume that the known code corresponds to the mid-cycle code of the circuit, while the end-cycle code is yet-to-be-determined.

More precisely, the morphing construction is defined by a series of *contraction circuits* F_i. Each contraction circuit F_i must be a Clifford circuit that takes a subset $S_i \subseteq S$ of the stabilizer generators of the known, mid-cycle code C and contracts each generator onto a single qubit. We call the stabilizers in S_i *contracting* stabilizers. The circuit F_i should *not* use any additional qubits since the mid-cycle code C is already encoded across all the data and ancilla qubits. Subsequently, each qubit that hosts a contracted stabilizer is measured in the X- or Z-basis to reveal the eigenvalue of the contracted stabilizer. We label this set of measurements M_i. At this point, the remaining nonmeasured qubits are encoded in the end-cycle code $\tilde{C}_i$, which is a new code determined by propagating the stabilizers of the known code C through the circuit $M_i \circ F_i$. Finally, we restore the mid-cycle code C by first resetting all the measured qubits (labeled R_i) and then applying the inverse circuit $F_i^\dagger$.

A set of contracting circuits F_i defines a valid morphing (parity check) protocol if every stabilizer generator is contained in at least one of the contracting sets S_i, i.e., $\cup_i S_i = S$. When this is the case, we can use the morphing protocol to implement a parity check schedule of the *end-cycle* code $\tilde{C}_i$, that simultaneously transforms the code into $\tilde{C}_{i+1}$ after each QEC round. The end-cycle codes $\tilde{C}_i$ are new and their parameters are $[[\tilde{n}_i, k, \tilde{d}_i]]$ when the known code C has parameters $[[n, k, d]]$. It is guaranteed that $\tilde{n}_i < n$ and, typically, the distance $\tilde{d}_i \le d$; in [14] we give a simple lower bound on $\tilde{d}_i$ given the circuits F_i.

For the codes considered in this Letter, we only need two contracting circuits F_1 and F_2, with each contracting subset S_i containing half of the generators in S. In this case, the parity check schedule from $\tilde{C}_1 \to \tilde{C}_2$ measures all of the stabilizers of $\tilde{C}_1$, and vice versa. In Fig. 1 we summarize the practical operation of such a morphing protocol. The parity check circuit for $\tilde{C}_1$ is given by $M_2 \circ F_2 \circ F_1^\dagger \circ R_1$, after which we are in the code $\tilde{C}_2$. Then, the parity check circuit for $\tilde{C}_2$ is $M_1 \circ F_1 \circ F_2^\dagger \circ R_2$, which returns us back to $\tilde{C}_1$.

Weight-6 Abelian 2BGA codes—We consider weight-6 Abelian 2BGA codes which includes the BB codes studied in Ref. [4]—see [14] for a generalization to all 2BGA codes. Each code is defined by an Abelian group G and two sets of group elements $A = \{a_1, a_2, a_3\}$ and $B = \{b_1, b_2, b_3\}$. The code is defined on $n = 2|G|$ physical qubits labeled $q(L, g)$ or $q(R, g)$ for $g \in G$, with L and R standing for "left" and "right" qubits, respectively. We write $X(P, Q)$ for the product of X operators on the left qubits with labels in the subset $P \subseteq G$ and on right qubits $Q \subseteq G$, and similarly for $Z(P, Q)$ for a Z operator. The stabilizer generators of the code are then given by $s(X, g) = X(Ag, Bg)$ and $s(Z, g) = Z(B^{-1}g, A^{-1}g)$ for $g \in G$, both of which have weight $w = |A| + |B| = 6$. Here, for any subset $H \subseteq G$, we interpret inverse and multiplication elementwise, i.e., $Hg = \{hg | h \in H\}$ and $H^{-1} = \{h^{-1} | h \in H\}$. Multiplying A or B by a group element leaves the code invariant [6]; so without loss of generality we can assume that $a_1 = b_1 = 1$.

In Ref. [4], the authors find a number of examples of weight-6 Abelian 2BGA codes that achieve comparable circuit-level performance to the surface code, listed in Table I. Each code has $G = \mathbb{Z}_\ell \times \mathbb{Z}_m$ for positive integers ℓ, m. The syndrome extraction schedule for this family of codes is highly optimized and requires seven rounds of CNOTs in total during which each qubit interacts with six other qubits. The Tanner graph of the codes—the bipartite graph with qubits and checks as nodes and an edge between a qubit and a check if the check acts on the qubit—is biplanar, meaning the edges can be split into two subsets each of which forms a planar graph.

Applying the morphing protocol—We now show how to construct a pair of contraction circuits F_1 and F_2 that measure all the stabilizers of a given weight-6 Abelian 2BGA code, whenever the code satisfies the following:

Criterion 1—There exists a group homomorphism $f: G \to \mathbb{Z}_2$ with the property that $f(a_1) \neq f(a_2) = f(a_3)$ and $f(b_1) \neq f(b_2) = f(b_3)$.

Under the assumption that $a_1 = b_1 = 1$, Criterion 1 becomes simply $f(a_2) = f(a_3) = f(b_2) = f(b_3) = u$,

TABLE I. Table of BB codes from [4], with the possible choices of homomorphism f_x, f_y, or f_{xy} [Eq. (1)] that satisfy Criterion 1, including the code parameters and circuit-level distance of the corresponding standard parity check schedule. When at least one homomorphism exists that satisfies Criterion 1, we list the code parameters of the end-cycle codes and the circuit-level distance upper bound of the morphing circuits, which are the same for all choices of f and for both end-cycle codes. Note here that n refers to the number of qubits in the BB code from [4], which includes *both* data *and* ancilla qubits when ran as a morphing circuit, but *only* includes data qubits when ran using the circuits from [4].

Code definition				BB code C [4]		End-cycle code $\tilde{C}_i$	
ℓ, m	A	B	f	$[[n, k, d]]$	d_{circ}	$[[\tilde{n}, k, \tilde{d}]]$	$\tilde{d}_{\mathrm{circ}}$
6, 6	$\{x^3, y, y^2\}$	$\{y^3, x, x^2\}$	f_x, f_y, f_{xy}	$[[72, 12, 6]]$	≤ 6	$[[36, 12, 3]]$	≤ 3
9, 6	$\{x^3, y, y^2\}$	$\{y^3, x, x^2\}$	f_y	$[[108, 8, 10]]$	≤ 8	$[[54, 8, 8]]$	≤ 7
12, 6	$\{x^3, y, y^2\}$	$\{y^3, x, x^2\}$	f_x, f_y, f_{xy}	$[[144, 12, 12]]$	≤ 10	$[[72, 12, 6]]$	≤ 6
12, 12	$\{x^3, y^7, y^2\}$	$\{y^3, x, x^2\}$	f_x, f_y, f_{xy}	$[[288, 12, 18]]$	≤ 18	$[[144, 12, 12]]$	≤ 12

where we write $\mathbb{Z}_2 = \{1, u\}$ with $u^2 = 1$. Since any group homomorphism obeys $f(xy) = f(x)f(y)$, f can be uniquely specified by how it acts upon the generators of G. As such, when $G = \mathbb{Z}_\ell \times \mathbb{Z}_m$, there are at most three possible choices of homomorphism that could satisfy Criterion 1, given by

$$f_x(x) = u, \qquad f_x(y) = 1, \qquad \text{if } \ell \equiv 0 \bmod 2, \tag{1a}$$

$$f_y(x) = 1, \qquad f_y(y) = u, \qquad \text{if } m \equiv 0 \bmod 2, \tag{1b}$$

$$f_{xy}(x) = u, \qquad f_{xy}(y) = u, \qquad \text{if } \ell \equiv m \equiv 0 \bmod 2. \tag{1c}$$

Note that the conditions on ℓ and m are necessary to ensure that the function f is a group homomorphism of G.

The BB codes from Ref. [4] are listed in Table I, along with the possible choices of homomorphism in Eq. (1) that satisfy Criterion 1 [21]. When a code satisfies Criterion 1, we define the two cosets $K = \ker f = \{g | f(g) = 1\}$ and $K^c = G \backslash K = \{g | f(g) = u\}$. Moreover, we call qubits and stabilizers "even" if they are labeled by an element $g \in K$ and "odd" if $g \in K^c$.

The morphing circuit is defined through the contracting circuits F_i, measurements M_i, and resets R_i listed in Table II. In particular, F_i consists of three rounds of CNOTs, such that the total CNOT depth of the parity check circuit $F_2 \circ F_1^\dagger$ is six.

TABLE II. Definition of the contracting circuits F_i and measurements M_i, assuming that Criterion 1 is satisfied. CNOT(q_1, q_2) indicates a CNOT gate with control q_1 and target q_2 and M_P represents a measurement in the P basis. The circuit F_1 (respectively, F_2) is defined by applying the gates below for each $g \in K$ $(g \in K^c)$ and $h \in K^c$ $(h \in K)$.

Round 1	CNOT$(q(L, g), q(R, b_3 g))$, CNOT$(q(R, a_3^{-1} h), q(L, h))$
Round 2	CNOT$(q(L, g), q(R, b_2 g))$, CNOT$(q(R, a_2^{-1} h), q(L, h))$
Round 3	CNOT$(q(R, g), q(L, g))$, CNOT$(q(L, h), q(R, h))$
Round 4	$M_X(q(R, g))$, $M_Z(q(R, h))$

The contracting stabilizers in the set S_1 are all even X stabilizers $s(X, g)$ $(g \in K)$ and the odd Z stabilizers $s(Z, g)$ $(g \in K^c)$, while the contracting stabilizers in S_2 are the odd X and even Z stabilizers. Both measurements M_1 and M_2 measure all of the right qubits of C, in the X or Z basis depending on whether the qubit is even or odd. One can see in Fig. 2 that the contracting stabilizers are indeed contracted and measured under the circuit $M_i \circ F_i$, we show this formally in [14].

Connectivity—One advantage of the morphing protocol is that the connectivity graph of the circuits—with vertices for each qubit and edges between qubits that participate in a CNOT—has degree 5, one fewer than the degree of the circuits in Ref. [4]. Indeed, by considering Table II for both circuits F_1 and F_2, we see that the connectivity graph is bipartite between the left and right qubits, with edges $\{q(L, g), q(R, a_i^{-1} b_j g)\}$ for all $g \in G$ and $(i, j) \in \{(1,1), (1,2), (1,3), (2,1), (3,1)\}$. Moreover, we explicitly prove that the connectivity graph is biplanar in [14]. For each of the codes listed in Table I, the standard circuit can be implemented in the "toric$^+$" layout using the four short-range connections of the toric code plus two long-range edges. Meanwhile, the morphing protocol requires the three short-range connections of the hex-grid rotated toric code [7] plus two long-range edges; see [14] for more details.

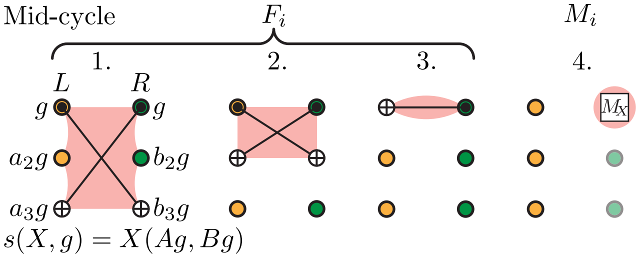

FIG. 2. Visual representation of a contracting X stabilizer, beginning with the mid-cycle support of $s(X, g)$, then, the three steps of the contraction circuit F_i, and the measurement step M_i, as given in Table II. The support of the stabilizer before each step is shown in red.

The end-cycle codes—The stabilizers that are *not* contracting are called *expanding* stabilizers, and these form a set of stabilizer generators for the end-cycle code $\tilde{C}_i$. Each end-cycle code has support only on the left qubits of C, and has stabilizer generators of the form $X(ABg, 0)$ and $Z(A^{-1}B^{-1}h, 0)$, where $g \in K$ and $h \in K^c$ for the code $\tilde{C}_1$ (and vice versa for $\tilde{C}_2$)—see Fig. 4 of [14]. Here we have inherited the multiplication operation from the group algebra representation $\mathbb{Z}_2[G]$ of the subsets A and B; explicitly, the product AB is simply the set $\{a_i b_j\}$ unless some $a_i b_j = a_{i'} b_{j'}$, in which case these two elements are removed from the set. Thus, if each of the products $a_i b_j$ is unique, each end-cycle stabilizer generator has weight nine. Moreover, the stabilizer groups of the codes $\tilde{C}_1$ and $\tilde{C}_2$ are identical up to a shift of the qubits by any element $s \in K^c$. Finally, each end-cycle code can be rewritten as a 2BGA code by identifying the end-cycle-left and end-cycle-right qubits as the even and odd left qubits, as shown in [14].

We summarize the parameters of the end-cycle codes alongside their corresponding mid-cycle BB codes in Table I. The distance of the end-cycle codes was calculated using a linear binary integer program [22], following the methods of Ref. [23]. Perhaps coincidentally, the end-cycle codes often have the same parameters as a different BB code. For example, the end-cycle code derived from the $[[288, 12, 18]]$ BB code has parameters $[[144, 12, 12]]$, presenting a second "gross" code that could be targeted by future experiments.

Circuit-level performance—We have estimated an upper bound of the circuit-level distance d_{circ} of the morphing parity check protocol using the BP-OSD decoder [10,11], following the methods of Ref. [4]. In Fig. 3 we numerically simulated the performance of each $k = 12$ parity check circuit in Table I under a uniform circuit-level depolarizing noise model using the BP-OSD decoder, see [14] for the parameters used and data for more codes. When the code parameters of the BB code and the end-cycle code match, we see that they perform similarly under this circuit-level noise model, demonstrating that the morphing protocol is a viable alternative to the circuits in Ref. [4].

Logical operations—In Ref. [4], Bravyi *et al.* show how to perform the logical input and output (I/O) between an arbitrary logical qubit of the BB code and a surface code ancilla system. This involves operations within the BB code using only already-existing connections, as well as the addition of a separate linking code [24] that preserves the biplanarity of the code. In [14] we show that, under some loose assumptions about the structure of the logical operators, the I/O of arbitrary logical qubits is also possible in a biplanar layout with the morphing protocol. We show how to perform shift automorphisms within the BB code without using additional connections [25]. Moreover, we explain how to perform general lattice-surgery-like operations within the framework of the morphing protocol and apply this to the logical I/O between the BB and linking code.

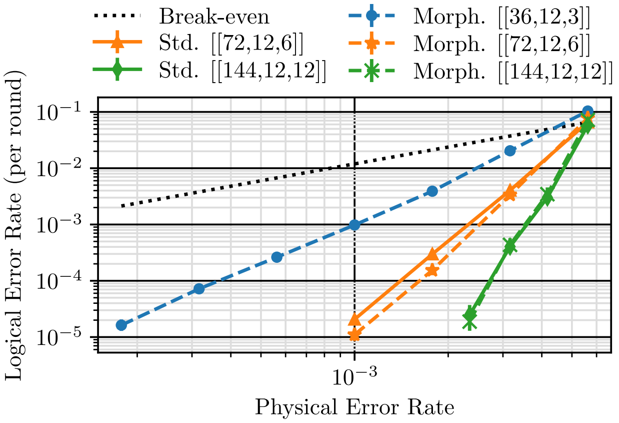

FIG. 3. Numerical logical performance of codes from Ref. [4] under standard parity check circuits (Std.) and the new codes designed from morphing circuits (Morph.) with respect to a uniform circuit-level depolarizing noise model and decoded using BP-OSD; $[[n, k, d]]$ here refers to the *end-cycle* code parameters. The break-even line represents the logical error rate of 12 bare physical qubits.

Modifications to F_i—The morphing protocol defined in Table II is far from unique. For example, one can reverse the direction of some of the CNOTs in Table II and still obtain a valid contraction circuit for the BB code. In [14] we show that a reversal of the CNOTs in round 3 can be used to swap the data and ancilla qubits in each round of QEC without compromising the parameters of the end-cycle codes. This could be used experimentally to mitigate the effects of leakage [7].

A less trivial modification involves reversing the CNOTs in round 2. In [14] we show that the resulting end-cycle codes $\tilde{C}_i$ are *not* equivalent to those derived from Table II, and we identify three circuits that have a larger distance and circuit-level distance upper-bound than the corresponding protocols presented in Table I. However, their numerical performance against circuit-level noise does not improve, as shown in [14]. We leave further investigation to future work.

Outlook—In this work we have developed a general framework to design morphing protocols for arbitrary codes and applied this framework to the BB codes presented in Ref. [4]. Similarly to surface codes and color codes [7,8], these new parity check circuits reduce the degree of the connectivity graph and allow for the swapping of data and ancilla qubits between each round. Moreover, these advantages are achieved without sacrificing the biplanarity of the connectivity graph, numerical performance, or logical capabilities of the original BB codes. An exciting area of future research is therefore to apply the morphing construction to more codes; for example, BB codes that do *not* satisfy Criterion 1, as well as other classes of codes such as hypergraph product codes and higher-dimensional topological codes.

Acknowledgments—This work is supported by QuTech NWO funding 2020-2024—Part I “Fundamental Research,” Project No. 601.QT.001-1, financed by the Dutch Research Council (NWO). B. M. T. thanks the OpenSuperQPlus100 Project (No. 101113946) of the EU Flagship on Quantum Technology (HORIZON- CL4-2022-QUANTUM-01-SGA) for support. We acknowledge the use of the DelftBlue supercomputer for running the decoding. We thank Marc Serra Peralta for the helpful conversations and assistance in running the decoding.

[1] A. Y. Kitaev, Ann. Phys. (N.Y.) **303**, 2 (2003).

[2] S. B. Bravyi and A. Y. Kitaev, arXiv:quant-ph/9811052.

[3] A. Cleland, SciPost Phys. Lect. Notes, 10.21468/scipost-physlectnotes.49 (2022).

[4] S. Bravyi, A. W. Cross, J. M. Gambetta, D. Maslov, P. Rall, and T. J. Yoder, Nature (London) **627**, 778 (2024).

[5] A. A. Kovalev and L. P. Pryadko, Phys. Rev. A **88**, 012311 (2013).

[6] H.-K. Lin and L. P. Pryadko, arXiv:2306.16400.

[7] M. McEwen, D. Bacon, and C. Gidney, Quantum **7**, 1172 (2023).

[8] C. Gidney and C. Jones, arXiv:2312.08813.

[9] M. Vasmer and A. Kubica, PRX Quantum **3**, 030319 (2022).

[10] P. Panteleev and G. Kalachev, Quantum **5**, 585 (2021).

[11] J. Roffe, D. R. White, S. Burton, and E. Campbell, Phys. Rev. Res. **2**, 043423 (2020).

[12] Strictly speaking of course the “mid-way” point of a parity check circuit is only well-defined if the circuit has an even depth—which is satisfied for all the new circuits constructed in this Letter.

[13] S. Aaronson and D. Gottesman, Phys. Rev. A **70**, 052328 (2004).

[14] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevLett.134.090602 for further details and proofs, which includes Refs. [15–20].

[15] https://github.com/Mac-Shaw/morphing_qec_circuits

[16] C. Gidney, Quantum **5**, 497 (2021).

[17] N. P. Breuckmann, C. Vuillot, E. Campbell, A. Krishna, and B. M. Terhal, Quantum Sci. Technol. **2**, 035007 (2017).

[18] Q. Xu, J. P. Bonilla Ataides, C. A. Pattison, N. Raveendran, D. Bluvstein, J. Wurtz, B. Vasić, M. D. Lukin, L. Jiang, and H. Zhou, Nat. Phys. **20**, 1084 (2024).

[19] D. Horsman, A. G. Fowler, S. Devitt, and R. Van Meter, New J. Phys. **14**, 123011 (2012).

[20] C. Vuillot, L. Lao, B. Criger, C. García Almudéver, K. Bertels, and B. M. Terhal, New J. Phys. **21**, 033028 (2019).

[21] Note that different homomorphisms applied to the same mid-cycle BB code can indeed lead to distinct end-cycle codes that are unrelated via the mappings of Ref. [6].

[22] Gurobi Optimization, LLC, Gurobi Optimizer Reference Manual (2024).

[23] A. J. Landahl, J. T. Anderson, and P. R. Rice, arXiv:1108.5738.

[24] L. Z. Cohen, I. H. Kim, S. D. Bartlett, and B. J. Brown, Sci. Adv. **8**, eabn1717 (2022).

[25] We leave investigations into the ZX-duality to future work, since even for the codes of Ref. [4], more work must be done to optimize the gate sequence before it can be implemented in practice.