Measurement-Free Code-Switching Protocol for Low-Overhead Quantum Computation Using Permutation-Invariant Codes

Yingkai Ouyang,[1,*] Yumang Jing,[2] and Gavin K. Brennen[2,3]

[1] *School of Mathematical and Physical Sciences, University of Sheffield, Sheffield S3 7RH, United Kingdom*
[2] *Center for Engineered Quantum Systems, Dept. of Physics and Astronomy, Macquarie University, 2109 NSW, Australia*
[3] *BTQ Technologies, 16-104 555 Burrard Street, Vancouver, British Columbia, Canada V7X 1M8*

(Received 15 January 2025; revised 10 June 2025; accepted 17 October 2025; published 20 November 2025)

Transversal gates on quantum error correction codes have been a promising approach for fault-tolerant quantum computing, but are limited by the Eastin-Knill no-go theorem. Existing solutions such as gate teleportation and magic state distillation are resource-intensive. We present a measurement-free code-switching protocol for universal quantum computation, switching between a stabilizer code for transversal Clifford gates and a permutation-invariant (PI) code for transversal non-Clifford gates that are logical Z rotations for any rational multiple of π. The novel non-Clifford gates enabled by this code-switching protocol provide for a lower gate count implementation of a universal gate set relative to the Clifford + T gate set. To achieve this, we present a protocol for performing controlled-NOT gates between the codes using near-term quantum control operations that employ a catalytic bosonic mode. We also present a new class of PI codes with tunable code distance, supporting transversal non-Clifford gates, and demonstrate their reduced gate count overhead relative to a comparable stabilizer code to stabilizer code-switching scheme.

DOI: 10.1103/p28m-hb6x

I. INTRODUCTION

Transversal gates [1] on quantum error correction (QEC) codes are the cornerstone of many promising approaches to realizing a fault-tolerant quantum computer [2]. Transversal gates have an elegant structure which makes them fault-tolerant since they act on groups of physical qubits in parallel, thus avoiding a catastrophic spread of single errors within a code block. While using transversal gates for fault-tolerant quantum computation is attractive, for any given QEC code, the Eastin-Knill no-go theorem [1] forbids a universal set of transversal gates.

In light of the Eastin-Knill no-go theorem, the path toward realizing fault-tolerant logical non-Clifford gates cannot rely on a single QEC code that implements universal quantum gates transversally. Instead, we need alternative methods. One such approach is gate teleportation [3,4], which consumes magic states. To ensure fault-tolerance, the magic states must have sufficiently high quality, achieved by distilling cleaner magic states from a larger quantity of noisy ones [5–11]. However, the magic state distillation procedure can be costly both in terms of computation time and the number of qubits used [12–14].

Code-switching [15–18] presents a conceptually simple alternative to gate teleportation using distilled magic states for enabling universal fault-tolerant quantum computations. In code-switching, different QEC codes are employed for logical Clifford and non-Clifford gates, which allows all these logical gates to be performed transversally. While code-switching offers a straightforward approach to achieve universal quantum gates fault-tolerantly, it can incur significant measurement overhead [13].

For instance, in code-switching between two stabilizer codes [19–22], one performs multiple fault-tolerant stabilizer measurements to transform one stabilizer code into another, and the number of these measurements increases with the number of errors that the stabilizer codes can correct. Ref. [23] uses two single transversal measurements instead of multiple fault-tolerant stabilizer measurements. Another solution, amenable to surface codes, achieves a fault-tolerant CCZ gate using local transversal gates and code deformations over a time that scales with the size of the qubit array [24].

A recent approach to measurement-free code-switching uses multi-qubit non-Clifford gates that are not transversal [25]. While elegant, compiling such nontransversal

*Contact author: y.ouyang@sheffield.ac.uk

non-Clifford gates via near-term high-fidelity quantum control operations can be a challenge. Hence, there remains the question regarding the possibility of measurement-free code-switching using only transversal gates.

In our paper, we show how to perform high-fidelity quantum computing with a universal gate set using a measurement-free code-switching protocol. In our protocol, we switch between QEC codes that encode a single logical qubit: a stabilizer code performs transversal logical Clifford gates and a permutation-invariant (PI) code that performs transversal logical non-Clifford gates. The types of such transversal non-Clifford gates include logical Z rotations for any rational multiple of π. The novel transversal non-Clifford gates that we introduce allow the implementation of gates in the binary icosahedral group and also an approximate τ_{60} gate [26] to allow the implementation of the most efficient known single-qubit universal gate set [27].

At the core of our proposed code-switching algorithm is (1) a measurement-free state-teleportation protocol that uses only logical controlled-NOT (CNOT) gates, or equivalent gate set, between a stabilizer code and a PI code and (2) our compilation of CNOT gates that act between a stabilizer code and a PI code in terms of near-term quantum control. The requirements of our protocol are modest. First, we need strong, linear coupling between the spins and the mode. In the case of coupling to a quantized cavity mode such as an optical or microwave cavity mode, this translates into the requirement for high cooperativity. Second, we require that the stabilizer code admits transversal implementations of logical Clifford gates. Third, the stabilizer code should be an *even-odd quantum code* meaning its logical zero (one) codeword can be written as a superposition of even (odd)-weight computational basis states. That is, we can write the logical zero as $|0_L = \sum_{\mathbf{x}\in\{0,1\}^n,|\mathbf{x}| \text{ even}} a_{\mathbf{x}}|\mathbf{x}$ and the logical one as $|1_L = \sum_{\mathbf{x}\in\{0,1\}^n,|\mathbf{x}| \text{ odd}} b_{\mathbf{x}}|\mathbf{x}$ for some complex coefficients $a_{\mathbf{x}}, b_{\mathbf{x}}$. As explained in the following, stabilizer codes with the second two properties include two-dimensional (2D) color codes with even stabilizer weights and an odd number of qubits [2,28] such as the Steane code [29] and also Bacon-Shor codes [30–32] that are concatenations of odd length repetition codes. Most of the PI codes we consider here will also satisfy the even-odd code property, though we show that it is possible to switch from an even-odd stabilizer code to a non-even-odd PI code using a nonlinear, dispersive coupling to a mode.

II. CODE SWITCHING VIA SWAPPING

A. Transversal gates on PI codes

PI codes [33–38] allow QEC to be done on quantum states that are invariant under any permutation of the underlying particles. PI codes are attractive for various reasons. First their controllability by global fields could allow for their scalable physical implementations [39] in near-term devices such as trapped ions or ultracold atoms where addressability is challenging due to cross-talk. Second, PI codes can correct deletions, i.e., erasures at unknown locations [40,41], along with insertion errors [42–44], which conventional QEC codes cannot correct.

The potential to implement PI codes for applications such as quantum sensing [41,45] and quantum storage [46] has been explored. However, it was only recently recognized that PI codes can also enable the transversal implementation of logical non-Clifford gates [47,48]. For simplicity, we first focus on two PI codes both of which encode a single logical qubit and have distance three, one on 7 qubits [47,48] and another on 11 qubits [47]. Both of these codes lie within the family of PI codes recently introduced by Aydin *et al.* [48]. For a definition, see Appendix A.

PI quantum states on N qubits are superpositions of the Dicke states $|D_w^N = \binom{N}{w}^{-1/2} \sum_{x_1,\ldots,x_N\in\{0,1\}, x_1+\cdots+x_N=w} |x_1,\ldots,x_N$, of weights $w = 0,\ldots,N$. The Dicke states span the $(N+1)$-dimensional maximum spin ($J = N/2$) angular momentum space of the qubits. The 7-qubit PI code that we consider has support on four Dicke states [48, Example 4] and is a multi-spin analog of the Gross code on a single spin [49, Eq. (14)]. This 7-qubit code has logical codewords

$$\begin{aligned} |0_{\text{pi7}} &:= \sqrt{3/10}|D_0^7 + \sqrt{7/10}|D_5^7 \\ |1_{\text{pi7}} &:= \sqrt{7/10}|D_2^7 - \sqrt{3/10}|D_7^7 , \end{aligned} \tag{1}$$

and has distance three, but note that since the logical code words are superpositions of Dicke states with different parity weights, this is not an even-odd code. An 11-qubit PI code with distance three that was introduced in Ref. [47] has logical codewords given by

$$|0_{\text{pi11}} = \tfrac{1}{4}(\sqrt{5}|D_0^{11} + \sqrt{11}|D_8^{11}) \tag{2}$$

$$|1_{\text{pi11}} = \tfrac{1}{4}(\sqrt{11}|D_3^{11} + \sqrt{5}|D_{11}^{11}). \tag{3}$$

This Kubischta-Teixeira PI code supports a transversal implementation of the logical $T = Z(\pi/4)$ gate, where $Z(\theta) := |0\rangle\langle 0| + |1\rangle\langle 1|e^{i\theta}$ denotes a qubit rotation operator about the Z-axis.

We denote a logical $Z(\theta)$ gate as $\bar{Z}_\Gamma(\theta)$ to denote a code logical operator on a set of physical qubits defined by a subsystem Γ that has the following action on the codespace spanned by logical codewords $|0_\Gamma\rangle$ and $|1_\Gamma\rangle$:

$$\bar{Z}_\Gamma(\theta)(c_0|0_\Gamma\rangle + c_1|1_\Gamma\rangle) = c_0|0_\Gamma\rangle + c_1 e^{i\theta}|1_\Gamma\rangle. \tag{4}$$

Here, we mostly use A to denote a subsystem that uses a stabilizer code and B to denote a subsystem that uses a PI

code, but, in general, we only require the codes in subsystems A and B to be even-odd codes. The 7-qubit PI code admits a logical non-Clifford gate $\bar{Z}_B(4\pi/5) = Z(2\pi/5)^{\otimes 7}$ that is transversal.

Now let us denote the transversal operators on subsystem Γ as $\bar{X}_\Gamma := X^{\otimes|\Gamma|}$, $\bar{Y}_\Gamma := Y^{\otimes|\Gamma|}$, and $\bar{Z}_\Gamma := Z^{\otimes|\Gamma|}$, where X, Y, Z denote qubit Pauli matrices and $|\Gamma|$ is the number of qubits in Γ. Since $\bar{Z}_B\bar{X}_B|0_{\text{pi7}}\rangle = |1_{\text{pi7}}\rangle$ and $\bar{Z}_B\bar{X}_B|1_{\text{pi7}}\rangle = -|0_{\text{pi7}}\rangle$, the logical $-iY$ operator on the 7-qubit PI code is $X_{\text{pi7}} := \bar{Z}_B\bar{X}_B = -i\bar{Y}_B$.

The Pollatsek-Ruskai code is another 7-qubit PI code of distance three which is distinct from the 7-qubit code in (1), because it is an even-odd code [34]. The Pollatsek-Ruskai 7-qubit PI code admits transversal gates in the group 2I, the binary icosahedral group with order $|2\text{I}| = 120$ [50]. For this code, the transversal X and transversal Z operators are also the logical X and Z operators, and the transversal F gate is also the logical F^* gate, where $F = HZ(-\pi/2)$ and H denotes the Hadamard operator. This code can also implement the logical $\Phi^\star$ code with transversal Φ gates, where Φ and $\Phi^\star$ are non-Clifford gates given by

$$\Phi = \frac{1}{2}\begin{pmatrix} \zeta & 1 \\ -1 & \zeta^* \end{pmatrix}, \quad \Phi^\star = \frac{1}{2}\begin{pmatrix} \zeta^\star & 1 \\ -1 & (\zeta^\star)^* \end{pmatrix}, \tag{5}$$

where $\zeta = \varphi + i\varphi^{-1}$, $\zeta^\star = \varphi^\star + i(\varphi^\star)^{-1}$, $\varphi = (1+\sqrt{5})/2$, and $\varphi^\star = (1-\sqrt{5})/2$. Together, the logical X, logical Z, logical F, and logical $\Phi^\star$ gates generate the algebra of a binary icosahedral group 2I. Combined with the τ_{60} gate given by

$$\tau_{60} := \frac{1}{\sqrt{5\varphi+7}}\begin{pmatrix} 2+\varphi & 1-i \\ 1+i & -2-\varphi \end{pmatrix}, \tag{6}$$

this allows us to obtain the most efficient known universal single-qubit gate set [50], providing a factor of ~ 5.9 complexity saving as compared with the usual Clifford + T gate set [51].

Turning our attention to the 11-qubit PI code, we see that $\bar{X}_B$ and $\bar{Z}_B$ implement the logical X and logical Z operators, respectively. Application of the transversal T gate gives us a non-Clifford logical gate $\bar{Z}_B(3\pi/4) = T^{\otimes 11}$. Thus, we can obtain a logical T gate by applying $Z^{\otimes 11}(T^\dagger)^{\otimes 11}$. Hence, we can obtain a logical T gate by applying $Z(3\pi/4)^{\otimes 11}$.

There are also PI codes that can implement logical Z rotations using transversal gates with different angles. These codes have logical codewords

$$\begin{aligned} |0_{b,g}\rangle &:= (\sqrt{2b-g}|D_0^{2b+g}\rangle) + \sqrt{2b+g}|D_{2b}^{2b+g}\rangle))/\sqrt{4b}, \\ |1_{b,g}\rangle &:= (\sqrt{2b-g}|D_{2b+g}^{2b+g}\rangle + \sqrt{2b+g}|D_g^{2b+g}\rangle))/\sqrt{4b}, \end{aligned} \tag{7}$$

where g is a positive integer and $2b \geq g+1$. We call such a code a (b,g) PI code. In Appendix B, we show that is a special case of the Aydin *et al.* code [48] that corrects 1 error if $g \geq 3$ and $2b-g \geq 3$. For this code, the transversal gate $Z(\pi/b)^{\otimes n}$ applies a logical Z-rotation with angle $\pi g/b$ as shown in Appendix C.

When $g = 3$, the (b,g) PI code is equal to the Kubischta-Teixeira PI code on $2b+3$ qubits and with distance 3 [47]. Hence, we can think of the (b,g) PI code as a generalization of the $(2b+3)$-qubit Kubischta-Teixeira PI code. Note that the $(4,3)$ PI code is the 11-qubit PI code. When $b = 2^{r-1}$ for $r \geq 3$, the Kubischta-Teixeira PI code allows one to implement a logical gate in the rth level of the Clifford hierarchy using transversal gates.

For longer-distance codes that support logical $Z(\pi g/b)$ rotations with transversal gates, we introduce (b,g,m) PI codes, which generalize (b,g) PI codes. These codes use $N = 2bm+g$ qubits. In general, for any positive integer m, we define

$$|0_{b,g,m}\rangle = \sum_{k=0}^{m} \sqrt{\binom{m}{k}} \frac{\gamma_{b,g,m,k}}{2^m\sqrt{(2m-1)!!}} |D_{2kb}^N\rangle, \tag{8}$$

and correspondingly $|1_{b,g,m}\rangle = X^{\otimes N}|0_{b,g,m}\rangle$, where for $k = 0,\ldots,m$, we define

$$\gamma_{b,g,m,k} = b^{-m/2} \prod_{i=k+1}^{m} \sqrt{2ib-g} \prod_{j=m-k+1}^{m} \sqrt{2jb+g}. \tag{9}$$

Note that the $(b,g,1)$ PI code is a (b,g) PI code. We prove in the Appendix B that when $g, 2b-g \geq 2t+1$ and $m \geq t$, a (b,g,m) PI code has distance at least $2t+1$, and hence corrects t errors. These (b,g,m) PI codes are equivalent to the code of Aydin *et al.* only when $m = 1$. For $m \geq 2$, the (b,g,m) PI codes form a new family of PI codes.

For $m = 2$, we find generalized (b,g) code of the form

$$\begin{aligned} |0_{b,g,2}\rangle = &\frac{\sqrt{(2b-g)(4b-g)}}{4b\sqrt{3}}|D_0^N\rangle \\ &+ \frac{\sqrt{(4b+g)(4b-g)}}{2b\sqrt{6}}|D_{2b}^N\rangle \\ &+ \frac{\sqrt{(2b+g)(4b+g)}}{4b\sqrt{3}}|D_{4b}^N\rangle. \end{aligned} \tag{10}$$

For $m = 3$, we find generalized (b, g) code of the form

$$|0_{b,g,3}\rangle = \frac{\sqrt{(2b-g)(4b-g)(6b-g)}}{8b\sqrt{15b}}|D_0^N\rangle + \frac{\sqrt{(4b-g)(6b+g)(6b-g)}}{8b\sqrt{5b}}|D_{2b}^N\rangle + \frac{\sqrt{(4b+g)(6b+g)(6b-g)}}{8b\sqrt{5b}}|D_{4b}^N\rangle + \frac{\sqrt{(2b+g)(4b+g)(6b+g)}}{8b\sqrt{15b}}|D_{6b}^N\rangle. \quad (11)$$

For $m = 4$, we find

$$|0_{b,g,4}\rangle = \frac{\sqrt{(2b-g)(4b-g)(6b-g)(8b-g)}}{16b^2\sqrt{105}}|D_0^N\rangle + \frac{\sqrt{(4b-g)(6b-g)(8b+g)(8b-g)}}{8b^2\sqrt{105}}|D_{2b}^N\rangle + \frac{\sqrt{(6b+g)(6b-g)(8b+g)(8b-g)}}{8b\sqrt{70}}|D_{4b}^N\rangle + \frac{\sqrt{(4b+g)(6b+g)(8b+g)(8b-g)}}{8b\sqrt{105}}|D_{6b}^N\rangle + \frac{\sqrt{(2b+g)(4b+g)(6b+g)(8b+g)}}{16b\sqrt{105}}|D_{8b}^N\rangle, \quad (12)$$

and $|1_{b,g,4}\rangle = X^{\otimes N}|0_{b,g,4}\rangle$.

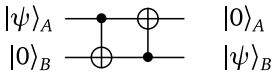
B. Even-odd quantum codes

Many PI codes are even-odd quantum codes. Indeed for g odd, the family of (b, g) PI codes introduced previously and the family of (g, n, u) PI codes described in Ref. [35] are even-odd quantum codes. We now show that many stabilizer codes share this property as well. Bacon-Shor codes [30–32] that are concatenations of odd length codes are even-odd quantum codes. The logical zero and one of a repetition code are $|0\rangle^{\otimes n}$ and $|1\rangle^{\otimes n}$, respectively, which have even and odd weight computational basis states for odd n. Moreover, in the Hadamard basis, the logical zero and one also are superpositions of only even and odd weight computational basis states for odd n. Since the concatenation of an even-odd quantum code with an even-odd quantum code results in an even-odd quantum code, the Bacon-Shor codes that are the concatenation of odd length repetition codes in the standard basis and the Hadamard basis must also be even-odd quantum codes.

The 2D color codes with even stabilizer weights and an odd number of qubits are also even-odd quantum codes. Such codes have stabilizers generated by Pauli operators with only X-type stabilizers and Z-type stabilizers, and admit transversal Clifford gates as their logical gates. Note that we can write the logical zero of such a code to be proportional to the state $\sum_{P\in S} P|0\rangle^{\otimes n}$, where S denotes the stabilizer of the code. The X-type generators have even weights, which implies that the logical zero operator must be a superposition over even weight computational basis states. Since the transversal X gate is the logical X gate which takes the logical zero state to the logical one state and there is an odd number of qubits, the logical one state must be a superposition over odd weight computational basis states. Hence, such 2D color codes are even-odd quantum codes.

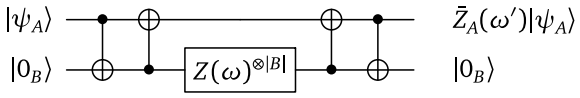
C. High-level code-switching protocol

One measurement-free approach to swap two qubits is to apply three CNOT gates. The following circuit shows how to swap a pair of qubits with two CNOT gates if one of the states is known [4, Eq. (1)]:



(13)

For our code-switching protocol, we treat system A as the stabilizer code on $|A|$ qubits and system B as a PI code on $|B|$ qubits. Based on (13), we can write the logical variant of the swap circuit using two CNOT gates and a known PI ancilla prepared in the $|0_B\rangle$. Using this idea, starting from a stabilizer code where we can perform logical Clifford gates transversally, we switch to a PI code to perform a logical non-Clifford gate transversally before switching back to the stabilizer code. We can achieve this using the following protocol:



(14)

Using the 11-qubit PI code, we can obtain the logical T gate on the stabilizer code with $\omega = 3\pi/4$ and $\omega' = \pi/4$. Using the 7-qubit PI code, we can obtain a logical non-Clifford on the stabilizer code with $\omega = 2\pi/5$ and $\omega' = 4\pi/5$.

Since both of these codes have distance equal to 3, they both allow the correction of any single-qubit error on the logical information.

We can code-switch to PI codes which admit transversal implementation of other exotic non-Clifford gates. Now let g and b be coprime so that there exists an integer k such that $\text{mod}(kg, b) = 1$. Then the (b, g) PI code on $(2b + g)$ qubits can implement the logical $Z(\pi u/b)$ gate transversally on the stabilizer code with $\omega = uk\pi/b$ for any integer u, while allowing the correction of a single error if $g \geq 3$ and $2b - g \geq 3$.

III. GEOMETRIC PHASE GATES

Our proposed mechanism to process quantum information in PI codes uses geometric phase gates (GPGs) arising from coupling quantum spins to a coherently controllable bosonic mode [52,53]. This flavor of GPG is a standard tool for entangling gates in trapped ion quantum processors [54], where the bosonic mode is a quantized motional mode. Other suitable architectures, as outlined in Ref. [55], include trapped atoms in optical cavities [56], superconducting qubits coupled via a driven microwave resonator [57], or polar molecules coupled to microwave stripline cavities [58].

To understand the action of the gate, consider the spin operator $\hat{w}_\Gamma = \frac{|\Gamma|}{2}\mathbf{1} - \hat{J}^z_\Gamma$ that counts the Hamming weight of spin states, where $\hat{J}^z_\Gamma = \frac{1}{2}\sum_{j=1}^{|\Gamma|} Z_{\Gamma,j}$ is the collective angular momentum operator. Here, $Z_{\Gamma,j}$ applies a phase flip on the jth qubit of system Γ and applies the identity operator on all other qubits of Γ. We describe three types of GPGs here. The first two use an interaction between the spins in Γ and the mode that is linear in the creation and annihilation operators of the mode, whereas the third uses a dispersive interaction that is quadratic in bosonic operators:

$$U = \begin{cases} e^{i\phi\hat{w}^2_\Gamma} & \text{Linear GPG-}A\ [52,55] \\ e^{-iI/(\delta-\hat{w}_\Gamma g^2/\Delta)} & \text{Linear GPG-}B\ [55] \\ e^{-i2\chi\,\sin(\theta\hat{w}_\Gamma+\beta)} & \text{Nonlinear GPG [59]} \end{cases} \tag{15}$$

In the linear GPG-A gate [Fig. 1(a)], also known as the Mølmer-Sørensen gate, the angle $\phi \in [0, 2\pi)$ is fully tunable through the control pulses used in the implementation. Recently it has been shown using optimal control methods that this gate together with global rotations of the spins provides for efficient exactly universal state and unitary synthesis in the Dicke state space [60–62]. In the linear GPG-B gate introduced in Ref. [55] the relevant parameters are intensity I, and the detunings δ, Δ which are tunable within a range whereas the coupling strength g between the spins and cavity is usually treated as constant. This gate enables a broader class of entangling gates such as the multi-controlled phase gate $\mathrm{C}_{N-1}(Z)$. For the nonlinear GPG [Fig. 1(b)], the angles $\chi, \theta, \beta \in [0, 2\pi)$ are fully controllable through a combination of dispersive interaction evolutions punctuated by displacement operations on the mode. Complemented by global spin rotations, this gate is also exactly universal for state and unitary synthesis in the Dicke space [39].

IV. PI ANCILLA STATE PREPARATION

Logical state preparation of a target PI state $|\Psi_t\rangle$ can be achieved using the linear GPG-A gate, which we simply denote as l-GPG [61,62]. Starting in the product state $|D^N_0\rangle$,

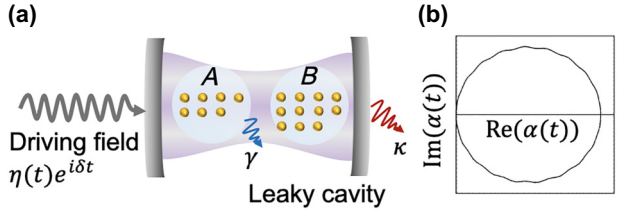


FIG. 1. (a) Two registers of qubits are coupled to a common cavity mode with strength g. The cavity mode decays at a rate κ, whereas the spins decay from their excited states at a rate γ. The cavity acts as a mediator of interactions between the qubits, allowing the two registers to influence one another through their coupling to the same cavity mode. (b) Illustration of a closed trajectory in phase space of a coherent state $\alpha(t)$ of the cavity mode associated to an eigenspace of the spin operator $\hat{w}_\Gamma$ for a linear GPG. After the gate, the mode and the spins are disentangled and the phase accumulated on the eigenspace depends on the area of the trajectory.

one applies a sequence of gates

$$|\Psi\rangle = \prod_{p=1}^{P}[R(\theta_p, \xi_p, \gamma_p)U(\phi_p)]|D^N_0\rangle \tag{16}$$

that prepares an output approximating $|\Psi_t\rangle$, where P denotes the number of pulses. The GPGs $U(\phi_p)$ are interleaved with global rotations around an arbitrary axis as

$$\begin{aligned} R(\theta_p, \xi_p, \gamma_p) &= R_z(\theta_p)R_y(\xi_p)R_z(\gamma_p) \\ &= e^{i\theta_p\hat{J}^z}e^{i\xi_p\hat{J}^y}e^{i\gamma_p\hat{J}^z}. \end{aligned} \tag{17}$$

The parameters θ_p, ξ_p, γ_p, and ϕ_p are chosen numerically to minimize the infidelity $1 - |\langle\Psi|\Psi_t\rangle|^2$. Here, for simplicity, instead of using the definition of the l-GPG above with $U(\phi) = e^{i\phi\hat{w}^2_\Gamma}$, we exploit the decomposition

$$e^{i\phi_p\hat{w}^2_\Gamma} = e^{i\phi_p(\frac{|\Gamma|^2}{4} - |\Gamma|\hat{J}^z_\Gamma + \hat{J}^{z^2}_\Gamma)}, \tag{18}$$

and instead define $U(\phi_p) = e^{i\phi_p\hat{J}^{z^2}_\Gamma}$, since the constant and the $\hat{J}^z_\Gamma$ term in the exponent can be absorbed into the global rotation R during the optimization. This allows us to treat the contribution by evolution generated by $\hat{J}^z_\Gamma$ as part of a uniform rotation without affecting the overall dynamics of the system.

We now discuss the state preparation using l-GPGs in the presence of errors. We assume that the rotations are noise-free, as they can typically be performed fast relative to the spin mode coupling g. We apply the error model presented in Ref. [55] in the presence of losses from cavity mode decay at rate κ and the spontaneous emission of the spin excited state at rate γ. In the Dicke subspace, the ideal

pth l-GPG $U(\phi_p)$ is modified to the erroneous mapping as

$$\mathcal{E}(\phi_p, \rho_{p-1}) = \sum_{n,m=0}^{N} \langle D_n^N|\rho_{p-1}|D_m^N\rangle f_{n,m}(\phi_p)|D_n^N\rangle\langle D_m^N|, \tag{19}$$

where

$$f_{n,m}(\phi_p) = e^{-(m-n)^2 \frac{|\phi_p|}{2}\sqrt{\frac{2(1+2^{-N})}{C}} - (m+n)\frac{|\phi_p|}{2}\frac{1}{\sqrt{2C(1+2^{-N})}}} e^{-i(n^2-m^2)\phi_p} \tag{20}$$

and $C = g^2/\kappa\gamma$ is the cooperativity of the cavity supporting the mode. Here, we apply the absolute value on the parameter ϕ_p as it can take negative values during numerical optimization. Note the map $\mathcal{E}$ is not trace preserving as it treats decay from the excited state as leakage. This provides a lower bound on the process fidelity [55].

The full preparation is a concatenation of erroneous GPGs $\mathcal{E}(\phi_p, \rho_{p-1})$ interleaved with error-free rotations $R(\theta_p, \xi_p, \gamma_p)$, which is written as

$$\mathcal{E}_1\text{-GPG} = \mathcal{E}_1^P\text{-GPG} \circ \mathcal{E}_1^{P-1}\text{-GPG} \circ \cdots \circ \mathcal{E}_1^1\text{-GPG}, \tag{21}$$

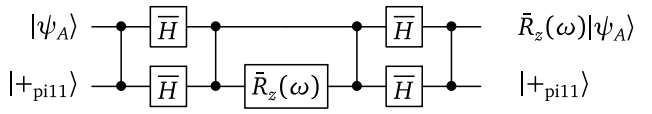
where $\mathcal{E}_1^p\text{-GPG} = R(\theta_p, \xi_p, \gamma_p)\mathcal{E}(\phi_p, \rho_{p-1})R^\dagger(\theta_p, \xi_p, \gamma_p)$. We apply Eq. (20) to the initial state $\rho_0 = |D_0^N\rangle\langle D_0^N|$ and use infidelity $1 - \langle\Psi_t|\rho_P|\Psi_t\rangle$ as the cost function, where ρ_P is the output state after P number of pulses are applied, and $|\Psi_t\rangle$ is the target state. In our optimization, we fix the total number of pulses P and utilize MATLAB's built-in toolbox, which employs the sequential quadratic programming (SQP) method, to minimize the cost function and obtain lists of parameters θ_p, ξ_p, γ_p, and ϕ_p. Note that the optimization method applied is nondeterministic.

V. IMPLEMENTING ENTANGLING LOGICAL GATES

The different GPGs discussed previously enable use of different logical operations for code-switching. The nonlinear GPG can implement logical CNOT gates between a non-even-odd PI code, such as the PI-7 code, and an even-odd stabilizer code (see Appendix G). However, the dispersive interactions between the spins and the cavity mode necessary to instantiate the nonlinear GPG are not accessible directly in many physical systems [39], and the dispersive interaction strengths tend to be smaller than linear interaction strengths implying slower gates. We therefore focus most of our analysis on using linear GPGs.

A. Using linear GPGs with the PI-11 code

Instead of the switching circuit shown in Sec. II C, we discuss a variation of it, with the idea that logical CZ gates are easier to be implemented than CNOT gates. In what follows, we show how to implement the following circuit for code-switching between the Steane code (system A) and the 11-qubit PI code (system B):



(22)

1. Implementing the $C_{A(B)}Z_{B(A)}$ gate

To realize the CZ, which acts symmetrically on the control and target, it suffices to construct a gate that up to global phase on the logical subspace, applies a minus sign on the logical basis state $|1\rangle_A|1\rangle_B$ and acts trivially on the other logical basis states. When both A and B are even-odd quantum codes, as is the case here, then we have the following decomposition in terms of three l-GPGs:

$$CZ = e^{i\frac{\pi}{2}\hat{w}_{A\cup B}^2} e^{-i\frac{\pi}{2}\hat{w}_A^2} e^{-i\frac{\pi}{2}\hat{w}_B^2}. \tag{23}$$

Here an l-GPG involving the operator $\hat{w}_{A\cup B} = \hat{w}_A + \hat{w}_B$ can be obtained by coupling all the spins constituting codes A and B to the same bosonic mode. The process infidelity for performing each of the three l-GPGs in this sequence is [55]

$$1 - F = \frac{\pi|\Gamma|}{2\sqrt{2(1+2^{-|\Gamma|})C}}. \tag{24}$$

.

2. Implementing the logical $\overline{H}$ gate

Next, we discuss a way to implement a logical Hadamard gate $\overline{H}$ for the PI-11 code. In the code space, the eigenvalues of $\overline{H}$ are ± 1 with corresponding eigenstates:

$$\begin{aligned} |\lambda_+\rangle &= \frac{1}{\sqrt{2(2+\sqrt{2})}}[(1+\sqrt{2})|0_{\text{pi11}}\rangle + |1_{\text{pi11}}\rangle], \\ |\lambda_-\rangle &= \frac{1}{\sqrt{2(2-\sqrt{2})}}[(1-\sqrt{2})|0_{\text{pi11}}\rangle + |1_{\text{pi11}}\rangle]. \end{aligned} \tag{25}$$

We can use the method of PI state preparation to construct a unitary on the entire Dicke space which acts correctly in the logical subspace. Using the spectral decomposition, $\overline{H}$ must have the form

$$\overline{H} = |\lambda_+\rangle\langle\lambda_+| - |\lambda_-\rangle\langle\lambda_-| + \sum_s e^{i\beta_s}|\beta_s\rangle\langle\beta_s|, \tag{26}$$

here $\{|\beta_s\rangle\}$ are any set of orthonormal vectors which are contained completely in the orthocomplement to the code space, and $\{\beta_s\}$ are any eigenvalues of on those states.

We can construct this gate with linear GPGs using the method of state preparations adapted for subspaces [63]. Specifically, we can write

$$\overline{H} = We^{i\pi|D_N^N\rangle\langle D_N^N|}W^\dagger \tag{27}$$

where W is any unitary extension of the state preparation: $W|D_N^N\rangle = |\lambda_-\rangle$. The diagonal phase gate $e^{i\pi|D_N^N\rangle\langle D_N^N|}$ is the $\mathrm{C}_{N-1}(Z)$ gate, where $N = 11$ for the PI-11 code. This gate can be synthesized from linear GPG-B gates using the methods in Ref. [55].

We characterize the performance of the implemented Hadamard gate described by Eq. (25) in the presence of losses using the fidelity-based measure $F_{\mathrm{pro}}(\mathcal{E}_{\overline{H}}, \overline{\mathcal{H}})$ proposed in Ref. [64] to compare the performance of actual realized process $\mathcal{E}_{\overline{H}}$ against the ideal unitary process $\overline{\mathcal{H}}(\rho) = \overline{H}_{\mathrm{ide}}\rho\overline{H}_{\mathrm{ide}}^\dagger$, where $\overline{H}_{\mathrm{ide}}$ is the ideal Hadamard gate in logical subspace. The actual process for the implementation of logical Hadamard gate is given by

$$\mathcal{E}_{\overline{H}} = \mathcal{E}_{\mathrm{l}}\text{-GPG} \circ \mathcal{E}_{\mathrm{ph}} \circ \mathcal{E}_{\mathrm{l}}^{\mathrm{R}}\text{-GPG}, \tag{28}$$

where $\mathcal{E}_{\mathrm{ph}}(\rho) = F_{\mathrm{ph}}\mathrm{C}_{N-1}(Z)\rho\mathrm{C}_{N-1}^\dagger(Z)$ is the erroneous map of the middle phase gate in Eq. (25). Here, F_{ph} denotes the fidelity of the multi-qubit phase gate. This multi-qubit phase gate is implemented in the weak-drive regime, where an adiabatic evolution of the joint cavity-qubit system is exploited to realize the operation. The dominant noise arises from cavity decay κ and spontaneous emission from the excited state γ, which are incorporated through the cooperativity C. Numerical simulations reported in Ref. [55, Sec. IV B, Fig. 3(b,c)] show that these error channels lead to an infidelity scaling of approximately $1 - F_{\mathrm{ph}} \sim 1.8 \times N/\sqrt{C}$, which we quote here for completeness. Here $\mathcal{E}_{\mathrm{l}}$-GPG ($\mathcal{E}_{\mathrm{l}}^{\mathrm{R}}$-GPG) is the (reverse of) preparation mapping associated with W ($W^\dagger$) in the presence of losses, as detailed in Eq. (20). Here, for simplicity, we optimize parameters solely for the preparation process and apply the same parameters to the reverse mapping without additional optimization. The fidelity measure $F_{\mathrm{pro}}(\mathcal{E}_{\overline{H}}, \overline{\mathcal{H}})$ is computed based on process matrices, which necessitates the superoperator formalism for each mapping in Eq. (26). The detailed calculations are provided in Appendix F.

In Fig. 2, we plot the infidelity for the implementation error of the Hadamard gate (orange) and the preparation error (blue) for the PI-11 code as a function of the cooperativity C. The simulation is performed in the presence of loss errors as described in Eq. (18). Following the approximation in Ref. [55, Eq. (25)], the infidelity of the $\overline{H}$ implementation can be estimated as $1 - F_{\mathrm{pro}}(\mathcal{E}_{\overline{H}}, \overline{\mathcal{H}}) \sim 21.78 \times N/\sqrt{C}$, where we consider 18 nontrivial GPG operations for the preparation step and its inverse, with $\theta = \pi/2$. Similarly, the infidelity for state preparation is estimated by $1 - \langle +_{\mathrm{pi11}}|\rho|+_{\mathrm{pi11}}\rangle \sim 9.99 \times N/\sqrt{C}$, where nine nontrivial GPG gates are considered, again with $\theta = \pi/2$. Our simulation results indicate a somewhat better performance with the infidelity scaling as $1 - F_{\mathrm{pro}}(\mathcal{E}_{\overline{H}}, \overline{\mathcal{H}}) \sim 8.29 \times N/C^{0.4985}$ and $1 - \langle +_{\mathrm{pi11}}|\rho|+_{\mathrm{pi11}}\rangle \sim 2.80 \times N/C^{0.4953}$, indicating lower infidelity compared with the approximation.

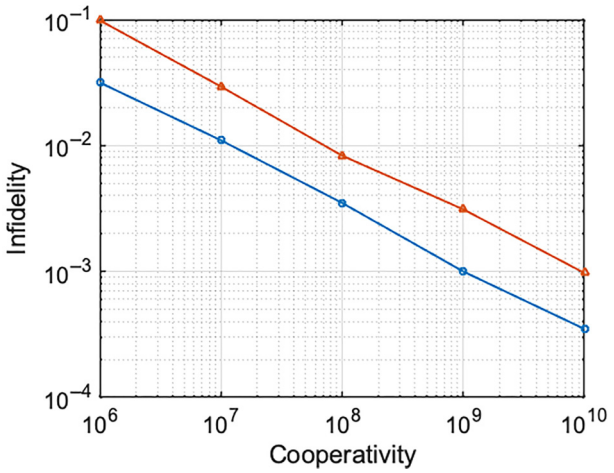


FIG. 2. Process infidelity $1 - F_{\mathrm{pro}}(\mathcal{E}_{\overline{H}}, \overline{\mathcal{H}})$ for implementing the logical Hadamard gate (orange, triangles) on the PI-11 code, and state infidelity $1 - \langle +_{\mathrm{pi11}}|\rho|+_{\mathrm{pi11}}\rangle$ for preparing the logical $|+_{\mathrm{pi11}}\rangle$ state (blue, circles), using l-GPGs as a function of cooperativity C. In both cases $P = 10$ gate sequences are used for the state preparation steps.

VI. APPROXIMATION OF THE SUPERGOLDEN GATE

The most efficient single-qubit gate set is given by representations of the binary icosahedral group $2I$ (which can be implemented with the Pollatsek-Ruskai code) and the supergolden gate τ_{60} given in (6). This efficiency is measured in terms of gate counts. Here we give an approximate decomposition of τ_{60} which can be achieved by code-switching between (b, g) PI code, the Pollatsek-Ruskai code that implements 2I and PI codes that implement the transversal T gate.

Let $S = Z(\pi/2)$, and $T = Z(\pi/4)$. Let H denote the Hadamard gate, and Z denote $Z(\pi)$. Noting that the supergolden gate squares to the identity, we can write it in an Euler decomposition as

$$\tau_{60} = ie^{-i\frac{\pi}{8}Z}e^{-i\frac{\theta}{2}Y}e^{-i\frac{3\pi}{8}Z},$$

where $\theta = 2\cos^{-1}(2+\varphi)/(\sqrt{5\varphi+7})$. We can further simplify

$$\begin{aligned}\tau_{60} &= Te^{-i\frac{\theta}{2}Y}ZT^\dagger \\ &= e^{-i\theta/2}TSHZ(\theta)HS^\dagger ZT^\dagger.\end{aligned}$$

Now θ/π can be approximated by a rational. Defining

$$\begin{aligned}\tilde{\tau}_{60}(\gamma) &= TSHZ(\gamma)HS^\dagger ZT^\dagger \\ &= TF^\dagger Z(\gamma)FZT^\dagger\end{aligned}$$

we find $\|\tilde{\tau}_{60}(\gamma) - \tau_{60}\| < 10^{-6}$ for $\gamma = \pi\, 167/704$, where $\|\cdot\|$ denotes the operator norm minimized over global phases. This could be achieved using a (704 167) PI code for the small angle Z rotation, a smaller (b, g) PI code for transversal T gates, and a PI code such as the Pollatsek-Ruskai code admitting transversal gates in the group 2I (noting $F^2 = F^\dagger$ up to a global phase).

VII. ON THE FAULT-TOLERANCE OF OUR SCHEME

A. Spin system errors

An operator Q is fault-tolerant on the spin-system for QEC codes on system A and B that correct t errors if for any multi-qubit Pauli operator P of weight t, there exists an operator A with weight at most t such that $QP = AQ$. Operationally, this means that when a weight t error P propagates across a quantum circuit Q, the error P evolves in the Heisenberg picture to an error A that does not have too many errors. This definition of fault-tolerance on the spin-system is consistent with the definition of fault-tolerance proposed in Ref. [65]. For us, the 7-qubit and 11-qubit PI codes correct a single error, and hence $t = 1$.

Each constituent gate in our GPGs and conditional GPGs is spin fault-tolerant, which demonstrates the spin fault-tolerance of our protocol. These constituent gates are the rotation operators $R(\theta\hat{J}_\Gamma^z)$ and $R(\theta\hat{J}_\Gamma^y)$ and the mode displacement operator. Mode displacement operators are trivially spin fault-tolerant because they commute with all spin errors.

To demonstrate the spin fault-tolerance of the rotation operators, it suffices to focus our attention on $R(\theta\hat{J}_\Gamma^z)$. The rotation operator $R(\theta\hat{J}_\Gamma^z)$ is trivially spin fault-tolerant with respect to Z errors because it commutes with Z errors on the spin system. From Ref. [66], the rotation operator also obeys the commutation relations $R(\theta\hat{J}_\Gamma^z)X_{\Gamma,k} = e^{i\theta Z_{\Gamma,k}}X_{\Gamma,k}R(\theta\hat{J}_\Gamma^z)$ and $R(\theta\hat{J}_\Gamma^z)Y_{\Gamma,k} = e^{i\theta Z_{\Gamma,k}}Y_{\Gamma,k}R(\theta\hat{J}_\Gamma^z)$ for X and Y errors, respectively. These commutation relations imply that both an X and Y error on the kth qubit evolves under the rotation operator into an error that remains localized on the kth qubit. Hence, the rotation operator $R(\theta\hat{J}_\Gamma^z)$ is indeed spin fault-tolerant.

B. Mode errors

Loss or gain errors can occur on our bosonic mode, which we model using the lowering operator $\hat{a}$ and the raising operator $\hat{a}^\dagger$ respectively. We also consider phase errors on the bosonic mode, which are generated by the number operator $\hat{a}^\dagger\hat{a}$. We say that an operator Q is fault-tolerant on the bosonic system, or Bose fault-tolerant, if for any operator $\bar{B}$ that is either a ladder operator or a number operator, there exists an operator A with weight at most t on the spin systems such that $Q\bar{B} = AQ$.

The rotation operator $R(\theta\hat{w}_\Gamma)$ obeys the commutation relations $R(\theta\hat{w}_\Gamma)\hat{a} = e^{-i\theta\hat{w}_\Gamma}\hat{a}R(\theta\hat{w}_\Gamma)$ and $R(\theta\hat{w}_\Gamma)\hat{a}^\dagger = e^{i\theta\hat{w}_\Gamma}\hat{a}^\dagger R(\theta\hat{w}_\Gamma)$. This shows that ladder operator errors on the bosonic mode can propagate to the spin system as unitaries. Hence the rotation operator is not Bose fault-tolerant with respect to ladder operators, unless θ is very close to an integer multiple of 2π. On the other hand, the rotation operator is trivially Bose fault-tolerant with respect to the number operator.

While the mode displacement operator does not commute with the ladder operators, it does not introduce additional errors to the spin system because ladder operator errors still propagate to ladder operator errors. Hence, the displacement operator is Bose fault-tolerant with respect to ladder operators. The displacement operator is furthermore Bose fault-tolerant with respect to number operators because it propagates the number operator to a linear combination of the number operator and ladder operators. Hence, the displacement operator is Bose fault-tolerant.

To mitigate the effect of the rotation gate not being Bose fault-tolerant in general, we should aim to perform both the displacement gate and the rotation gate as quickly and as accurately as possible, minimizing the number of gain or loss errors. On the other hand, our protocol is fault-tolerant with respect to bosonic phase errors.

The dominant error of our scheme arises from finite cooperativity of the cavity. Recent experimental work [67] demonstrates precision control with a high-finesse Fabry-Pérot cavity which provides for a single-particle cooperativity of $C = 6.75 \times 10^5$ utilizing circular Rydberg transitions coupled to a microwave transition. Higher cooperativities of 10^8–10^9 are achievable [68] using millimeter-wave cavities, though simultaneously achieving optical access is challenging. An alternative to using photonic modes is to use phononic modes, such as a motional mode of an ion trap array. Variational quantum circuits using the same gates as our scheme here have already been demonstrated [69] for the preparation of squeezed quantum states for quantum metrology. For this architecture, the mode is very long-lived and the cooperativity is not the relevant parameter; rather, the error associated with coupling to the motional mode is dominated by laser fluctuation noise. The analysis in Ref. [61] shows that in the presence of laser noise that induces a fluctuation in the phase ϕ of the linear GPG-A by an amount $\delta\phi = 0.01\%$, the full state preparation infidelity of a random state in the Dicke subspace of an $N = 10$ qubit system is below 10^{-6}. If we consider larger codes, such as the $(6, 5, 2)$ PI code, referred to as the PI-29 code and capable of correcting two errors, the same reference reports that the corresponding error can remain below 10^{-4} for $N = 30$.

VIII. PERFORMANCE

To highlight the advantages of our measurement-free code-switching scheme relative to others based on measuring stabilizers, we can compare the control complexity required. For simplicity, we consider switching between two even-odd codes, one code A having N_A qubits and admitting transversal Clifford gates and code B with N_B qubits admitting a transversal non-Clifford gate.

The gate cost, counted by number of transversal gates or cavity pulses, to switch into a PI code B using our method is as follows. State preparation requires at most $2N_B$ global pulses [61]. The logical Hadamards can be done as one transversal gate on the stabilizer code, and on the PI code via (25) $2N_B$ gates for the state preparation unitary W, $2N_B$ gates for $W^\dagger$, and $N_B - 1$ gates for the $\mathrm{C}_{N_B-1}(Z)$ gate [55]. Finally the logical CZ gate is done with 3 cavity pulses and there are two of them. This gives a total count of $7N_B + 6$ gates for switching into the PI code. The total round-trip switching cost, with the transversal implementation of the non-Clifford gate is then at most $14N_B + 13$. Two noteworthy points are that the gate count is independent of N_A and the native nonlocality of the cavity-mediated entangling gates means no qubit swapping or shuttling is needed at any point. In addition, only minimal addressability is required; namely it is only needed to distinguish qubits in one code versus the other.

The cost to switch from one stabilizer code A with some transversal Clifford gates to another stabilizer code B with a transversal non-Clifford gate can be fairly compared with our scheme by considering a non-fault-tolerant set of stabilizer measurements: first on the stabilizers for B to prepare a logical state, and second for parity checks on $A \cup B$. Following the construction of Ref. [70], we choose for code A a doubly even CSS code that includes the Hadamard among its set of transversal logical gates and for code B a triorthogonal code that supports a transversal logical T gate. The double-even CSS codes in Ref. [70] are even-odd codes, because the classical codes used in the CSS construction have even weights, and any doubly even CSS code admits a strongly transversal logical X and has odd length.

We analyze the gate complexity of code-switching from code A to code B, where code B is either a triorthogonal code of distance d or our $(4, 3, (d-1)/2)$ PI code. Note that such PI codes support a transversal logical T gate and are furthermore even-odd codes because the parameter g in the (b, g, m) PI code is odd. We supply a lower bound and an upper bound for the gate count of using a triorthogonal code and a PI code for code B, respectively. For the lower bound, the gate count is at least $w(N_B - 1)$ considering non-fault-tolerant stabilizer measurements, where w is the minimum weight of the triorthogonal code's checks. Table I lists the shortest triorthogonal codes for a given distance $d = 3, 5, 7, 9, 11$. For these triorthogonal codes, we prove that $w \geq 8$. For $d = 3, 5, 7$, the triothogonal codes here are triply even, and hence $w \geq 8$. For the higher distance codes here, the average weight of the triorthogonal code of distance d is at least the minimum of the weights of the triorthogonal codes here of distance $d-2$, the length of the triorthogonal code of distance $d-2$, and twice the distance of the self-dual classical code used in the doubling construction of Ref. [70]. By induction, this minimum is at least eight for all odd $d \geq 9$. Therefore, the gate count of using triorthogonal codes according to the doubling construction of Ref. [70] in Table I is at least $8(N_B - 1)$ considering non-fault-tolerant stabilizer measurements. The simple lower bounds we supply apply not only to both measurement-based and measurement-free code-switching, but also to magic-state distillation techniques. The gate count would be even higher if we consider (1) spatial locality constraints on the implementation for instance in two dimensions, which would impose an additional $O(\sqrt{N_B})$ overhead, and (2) the fact that the weights of the stabilizer checks grow as $O(\sqrt{N_B})$. The gate count for our scheme, on the other hand, is upper bounded by $7N_B + 6$. A key benefit of PI codes therefore is most pronounced when their lengths are considerably shorter than their corresponding triorthogonal codes. This is evidenced by the inequality $7N_{\mathrm{PI},B} + 6 < 8(N_{\mathrm{tri},B} - 1)$, for all length $N_{\mathrm{tri},B}$ triorthogonal and length $N_{\mathrm{PI},B}$ PI codes of distances $d = 3, 5, 7, 9, 11$ involved.

IX. CONCLUSION

We have presented a method for code-switching between a stabilizer code and a permutation invariant code, and along the way have introduced a family of PI codes with a tunable set of transversal, non-Clifford gates. It is noteworthy that our measurement-free code-switching scheme is also applicable to transitions between two stabilizer codes, provided they satisfy the even-odd quantum code structure. In comparison with the measurement-based approach, our scheme eliminates the overhead associated with measurements and reduces the gate operation complexity by utilizing collective gate implementations. Given the recent development on how QEC on arbitrary PI codes can be implemented using GPGs [41], we expect it to be possible to extend our work to a fully fault-tolerant setting, using ideas from fault-tolerance theory [65,71–73]. In addition, a numerical study of the efficiency of our protocol for fault-tolerant quantum computation is an interesting line of future line of enquiry that is currently beyond the scope of our paper.

ACKNOWLEDGMENTS

Y.O. acknowledges support from EPSRC (Grant No. EP/W028115/1) and also the EPSRC funded QCI3 Hub under Grant No. EP/Z53318X/1. G.K.B. and Y.J. acknowledge support from the Australian Research Council Centre

TABLE I. Comparison of gate costs for switching from code A to code B, where code B is either a triply even stabilizer code or a PI code. The stabilizer codes of type A and B are doubly even and triorthogonal codes given in [70, Tabled 1 and 2], respectively. We calculate a gate lower bound and upper bound for switching from code A to stabilizer code B and a PI-$(4,3,m)$ code B, respectively. Here, code A supports transversal logical Clifford gates and code B supports transversal logical T gates. All codes involved here encode a single logical qubit.

Distance d	Stab. Code A (Clifford)	Stab. Code B	PI Code B	Gate Count ($A \to B$)
3	[[7, 1, 3]]	[[15, 1, 3]]	PI-(4,3,1)=((11,2,3))	$\geq 112, \leq 83$
5	[[17, 1, 5]]	[[49, 1, 5]]	PI-(4,3,2) = ((19,2,5))	$\geq 384, \leq 139$
7	[[23, 1, 7]]	[[95, 1, 7]]	PI-(4,3,3) = ((27,2,7))	$\geq 752, \leq 195$
9	[[45, 1, 9]]	[[185, 1, 9]]	PI-(4,3,4) = ((35,2,9))	$\geq 1472, \leq 251$
11	[[47, 1, 11]]	[[279, 1, 11]]	PI-(4,3,5) = ((43,2,11))	$\geq 2224, \leq 307$

of Excellence for Engineered Quantum Systems (Grant No. CE 170100009).

DATA AVAILABILITY

No data were created or analyzed in this study.

APPENDIX A: AAB+ CODE

The AAB+ type code is specified by three integer parameters g, m, and δ, with logical codewords given by

$$|0_{(g,m,\delta,+)}\rangle = \sum_{l \text{ even}} \gamma b_l |D^n_{gl}\rangle + \sum_{l \text{ odd}} \gamma b_l |D^n_{n-gl}\rangle, \tag{A1}$$

$$|1_{(g,m,\delta,+)}\rangle = \sum_{l \text{ even}} \gamma b_l |D^n_{n-gl}\rangle + \sum_{l \text{ odd}} \gamma b_l |D^n_{gl}\rangle, \tag{A2}$$

where $b_l = \sqrt{\binom{m}{l}/\binom{n/g-l}{m+1}}$, $\gamma = \sqrt{\binom{n/(2g)}{m}\frac{n-2gm}{g(m+1)}}$, and $n = 2gm + \delta + 1$. The code corrects t errors when $g \geq 2t+1$, $m \geq t$, and $\delta \geq 2t$.

In Appendix B, we show that the (b,g) PI code is the $(g, 1, 2b-g-1)$ AAB+ code. From the property of the AAB+ code, the (b,g) PI code has distance 3 if $g \geq 3$ and $2b - g \geq 2$.

Next we turn our attention to the transversal non-Clifford gate. We consider the gate $T_b = Z(\pi/b)$. Then we see that

$$T_b^{\otimes n}|0_{b,g}\rangle = |0_{b,g}\rangle, \tag{A3}$$

$$T_b^{\otimes n}|1_{b,g}\rangle = e^{ig\pi/b}|1_{b,g}\rangle. \tag{A4}$$

From this, we see that the transversal gate $T_b^{\otimes n}$ gate applies a logical Z-rotation with angle $\pi g/b$.

APPENDIX B: PROOF THAT THE (b,g) PI CODE IS AN AAB+ CODE

To have the (b,g) PI code, the parameter m in the AAB+ type code must be equal to 1, because each logical codeword is a superposition of two Dicke states. By setting $m = 1$, we have

$$b_0 = 1/\sqrt{\binom{n/g}{2}} = g/\sqrt{n(n-g)/2}, \tag{B1}$$

$$b_1 = 1/\sqrt{\binom{n/g-1}{2}} = g/\sqrt{(n-g)(n-2g)/2}, \tag{B2}$$

$$\gamma = \sqrt{\frac{n}{2g}\frac{\delta+1}{2g}} = \sqrt{n(\delta+1)}/(2g). \tag{B3}$$

Then we have

$$\gamma b_0 = \sqrt{n(\delta+1)}/\sqrt{2n(n-g)} = \sqrt{\frac{\delta+1}{2(n-g)}}, \tag{B4}$$

$$\gamma b_1 = \sqrt{n(\delta+1)}/\sqrt{2(n-g)(n-2g)} = \sqrt{\frac{n(\delta+1)}{2(n-g)(n-2g)}}. \tag{B5}$$

Now consider having $n = 2b + g$ for some positive integer b so that $\delta = 2b - g - 1$, $n - g = 2b, n - 2g = 2b - g$. Then

$$\gamma b_0 = \sqrt{\frac{2b-g}{4b}} = \sqrt{\frac{1}{2} - \frac{g}{4b}}, \tag{B6}$$

$$\gamma b_1 = \sqrt{\frac{n(2b-g)}{4b(2b-g)}} = \sqrt{\frac{2b+g}{4b}} = \sqrt{\tfrac{1}{2} + \frac{g}{4b}}. \tag{B7}$$

This shows that the (b,g) PI codes are just $(g, 1, 2b-g-1)$-AAB+ codes. The condition for (b,g) PI codes to correct a single error is that $g \geq 3$ and $2b - g \geq 3$. When $g = 3, m = 1, \delta = 2^r - 4$, the AAB+ code can implement the logical gate for $T^{2^{3-r}}$ transversally for $r \geq 3$ [48].

APPENDIX C: TRANSVERSAL NON-CLIFFORD LOGICAL OPERATIONS

Our code supports transversal X as the logical X. If n is odd (which is when g is odd), then the transversal Z is the logical Z. If n is even (when g is even), the transversal Z gate stabilizes the code.

Whenever b and g are coprime, we can implement the logical $\pi k/b$ rotation transversally for any $k = 0, \ldots, b-1$. For the (b, g) PI code to correct 1 error, it suffices to have $2b - g \geq 3$ and $g \geq 3$. When b and g are coprime, we can implement a logical T_b^k gate transversally for any $k = 0, \ldots, b-1$ by setting g as the smallest coprime number to a fixed b.

APPENDIX D: CONSTRUCTION OF CODES FROM LINEAR PROGRAMMING

Let us consider the problem of constructing quantum codes encoding a single logical qubit that correct against a set of Kraus operators $\{K_i\}$ given the promise that the orthogonality of the Knill-Laflamme QEC conditions are satisfied.

Now suppose that the codespace resides in a subspace of the space spanned by the orthonormal vectors $\{|c_j\rangle\}$, and we write the logical codewords as linear combinations of these orthonormal vectors. Specifically, we demand that the logical codewords are linear combinations of distinct basis states given by

$$|0_L\rangle = \sum_{j \text{ even}} \sqrt{x_j}|c_j\rangle \tag{D1}$$

$$|1_L\rangle = \sum_{j \text{ odd}} \sqrt{x_j}|c_j\rangle, \tag{D2}$$

where the coefficients x_j are real and nonnegative.

Such a method has been described in [38], and we explain in the next appendix how to apply this idea explicitly to construct extended (b, g) code that correct errors of increasing distance, and hence ensure the scalability of the codes.

APPENDIX E: (b, g, m) PI CODES

Here, we prove that (b, g, m) PI codes have distance $d \geq 2t + 1$ whenever $g, 2b - g \geq 2t + 1$ and when $m \geq t$. Clearly, when $g, |2b - g| \geq d$, the code has a bit-flip distance of d. Hence, for the code to have a distance of d, it suffices to ascertain that the nondeformation conditions of the Knill-Laflamme conditions holds for the diagonal errors, that is,

$$\langle 0_{b,g,m}|P_j|0_{b,g,m}\rangle = \langle 1_{b,g,m}|P_j|1_{b,g,m}\rangle, \tag{E1}$$

for all $k = 0, \ldots, d-1$, where $P_j = Z^{\otimes j} \otimes I^{\otimes N-j}$. Now we consider a code with logical codewords

$$\begin{aligned} |0_{b,g,m}\rangle &= \sum_{k=0}^{m} a_k |D^N_{2bk}\rangle, \\ |1_{b,g,m}\rangle &= \sum_{k=0}^{m} b_k |D^N_{2bk+g}\rangle, \end{aligned} \tag{E2}$$

for real coefficients a_k, b_k, with the idea of solving for a_k and b_k using linear algebra techniques, akin to the idea in Refs. [38,74]. Then we can write the nondeformation conditions as

$$\sum_{0\leq k\leq m} a_k^2 \langle D^N_{2kb}|P_k|D^N_{2kb}\rangle = \sum_{0\leq k\leq m} b_k^2 \langle D^N_{g+2kb}|P_k|D^N_{g+2kb}\rangle, \tag{E3}$$

which is a system of linear equations in the variables a_k^2 and b_k^2. The Dicke inner products $\langle D^N_w|P_k|D^N_w\rangle$ can be written as Krawtchouk polynomials [45, Lemma 6], namely

$$\langle D^N_w|P_k|D^N_w\rangle = K^N_w(k)/\binom{N}{w}, \tag{E4}$$

where

$$K^N_k(z) = \sum_{j=0}^{z} \binom{z}{j}\binom{N-z}{k-j}(-1)^j \tag{E5}$$

is a binary Krawtchouk polynomial. In the language of generating functions,

$$K^N_k(z) = [x^k](1-x)^z(1+x)^{N-z}, \tag{E6}$$

where $[x^k]f(x)$ denotes the coefficient of x^k of a polynomial $f(x)$. From the above generating function, we can also see that

$$K^N_k(z) = [y^{N-k}](y-1)^z(y+1)^{N-z}, \tag{E7}$$

and hence we have the symmetry relation

$$\begin{aligned} K^N_{N-k}(z) &= [x^{N-k}](1-x)^z(1+x)^{N-z} \\ &= [y^{N-k}](-1)^z(y-1)^z(y+1)^{N-z} \\ &= (-1)^z[y^k](y-1)^z(y+1)^{N-z} \\ &= (-1)^z K^N_k(z). \end{aligned} \tag{E8}$$

We can again rewrite this as a system of homogeneous linear equations, $A\mathbf{x} = 0$, where

$$\begin{aligned} A = &\sum_{j=0}^{d-1} \sum_{0\le k\le m} |j\rangle\langle 2k| \langle D^N_{2kb}|P_j|D^N_{2kb}\rangle \\ &+ \sum_{j=0}^{d-1} \sum_{0\le k\le m} |j\rangle\langle 2k+1| \langle D^N_{g+2kb}|P_j|D^N_{g+2kb}\rangle, \\ = &\sum_{j=0}^{d-1} \sum_{0\le k\le m} |j\rangle\langle 2k| K^N_{2kb}(j)/\binom{N}{2kb} \\ &+ \sum_{j=0}^{d-1} \sum_{0\le k\le m} |j\rangle\langle 2k+1| K^N_{g+2kb}(j)/\binom{N}{g+2kb}, \end{aligned} \tag{E9}$$

and where $\mathbf{x} = (x_0, \ldots, x_{2m+1})$. Now define the length $2(m+1)$ vector of ones as $\mathbf{e}$. Define the length $2(m+1)$ vector $\mathbf{f} = (f_0, \ldots, f_{2m+1})$ to be such that $f_j = (-1)^j$ for all $j = 0, \ldots, m$. The equation (E9) shows that we can find the coefficients a_k by solving the linear program

$$\begin{aligned} \underset{\mathbf{x}}{\text{maximize}} \quad & 0 \\ \text{subject to} \quad & A\mathbf{x} = 0 \\ & \mathbf{f}^T\mathbf{x} = 2 \\ & (-1)^k x_k \ge 0, \quad k = 0, \ldots, 2m+1 \end{aligned} \tag{E10}$$

Exploiting the structure of the matrix A, we can derive the form of the vector $\mathbf{x}$ that satisfies the constraints in the above linear program. Namely, when $(-1)^k x_k \ge 0$, we can set the coefficients of the logical codewords of the generalized (b, g) code as

$$a_k = \sqrt{x_{2k}} \quad k \text{ even}, \tag{E11}$$

$$b_k = \sqrt{-x_{2k+1}} \quad k \text{ odd}. \tag{E12}$$

We choose m as the smallest integer for which (E10) has a solution.

Next, let us simplify the linear program (E10). Consider a vector $\mathbf{y} = (y_0, y_1, \ldots, y_{2(m+1)-1})$ such that

$$y_k = -y_{2m+1-k} \tag{E13}$$

for all $k = 0, \ldots, 2m+1$. Then we can write $\mathbf{y}$ as a vector that depends on $(m+1)$ variables, in the sense that

$$\mathbf{y} = (y_0, -y_m, y_1, -y_{m-1}, \ldots, y_m, -y_0). \tag{E14}$$

For any vector $\mathbf{a}$ of the form

$$\mathbf{a} = (a_0, a_m, a_1, a_{m-1}, \ldots, a_m, a_0), \tag{E15}$$

it follows that

$$\begin{aligned} \mathbf{a}^T\mathbf{y} = &y_0a_0 - y_ma_m \\ &+ y_1a_1 - y_{m-1}a_m - 1 \\ &+ \cdots + + y_ma_m - y_0a_0 \\ = &0. \end{aligned} \tag{E16}$$

From (E8) and $\binom{N}{g+2bk} = \binom{N}{2bk}$, every odd row of A has the form of (E15). Hence, every odd row is orthogonal to $\mathbf{y}$, and it suffices to find a $\mathbf{y}$ that is orthogonal to every even row in A.

For any vector $\mathbf{b}$ of the form

$$\mathbf{b} = (b_0, -b_m, b_1, -b_{m-1}, \ldots, b_m, -b_0), \tag{E17}$$

it follows that

$$\begin{aligned} \mathbf{b}^T\mathbf{y} = &y_0b_0 + y_mb_m \\ &+ y_1b_1 + y_{m-1}b_m - 1 \\ &+ \cdots + \\ &+ y_mb_m + y_0b_0, \end{aligned} \tag{E18}$$

and the constraint that $\mathbf{b}^T\mathbf{y} = 0$ is then equivalent to showing that

$$\sum_{k=0}^{m} y_kb_k = 0. \tag{E19}$$

Hence, we consider a submatrix of A given by

$$B = \sum_{\substack{0\le j\le d-1 \\ j \text{ odd}}} \sum_{k=0}^{m} K^N_{2bk}(j)/\binom{N}{2bk} |j\rangle\langle k|. \tag{E20}$$

Then, any $(y_0, \ldots, y_m)$ in the nullspace of B gives a vector $\mathbf{y} = (y_0, -y_m, y_1, -y_{m-1}, \ldots, y_m, -y_0)$ that is also in the nullspace of A. To construct a valid generalized (b, g) PI code, it suffices to require that $(y_0, \ldots, y_m)$ is a nonnegative vector, that is, $(y_0, \ldots, y_m) \ge 0$.

When $d = 2t+1$ and $m = t$, the matrix B has $t+1$ columns and t rows. Then, B is guaranteed to have a nontrivial nullspace. Numerically, in all of the examples we explored, we find that B is a full-rank matrix, and its nullspace of B always admits nonnegative vectors. Namely, $B = (\mathbf{e}, \bar{B})$, where $\bar{B}$ is a square invertible matrix, and hence a valid $(y_0, \ldots, y_m)$ is proportional to $\bar{B}^{-1}(y_0, \ldots, y_m)$. Hence, by setting $m = t$, it is often possible to have generalized (b, g) code.

We like to prove that this generalized (b, g) PI code has distance at least $2t+1$, thereby correcting t errors, whenever (1) $g, (b-g) \ge 2t+1$ and (2) $m \ge t$. Note that condition (1) imposes the bit-flip distance, and hence it remains to show the nondeformation condition for the

phase errors. From the above A-matrix type of argument, we just need to ascertain that the vector

$$\gamma_{b,g,m} = (\gamma^2_{b,g,m,0}, \ldots, \gamma^2_{b,g,m,m}) \tag{E21}$$

resides in the null space of the matrix B, that is, $B\gamma_{b,g,m} = 0$. For this, we need to show that the following is true for all $x = 1, \ldots, t$:

$$\sum_{k=0}^{m} \frac{K^N_{2bk}(2x-1)}{\binom{N}{2bk}} \binom{m}{k} \gamma^2_{b,g,m,k} = 0. \tag{E22}$$

The system of linear equations can equivalently be written as

$$\sum_{k=0}^{m} \binom{m}{k} K^N_{2x-1}(2bk) \gamma^2_{b,g,m,k} = 0, \tag{E23}$$

which is

$$\sum_{k=0}^{m} \binom{m}{k} K^N_{2x-1}(2bk)(N/2b - g/b)_{(m-k)}(N/2b)_{(k)} = 0, \tag{E24}$$

It then remains to prove the following combinatorial lemma.

Lemma E1. Let b, g, m and x be positive integers where $x \leq m$. Let $N = 2bm + g$. Then

$$S := \sum_{k=0}^{m} \binom{m}{k} K^N_{2x-1}(2bk)(N/2b - g/b)_{(m-k)}(N/2b)_{(k)} = 0. \tag{E25}$$

Proof. The Krawtchouk polynomial $K^N_{2x-1}(2bk)$ can be written as a coefficient of a generating function, in the sense that

$$K^N_{2x-1}(2bk) = [t^{2x-1}](1-t)^{2bk}(1+t)^{N-2bk}. \tag{E26}$$

Now let $\alpha = N/2b - g/b$ and $\beta = N/2b$. Note that $\alpha + \beta = 2N/2b - g/b = (N-g)/b = 2m$. In addition, we can write

$$\binom{m}{k} \alpha_{(m-k)} \beta_{(k)} = m! \binom{\alpha}{m-k} \binom{\beta}{k}. \tag{E27}$$

Then we can write

$$\begin{aligned} S &= m! \sum_{k=0}^{m} \binom{\alpha}{m-k} \binom{\beta}{k} [t^{2x-1}](1-t)^{2bk}(1+t)^{N-2bk} \\ &= m![t^{2x-1}](1+t)^N \sum_{k=0}^{m} \binom{\alpha}{m-k} \binom{\beta}{k} ((1-t)/(1+t))^{2bk}. \end{aligned} \tag{E28}$$

Now

$$\sum_{k=0}^{m} \binom{\alpha}{m-k} \binom{\beta}{k} y^k = [s^m](1+s)^\alpha (1+sy)^\beta. \tag{E29}$$

Identifying $y = (1-t)/(1+t)$, to prevent higher powers of $(1-t)/(1+t)$ from appearing in the generating function when we make the substitution $y = (1-t)/(1+t)$, we have that whenever $x \leq m$,

$$\begin{aligned} S &= m![t^{2x-1}s^m](1+t)^N (1+s)^\alpha (1+s((1-t)/(1+t)))^\beta \\ &= m![t^{2x-1}s^m]E(t,s), \end{aligned} \tag{E30}$$

where

$$E(t,s) = (1+s)^\alpha ((1+t)^{2b} + s(1-t)^{2b})^\beta \tag{E31}$$

is a formal power series in the variables t and s, which allows us to write

$$E(t,s) = \sum_{u,v \geq 0} e_{u,v} t^u s^v. \tag{E32}$$

Now

$$\begin{aligned} E(-t, 1/s) &= (1+1/s)^\alpha ((1-t)^{2b} + s^{-1}(1+t)^{2b})^\beta \\ &= s^{-\alpha-\beta}(s+1)^\alpha (s(1-t)^{2b} + (1+t)^{2b})^\beta \\ &= s^{-\alpha-\beta} E(t,s) \\ &= s^{-2m} E(t,s). \end{aligned} \tag{E33}$$

Using (E33), it follows that

$$\sum_{u',v' \geq 0} e_{u,v} (-t)^{u'} s^{-v'} = \sum_{u,v \geq 0} e_{u,v} t^u s^{v-2m}. \tag{E34}$$

We are interested in the scenario where $v - 2m = -m$ and $-v' = -m$, which is equivalent to $v' = v = m$. Then we equate the coefficients of the left- and right-hand side of the above equation, focusing on the coefficient of $s^{-m}t^u$ for odd u. Then we get

$$e_{u,m} = -e_{u,m} \tag{E35}$$

for odd u, which means that $e_{u,m} = 0$ for odd u. Recalling our constraint $x \leq m$, which means that we also need to have $u \leq 2m - 1$, this proves the result. ■

Since Lemma E1 is true, this shows that our (b, g, m) PI code has the stipulated distance property.

APPENDIX F: FIDELITY MEASURE FOR HADAMARD GATE PROCESS

In this appendix we discuss how to evaluate the fidelity measure using process matrices. The process matrix gives a convenient way of describing the evolution of quantum states, especially when dealing with mixed states or open quantum systems where the state is described by a density matrix. In the case of a unitary operation $\rho' = U\rho U^\dagger$, if we vectorize the density matrix ρ using the row-stacking operation, the action of the superoperator can be expressed in a matrix form as

$$|\rho'\rangle\rangle = \mathcal{U}|\rho\rangle\rangle, \tag{F1}$$

where $\mathcal{U} = U \otimes U^*$. Here U^* is the complex conjugate of U and $\otimes$ denotes the Kronecker product.

In our case, for the preparation mapping described in Eq. (20), the global rotations R are unitary and their associated superoperators are represented in the Kronecker product form. Though the erroneous GPG mapping in Eq. (18) is nonunitary, it does not induce coherence between different state components, and can thus be constructed as a diagonal matrix in the superoperator formalism. Consequently, the superoperator associated with the preparation mapping $\mathcal{E}_1$-GPG is given by

$$E_1\text{-GPG} = \prod_{p=P}^{1} [R(\theta_p, \xi_p, \gamma_p) \otimes R^*(\theta_p, \xi_p, \gamma_p)]\mathcal{G}_p(\phi_p), \tag{F2}$$

where $\mathcal{G}_p(\phi_p)$ is a $(N+1)^2 \times (N+1)^2$ diagonal matrix that has elements

$$\langle D_n^N|\langle D_m^N|\mathcal{G}_p(\phi_p)|D_n^N\rangle|D_m^N\rangle = f_{n,m}(\phi_p) \tag{F3}$$

on each diagonal position accordingly.

Analogously, the superoperator associated with the reverse mapping of the preparation $\mathcal{E}_1^{\mathrm{R}}$-GPG is given by

$$E_1^{\mathrm{R}}\text{-GPG} = \prod_{p=1}^{P} \mathcal{G}_p(-\phi_p)[R^\dagger(\theta_p, \xi_p, \gamma_p) \otimes (R^\dagger(\theta_p, \xi_p, \gamma_p))^*]. \tag{F4}$$

The middle phase gate in Eq. (25) is unitary but has an overall fidelity factor F_{ph}. We express its superoperator as

$$E_{\mathrm{ph}} = F_{\mathrm{ph}}\mathrm{C}_{N-1}(Z) \otimes \mathrm{C}^*_{N-1}(Z). \tag{F5}$$

Thus, the actual mapping for the implementation of the logical Hadamard gate in the superoperator formalism is given by

$$E = E_1\text{-GPG}E_{\mathrm{ph}}E_1^{\mathrm{R}}\text{-GPG}. \tag{F6}$$

Since our primary concern is whether the implemented Hadamard gate acts correctly in the logical subspace, we project E from the Dicke subspace onto the logical $\{|0_{\mathrm{L}}\rangle, |1_{\mathrm{L}}\rangle\}$ subspace, resulting in

$$E_{\mathrm{L}} = \begin{pmatrix} E_{00,00} & E_{00,01} & E_{00,10} & E_{00,11} \\ E_{01,00} & E_{01,01} & E_{01,10} & E_{01,11} \\ E_{10,00} & E_{10,01} & E_{10,10} & E_{10,11} \\ E_{11,00} & E_{11,01} & E_{11,10} & E_{11,11} \end{pmatrix}, \tag{F7}$$

where $E_{x'y',xy} = \langle x'_{\mathrm{L}}|\langle y'_{\mathrm{L}}|E|x_{\mathrm{L}}\rangle|y_{\mathrm{L}}\rangle$. Following the method in Ref. [64], the process fidelity between $\mathcal{E}_{\overline{H}}$ and $\overline{\mathcal{H}}$ can be simply calculated as

$$F_{\mathrm{pro}}(\mathcal{E}_{\overline{H}}, \overline{\mathcal{H}}) = \tfrac{1}{8}\sum_j \mathrm{Tr}[\overline{H}_{\mathrm{ide}}U_j^\dagger \overline{H}_{\mathrm{ide}}\mathcal{E}_{\overline{H}}(U_j)], \tag{F8}$$

where $\{U_j\}$ are orthonormal bases of unitary operators. Here, in the logical subspace, we select $U_0 = I_2$, $U_1 = X$, $U_2 = Y$, and $U_3 = Z$ (where I_2, X, Y, and Z are the identity and Pauli matrices). Substitution into Eq. (F8) gives

$$\begin{aligned} F_{\mathrm{pro}}(\mathcal{E}_{\overline{H}}, \overline{\mathcal{H}}) &= \frac{1}{8}(\mathrm{Tr}[\mathcal{E}_{\overline{H}}(I_2)] + \mathrm{Tr}[Z\mathcal{E}_{\overline{H}}(X)] + \mathrm{Tr}[-Y\mathcal{E}_{\overline{H}}(Y)] + \mathrm{Tr}[X\mathcal{E}_{\overline{H}}(Z)]) \\ &= \tfrac{1}{8}\left[\begin{pmatrix}1 & 0 & 0 & 1\end{pmatrix}E_{\mathrm{L}}\begin{pmatrix}1\\0\\0\\1\end{pmatrix} + \begin{pmatrix}1 & 0 & 0 & -1\end{pmatrix}E_{\mathrm{L}}\begin{pmatrix}0\\1\\1\\0\end{pmatrix} + \begin{pmatrix}0 & i & -i & 0\end{pmatrix}E_{\mathrm{L}}\begin{pmatrix}0\\-i\\i\\0\end{pmatrix} + \begin{pmatrix}0 & 1 & 1 & 0\end{pmatrix}E_{\mathrm{L}}\begin{pmatrix}1\\0\\0\\-1\end{pmatrix}\right] \\ &= \tfrac{1}{8}(E_{00,00} + E_{00,11} + E_{11,00} + E_{11,11} + E_{00,01} + E_{00,10} - E_{11,01} - E_{11,10} \\ &\quad - E_{01,01} + E_{01,10} + E_{10,01} - E_{10,10} + E_{01,00} - E_{01,11} + E_{10,00} - E_{10,11}) \end{aligned} \tag{F9}$$

APPENDIX G: NONLINEAR GPGS FOR SWITCHING BETWEEN AN EVEN-ODD CODE AND A NON-EVEN-ODD CODE

In this appendix we describe how to directly implement code-switching using logical CNOT gates which works even when the PI code is not an even-odd code, as is the case for the PI-7 code. This requires use of nonlinear GPGs and to explain the mechanism we first review how this highly nonlinear spin gate is implemented from elementary interactions.

1. Implementing the nonlinear GPG

The dispersive interaction between spins and the bosonic mode we consider takes the form $H = g\hat{a}^\dagger\hat{a} \otimes \hat{w}_\Gamma$. Evolving the dispersive interaction H for a time t generates the operator $R(\theta\hat{w}_\Gamma) := e^{i\theta\hat{w}_\Gamma \otimes \hat{a}^\dagger\hat{a}}$ where $\theta = gt$. This applies a rotation in phase space by an amount proportional to the eigenvalues associated to eigenspaces of the Hermitian operator $\hat{w}$ acting on subsystem Γ. Note it is possible to generate $R(-\theta\hat{w}_\Gamma)$ by reversing the coupling strength $g \to -g$ which can be done in some physical setups by, e.g., changing the sign of detuning of the cavity mode from the spin transition frequency. In this appendix we consider mode-spin interactions H where the spin operator is $\hat{w}_\Gamma = \hat{J}^z_\Gamma$. This could arise from a native coupling [39], or, as described in Sec. IV, is locally equivalent to the interaction involving the Hamming weight operator.

Under conjugation by rotations, displacements can be made conditional on the spin states:

$$D(\alpha e^{i\theta\hat{w}_\Gamma}) = R(\theta\hat{w}_\Gamma)D(\alpha)R(-\theta\hat{w}_\Gamma). \tag{G1}$$

We can similarly make mode rotations conditional on the spin states using the operator

$$\Lambda_\Gamma(\alpha,\theta) := D(\alpha)R(\theta\hat{w}_\Gamma)D(-\alpha)R(-\theta\hat{w}_\Gamma). \tag{G2}$$

A composition of displacement and conditional displacement operators around a closed trajectory in phase space, provides for an identity operator on the mode and a nonlinear GPG on the spins:

$$\begin{aligned} U_{\text{nl}}\text{-GPG}(\theta,\phi,\chi) &= D(-\beta)R(\theta\hat{w}_\Gamma)D(-\alpha)R(-\theta\hat{w}_\Gamma) \\ &\quad \times D(\beta)R(\theta\hat{w}_\Gamma)D(\alpha)R(-\theta\hat{w}_\Gamma) \\ &= e^{-i2\chi\,\sin(\theta\hat{w}_\Gamma+\phi)}. \end{aligned}$$

Here $\phi = \arg(\alpha) - \arg(\beta)$. If the mode begins as the vacuum state, then the first rotation operator is not needed.

2. Implementing the C_BX_A gate

First, consider the gate C_BX_A which has the PI code as control and a stabilizer code as target. The stabilizer code is assumed to be an even-odd code, while The PI code has its logical zero codeword $|0_B\rangle$ and logical one codeword $|1_B\rangle$ consisting of superpositions of states with Hamming weight 0 mod q and s mod q, respectively. In the case of the PI-7 code, $q = 5$ and $s = 2$.

Here, we construct a controlled GPG that traverses a path in phase space with zero area if the control has weight 0 mod q, while producing the area required to generate a transversal X gate on the target if the control has s mod q.

If we pick $\gamma = s\pi/q$, then the action of the controlled displacement operator $\Lambda_B(\alpha, s\pi/q)$ on the codespace of the PI code is

$$\begin{aligned} &\Lambda_B(\alpha,s\pi/q)(c_0|0_B\rangle + c_1|1_B\rangle) \otimes |\psi_{\text{mode}}\rangle \\ &\quad = c_0|0_B\rangle \otimes |\psi_{\text{mode}}\rangle + c_1|1_B\rangle \\ &\quad \otimes D(\alpha(1-e^{2is\pi/q}))|\psi_{\text{mode}}\rangle. \end{aligned} \tag{G3}$$

We can use these controlled displacements between the PI code and the mode to produce a GPG on the stabilizer code conditional on the logical state of PI code:

$$\begin{aligned} &\Lambda_B(U_{\text{nl}}\text{-GPG}(\theta,\phi,\chi)) \\ &\quad := \Lambda_B(-\beta,s\pi/q)R(\theta\hat{J}^z_{\mathsf{A}})\Lambda_B(-\alpha,s\pi/q) \\ &\qquad \times R(-\theta\hat{J}^z_{\mathsf{A}})\Lambda_B(\beta,s\pi/q)R(\theta\hat{J}^z_{\mathsf{A}}) \\ &\qquad \times \Lambda_B(\alpha,s\pi/q)R(-\theta\hat{J}^z_{\mathsf{A}}), \end{aligned} \tag{G4}$$

where $\chi = |\alpha\beta||1-e^{2is\pi/q}|^2 = 4|\alpha\beta|\sin^2(s\pi/q)$ and $\phi = \arg(\alpha) - \arg(\beta)$. The action of this conditional nl-GPG on the codespace of the PI code and stabilizer code is

$$\begin{aligned} &\Lambda_B(U_{\text{nl}}\text{-GPG}(\theta,\phi,\chi))(c_0|0_B\rangle + c_1|1_B\rangle) \otimes |\psi_A\rangle \\ &\quad = c_0|0_B\rangle \otimes |\psi_A\rangle + c_1|1_B\rangle \otimes e^{-i2\chi\,\sin(\theta\hat{J}^z_A+\phi)}|\psi_A\rangle. \end{aligned} \tag{G5}$$

We pick $\alpha = \beta$ so that $\phi = 0$. With the choice $\theta = \pi$, when the control is in $|1_B\rangle$, the mode experiences a trajectory in phase space that is a rotated square with area χ, and when the control is in $|0_B\rangle$ the trajectory has zero area. If we further pick $\chi = \pi/4$, i.e., $\alpha = \sqrt{\pi}/(4|\sin(s\pi/q)|)$, then $e^{-i2\chi\,\sin(\theta\hat{J}^z_A)} = i\prod_{j=1}^7 Z_j = i\bar{Z}_A$, where $\bar{Z}_A$ is the logical Z operator on the stabilizer code.

The final gate is then

$$C_BX_A = \bar{S}_A\bar{H}_A\Lambda_1(U_{\text{nl}}\text{-GPG}(\pi,0,\frac{\pi}{4}))\bar{H}_A, \tag{G6}$$

where $\bar{H}_A = H^{\otimes|A|}$ is the transversal logical Hadamard and $\bar{S}_A = \bar{Z}S^{\otimes|A|}$ is the logical $S = Z(\pi/2)$ gate on the stabilizer code.

3. Implementing the C_AX_B gate

Second, consider the gate C_AX_B which has the stabilizer code as the control and the PI code as target. We can use the same procedure as in Appendix G 2, with the roles of A and B reversed, except we choose $g\tau = \gamma = \pi$. In

this case the action angle in phase space will be $\chi = 4|\alpha|^2$ and the GPG will have the parameters $\phi = \pi$, $\theta = \pi$, and $\alpha = \beta = \sqrt{\pi}/4$. This will achieve the operation $-i\bar{Y}_B$ if the control is in $|1_A\rangle$, and acts trivially if the control is in $|0_A\rangle$. In summary,

$$\begin{aligned}
\mathrm{C_A X_B} &= \Lambda_A\left(U_{\mathrm{nl}}\text{-GPG}\left(\pi,\pi,\frac{\pi}{4}\right)\right)\\
&= \Lambda_A\left(-\frac{\sqrt{\pi}}{4},\pi\right) e^{i\frac{\pi}{2}\hat{J}_B^x} R\left(\pi\hat{J}_{\mathsf{B}}^z\right) e^{-i\frac{\pi}{2}\hat{J}_B^x}\Lambda_A\left(-\frac{\sqrt{\pi}}{4},\pi\right)\\
&\quad\times e^{i\frac{\pi}{2}\hat{J}_B^x} R(-\pi\hat{J}_{\mathsf{B}}^z) e^{-i\frac{\pi}{2}\hat{J}_B^x}\Lambda_A\left(\frac{\sqrt{\pi}}{4},\pi\right) e^{i\frac{\pi}{2}\hat{J}_B^x} R(\pi\hat{J}_{\mathsf{B}}^z)\\
&\quad\times e^{-i\frac{\pi}{2}\hat{J}_B^x}\Lambda_A\left(\frac{\sqrt{\pi}}{4},\pi\right) e^{i\frac{\pi}{2}\hat{J}_B^x} R(-\pi\hat{J}_{\mathsf{B}}^z) e^{-i\frac{\pi}{2}\hat{J}_B^x}. \qquad \text{(G7)}
\end{aligned}$$

[1] B. Eastin, and E. Knill, Restrictions on transversal encoded quantum gate sets, Phys. Rev. Lett. **102**, 110502 (2009).

[2] E. T. Campbell, B. M. Terhal, and C. Vuillot, Roads towards fault-tolerant universal quantum computation, Nature **549**, 172 (2017).

[3] D. Gottesman, and I. L. Chuang, Demonstrating the viability of universal quantum computation using teleportation and single-qubit operations, Nature **402**, 390 (1999).

[4] X. Zhou, D. W. Leung, and I. L. Chuang, Methodology for quantum logic gate construction, Phys. Rev. A **62**, 052316 (2000).

[5] S. Bravyi, and A. Kitaev, Universal quantum computation with ideal Clifford gates and noisy ancillas, Phys. Rev. A **71**, 022316 (2005).

[6] S. Bravyi, and J. Haah, Magic-state distillation with low overhead, Phys. Rev. A **86**, 052329 (2012).

[7] A. Krishna, and J.-P. Tillich, Towards low overhead magic state distillation, Phys. Rev. Lett. **123**, 070507 (2019).

[8] E. T. Campbell, and M. Howard, Unified framework for magic state distillation and multiqubit gate synthesis with reduced resource cost, Phys. Rev. A **95**, 022316 (2017).

[9] E. T. Campbell, and M. Howard, Unifying gate synthesis and magic state distillation, Phys. Rev. Lett. **118**, 060501 (2017).

[10] B. Eastin, Distilling one-qubit magic states into Toffoli states, Phys. Rev. A **87**, 032321 (2013).

[11] M. B. Hastings, and J. Haah, Distillation with sublogarithmic overhead, Phys. Rev. Lett. **120**, 050504 (2018).

[12] J. O'Gorman, and E. T. Campbell, Quantum computation with realistic magic-state factories, Phys. Rev. A **95**, 032338 (2017).

[13] M. E. Beverland, A. Kubica, and K. M. Svore, Cost of universality: A comparative study of the overhead of state distillation and code switching with color codes, PRX Quantum **2**, 020341 (2021).

[14] D. Litinski, Magic state distillation: Not as costly as you think, Quantum **3**, 205 (2019).

[15] J. T. Anderson, G. Duclos-Cianci, and D. Poulin, Fault-tolerant conversion between the Steane and Reed-Muller quantum codes, Phys. Rev. Lett. **113**, 080501 (2014).

[16] H. Bombín, Dimensional jump in quantum error correction, New J. Phys. **18**, 043038 (2016).

[17] A. Kubica, and M. E. Beverland, Universal transversal gates with color codes: A simplified approach, Phys. Rev. A **91**, 032330 (2015).

[18] H. Poulsen Nautrup, N. Friis, and H. J. Briegel, Fault-tolerant interface between quantum memories and quantum processors, Nat. Commun. **8**, 1321 (2017).

[19] F. Butt, S. Heußen, M. Rispler, and M. Müller, Fault-tolerant code-switching protocols for near-term quantum processors, PRX Quantum **5**, 020345 (2024).

[20] I. Pogorelov, F. Butt, L. Postler, C. D. Marciniak, P. Schindler, M. Müller, and T. Monz, Experimental fault-tolerant code switching, ArXiv:2403.13732.

[21] J. Huang, S. M. Li, L. Yeh, A. Kissinger, M. Mosca, and M. Vasmer, Graphical CSS code transformation using ZX calculus, ArXiv:2307.02437.

[22] L. Daguerre, and I. H. Kim, Code switching revisited: Low-overhead magic state preparation using color codes, ArXiv:2410.07327.

[23] S. Heußen, and J. Hilder, Efficient fault-tolerant code switching via one-way transversal CNOT gates, ArXiv:2409.13465.

[24] B. J. Brown, A fault-tolerant non-Clifford gate for the surface code in two dimensions, Sci. Adv. **6**, eaay4929 (2020).

[25] S. Heußen, D. F. Locher, and M. Müller, Measurement-free fault-tolerant quantum error correction in near-term devices, PRX Quantum **5**, 010333 (2024).

[26] E. Kubischta, and I. Teixeira, Family of quantum codes with exotic transversal gates, Phys. Rev. Lett. **131**, 240601 (2023).

[27] O. Parzanchevski, and P. Sarnak, Super-golden-gates for PU(2), Adv. Math. **327**, 869 (2018), special volume honoring David Kazhdan.

[28] H. Bombín, Gauge color codes: Optimal transversal gates and gauge fixing in topological stabilizer codes, New J. Phys. **17**, 083002 (2015).

[29] A. M. Steane, Error correcting codes in quantum theory, Phys. Rev. Lett. **77**, 793 (1996).

[30] P. W. Shor, Scheme for reducing decoherence in quantum computer memory, Phys. Rev. A **52**, R2493 (1995).

[31] D. Bacon, Operator quantum error-correcting subsystems for self-correcting quantum memories, Phys. Rev. A **73**, 012340 (2006).

[32] P. Aliferis, and A. W. Cross, Subsystem fault tolerance with the Bacon-Shor code, Phys. Rev. Lett. **98**, 220502 (2007).

[33] M. B. Ruskai, Pauli exchange errors in quantum computation, Phys. Rev. Lett. **85**, 194 (2000).

[34] H. Pollatsek, and M. B. Ruskai, Permutationally invariant codes for quantum error correction, Linear Algebra Appl. **392**, 255 (2004).

[35] Y. Ouyang, Permutation-invariant quantum codes, Phys. Rev. A **90**, 062317 (2014).

[36] Y. Ouyang, and J. Fitzsimons, Permutation-invariant codes encoding more than one qubit, Phys. Rev. A **93**, 042340 (2016).

[37] Y. Ouyang, Permutation-invariant qudit codes from polynomials, Linear Algebra Appl. **532**, 43 (2017).

[38] R. Movassagh, and Y. Ouyang, Constructing quantum codes from any classical code and their embedding in

ground space of local Hamiltonians, Quantum **8**, 1541 (2024).
[39] M. T. Johnsson, N. R. Mukty, D. Burgarth, T. Volz, and G. K. Brennen, Geometric pathway to scalable quantum sensing, Phys. Rev. Lett. **125**, 190403 (2020).
[40] Y. Ouyang, Permutation-invariant quantum coding for quantum deletion channels, in *2021 IEEE International Symposium on Information Theory (ISIT)* (IEEE, Melbourne, 2021), p. 1499.
[41] Y. Ouyang, and G. K. Brennen, Finite round quantum error correction on symmetric quantum sensors, ArXiv:2212.06285.
[42] M. Hagiwara, The four qubits deletion code is the first quantum insertion code, IEICE Commun. Express **10**, 243 (2021).
[43] T. Shibayama, and Y. Ouyang, The equivalence between correctability of deletions and insertions of separable states in quantum codes, in *2021 IEEE Information Theory Workshop (ITW)* (IEEE, Kanazawa, 2021), p. 1.
[44] T. Shibayama, and M. Hagiwara, Equivalence of quantum single insertion and single deletion error-correctabilities, and construction of codes and decoders, in *2022 IEEE International Symposium on Information Theory (ISIT)* (IEEE, Espoo, Finland, 2022), p. 2957.
[45] Y. Ouyang, N. Shettell, and D. Markham, Robust quantum metrology with explicit symmetric states, IEEE Trans. Inf. Theory **68**, 1809 (2022).
[46] Y. Ouyang, Quantum storage in quantum ferromagnets, Phys. Rev. B **103**, 144417 (2021).
[47] E. Kubischta, and I. Teixeira, The not-so-secret fourth parameter of quantum codes, ArXiv:2310.17652.
[48] A. Aydin, M. A. Alekseyev, and A. Barg, A family of permutationally invariant quantum codes, Quantum **8**, 1321 (2024).
[49] J. A. Gross, Designing codes around interactions: The case of a spin, Phys. Rev. Lett. **127**, 010504 (2021).
[50] E. Kubischta, and I. Teixeira, Family of quantum codes with exotic transversal gates, Phys. Rev. Lett. **131**, 240601 (2023).
[51] P. O. Boykin, T. Mor, M. Pulver, V. Roychowdhury, and F. Vatan, On universal and fault-tolerant quantum computing: A novel basis and a new constructive proof of universality for Shor's basis, in *40th Annual Symposium on Foundations of Computer Science, 1999* (IEEE Computer Society, New York City, NY, 1999), p. 486.
[52] K. Mølmer, and A. Sørensen, Multiparticle Entanglement of Hot Trapped Ions, Phys. Rev. Lett. **82**, 1835 (1999).
[53] A. Luis, Quantum mechanics as a geometric phase: phase-space interferometers, J. Phys. A: Math. Gen. **34**, 7677 (2001).
[54] D. Leibfried, B. DeMarco, V. Meyer, D. Lucas, M. Barrett, J. Britton, W. M. Itano, B. Jelenković, C. Langer, T. Rosenband, and D. J. Wineland, Experimental demonstration of a robust, high-fidelity geometric two ion-qubit phase gate, Nature **422**, 412 (2003).
[55] S. Jandura, V. Srivastava, L. Pecorari, G. K. Brennen, and G. Pupillo, Nonlocal multiqubit quantum gates via a driven cavity, Phys. Rev. A **110**, 062610 (2024).
[56] A. S. Sørensen, and K. Mølmer, Measurement induced entanglement and quantum computation with atoms in optical cavities, Phys. Rev. Lett. **91**, 097905 (2003).
[57] L. B. Nguyen, Y.-H. Lin, A. Somoroff, R. Mencia, N. Grabon, and V. E. Manucharyan, High-coherence fluxonium qubit, Phys. Rev. X **9**, 041041 (2019).
[58] A. André, D. DeMille, J. M. Doyle, M. D. Lukin, S. E. Maxwell, P. Rabl, R. J. Schoelkopf, and P. Zoller, A coherent all-electrical interface between polar molecules and mesoscopic superconducting resonators, Nat. Phys. **2**, 636 (2006).
[59] X. Wang, and P. Zanardi, Simulation of many-body interactions by conditional geometric phases, Phys. Rev. A **65**, 032327 (2002).
[60] N. Gutman, A. Gorlach, O. Tziperman, R. Ruimy, and I. Kaminer, Universal control of symmetric states using spin squeezing, Phys. Rev. Lett. **132**, 153601 (2024).
[61] L. J. Bond, M. J. Davis, J. c. v. Minář, R. Gerritsma, G. K. Brennen, and A. Safavi-Naini, Global variational quantum circuits for arbitrary symmetric state preparation, Phys. Rev. Res. **7**, L022072 (2025).
[62] V. Srivastava, S. Jandura, G. K. Brennen, and G. Pupillo, Entanglement-enhanced quantum sensing via optimal global control, ArXiv:2409.12932.
[63] G. K. Brennen, D. P. O'Leary, and S. S. Bullock, Criteria for exact qudit universality, Phys. Rev. A **71**, 052318 (2005).
[64] A. Gilchrist, N. K. Langford, and M. A. Nielsen, Distance measures to compare real and ideal quantum processes, Phys. Rev. A–At., Mol. Opt. Phys. **71**, 062310 (2005).
[65] P. Aliferis, D. Gottesman, and J. Preskill, Quantum accuracy threshold for concatenated distance-3 codes, Quantum. Inf. Comput. **6**, 97 (2006).
[66] Y. Ouyang, Avoiding coherent errors with rotated concatenated stabilizer codes, Npj Quantum Inf. **7** (2021).
[67] T. Zhang, M. Wu, S. R. Cohen, L. Xin, D. Das, K. K. S. Multani, N. Peard, A.-M. Valente-Feliciano, P. B. Welander, A. H. Safavi-Naeini, E. A. Nanni, and M. Schleier-Smith, Optically accessible high-finesse millimeter-wave resonator for cavity quantum electrodynamics with atom arrays, ArXiv:2506.05804.
[68] S. Kuhr, S. Gleyzes, C. Guerlin, J. Bernu, U. B. Hoff, S. Deléglise, S. Osnaghi, M. Brune, J.-M. Raimond, S. Haroche, E. Jacques, P. Bosland, and B. Visentin, Ultrahigh finesse fabry-pérot superconducting resonator, Appl. Phys. Lett. **90**, 164101 (2007).
[69] C. D. Marciniak, T. Feldker, I. Pogorelov, R. Kaubruegger, D. V. Vasilyev, R. van Bijnen, P. Schindler, P. Zoller, R. Blatt, and T. Monz, Optimal metrology with programmable quantum sensors, Nature **603**, 604 (2022).
[70] S. P. Jain, and V. V. Albert, Transversal Clifford and T-gate codes of short length and high distance, IEEE J. Sel. Areas Inf. Theory **6**, 127 (2025).
[71] D. Gottesman, Fault-tolerant quantum computation with constant overhead, Quantum Inf. Comput. **14**, 1338 (2014).
[72] R. Chao, and B. W. Reichardt, Quantum error correction with only two extra qubits, Phys. Rev. Lett. **121**, 050502 (2018).
[73] R. Chao, and B. W. Reichardt, Flag fault-tolerant error correction for any stabilizer code, PRX Quantum **1**, 010302 (2020).
[74] Y. Ouyang, and R. Chao, Permutation-invariant constant-excitation quantum codes for amplitude damping, IEEE Trans. Inf. Theory **66**, 2921 (2019).