Efficient fault-tolerant code switching via one-way transversal CNOT gates

Sascha Heußen[1] and Janine Hilder[2,3]

[1]neQxt, 50670 Cologne, Germany
[2]neQxt, 63906 Erlenbach am Main, Germany
[3]QUANTUM, Institut für Physik, Universität Mainz, 55128 Mainz, Germany

September 20, 2024

Code switching is an established technique that facilitates a universal set of fault-tolerant (FT) quantum gate operations by combining two quantum error correcting (QEC) codes with complementary sets of gates, which each by themselves are easy to implement fault-tolerantly. In this work, we present a code switching scheme that respects the constraints of FT circuit design by only making use of transversal gates. These gates are intrinsically FT without additional qubit overhead. We analyze application of the scheme to low-distance color codes, which are suitable for operation in existing quantum processors, for instance based on trapped ions or neutral atoms. We also briefly discuss connectivity constraints that arise for architectures based on superconducting qubits. Numerical simulations of circuit-level noise indicate that a logical T-gate, facilitated by our scheme, could outperform both flag-FT magic state injection protocols and a physical T-gate at low physical error rates. Transversal code switching naturally scales to code pairs of arbitrary code distance. We observe improved performance of a distance-5 protocol compared to both the distance-3 implementation and the physical gate for realistic error rates. We discuss how the scheme can be implemented with a large degree of parallelization, provided that logical auxiliary qubits can be prepared reliably enough. Our logical T-gate circumvents the need for potentially costly magic state factories. The requirements to perform QEC and to achieve an FT universal gate set are then essentially the same: Prepare logical auxiliary qubits offline, execute transversal gates (ideally in parallel) and perform fast-enough measurements. Transversal code switching thus serves to enable more practical hardware realizations of FT universal quantum computation. The scheme alleviates resource requirements for experimental demonstrations of quantum algorithms run on logical qubits.

Sascha Heußen: s.heussen@neqxt.org

1 Introduction

Universal quantum computation offers the potential to efficiently solve computational tasks that classical computers require an exponentially growing time or memory for. Crucially, a discrete universal set of quantum gate operations should be implemented on error-corrected logical qubits in a fault-tolerant (FT) manner [1]. Quantum error correction (QEC) allows one to systematically suppress noise such that the quantum computation can in principle be upheld for arbitrarily long times, provided the physical noise level is below some threshold value [2]. The formal requirements of quantum fault tolerance add an additional gate and/or qubit overhead to any quantum circuit, which – however – has been shown to not scale unfavorably with the size of such circuit [3]. Remarkably, these added components have demonstrated improvements of overall circuit performance in real devices compared to a non-FT implementation, despite being noisy themselves [4–11].

A prominent modern technique to render quantum circuits FT with little qubit and gate overhead is the flag qubit paradigm [12–20]. Faults within the circuit that may cause uncorrectable errors on the logical qubit are heralded by controlled propagation onto extra auxiliary qubits via additional CNOT gates, which are interleaved into the quantum circuit in an intricate way. When such a flag qubit is measured in a particular state, one may employ further circuity to treat such faults or repeat the circuit altogether until the flag is clear. On the one hand, flag circuits may be favorable due to their conceptual simplicity and typically small qubit overhead. On the other hand, dynamical circuit branching with relatively long circuit sequences might pose a challenge for practical applications of QEC with flag circuits. Also, recent developments of quantum computing hardware suggest that the number of qubits may no longer be the most critical limiting factor [21–25].

Although possibly requiring a larger number of physical qubits, using *transversal* gate operations could offer numerous operational advantages. Not only are they inherently FT but also straightforwardly scale up to larger code distances and can directly be parallelized if the physical qubit architec-

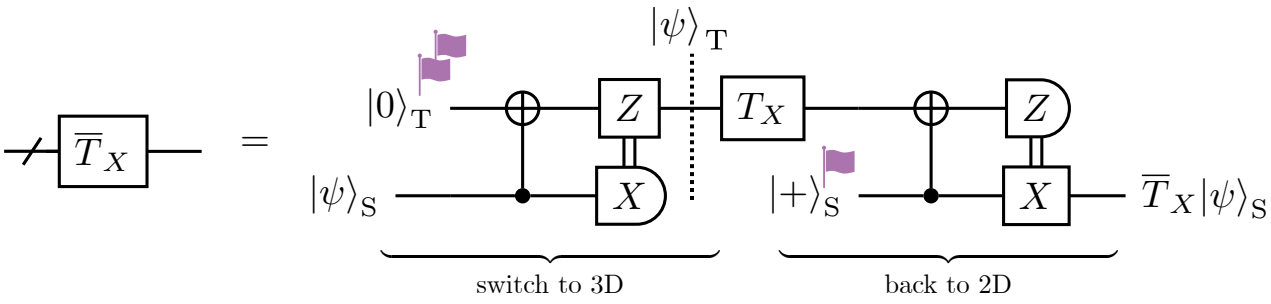

Figure 1: Implementation of the logical T-gate on a self-dual code via transversal code switching between the self-dual code (S) on the lower wire and a doubled triorthogonal code (T) on the upper wire. An arbitrary input state $|\psi\rangle$ is moved to the upper wire with a teleportation subroutine that only requires transversal logical operations and preparation of an auxiliary qubit in the logical zero state. The triorthogonal code implements the non-Clifford T-gate transversally. A second teleportation subroutine moves the logical state $T_X|\psi\rangle$ back to the lower wire where, in the meantime, a logical plus state has been prepared. The state preparation subroutines might require flag qubits or another type of verification to comply with fault tolerance. A concrete application of the scheme to the 2D/3D color codes of distance $d = 3$, the Steane code and the Tetrahedral code, is described in Sec. 2.2.

ture permits. Other FT QEC algorithms, like Steane-type [26, 27] or Knill-type [10, 28, 29] error correction, make use of transversal gates. For instance, Steane-type EC has been shown to outperform flag EC if the limiting factor is the entangling gate error rate p_2 in the limit of small p_2 ($\lesssim 3\%$) even though using more physical qubits [27].

For the controlled manipulation of encoded information, FT gate operations must be performed on the logical qubits. Transversal CNOT gates are appealing because they need not induce a time-overhead to respect fault tolerance such as, e.g., repeated operator measurements in lattice surgery [30–32]. It is known that no non-trivial QEC code offers a transversal universal gate set [33]. Therefore, given any code with a set of transversal gates, additional strategies are imperative to establish universality. There exist several procedures such as gate teleportation [34] or code conversion [35, 36] that vary in intricacy from both a theoretical and experimental point of view. Code switching in particular refers to the idea of employing *two* different QEC codes, which each on their own have a different set of transversal gates but in conjunction offer an FT universal gate set that is simple to perform on the logical level [37, 38]. The difficulty of implementing a particular gate is traded in for transferring the logical information between the two codes at will in an FT fashion.

All Clifford gates have a transversal implementation in two-dimensional (2D) color codes [39]. To unlock universality, an efficient FT construction of a non-Clifford gate, such as the T-gate, is required. The T-gate is transversal in three-dimensional (3D) color codes (as well as the CNOT gate but the Hadamard gate H is not) [40]. An FT code switching technique based on measuring potentially long sequences of stabilizers with flag qubits has been introduced recently for distance-3 color codes, which serve to correct arbitrary single Pauli errors [41]. It has subsequently been demonstrated in practice on an ion-trap quantum processor [42].

In this work, we investigate a new approach to implement the non-Clifford T-gate fault-tolerantly on the 2D color code via FT code switching to the 3D color code that is based on code doubling [43–46]. A notable strength of the scheme is that it can be viewed as *deterministic* in practice: Only QEC codes of distance $d \geq 3$ are employed, instead of mere quantum error *detecting* codes with $d = 2$, and no post-selection is required. The scheme is considered efficient because it relies on a transversal construction of the logical CNOT gate, which works as long as one restricts the connection of the gate to be one-directional. Transversal code switching is performed via logical state teleportation. The logical T-gate implementation is then built on two switching steps "2D $\rightarrow$ 3D $\rightarrow$ 2D", depicted with logical building blocks in Fig. 1. Logical auxiliary qubits in both codes can be prepared with Clifford circuits only, so that no complicated magic state subroutines are needed. We find that transversal code switching can outperform other universality schemes in terms of logical failure rate, entangling gate count and circuit depth while only slightly increasing the number of required physical qubits.

This manuscript is organized as follows: The efficient FT T-gate construction is discussed in detail in Sec. 2. We offer insights on parallelization and embedding into a reduced-connectivity hardware architecture. In Sec. 3, we present simulations of incoherent circuit-level Pauli noise with both a simple single-parameter depolarizing noise model and a more elaborate multi-parameter noise model that can be used to capture the essential noise processes of a trapped-ion quantum computer. A comparison to other strategies for establishing an FT universal gate set is provided in

Sec. 4. In Sec. 5, we consider scaling up our scheme to QEC codes of arbitrary distance and lastly draw conclusions and provide an outlook on future work in Sec. 6.

2 Transversal code switching

Code switching has originally been deployed as a method relying on two topological stabilizer codes each manifesting as a particular gauge of the same, common subsystem code [37]. The switching is performed by gauge fixing in this framework and a simplified switching scheme has been developed soon after [38].

Essential for any distance-d QEC code, FT code switching must ensure that no more than $t = \lfloor \frac{d-1}{2} \rfloor$ *errors* of any Pauli type X or Z can result from up to t *fault* events in a circuit-level noise model. Such faults must be assumed to happen during the execution of the actual quantum circuits of the switching algorithm where every physical operation acting on q qubits is subjected to a q-qubit depolarizing noise channel. A logical gate is deemed transversal if it can be implemented by applying bitwise physical gate operations to all physical qubits and, crucially, no interactions between physical qubits happen within the same logical block. Therefore, faults are confined to the qubits on which they initially occur and never spread uncontrollably within a code block.

The Hadamard gate is not directly transversal in the 3D color code because the X- and Z-type stabilizers are not invariant under the exchange of X and Z, i.e., it is not self-dual, as opposed to the 2D color code. The T-gate has a transversal implementation for the class of triorthogonal codes, to which the 3D color code belongs but the 2D color code does not. By using more elaborate transversal operations and an auxiliary system, it was shown how to perform the FT Hadamard gate on triorthogonal codes [47]. The FT T-gate can be implemented on a small 2D color code via flag-assisted code switching, which requires an intermediate (half-)cycle of QEC [41].

In the following, we describe a novel FT T-gate implementation for 2D color codes based on transversal code switching. We first review the construction of so-called doubled codes, which are appropriate triorthogonal codes to enable the transversal T-gate and efficient switching procedures. Then, we focus on FT code switching between 2D and 3D color codes. Usage of the smallest code representatives of distance $d = 3$ is explained in detail on the level of physical qubit quantum circuits. Practical aspects of the scheme such as logical auxiliary state preparation, parallelized gate operations and limited qubit-connectivity are discussed subsequently.

2.1 Doubled-code construction

In Ref. [46], a code construction is described that allows one to perform FT code switching with the help of logical auxiliary qubits and one-way transversal CNOT gates. The central ingredients are a $[[n, k = 1, d]]$ self-dual Calderbank-Shor-Steane (CSS) code[1] and a triorthogonal $[[\tilde{n}, 1, \tilde{d}]]$ code, from which a so-called *doubled* code is constructed, which is itself again a triorthogonal code with parameters $[[n' = 2n + \tilde{n}, 1, d' = \min(d, \tilde{d} + 2)]]$. The stabilizers and logical operators of the two ingredient codes are recombined to a new, bigger triorthogonal matrix in the process of code doubling. The initial self-dual code is complemented with additional Z-type stabilizers to yield the new triorthogonal code, which may have non-local stabilizers. Importantly, the original Z-type stabilizers of the self-dual code remain intact and are incorporated into the doubled code. As a consequence, the one-way logical CNOT gate can be executed by connecting n physical CNOT gates in a pair-wise fashion between the n physical qubits of the self-dual code and a subset of n physical qubits of the doubled code. This enables bi-directional switching between the two codes via standard quantum state teleportation circuits at the logical level. Transversality ensures that fault tolerance is respected for a code switching step. Additionally, it must be guaranteed that no uncorrectable error can ever propagate between the two codes from the preparation of the logical auxiliary qubits. While code doubling has been investigated before [43–45], the one-way transversal CNOT gate is a new ingredient identified in Ref. [46].

2.2 Reduction to 2D/3D color code combination

The logical CNOT gate can be performed directly between a 2D and a 3D color code at the price of restricting it to only one direction. The reason for this is that the 3D color code already is a doubled code. For the code definitions used in Ref. [46], this means that the control of the logical CNOT may only be encoded in the 3D color code and the target of the logical CNOT must be encoded in the 2D color code. It can be ensured that all stabilizer operators of each code propagate through the logical CNOT onto stabilizer operators of the respective other code, using only n physical CNOT gates. The product of the stabilizer groups of both codes is therefore left invariant. Simultaneously, logical operators of both X- and Z-type are mapped between the two codes as prescribed by the action of a logical CNOT gate.

The remainder of this section is dedicated to applying the aforementioned code switching protocol to the smallest instances of the 2D and 3D color code respectively depicted in Fig. 2, i.e., the seven-qubit

[1] This need not be a color code.

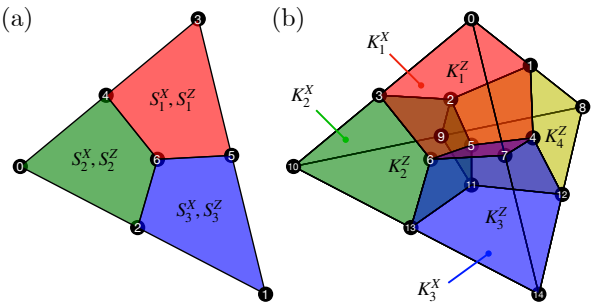

Figure 2: The smallest representatives of the code families of (a) two-dimensional and (b) three-dimensional CSS color codes. These distance-3 codes are commonly referred to as the Steane code and the Tetrahedral code respectively. The Clifford gates $\{H, S, \text{CNOT}\}$ are transversal in the Steane code and the gate set $\{X, T, \text{CNOT}\}$ is transversal in the Tetrahedral code. The weight-8 cell operators are defined as the Z-type stabilizer generators of the Tetrahedral code; three out of the 10 weight-4 face X-type stabilizer generators are labelled explicitly. They are inherited from the X-plaquettes of the Steane code.

Steane code and the fifteen-qubit Tetrahedral code. Both QEC codes have distance $d = 3$ and thus correct $t = 1$ arbitrary Pauli error. FT state preparation of the logical auxiliary qubit states will be discussed in Sec. 2.2.2. After providing a detailed discussion of this small-scale implementation, we broaden the horizon again and discuss scaling up to large distances in Sec. 5.

2.2.1 Distance-3 transversal code switching

We depict the scheme with logical building blocks in Fig. 1. The workhorse of the code switching procedure is the reduced-support transversal CNOT gate that can be applied between the Steane code and the Tetrahedral code [46]. The $[[7, 1, 3]]$ Steane code [48] is defined by the stabilizer generators

$$\begin{aligned} S_1^X &= X_3X_4X_5X_6, & S_1^Z &= Z_3Z_4Z_5Z_6, \\ S_2^X &= X_0X_2X_4X_6, & S_2^Z &= Z_0Z_2Z_4Z_6, \\ S_3^X &= X_1X_2X_5X_6, & S_3^Z &= Z_1Z_2Z_5Z_6. \end{aligned} \tag{1}$$

The logical operators have minimal weight 3. For example, the edges of the Steane code triangle, such as $\overline{X}_S = X_0X_3X_4$, correspond to logical operators. We use a definition of the Tetrahedral code, where all X- and Z-type stabilizers are interchanged compared to the conventional definition [49]. This way, we achieve protection against $t = 3$ Z-errors since the $[[15, 1, 3]]$ code defined by the stabilizer generators

$$\begin{aligned} K_1^X &= X_0X_3X_6X_7, & K_2^X &= X_3X_6X_{10}X_{13}, \\ K_3^X &= X_6X_7X_{13}X_{14}, & K_4^X &= X_8X_9X_{11}X_{12}, \\ K_5^X &= X_1X_2X_4X_5, & K_6^X &= X_4X_5X_6X_7, \\ K_7^X &= X_2X_3X_5X_6, & K_8^X &= X_4X_5X_{11}X_{12}, \\ K_9^X &= X_2X_5X_9X_{11}, & K_{10}^X &= X_5X_6X_{11}X_{13} \end{aligned} \tag{2}$$

and

$$\begin{aligned} K_1^Z &= Z_0Z_1Z_2Z_3Z_4Z_5Z_6Z_7, \\ K_2^Z &= Z_2Z_3Z_5Z_6Z_9Z_{10}Z_{11}Z_{13}, \\ K_3^Z &= Z_4Z_5Z_6Z_7Z_{11}Z_{12}Z_{13}Z_{14}, \\ K_4^Z &= Z_1Z_2Z_4Z_5Z_8Z_9Z_{11}Z_{12} \end{aligned} \tag{3}$$

has distance $d_Z = 7$ and distance $d_X = 3$. This reflects the fact that the 3D color code is not self-dual. The ten X-type stabilizers are faces of the tetrahedron of weight 4 in Fig. 2 and the four Z-type stabilizers correspond to weight-8 cells. The interchange of X- and Z-type stabilizers has the consequence that, for our implementation, the one-way transversal CNOT may only connect the logical control to the Steane code and the logical target to the Tetrahedral code. Logical $X(Z)$- operators can be chosen to correspond to any edge (side) of the tetrahedron. For example, $\overline{X}_T = X_0X_3X_{10}$ and $\overline{Z}_T = Z_0Z_1Z_4Z_7Z_8Z_{12}Z_{14}$ are logical operators. Other representatives can be obtained by multiplication with stabilizers. As noted in Ref. [46], the Tetrahedral code can be constructed as a doubled code from the Steane code and the trivial triorthogonal code with parameters $[[\tilde{n} = 1, 1, \tilde{d} = 1]]$. This is why the X-generators of the Tetrahedral code explicitly contain the Steane code's X-plaquettes (also see Fig. 2).

Let us now explain why this one-way transversal CNOT gate works; Figs. 3 and 4 show the propagation of stabilizers and logical operators through the seven physical CNOTs wired between the seven qubits of the Steane code and one side of the Tetrahedron. With the given qubit labels, the logical CNOT is implemented as

$$\begin{aligned} \overline{\text{CNOT}} = &\,\text{CNOT}(0, 10)\,\text{CNOT}(1, 14)\,\text{CNOT}(2, 13) \\ &\,\text{CNOT}(3, 0)\,\text{CNOT}(4, 3)\,\text{CNOT}(5, 7) \\ &\,\text{CNOT}(6, 6). \end{aligned} \tag{4}$$

The labels of the physical controls refer to the Steane code and the labels of the physical targets refer to the Tetrahedral code (see also the circuit diagram in Fig. 14). A physical CNOT gate propagates a single X from the control to its target. A single Z is propagated from the target to the control qubit. A Pauli $X(Z)$ on the target (control) does not propagate. Consequentially, the one-way logical CNOT in Eq. (4) leaves the respective stabilizers of the Steane code and the Tetrahedral code invariant. The X-type

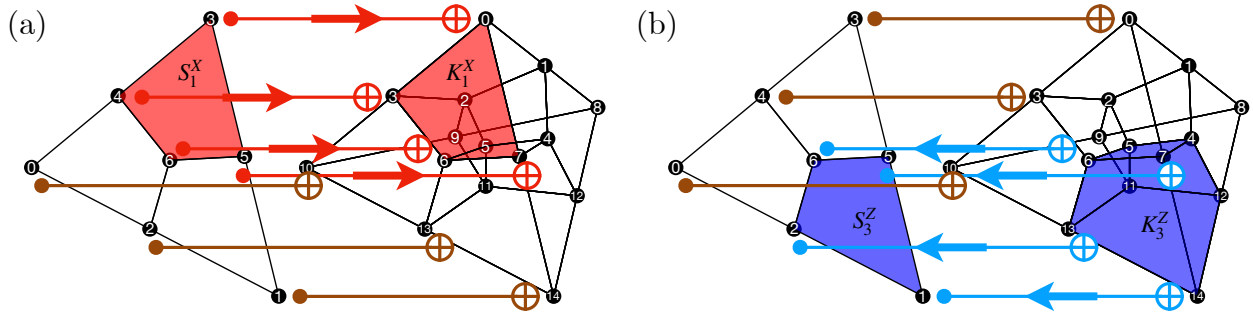

Figure 3: Propagation of stabilizers through the one-way transversal CNOT gate. (a) The red X-plaquette of the Steane code S_1^X is mapped to the red face of the Tetrahedral code K_1^X. (b) The blue cell of the Tetrahedral code K_3^Z propagates partly to the Steane code such that it corresponds to the blue Z-plaquette S_3^Z.

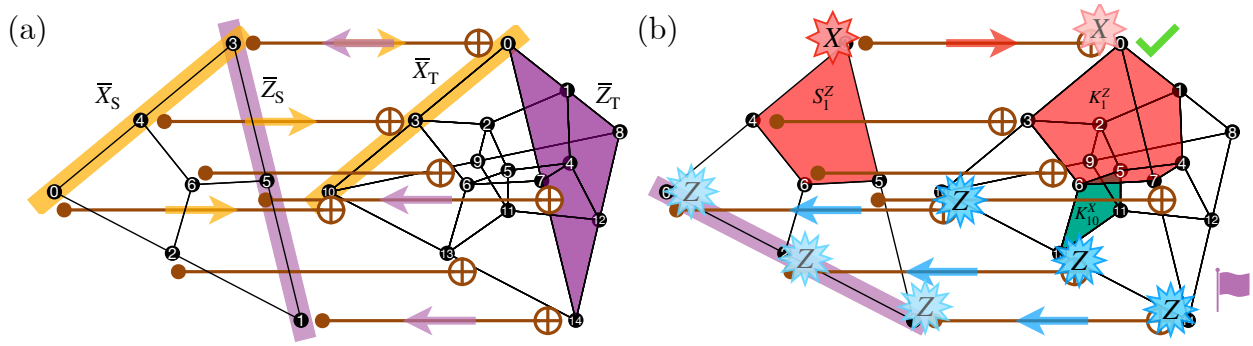

Figure 4: Propagation of logical operators and errors through the one-way transversal CNOT gate. (a) Propagating logical operators (see Eq. (7)) match the codes' geometries: The logical X-operator $\overline{X}_S = X_0X_3X_4$ on the Steane code's left edge is mapped to an edge of the tetrahedron (both yellow), reflecting the fact that it has $d_X = 3$. Three Pauli operators of the logical Z-operator $Z_0Z_1Z_4Z_7Z_8Z_{12}Z_{14}$ on the right side of the Tetrahedral code (purple) propagate to the right edge of the Steane code (light purple) The logical operator $Z_0Z_3Z_6Z_7Z_{10}Z_{13}Z_{14}$ on the front side of the Tetrahedral code (not marked) maps to the weight-7 operator $Z^{\otimes 7}$ on the Steane code, which is also a logical operator. Every logical Z-operator on the Tetrahedral code has overlap 3 or 7 with the physical CNOT gates. It is important that a weight-3 logical X-operator on the Tetrahedral code, which may only have overlap 1 with the physical CNOT gates, such as $X_0X_1X_8$ (not marked), does *not* propagate to the Steane code. (b) A single X-error on the Steane code (red star) propagates to a single correctable X-error on the Tetrahedral code (light red star). A weight-3 Z-error (blue stars), which is correctable in the $d_Z = 7$-version of the Tetrahedral code, propagates directly to a logical Z-error on the Steane code (light blue stars) and must therefore be prevented; for instance by utilizing a flag qubit that heralds such errors. Stabilizers that anticommute with the given correctable errors are colored in red and turquoise respectively.

stabilizers can only propagate from the Steane code to the Tetrahedral code but not vice versa; plaquettes of the Steane code are mapped to faces of the Tetrahedral code, which indeed are stabilizers by construction [46]. As an example, the propagation

$$S_1^X \otimes I \to S_1^X \otimes K_1^X \tag{5}$$

is depicted in Fig. 3a. Since four physical CNOT gates are connected to three cells each, the weight-8 Z-type stabilizers of the Tetrahedral code propagate to only weight-4 plaquettes of the Steane code. Consider, for instance, the propagation

$$I \otimes K_3^Z \to S_3^Z \otimes K_3^Z \tag{6}$$

as shown in Fig. 3b.

A weight-3 logical X-operator on one edge of the Steane code is correctly mapped to a weight-3 logical X-operator on one edge of the Tetrahedral code. Logical Z-operators have minimal weight 7 and correspond to any side of the tetrahedron. They propagate to weight-3 logical Z-operators on the Steane code. These two propagation directions are illustrated with example operators in Fig. 4a. Since *all* logical operators propagate like this up to stabilizer-equivalences (or do not propagate at all), Eq. (4) implements the map

$$\begin{aligned}
\overline{X}_\mathrm{S} &\to \overline{X}_\mathrm{S}\overline{X}_\mathrm{T} \\
\overline{X}_\mathrm{T} &\to \overline{X}_\mathrm{T} \\
\overline{Z}_\mathrm{S} &\to \overline{Z}_\mathrm{S} \\
\overline{Z}_\mathrm{T} &\to \overline{Z}_\mathrm{S}\overline{Z}_\mathrm{T},
\end{aligned} \tag{7}$$

which is the action of a logical $\mathrm{C_SNOT_T}$ gate between a Steane code (S) and a Tetrahedral code (T) state.

Equipped with this logical CNOT gate, we now consider the FT T-gate implementation based on two FT code switching steps, depicted in Fig. 1. First, a logical zero state $|0\rangle_\mathrm{T}$ is prepared in the Tetrahedral code. Then, a logical state teleportation step moves the arbitrary input state $|\psi\rangle_\mathrm{S}$, encoded in the Steane code, onto the upper logical wire by the transversal logical CNOT gate, transversal measurements and transversal correction operations that are classically-conditioned on the logical measurement outcome. On the Tetrahedral code, the logical T-gate can simply be executed by performing fifteen physical X-rotations $R^X(\theta)$ of angles $\theta = \pm\pi/4$ since $\overline{T}_X = R^X_{\{0,2,4,6,8,10,11,14\}}(\pi/4)R^X_{\{1,3,5,7,9,12,13\}}(-\pi/4)$ [39]. After preparing a logical auxiliary qubit in the $|+\rangle_\mathrm{S}$ state of the Steane code, the second code switching step moves the logical state back to the lower logical wire. Note that the logical CNOT is connected in the same direction as for the first step. Transversal measurements of the Tetrahedral code state are performed and a conditional logical X-correction may be applied in the teleportation subroutine. As a result, the logical T-gate

$$\overline{T}_X = \cos(\pi/8) - \mathrm{i}\sin(\pi/8)\,\overline{X}_\mathrm{S} \tag{8}$$

has been applied to the arbitrary Steane code state $|\psi\rangle_\mathrm{S} = \alpha\,|0\rangle_\mathrm{S} + \beta\,|1\rangle_\mathrm{S}$.

We now also consider the propagation of errors. It is obvious that a single correctable error in the Steane code can, due to transversality, at most propagate to a single error on the Tetrahedral code, which is therein also correctable. Correctable Z-errors of weight-3 in the Tetrahedral code may either propagate to correctable weight-1 errors in the Steane code through a single physical CNOT gate but can also, and more severely, directly propagate to uncorrectable weight-3 Z-errors on the Steane code, as shown in Fig. 4b. Their appearance must therefore be prevented by an appropriate FT state preparation circuit. It is not enough to only fault-tolerantly prepare the 3D color code state but the state preparation routine must be FT with respect to the 2D color code as well in order to render the whole code switching scheme FT.

It comes in handy that some correctable errors can be physically removed within a single code switching step at no extra cost: Instead of just conditionally applying logical operators during the teleportation part, one may also infer syndromes from the classical measurement result to adapt the feed-forward logical operation to include error correction. For instance, a single $Z_0(X_0)$-error after preparation of $|0\rangle_\mathrm{T}$ ($|+\rangle_\mathrm{S}$) propagates down (up) through the subsequent CNOT gate and will cause the non-trivial $X(Z)$-syndrome $\{+1,-1,+1\}$ ($\{-1,+1,+1,+1\}$) on the Steane (Tetrahedral) code, which can be obtained in classical post-processing from the 7(15)-bit string output by the transversal $X(Z)$-basis measurement. Instead of, say, the logical operators $Z^{\otimes 15}(X^{\otimes 7})$, in this particular case we would apply $Z^{\otimes 15}/Z_0 = Z_1Z_2\cdots Z_{14}(X^{\otimes 7}/X_0 = X_1X_2\cdots X_6)$ in case the logical measurement yields 1 and $Z_0(X_0)$ in case the logical measurement yields 0.

Not *all* correctable errors can be removed this way but only the ones that are detectable by the respective measurements. One example is the single error Z_2. Since the logical CNOT gate has no support on qubit 2 of the Tetrahedral code, see Eq. (4), the error Z_2 after preparing $|0\rangle_\mathrm{T}$ does not propagate to the Steane code at all and hence can not be detected and corrected via the transversal X-measurement on the Steane code qubit. Advantageously however, an error Z_2 after the transversal T-gate also does not propagate to the Steane code in the second switching step (but a Z_0 error would propagate, undetected by the transversal Z-measurements).

2.2.2 FT state preparation

As stressed in Ref. [41], FT state preparation of the 3D color code state must be reviewed in the light of

switching to the 2D color code: While the 2D color code corrects $t = \lfloor \frac{d-1}{2} \rfloor$ errors of each Pauli type X or Z, the corresponding 3D color code corrects t errors of one Pauli type and a larger number of errors for the respective other type. This implies that one type of high-weight errors – correctable in the 3D color code – might propagate onto the 2D color code where they are uncorrectable but cause logical failure. Such high-weight errors should be avoided by a more restrictive FT state preparation of the 3D color code auxiliary qubit [15, 50, 51].

Regarding the Tetrahedral code, the switching scheme only requires preparation of the logical zero state (see Fig. 1). A unitary encoding circuit, constructed via the Latin rectangle method, amended by a single flag qubit that catches one type of Pauli faults has been given before [41]. While this construction would suffice to fault-tolerantly prepare the state $|0\rangle_{\mathrm{T}}$, it is not enough to ensure fault tolerance of the full code switching scheme. Three kinds of dangerous faults must be prevented in our scheme and we now describe these in detail.

1) High-weight X-errors: Since the Tetrahedral code can only correct a single X-error, all uncorrectable X-errors of weight 2 or higher must be sorted out. This can be done by an appropriate flag qubit that measures the logical Z-operator $Z_1Z_3Z_5Z_7Z_9Z_{12}Z_{13}$ [41].

2) High-weight Z-errors: Z-errors of weight larger than 1 on $|0\rangle_{\mathrm{T}}$ must be prevented from propagating to the Steane code during the course of code switching although they could be correctable in the Tetrahedral code. It is clear from Figs. 1 and 4b that propagation of such an error to the Steane code via the first logical CNOT gate, may cause a logically-flipped measurement result and subsequent erroneous application of a logical Z-operator. We find that, in our state preparation subroutine, three flags are necessary to catch all high-weight Z-errors. These flags can be chosen to measure the stabilizers $X_8X_9X_{11}X_{12}$, $X_2X_9X_{12}X_{14}$ and $X_0X_3X_6X_7$.

3) Mixed-type X_iZ_j-errors with $i \neq j$: Although in CSS codes a single X-error on qubit i and another single Z-error on qubit j can in principle be corrected independently, our FT T-gate circuit may not respect the separation of errors into distinct X- and Z-sectors. Since we apply physical T-gates to the 15-qubit code state, the logical T-gate is not a Clifford circuit. A single Z-error on qubit j is mapped by a single T-gate[2] to the operator $(Z_j - Y_j)/\sqrt{2}$. Combined with another single X-operator on qubit i, there is a finite probability that the error superposition in conjunction with such X-error has the effect of a weight-2 uncorrectable error X_iX_j ($i \neq j$), which breaks fault tolerance. If not sorted out, such an error would erroneously flip the logical Z-measurement after the second logical CNOT in Fig. 1 and a logical X-operator would wrongly be applied when switching back to the Steane code. An additional flag qubit that measures the complementary Z-operator $Z_0Z_2Z_4Z_6Z_8Z_{10}Z_{11}Z_{14}$ achieves this goal. As mentioned above, some weight-1 Z-errors propagate through the first logical CNOT of Fig. 1 and could be corrected in the teleportation step by decoding the measurement result of the Steane code logical qubit. Since the logical CNOT is only comprised of seven physical CNOT gates, unfortunately not *all* weight-1 Z-errors can be eliminated this way to prevent the occurrence of mixed-type X_iZ_j-errors.

In App. A, we give the non-FT state preparation circuit (Fig. 12) and the total flag verification circuit (Fig. 13), which together render the logical auxiliary state preparation subroutine FT for transversal code switching. Note that measurements of the three X-flags with bare auxiliary qubits could again cause propagation of X-faults back to high-weight X-errors on the data qubits. Therefore two additional CNOT gates per flag measurement are required that again act as flag qubits in these subcircuits. However, no new qubits are needed but the measurement qubits employed previously to flag high-weight X-errors can be re-used here without reset. These three X-flags are also tailored to detect high-weight Z-errors that may have been propagated from the measurement qubits for the two Z-operators described above.

For practical purposes, we suggest to use these circuits for non-deterministic logical state preparation because the acceptance rate of approximately 95%[3] is reasonably close to unity for state-of-the-art quantum processors where the entangling gate error rate of order $p_2 \approx 10^{-3}$ is the limiting factor [52]. No mid-circuit measurements or branching logic are required to correctly flag all dangerous faults. Deterministic variants of logical state preparation might as well be employed but we suspect this approach to require a larger number of entangling gates in general for low physical error rates.

At this point, we already notice that a large portion of the overall noise will stem from the logical state preparation of the triorthogonal, i.e. the Tetrahedral, code state because the largest fraction of entangling gates is required in this step. In the distance-3 implementation, 69 CNOT gates out of 83 CNOT gates for the whole scheme are taken up by state preparation of both the Tetrahedral code state and the Steane code state, which we teleport onto in the second code switching step. In spite of this gate overhead, performing logical state teleportation onto a (relatively) fresh set of auxiliary qubits could yield lower failure rates than keeping a logical state on the same physical qubits throughout the whole scheme, as, e.g., in

[2]Remember that our physical T-gates are X-rotations.

[3]This value is estimated from a circuit-level depolarizing noise model with parameters $p_1 = p_i = p_m = 10^{-4}$ and $p_2 = 10^{-3}$. More detail will be provided in Sec. 3.

FT magic state preparation [6, 12, 53], because intermediate errors may be erased this way; similar to teleportation-based Knill-type QEC compared to Steane-EC [10].

2.3 Parallelized implementation

Transversality arguably constitutes the simplest method to render logical gates fault-tolerant. Not only are transversal gates conceptually elegant but they also do not require any additional qubit or repetition overhead. A small number of physical gates is typically sufficient to implement a transversal gate that appears hard to beat with alternative FT gate constructions. Within physical hardware architectures that carry out quantum error correction routines relying on transversal CNOT gates anyway, it is advisable to also use our transversal code switching scheme to implement a universal set of gates. It has already been shown that using transversal gates in general and especially a transversal entangling gate can have advantages in hardware platforms that potentially allow for highly parallelized gate operations such as neutral atom [22, 24, 54] or ion trap devices [55–58].

Given logical auxiliary qubits and parallelized physical-level gates, the transversal code switching scheme can be implemented in constant circuit depth w.r.t. the code size. The first logical CNOT requires one time step, the transversal measurements and corrections require one time step each, the physical T-gates are all applied in parallel and the second code switching step then is done again in three time steps; the full logical T-gate circuitry is thus performed in seven steps regardless of the code distances, assuming a supply of logical auxiliary qubits is available. Unitary encoding circuits for the preparation of n-qubit logical auxiliary states with depth $\mathcal{O}(n)$ can be found for 2D topological codes with nearest-neighbor connectivity [59] and might be improved to $\mathcal{O}(\log n)$ by allowing for non-local qubit connectivity, exploiting symmetries [60] or with the help of physical auxiliary qubits [61].

We emphasize that the logical qubit states have an auxiliary function only. This means that the physical qubits that hold these states can in principle be reused an arbitrary number of times. Also, if the physical hardware architecture is capable of such operations, many logical auxiliary qubit states for both codes could be prepared and stored in separate regions of the quantum computer in parallel and be supplied to the processing unit on demand. While conceptually similar to the idea of magic state factories, only Clifford circuits are necessary to prepare such logical auxiliary qubit states. Since our auxiliary states are stabilizer states, no-go theorems related to the non-reusability of magic states do not apply [62].

In a practical implementation, the time t_M to perform a mid-circuit measurement might be much longer than the time t_G it takes to apply a single two-qubit gate. Therefore, the relatively large gate overhead induced by the necessity to prepare logical auxiliary qubits compared to the one-way transversal CNOT gate may not slow down the logical T-gate at all as long as the state preparation time $t_G \times L$ is smaller than the measurement time t_M, where L is the number of CNOT layers in the logical auxiliary state preparation subroutine. Especially for quantum processors based on neutral atoms, where entangling gates only take time on the order of nanoseconds but measurements require several hundred microseconds, execution of the transversal code switching protocol will be limited by the mid-circuit measurement for low-distance codes [24, 54]. In modern ion trap quantum processors on the other hand, single-qubit gates are generally fast but entangling gates and measurements can both take several hundred microseconds such that, here, offline preparation of logical auxiliary qubit states may provide an advantage.

2.4 Limited connectivity

Today's quantum computers are to varying degrees limited in their capabilities to perform entangling gates between arbitrary pairs of the individual qubits. We now briefly discuss some implications from limited connectivity between physical qubits in the most prevalent types of quantum computing hardware.

2.4.1 Superconducting architecture

State-of-the-art superconducting architectures are essentially restricted to nearest-neighbor connectivity and long-range coupling is a work in progress [21, 63]. One option to practically implement transversal entangling gates nonetheless is to stack two 2D layers of superconducting transmon qubits and drive the nearest-neighbor entangling gate in the third, interlayer, spatial dimension. If the spatial dimensionality shall be retained, one can instead make use of a lattice surgery protocol to perform logical CNOT gates at the price of an increased FT overhead [30].

The long-range connectivity required for unitary encoding circuits could in theory be circumvented with the help of circuit knitting [64]. One may resort to a stabilizer-measurement-based state preparation routine altogether if non-destructive operator measurements can be executed fast and with high fidelity. These can be executed largely in parallel on non-overlapping stabilizers. Parallelization of flag-FT stabilizer measurements has been recently shown for distance-5 codes [20]. Also on stabilizers, which overlap on a number of data qubits, measurement circuits may be executed in parallel with appropriate gate schedules. Note that in a stabilizer-measurement-based state preparation routine, entangling gates are

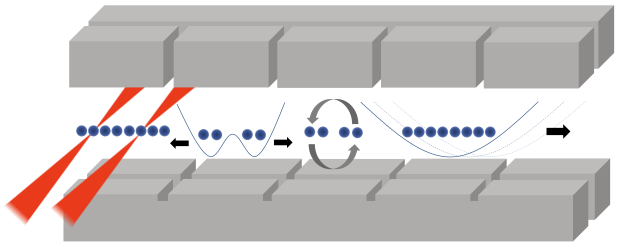

Figure 5: Illustration of a segmented ion trap with five zones (grey bars), each holding a linear ion crystal (blue dots). Focused laser beams can be pointed at any pair of ions to perform an entangling gate (red). Any ion chain can be reconfigured within a single crystal (grey arrows) and can be shuttled from one zone to another (black arrow) by modifying the trapping potential (blue lines).

always only applied between data qubits and auxiliary qubits but never between two data qubits. As a consequence, the occurrence of dangerous mixed-type X_iZ_j errors, as described in Sec. 2.2.2, is naturally prevented for $d = 3$ without additional flag qubits (apart from those that might be needed for FT stabilizer measurements, depending on the specific code).

2.4.2 Neutral atom platform

Atomic platforms natively implement long-range interactions due to their ability to shuttle large registers of atoms in parallel combined with entangling operations via Rydberg states [65, 66]. We suggest that our scheme, and especially the transversal CNOT gate, would ideally be implemented in a neutral atom quantum processor [22]. Their often-claimed "all-to-all" connectivity might be limited in practice by the necessity to physically move qubits into close proximity because the physical distance to cover by such operations grows with the code distance.

While a relatively small code state could be prepared with flag qubits, we alternatively envision a Steane-type verification of logical auxiliary qubit states, which leverages the parallelized transversal CNOT gate, as demonstrated recently [24]. Here, several $|0\rangle_{\mathrm{T}}$ states are prepared non-fault-tolerantly and are then connected via transversal CNOT gates. Transversal measurements can be used to verify the absence of uncorrectable errors. Also, a combination of, say, flag qubits that catch X-errors and a half-cycle of Steane-type EC that removes all correctable Z-errors could be advantageous in practice.

2.4.3 Ion trap processor

Entangling gates in trapped-ion quantum processors, such as the one sketched in Fig. 5, are conducted by leveraging the Coulomb interaction between the electrically charged particles [67–69]. This way one achieves the ability to entangle arbitrary pairs of ion qubits in a single ion crystal. Technical problems may arise when aiming to execute such geometric phase gates with high fidelity once more than 10-20 ions are trapped in the same potential well.

Our transversal code switching scheme can be conveniently partitioned into a segmented ion trap architecture where several ion crystals are connected via junctions that allow one to physically move ions between laser-interaction zones. We provide a detailed example of partitioning the $d = 3$ instance of transversal code switching into a segmented ion trap in App. B.

The transversal CNOT gate can be performed with any number of trap segments as long as one control qubit and its corresponding target qubit can be brought into same zone. In the extremal case of two-ion crystals, all physical entangling gates could be performed in parallel in just a single time step. For an implementation in just two size-n ion crystals, $\lceil n/2 \rceil$ sequential physical entangling gates must be performed (see Fig. 14).

Regarding logical auxiliary state preparation, let us consider unitary encoding circuits created with the Latin rectangle method or related techniques [70, 71]. To construct these densely connected circuits, for each X-type stabilizer a single physical qubit is initialized in the $|+\rangle$ state and serves as control qubit for several physical CNOT gates connected to the rest of the qubits that form the stabilizer. In general, we can assign a trap segment to each control qubit/stabilizer[4]. Then, by devising a shuttling schedule that allows each target qubit to get entangled with the control qubit that represents the stabilizer it belongs to, the state preparation circuit can be implemented with the help of ion reconfigurations. Additional shuttling moves that enable measurement of logical operators, as for flag verification, might be required. An embedding to initialize the logical zero state of the Tetrahedral code is given in App. B.

On a more general note, when performing state preparation via stabilizer measurements, it is recommended to assign qubits of disjoint stabilizer operators to ions in different gate zones in order to be able to measure them in parallel. This works as long as the highest-weight stabilizer to be measured fits into a single trap segment.

Wrapping up, we point out that all numerical tools are publicly available to construct the code switching protocol on the physical qubit level. The python package *StabGraph* can be used to find unitary encoding circuits [72, 73]. Fault-tolerant versions of these circuits may be found in an automated way with the help of Refs. [74–76]. A shuttling compiler assists in devising a concrete schedule of moving ions or atoms with short shuttling paths [77, 78].

[4]For this reason, the original Tetrahedral code defined by four X- and ten Z-stabilizers [41] might be preferable to reduce the number of CNOT gates and ion reconfigurations if this state preparation method is employed.

3 Numerical estimation of logical failure rates

In this section we investigate the logical failure rate of the fault-tolerant T-gate implementation for the Steane code enabled via transversal code switching. The relatively low number of physical CNOT gates and its transversality properties play out in favor of achieving lower logical failure rates than the state-of-the-art flag-FT code switching scheme [41, 42] and even magic state injection [6, 52]. We do not include the intermediate error correction, described in Sec. 2.2.1, in our numerical analysis.

In what follows we first employ a generic, architecture-agnostic depolarizing circuit-level noise model with a single parameter p, which represents the probability of applying faults to circuit operations. In particular, we apply one of the three possible Pauli operators X, Y or Z after any physical single-qubit gate, state initialization or before a measurement according to the channel $\mathcal{E}(\rho) = (1-p)\rho + \frac{p}{3}(X\rho X + Y\rho Y + Z\rho Z)$. In addition, physical two-qubit gates are followed by a two-qubit fault operator that is randomly drawn from the set $\Lambda = \{I, X, Y, Z\} \otimes \{I, X, Y, Z\} \setminus \{I \otimes I\}$ as prescribed by the depolarizing channel $\mathcal{E}(\rho) = (1-p)\rho + \frac{p}{15}\sum_{P\in\Lambda} P\rho P$.

Subsequently, we employ a more detailed incoherent multi-parameter Pauli noise model. It has been shown before that such twirled noise models are capable of approximating logical failure rates in state-of-the-art ion trap quantum processors well enough [6] to estimate the scaling behavior of the noisy circuit's failure rate in the low-error-rate-limit and allow to conduct a break-even analysis [52], while granting numerically efficient classical stabilizer simulations [79]. In our noise model we include depolarizing noise with four different parameters on single-qubit gates (p_1), two-qubit gates (p_2), initializations (p_i) and measurements (p_m) as well as different error rates on idling locations that belong to different types of operations ($p_{\text{idle},1}, p_{\text{idle},2}, p_{\text{idle},m}$ respectively), during which an idling qubit experiences a dephasing channel $\mathcal{E}(\rho) = (1-p_{\text{idle}})\rho + p_{\text{idle}} Z\rho Z$. In an ion trap, this could be caused by magnetic field fluctuations that are assumed to be uncorrelated between individual ion positions.

Two sets of error rates, called *high* and *low*, that we use for our analysis are given in Tab. 1. Statevector simulations in `PECOS` [80, 81] of the logical T-gate from Fig. 1 applied to the logical zero state of the Steane code yield failure rates of $(16.6 \pm 1.2) \times 10^{-2}$ and $(8.2 \pm 1.6) \times 10^{-4}$ for the two parameter sets respectively. We find that a stabilizer simulation of the Clifford circuit where we replace the physical T-gates with noisy identity gates and apply the circuit to the logical Y-eigenstate $|{+\mathrm{i}}\rangle_{\mathrm{S}} = (|0\rangle_{\mathrm{S}} + \mathrm{i}\,|1\rangle_{\mathrm{S}})/\sqrt{2}$ yields logical failure rates $(15.3 \pm 1.1) \times 10^{-2}$ and $(6.2 \pm 0.8) \times 10^{-4}$. These are in good agreement with the statevector simulation results. The state $|{+\mathrm{i}}\rangle_{\mathrm{S}}$ is susceptible to both X- and Z-errors, just as the intermediate non-stabilizer state generated in the code switching scheme. We therefore use the stabilizer simulation as a proxy to full statevector simulations in order to allow for faster numerical simulations. The following simulation results are obtained with the python package `qsample` [82, 83]. All errorbars in this work are 68% confidence intervals.

Rate set	*high* ($T_2 = 100\,\text{ms}$)	*low* ($T_2 = 2\,\text{s}$)
p_1	5×10^{-3}	10^{-4}
p_2	2.5×10^{-2}	10^{-3}
p_i	4.5×10^{-3}	10^{-4}
p_m	4.5×10^{-3}	10^{-4}
$p_{\text{idle},1}$	7.5×10^{-5}	3.75×10^{-6}
$p_{\text{idle},2}$	10^{-3}	10^{-4}
$p_{\text{idle},m}$	1.5×10^{-3}	10^{-4}

Table 1: We perform numerical simulations of incoherent Pauli noise with two sets of parameters that are representative for state-of-the-art or near-term ion trap quantum processors. The operation times t translate to physical idling error rates p_{idle} via the coherence time T_2 as $p_{\text{idle}} = [1 - \exp(-t/T_2)]/2$ for a dephasing channel applied on idling qubits. We assume operation times $t_1 = 15\,\mu\text{s}, t_2 = 200\,\mu\text{s}$ and $t_m = 300\,\mu\text{s}$ for the *high* rates and $t_1 = 15\,\mu\text{s}$ and $t_2 = t_m = 400\,\mu\text{s}$ for the *low* rates.

3.1 Depolarizing noise model

We show numerical estimations of logical failure rates for the single-parameter depolarizing noise model in Fig. 6. Both a direct Monte Carlo (MC) sampling approach and dynamical subset sampling (DSS) are employed and yield consistent results. The former is used to accurately sample data points at relatively large p where the sampling error can be made small with at most 10^6 shots. The latter enables extracting the quadratic low-p scaling behavior analytically from only a small number of 10^3 shots (see Ref. [82] for details).

The break-even point where $p_L = p$ can be identified at $p \approx 2 \times 10^{-3}$. This point is a meaningful comparison of an FT vs. physical-qubit implementation in the following sense: In a current or near-term quantum processor, a quantum algorithm performed with physical qubits will be dominated by the entangling gate error $p \sim p_2$. For an implementation with 2D color codes, a quantum algorithm performed with logical qubits will be dominated by the logical T-gate error since this is by far the most complicated logical gate to execute for this class of QEC codes. Note that this break-even point is very close to the threshold of QEC cycles with the 2D color code under circuit-level depolarizing noise, $p \approx 2 - 3 \times 10^{-3}$, reported in

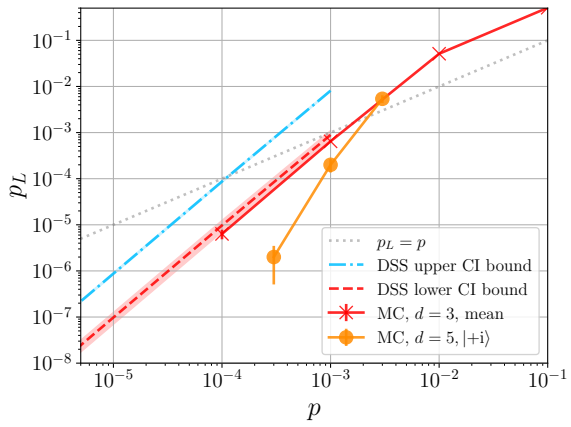

Figure 6: Logical failure rate estimation of the logical T-gate for a single-parameter depolarizing noise model. Direct Monte Carlo (MC) simulation with up to 10^6 shots suggests that the break-even point $p_L = p$ lies at $p \approx 2 \times 10^{-3}$ when averaging failure rates (red line with crosses) for the three logical input states $|+\rangle_S, |+i\rangle_S$ and $|0\rangle_S$. Individual failure rates are $(3.1 \pm 0.2) \times 10^{-4}, (9.3 \pm 0.5) \times 10^{-4}$ and $(7.0 \pm 0.3) \times 10^{-4}$ respectively at $p = 10^{-3}$. For the $[[17, 1, 5]]$ code, discussed in Sec. 5, we observe that a better performance can be expected for the $|+i\rangle_S$ state, which is the most noise-susceptible of the three Pauli eigenstates, at $p = 10^{-3}$ (orange line with circles). With 10^3 shots of dynamical subset sampling (DSS), we extract the low-p scaling behavior $p_L = \mathcal{O}(p^2)$ for the FT protocol of distance $d = 3$. In this regime, the MC logical failure rate estimation coincides with the lower bound of the DSS confidence interval (CI, dashed line). The relatively large gap to the DSS upper CI bound (dash-dotted line) is an artifact of the DSS implementation in `qsample`.

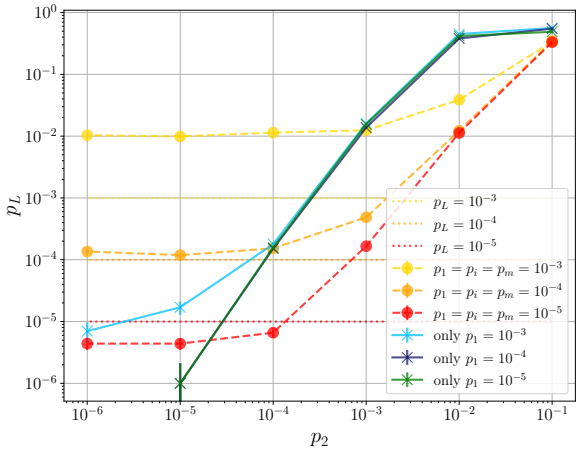

Figure 7: Logical failure rate estimation of the logical T-gate for various ion trap parameters. Dashed lines with circles represent a scenario where all but the entangling gate error rate p_2 are fixed and p_2 is varied. For all these curves we set $p_{\text{idle}} = 10^{-4}$. At low p_2, we observe saturation since noisy single-qubit operations do not permit a further decrease of the logical failure rate. When we also scale the other parameters $p_i = p_m = p_{\text{idle}} = p_2$ and *only* p_1 is kept fixed, as shown by the solid lines with crosses, logical failure rates can be suppressed below the error rate of a physical T-gate. For instance, $p_2 \leq 2 \times 10^{-4}$ (7×10^{-5}) is sufficient to yield $p_L < p_1 = 10^{-3}$ (10^{-4}).

Refs. [84–86], which reflects the low overhead of the code switching scheme and its advantageous design for operational noise.

Simulation of a distance-5 version of the protocol yields a lower logical failure rate than the distance-3 implementation for a physical error rate below $p \approx 2 \times 10^{-3}$. A more detailed discussion of the distance-5 simulation is postponed to Sec. 5.

3.2 Multi-parameter noise model

Let us now employ a more elaborate noise model described by operational noise as well as dephasing noise on idling qubits. We restrict ourselves to a single idling noise rate p_{idle} in the following since the dominant idling noise stems from waiting during relatively slow two-qubit gates and measurements.

Figure 7 depicts a collection of scaling different parameter subsets. When only the entangling gate error rate p_2 is scaled and all other parameters are held constant, the failure rate of the logical T-gate exhibits a noise floor that cannot be undercut. This noise floor is determined by the number of uncorrectable errors that stem from two single-qubit operation/idling locations. Especially, for our example rates $p_1 = p_i = p_m = 10^{-3}$ and 10^{-4}, we see that the logical failure rate does not drop below the physical T-gate error rate p_1. Only $p_1 = p_i = p_m = 10^{-5}$ is small enough so that the total scheme can actually yield a lower logical failure rate. We keep $p_{\text{idle}} = 10^{-4}$ fixed for all these three cases.

In a different scenario, we only keep p_1 fixed and scale all other error rates $p_2 = p_i = p_m = p_{\text{idle}}$ uniformly. A clear advantage $p_L < p_1$ can be seen in this case in Fig. 7 as soon as p_2 drops below 2×10^{-4} (7×10^{-5}) for $p_1 = 10^{-3}$ (10^{-4}) although here, also, a saturation of logical failure rates seems to set in eventually as $p_2 \to 0$. The logical T-gate performs advantageously even when comparing to physical T-gates that only fail with a rate of $p_1 = 10^{-5}$.

We acknowledge that reaching entangling gate error rates this low presents considerable experimental challenges. Note, however, that it is not strictly deemed necessary to run a logical T-gate with lower failure rate than the physical T-gate for a fault tolerance advantage. It is sufficient that the failure rate of a logical state after applying the logical T-gate is low enough such that it can be further reduced by a subsequent round of QEC that will likely be required in any practical algorithm performed with logical qubits anyway.

4 Comparison to other universality strategies

In this section we provide a quantitative comparison of our transversal code switching scheme with other state-of-the-art techniques to achieve an FT universal gate set: Flag-based code switching in Sec. 4.1, magic state injection in Sec. 4.2 and concatenation in Sec. 4.3. All three techniques allow one to establish FT circuits on the physical level without employing complicated magic state distillation schemes, which are only FT when applied to logical qubits. A comparison of logical failure rates and resource count for specific small instances of these schemes is given in Tab. 2. Transversal code switching from the Steane code to the Tetrahedral code and back requires at least 24 qubits[5] and 83 CNOT gates.

For comparison with flag-based code switching, we consult the numerical simulation results for logical failure rates that are given in Ref. [41]. The authors use a similar noise model but different error rates so that we only extract a rough lower bound estimation for our noise parameters based on their results. They conclude that flag-based code switching does not outperform FT magic state preparation and injection in near-term devices.

To compare our scheme to FT magic state preparation and injection [6], which implements a logical $\pi/4$-rotation about the Y-axis $T_Y = R^Y(\pi/4)$, we run our own simulations with the circuitry and noise model of Ref. [6] and report the mean logical infidelity of the states $T_Y|0\rangle_S$ and $T_Y|1\rangle_S$ in Tab. 2.

The comparison to a concatenation-enabled universal set of gates indicates that a prohibitive number of CNOT gates and physical qubits would be required to implement such a scheme with state-of-the-art quantum processors. Therefore, we refrain from explicitly investigating their logical failures rates numerically in this work.

4.1 Flag-style code switching

The first fully FT code switching scheme was given in Ref. [41] and shortly after demonstrated experimentally [42]. Here, the authors provide deterministic protocols to switch between the Steane code and the Tetrahedral code that rely on stabilizer measurements with flag circuits, sketched in Fig. 8a, and the application of gauge operators in the subsystem code that contains both stabilizer codes, each as a particular gauge. A large reduction of resource overhead is achieved by allowing for post-selection and using a morphed 10-qubit code that has distance $d = 2$ and therefore only serves to detect errors but can not correct them [88].

Using the Tetrahedral code, the given scheme requires more CNOT gates than our transversal code switching protocol: In case that no faults occur, 108 CNOT gates are needed for this deterministic variant. However, in the pathological case where flags are triggered early in the protocol, a large portion of measurements can subsequently be omitted and in these special cases the protocol might terminate faster and with fewer CNOT gates. This can lead to a better performance at *high* physical error rates. With entangling gates typically being the most noisy operations, we anticipate a worse overall performance of this scheme in the *low* physical error rate regime. However, a major advantage of the protocol for the implementation in small-scale devices is that two auxiliary qubits are sufficient for the flagged stabilizer measurements. Additionally, decoding of higher weight errors is relatively simple; only the flag error set needs to be determined. Since all circuits only contain Clifford gates, this step can be assisted by efficient numerical stabilizer simulations when scaling up to larger codes.

We point out that the number of stabilizers $n - k$ that might need to be measured quickly grows for topological codes with $k = 1$ since the number of qubits n in a 2D code scales as $\mathcal{O}(d^2)$. With a Shor-type measurement routine, each stabilizer needs to be measured $\mathcal{O}(d)$ times to maintain fault tolerance so the total number of CNOT gates is at least $\mathcal{O}(d^3)$, assuming the number of CNOTs per stabilizer measurement is constant as for qLDPC codes. It might also be challenging in practice to measure high-weight operators with flag circuits.

4.2 Magic state injection

Another commonly used technique to implement the T-gate using only Clifford gates is known as magic state preparation and injection [6, 53, 89–91]. Fault-tolerant preparation of a logical magic state is followed by a teleportation circuit, through which the T-gate is injected and the magic state is consumed[6]. For its first FT experimental demonstration, a non-deterministic FT magic state preparation scheme for the Steane code that uses a total of eight flags was employed [6]. Fault-tolerant measurement of the logical Hadamard operator is a centerpiece subroutine of this protocol and depicted in Fig. 8b. A large fraction of runs is discarded due to post-selection but a high-fidelity magic state is achieved this way using only 10 qubits. Subsequently, an additional logical CNOT is required between the Steane code magic

[5] It is assumed that flag qubits are re-used after state preparation of either logical auxiliary qubit, otherwise the qubit count might increase to 27. One may use a completely fresh set of qubits to prepare the Steane code auxiliary qubit in the second switching step if no fast reset operation is available. This would add another 7 data qubits and 1 flag qubit.

[6] Referring back to our transversal code switching scheme, we note that one may prepare a magic state directly in the Tetrahedral code and teleport it one-way to the Steane code to obtain the logical magic state.

$(d=3)$ T-gate via...	Failure rate p_L		#CNOTs	#qubits
	high	*low*		
Transversal CS	$(16 \pm 1) \times 10^{-2}$	$(8 \pm 2) \times 10^{-4}$	83	24
Flag-based CS	$\gtrsim 3 \times 10^{-2}$	$\gtrsim 2 \times 10^{-3}$	108 (70)	17 (12)
Magic state injection	$(4.9 \pm 0.4) \times 10^{-2}$	$(3.1 \pm 0.1) \times 10^{-3}$	119 (55)	18 (16)
Concatenation	-	-	$\gtrsim$350 (154)	105 (70)

Table 2: Comparison of performance and required resources for different small-scale universality strategies. For *low* physical error rates, transversal code switching (CS) yields the lowest logical failure rate, obtained via statevector simulations, but can be improved upon by deterministic flag-based CS or non-deterministic magic state injection for the *high* physical error rate set. Nonetheless, transversal CS is implemented with the lowest number of CNOT gates in case deterministic protocols are used. We assume the typical protocol case where no errors occur for this figure of merit. The values in brackets refer to non-deterministic versions of the indicated schemes, which may save CNOT gates at the price of introducing post-selection. Transversal CS is only mildly non-deterministic in the sense that logical state preparations may need to be repeated in case flags are triggered. The slightly higher number of physical qubits seems acceptable in the light of recent hardware developments.

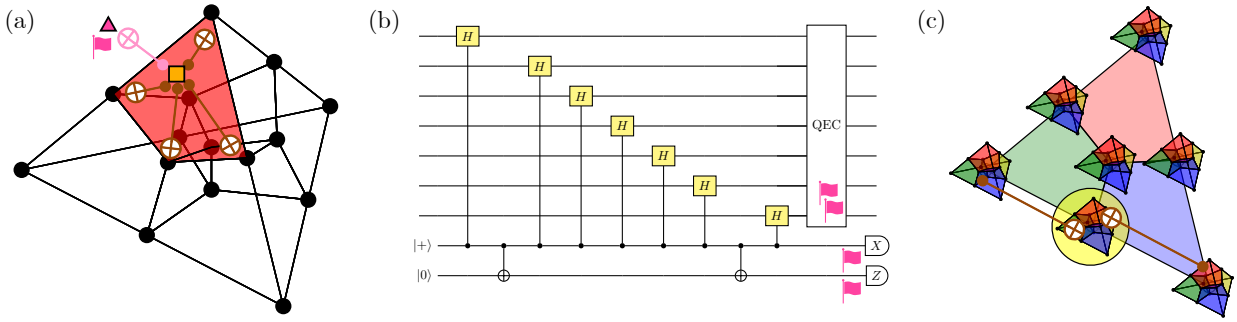

Figure 8: Illustrations of common FT universality strategies. (a) For flag-based code switching, individual stabilizers are measured with the help of single auxiliary qubits (orange square) that are also connected to flag qubits (pink triangle). (b) FT magic state preparation relies on fault-tolerantly measuring an operator whose eigenstate is a magic state. One example is the Hadamard operator, which has eigenbasis $\rho_{\pm H} = (I \pm (X+Z)/\sqrt{2})/2$. (c) Replacing each physical qubit of the Steane code with a logical qubit encoded in the Tetrahedral code, provides a recipe to perform the combined set of transversal gates of both codes fault-tolerantly. The authors of Ref. [87] propose a simplification that uses only three Tetrahedral codes. Fault tolerance must be ensured on the circuit level, for instance by performing a round of QEC on the outer Steane code.

state and the target state $|\psi\rangle_S$, which is encoded in another Steane code. Simulations suggest that the logical state fidelity worsens dramatically if the FT magic state preparation scheme is made deterministic by repeatedly measuring the logical Hadamard operator [52].

Even more flags would be required to go to larger distances [15, 53]. Unfortunately, measurement of the logical Hadamard operator and full rounds of FT QEC need to be repeated $(d-1)/2$ times [92]. Since the controlled-H gate is not a Clifford gate, decoding under circuit-level noise may become more complicated [52].

We note that one may directly opt to use a tri-orthogonal code as the "default" code so that the universal gate set, except the Hadamard gate, is transversal. The Hadamard gate might be performed fault-tolerantly via magic state injection [93], lattice surgery [31, 32] or using logical auxiliary qubits [47].

4.3 Concatenation

By concatenating two different QEC codes that each have a set of transversal gates, one obtains a larger code on which logical gates – previously non-FT on the separated codes – may naturally become FT [94, 95]. A dangerous error on the inner code, resulting in logical failure, can be corrected by the outer code. Faults on the outer code only spread between code blocks of the inner code and can be separately corrected by each instance of the inner code. Decoding of concatenated codes is straight-forward by decoding the individual QEC codes on each level.

If both codes can be implemented in, say, a 1D hardware architecture, then the concatenated code has a natural embedding in 2D. A very simple example for this could be a 1D bitflip code in the horizontal direction and a 1D phaseflip code in the vertical direction of a 2D square lattice, which, concatenated together, can correct an arbitrary Pauli error.

However, a potentially large gate overhead may be induced by the necessity to perform QEC on each level of concatenation. Figure 8c illustrates a well-known implementation of a logical T-gate on the Steane code, which only becomes FT when applied to logical qubits encoded in the Tetrahedral code [94][7]. Note that intermediate rounds of QEC in each of the used codes may further increase the CNOT gate overhead. Crucially, any logical operator at the end of a Tetrahedral code's encoding procedure must be corrected by performing QEC on the Steane code. Otherwise, for instance, a logical Y error on a Tetrahedral code may spread maliciously through the subsequent logical CNOT gates and lead to failure of the Steane code.

We emphasize that, by concatenating two codes to achieve a universal gate set, the minimal distance of the resulting code might not change. Only further concatenation may yield the famous double-exponential suppression of noise [97]. Here lies the central practical problem with concatenation: While, already for distance $d = 3$, the concatenation of the Steane code and the Tetrahedral code (see Fig. 8c) leads to a 105-qubit code – or 70-qubit code when the $[[10, 1, 2]]$ code is used instead of the Tetrahedral code –, the number of qubits keeps scaling exponentially with d [87]. Recent developments suggest, however, that an advantage of concatenated codes might not only be expected asymptotically but also in finite size realizations that could become available in real hardware in the near future [98–100].

5 Scaling up

The challenge to build a scalable quantum computer requires the development of practical schemes that can be adjusted to a given number of qubits at will. Physical hardware should be designed in a modular fashion so that existing small-scale architectures can be upgraded to larger systems in the future. The QEC primitives that can be run on such devices should scale in conjunction with hardware improvements.

Today's hardware typically suffers from deterioration of fidelities of physical operations when adding more qubits to the system [101]. It is already a noticeable experimental achievement to keep physical error rates constant while adding more qubits to the setup [102]. Once physical error rates p fall below the fault tolerance threshold, it is useful to employ codes with larger distances d in order to achieve stronger suppression of noise in their logical failure rates $p_L = \mathcal{O}(p^{\lfloor (d+1)/2 \rfloor})$ as $p \to 0$.

In this section we discuss the scalability of transversal code switching and aspects of embedding it into a 2D hardware layout.

5.1 Larger-distance codes

The number of physical CNOT gates per logical CNOT gate is $n = \mathcal{O}(d^2)$ for the transversal CNOT gate and preparation of an n-qubit logical code state is known to be possible with at most $\mathcal{O}(n^2/\log n)$ physical CNOT gates[8] [79]. Therefore, one can expect the fraction of physical CNOT gates taken up by logical state preparation in transversal code switching to ap-

[7] Concatenation of a Tetrahedral code state with the Steane code can be done via the circuit given in Ref. [96]. We are not aware of a state-agnostic encoding circuit for the Tetrahedral code or the morphed 10-qubit code [88] but use the CNOT count of their logical zero state preparation circuits as a lower bound estimation in Tab. 2.

[8] The logical auxiliary qubit state for the used 3D code requires $n' = \mathcal{O}(d^3) = \mathcal{O}(n^{3/2})$ physical qubits. A potential gate overhead for fault tolerance must be added to the above given CNOT count.

proach 1 when scaling up our scheme if no specialized state preparation subroutine is used.

A stricter definition of fault tolerance is required in order to patch subroutines of FT circuits together into a larger FT algorithm, as is the case for our transversal code switching scheme: Here we combine the FT preparation of logical auxiliary qubits with transversal CNOT gates. The latter are FT in the strictest sense [15, 50, 51]: Any number $s \leq t$ of faults that happen within the transversal CNOT gate leads to at most s errors on any output code state. This need not be true for a flagged encoding circuit. Here, one typically only requires that for all integer number of faults $0 < s \leq t$ within the circuit, the correct logical state is prepared with at most t errors that are correctable by the QEC code. It is obvious that in this scenario the combined execution of state preparation and transversal CNOT may break fault tolerance.

Consider, as an example, a $d = 5$ QEC code that corrects $t = 2$ errors. Say that there exists a single fault in the encoding circuit that causes two errors. For the individual encoding circuit, this is sufficient to call the circuit FT since these two errors will be correctable by the distance-5 code. Now assume that a second fault may happen during the subsequent transversal CNOT such that the total number of errors after both circuits is 3 and can thus lead to failure of the code. In this stricter sense of "sequential fault tolerance", the single fault in the encoding circuit should at most propagate to a single error, despite $t = 2$. This could be achieved by using flag circuits [15] or Steane-type state preparation, as mentioned in Sec. 2.4.

Transversal code switching with distance $d = 5$ can, for instance, be performed with the 2D self-dual $[[17, 1, 5]]$ color code [39] and the triorthogonal $[[49, 1, 5]]$ code [103]. The latter, despite being already established in the literature, can be obtained via code doubling from the $[[15, 1, 3]]$ code. We recommend to consult Ref. [46] for more detail and another example for distance $d = 7$. Additionally, we also consider the triangular $[[19, 1, 5]]$ color code [37, 38] in two spatial dimensions, for which the tetrahedral $[[65, 1, 5]]$ three-dimensional color code [104] can be used to perform the transversal T-gate.

We numerically verify that transversal code switching in both directions indeed works for $d = 5$ with the aforementioned codes and the one-way transversal CNOT gates consisting of 17 (19) physical CNOT gates respectively. To prepare both code switching auxiliary qubits fault-tolerantly, we employ non-FT unitary state preparation followed by Steane-type verification (see Fig. 17). In total, the protocols operate with 164 (217) physical qubits and 2317 (3797) CNOT gates[9] in the typical case where logical auxiliary qubits are verified at the first trial; only 34 (38) CNOTs are used for the actual code switching steps.

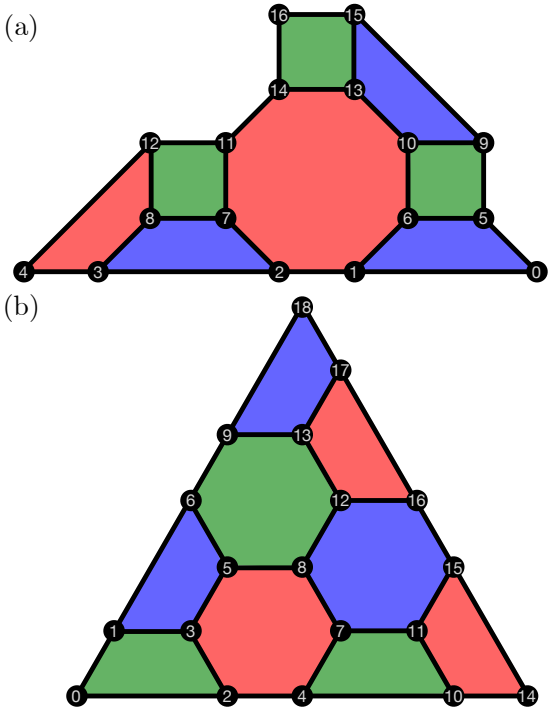

Figure 9: Two common distance-5 2D color codes with qubit labels according to the parity check matrices stated in App. A. (a) A 4.8.8 color code instance with code parameters $[[17, 1, 5]]$. A unitary encoding circuit is given in Fig. 15. Its complementing doubled code, which has a transversal T-gate, is the $[[49, 1, 5]]$ code. (b) The $[[19, 1, 5]]$ color code, also referred to as the 6.6.6 color code, corresponds to a hexagonal tiling of the 2D plane. A unitary encoding circuit is given in Fig. 16. In 3D, a $[[65, 1, 5]]$ triorthogonal code is used to perform the logical T-gate.

We show the scaling behavior $p_L = \mathcal{O}(p^3)$ for the $[[17, 1, 5]]$ code in Fig. 6. The code switching protocol is only applied to the logical input state, which we expect to be least protected by the triorthogonal code, i.e. $|+\mathrm{i}\rangle_S$ (see App. A). It is numerically challenging to collect enough Monte Carlo samples in the low-p regime [82]. We stress that, at a realistically attainable entangling gate error rate of $p_2 = 10^{-3}$, our stabilizer simulations suggest lower logical failure rates for the distance-5 codes, $p_L = (2.0 \pm 0.6) \times 10^{-4}$ and $p_L = (5.8 \pm 1.1) \times 10^{-4}$ respectively, as compared to the distance-3 logical T-gate, which achieves $p_L = (9.3 \pm 0.5) \times 10^{-4}$ for the least protected logical input state[10]. It is expected that the scheme performs slightly worse for the $[[19, 1, 5]]$ code than for the

[9] Our non-FT encoding circuits of the $|0\rangle_S$ state of the $[[17, 1, 5]]$ ($[[19, 1, 5]]$) and the $|+\rangle_T$ state of the $[[49, 1, 5]]$ ($[[65, 1, 5]]$) codes have 36 (45) and 159 (298) CNOT gates respectively. Physical qubits from auxiliary state verification are reset and reused where possible.

[10] For the three input logical Pauli states on the $[[17, 1, 5]]$ code, we find failure rates $(7.0 \pm 2.6) \times 10^{-6}$, $(2.0 \pm 0.6) \times 10^{-4}$ and $(1.5 \pm 0.4) \times 10^{-4}$ respectively. We also point out that our scheme seems competitive with Ref. [92] for $d = 5$, where only non-deterministic FT magic state preparation was considered but not the full T-gate implementation.

[[17, 1, 5]] code due to the larger CNOT gate and qubit overhead. For larger physical error rates p, the acceptance rate of logical auxiliary states vanishes quickly for Steane-type state preparation (see Tab. 3). So, it is imperative to devise FT state preparation circuits for larger-distance codes and methods to synthesize such circuits that are practical for the relatively large numbers of physical qubits required by the appropriate 3D codes [76].

5.2 Two-dimensional hardware layout

As a concrete scenario for operating our proposed scheme on a future scalable architecture, we consider a trapped-ion quantum processor based on one-dimensional segmented ion traps, which are arranged in a two-dimensional square lattice, similar to the architectures described in [105–112] (an alternative could be, e.g., the Quantum Spring Array [58]). Ion shuttling operations enable dynamic reconfiguration of the qubits, establishing effective all-to-all connectivity. On such a layout, we envisage at least two different use cases of the transversal code switching scheme with an arbitrary number of distance-3 logical qubits.

One option would be to keep the ions of the logical qubit $|\psi\rangle_S$ static and move the ions that constitute the Tetrahedral code auxiliary qubit $|0\rangle_T$ into close proximity when a logical T-gate should be executed. The transversal CNOT gate is enabled by local ion crystal reconfigurations that establish the connectivity between the Steane code ions and a suitable subset of the Tetrahedral code ions: One may move some ions of the 15-qubit state into the Steane code zone or move subsets of ions from both crystals into adjacent empty zones in order to apply the appropriate physical entangling gates. After the first code switching step, the Steane-qubit is (re-)initialized to $|+\rangle_S$ on the same physical qubits that held the Steane code state $|\psi\rangle_S$ previously. This is illustrated in Fig. 10a+b. The Tetrahedral code ions may be moved away again after having performed the second code switching step.

Apart from implementing such a stationary logical T-gate fault-tolerantly and enabling universal FT quantum computation via color codes, the teleportation subroutines could be utilized to move the logical Steane code state $|\psi\rangle_S$ through the lattice quickly while applying the FT T-gate, as shown in Fig. 10c+d. Different subsets of physical qubits of the 15-qubit code, corresponding to different sides of the Tetrahedral code, can be used to perform the transversal CNOT gates in the two code switching steps.

The above described routines could be used in parallel in distinct regions of the lattice, allowing one to scale the system up to an arbitrary number of logical qubits that can each be controlled via an FT universal gate set. Since our Tetrahedral code has $d_Z = 7$, the logical auxiliary qubit offers a better protection

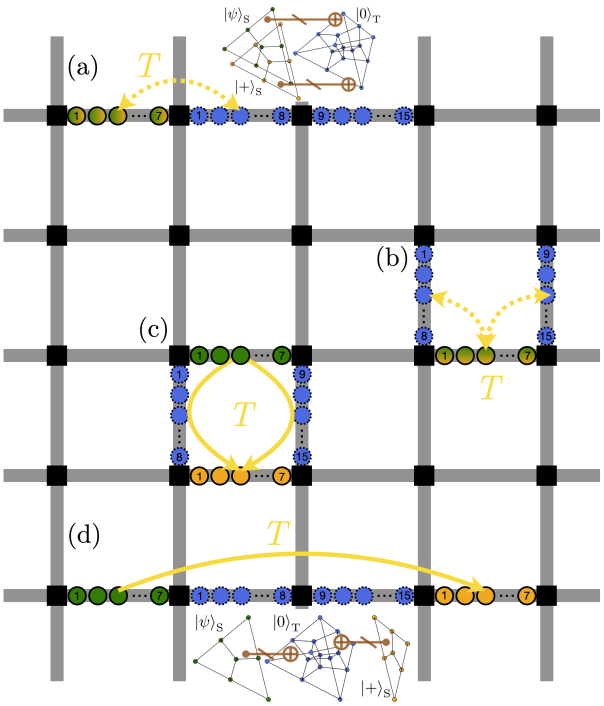

Figure 10: Possible arrangements of 1D ion crystals (colored circles) in a 2D lattice structure where trap segments (grey, cf. Fig. 5) are connected via junctions (black). (a) A linear arrangement of three zones can be used to teleport (arrows) a Steane code state (solid-line circles) to the 15-qubit state (blue dashed-line circles) and back while executing the logical T-gate. We split the 15-qubit state into two ion crystal due to limited control capabilities when crystals are too large (see App. B). These ion chains can also be shuttled close to the Steane code state from two sides as depicted by the configuration in (b). A circular four-zone arrangement (c) can be used to move the initial Steane code state (green, $|\psi\rangle_S$) down by one row (orange, $T|\psi\rangle_S$) while applying the logical T-gate via switching to the 15-qubit state. (d) Arranging the four zones linearly allows one to quickly move the state $|\psi\rangle_S$ three columns to the right while transforming it to $T|\psi\rangle_S$.

against dephasing noise. Up to 3 Z-errors could, in principle, be removed from the logical auxiliary qubit *before* it is coupled to the Steane code state.

6 Conclusions and outlook

In this work, we have investigated a new FT code switching scheme [46] based on transversal CNOT gates, which are applied in one orientation, between the two quantum error correcting codes. While it has been known before that teleportation-based error correction can be combined with logical gate operations [3, 28, 29], the one-way transversal CNOT gate between the Steane code and the Tetrahedral code has been missing to construct a protocol that is practically useful [46].

We focus on using the scheme as an enabler to perform the logical T-gate fault-tolerantly on a low-distance 2D color code state with smaller gate over-

head than previously known deterministic schemes such as flag-FT code switching and FT magic state preparation and injection. The FT code switching protocol has been shown suitable for parallelized implementation in hardware architectures with limited qubit connectivity. Microscopically motivated noise simulations suggest that our logical T-gate could be applied successfully in near-term quantum processors with improved entangling gate error rates and fast measurements on the order of the entangling gate time. Only a moderate increase of qubits is required compared to other state-of-the-art universality strategies when using QEC codes of distance $d = 3$. Since the actual code switching procedure entirely consists of transversal operations, it naturally allows for scaling up to larger code distances. We identify the repetition overhead of "repeat until success" logical auxiliary state preparation as the relevant bottleneck for the practical implementation of distance-5 protocols on near-term hardware.

We have not explored the effect of coherent errors in the transversal code switching circuit since we expect the mid-circuit measurements to destroy such unwanted coherences so that no build-up of noise takes place.

In all shown code instances, we made use of standard look-up-table decoders, which are known to grow exponentially in size when increasing the code distance. Exploring the use of efficient decoders and the potential of correlated decoding in the context of logical teleportation could be a promising direction for further improving scalability by, for instance, simplifying the logical auxiliary state verification [46, 113–115]. Additionally, further attention is needed to optimally leverage the potential of using the logical measurements of the teleportation subroutines for intermediate error correction. Possibilities of re-using the logical auxiliary qubits or at least restore their state after the logical teleportation step with potentially fewer physical operations could be another line of future research.

A measurement-free code switching scheme would facilitate the practical implementation of the FT T-gate in near-term quantum computers by avoiding potentially long measurement times, measurement-induced errors or the need for feed-forward logical operations. We spare some additional thoughts in App. C.

With the help of systematic circuit synthesis tools, such as a recently presented approach based on reinforcement learning [74], a more economical and/or deterministic FT state preparation circuit, for instance using single-shot flags [116], could be discovered. Efficient state preparation circuits would be needed to further decrease logical failure rates and increase acceptance rates of logical auxiliary qubit states in practical implementations beyond distance-3 [76].

Our perspective on concatenated codes might be overly pessimistic as there could exist decoding strategies that allow for the correction of higher-weight errors even if only small-distance QEC codes, such as C_4, C_6 and/or the Steane code as suggested in Ref. [98], are concatenated to obtain an FT universal gate set.

An interesting avenue of future work would be to shed more light on a systematic categorization of appropriate codes for transversal code switching, potentially without the need of explicitly constructing a new doubled code [117]. This should include codes with more than $k = 1$ logical qubit. We speculate that this could be possible for surface-code-like QEC codes, which may even include high-rate qLDPC codes [118], such as hypergraph product codes or lift-connected surface codes [119]. The homomorphic CNOT gate, a generalization of the transversal CNOT gate recently employed to perform generalized lattice surgery with encoded auxiliary qubits, could offer a path forward [120–122].

Despite not being LDPC, an emphasis is put on the distance-7 Golay code in Ref. [46], which could be a good candidate to take the first steps towards larger-distance FT quantum computation once enough physical qubits with (almost) all-to-all connectivity are available in practice [123]. Exploration of systematic scale-up of all transversal operations in the 2D color code, including the transversal code-switching-based T-gate, in an appropriate Clifford+T quantum algorithm that solves a practical problem, should be a future goal.

Code availability

All software code used in this project is available from the corresponding author upon reasonable request.

Author contributions

SH conceptualized the project, analyzed the scheme, designed the circuits and performed all numerical simulations. SH wrote the manuscript with input and feedback from JH.

Acknowledgements

We thank Thomas Scruby, Arthur Pesah, Tom Peham and Janis Wagner for fruitful discussions. We appreciate useful feedback on the manuscript by Ulrich Poschinger and Markus Müller. The python code to create Fig. 2 has initially been drafted with the help of ChatGPT and was then manually adjusted and extended.

JH gratefully acknowledges support by the European Union's Horizon Europe research and innovation program under Grant Agreement Number 101114305 ("MILLENION-SGA1" EU Project) and the US Army

Research Office through Grant Number W911NF-21-1-0007. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of IARPA, the Army Research Office, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

References

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

[2] B. M. Terhal, *Quantum error correction for quantum memories*, Reviews of Modern Physics **87**, 307 (2015).

[3] D. Gottesman, *Fault-tolerant quantum computation with constant overhead* (2014), arXiv:1310.2984 [quant-ph] .

[4] C. Ryan-Anderson, et al., *Realization of real-time fault-tolerant quantum error correction*, Physical Review X **11**, 041058 (2021).

[5] C. Ryan-Anderson, et al., *Implementing fault-tolerant entangling gates on the five-qubit code and the color code* (2022), arXiv:2208.01863 [quant-ph] .

[6] L. Postler, et al., *Demonstration of fault-tolerant universal quantum gate operations*, Nature **605**, 675 (2022).

[7] Google Quantum AI, *Suppressing quantum errors by scaling a surface code logical qubit*, Nature **614**, 676 (2023).

[8] M. P. da Silva, et al., *Demonstration of logical qubits and repeated error correction with better-than-physical error rates* (2024), arXiv:2404.02280 [quant-ph] .

[9] K. Mayer, et al., *Benchmarking logical three-qubit quantum Fourier transform encoded in the Steane code on a trapped-ion quantum computer* (2024), arXiv:2404.08616 [quant-ph] .

[10] C. Ryan-Anderson, et al., *High-fidelity and fault-tolerant teleportation of a logical qubit using transversal gates and lattice surgery on a trapped-ion quantum computer* (2024), arXiv:2404.16728 [quant-ph] .

[11] *Quantum error correction below the surface code threshold*, Nature **638**, 920 (2025).

[12] H. Goto, *Minimizing resource overheads for fault-tolerant preparation of encoded states of the Steane code*, Scientific Reports **6**, 1 (2016).

[13] T. J. Yoder and I. H. Kim, *The surface code with a twist*, Quantum **1**, 2 (2017).

[14] R. Chao and B. W. Reichardt, *Fault-tolerant quantum computation with few qubits*, npj Quantum Information **4**, 42 (2018).

[15] C. Chamberland and M. E. Beverland, *Flag fault-tolerant error correction with arbitrary distance codes*, Quantum **2**, 53 (2018).

[16] B. W. Reichardt, *Fault-tolerant quantum error correction for Steane's seven-qubit color code with few or no extra qubits*, Quantum Science and Technology **6**, 015007 (2020).

[17] P. Prabhu and B. W. Reichardt, *Fault-tolerant syndrome extraction and cat state preparation with fewer qubits*, 16th Conference on the Theory of Quantum Computation, Communication and Cryptography, 5:1 (2021).

[18] D. Bhatnagar, M. Steinberg, D. Elkouss, C. G. Almudever, and S. Feld, *Low-depth flag-style syndrome extraction for small quantum error-correction cdes*, 2023 IEEE International Conference on Quantum Computing and Engineering , 63 (2023).

[19] C. Du, Z. Ma, Y. Liu, Q. Duan, and Y. Fei, *Parallel flagged fault-tolerant error correction for stabilizer codes of distance 5*, Advanced Quantum Technologies , 2300291 (2024).

[20] P.-H. Liou and C.-Y. Lai, *Reducing quantum error correction overhead with versatile flag-sharing syndrome extraction circuits* (2024), arXiv:2407.00607 [quant-ph] .

[21] S. Bravyi, O. Dial, J. M. Gambetta, D. Gil, and Z. Nazario, *The future of quantum computing with superconducting qubits*, Journal of Applied Physics **132** (2022).

[22] D. Bluvstein, et al., *A quantum processor based on coherent transport of entangled atom arrays*, Nature **604**, 451 (2022).

[23] L. Pause, et al., *Supercharged two-dimensional tweezer array with more than 1000 atomic qubits*, Optica **11**, 222 (2024).

[24] D. Bluvstein, et al., *Logical quantum processor based on reconfigurable atom arrays*, Nature **626**, 58 (2024).

[25] H. J. Manetsch, et al., *A tweezer array with 6100 highly coherent atomic qubits* (2024), arXiv:2403.12021 [quant-ph] .

[26] A. M. Steane, *Active stabilization, quantum computation, and quantum state synthesis*, Physical Review Letters **78**, 2252 (1997).

[27] L. Postler, et al., *Demonstration of fault-tolerant Steane quantum error correction*, PRX Quantum **5**, 030326 (2024).

[28] E. Knill, *Scalable quantum computation in the presence of large detected-error rates* (2004), arXiv:quant-ph/0312190 [quant-ph] .

[29] E. Knill, *Quantum computing with realistically noisy devices*, Nature **434**, 39 (2005).

[30] C. Horsman, A. G. Fowler, S. Devitt, and R. Van Meter, *Surface code quantum computing by lattice surgery*, New Journal of Physics **14**, 123011 (2012).

[31] L. Z. Cohen, I. H. Kim, S. D. Bartlett, and B. J. Brown, *Low-overhead fault-tolerant quantum computing using long-range connectivity*, Science Advances **8**, eabn1717 (2022).

[32] A. Cowtan and S. Burton, *CSS code surgery as a universal construction*, Quantum **8**, 1344 (2024).

[33] B. Eastin and E. Knill, *Restrictions on transversal encoded quantum gate sets*, Physical Review Letters **102**, 110502 (2009).

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

[35] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, *Surface codes: Towards practical large-scale quantum computation*, Physical Review A **86**, 032324 (2012).

[36] J. T. Anderson, G. Duclos-Cianci, and D. Poulin, *Fault-tolerant conversion between the Steane and Reed-Muller quantum codes*, Physical Review Letters **113**, 080501 (2014).

[37] H. Bombín, *Gauge color codes: Optimal transversal gates and gauge fixing in topological stabilizer codes*, New Journal of Physics **17**, 083002 (2015).

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

[39] H. Bombín and M. A. Martín-Delgado, *Topological quantum distillation*, Physical Review Letters **97**, 180501 (2006).

[40] H. Bombin and M. A. Martin-Delgado, *Exact topological quantum order in $D = 3$ and beyond: Branyons and brane-net condensates*, Phys. Rev. B **75**, 075103 (2007).

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

[42] I. Pogorelov, et al., *Experimental fault-tolerant code switching*, Nature Physics **21**, 298 (2025).

[43] K. Betsumiya and A. Munemasa, *On triply even binary codes*, Journal of the London Mathematical Society **86**, 1 (2012).

[44] S. Bravyi and A. Cross, *Doubled color codes* (2015), arXiv:1509.03239 [quant-ph] .

[45] M. Shi, H. Lu, J.-L. Kim, and P. Solé, *Triorthogonal codes and self-dual codes*, Quantum Information Processing **23**, 1 (2024).

[46] M. Sullivan, *Code conversion with the quantum Golay code for a universal transversal gate set*, Physical Review A **109**, 042416 (2024).

[47] A. Paetznick and B. W. Reichardt, *Universal fault-tolerant quantum computation with only transversal gates and error correction*, Physical Review Letters **111**, 090505 (2013).

[48] A. Steane, *Multiple-particle interference and quantum error correction*, Proceedings of the Royal Society A **452**, 2551 (1996).

[49] A. M. Steane, *Quantum Reed-Muller codes*, IEEE Transactions on Information Theory **45**, 1701 (1999).

[50] D. Gottesman, *An introduction to quantum error correction and fault-tolerant quantum computation* (2009), arXiv:0904.2557 [quant-ph] .

[51] H. Zhou, et al., *Algorithmic fault tolerance for fast quantum computing* (2024), arXiv:2406.17653 [quant-ph] .

[52] S. Heußen, et al., *Strategies for a practical advantage of fault-tolerant circuit design in noisy trapped-ion quantum computers*, Physical Review A **107**, 042422 (2023).

[53] C. Chamberland and A. W. Cross, *Fault-tolerant magic state preparation with flag qubits*, Quantum **3**, 143 (2019).

[54] S. J. Evered, et al., *High-fidelity parallel entangling gates on a neutral-atom quantum computer*, Nature **622**, 268 (2023).

[55] C. Figgatt, et al., *Parallel entangling operations on a universal ion-trap quantum computer*, Nature **572**, 368 (2019).

[56] J. M. Pino, et al., *Demonstration of the trapped-ion quantum CCD computer architecture*, Nature **592**, 209 (2021).

[57] Y. Zhu, et al., *Pairwise-parallel entangling gates on orthogonal modes in a trapped-ion chain*, Advanced Quantum Technologies **6**, 2300056 (2023).

[58] M. Valentini, et al., *Demonstration of two-dimensional connectivity for a scalable error-corrected ion-trap quantum processor architecture* (2024), arXiv:2406.02406 [quant-ph] .

[59] O. Higgott, et al., *Optimal local unitary encoding circuits for the surface code*, Quantum **5**, 517 (2021).

[60] C. Moore and M. Nilsson, *Parallel quantum computation and quantum codes* (1998), arXiv:quant-ph/9808027 [quant-ph] .

[61] X.-M. Zhang, T. Li, and X. Yuan, *Quantum state preparation with optimal circuit depth: Implementations and applications*, Physical Review Letters **129**, 230504 (2022).

[62] J. T. Anderson, *On the power of reusable magic states* (2012), arXiv:1205.0289 [quant-ph] .

[63] M. Kjaergaard, et al., *Superconducting qubits: Current state of play*, Annual Review of Condensed Matter Physics **11**, 369 (2020).

[64] C. Piveteau and D. Sutter, *Circuit knitting with classical communication*, IEEE Transactions on Information Theory **70**, 2734 (2024).

[65] L. Henriet, et al., *Quantum computing with neutral atoms*, Quantum **4**, 327 (2020).

[66] K. Wintersperger, et al., *Neutral atom quantum computing hardware: Performance and end-*

user perspective, EPJ Quantum Technology **10**, 1 (2023).

[67] K. Mølmer and A. Sørensen, *Multiparticle entanglement of hot trapped ions*, Physical Review Letters **82**, 1835 (1999).

[68] A. Sørensen and K. Mølmer, *Entanglement and quantum computation with ions in thermal motion*, Physical Review A **62**, 022311 (2000).

[69] C. J. Ballance, *High-fidelity quantum logic in Ca^+* (Springer, 2017).

[70] A. M. Steane, *Fast fault-tolerant filtering of quantum codewords* (2004), arXiv:quant-ph/0202036 [quant-ph] .

[71] A. W. Cross, D. P. DiVincenzo, and B. M. Terhal, *A comparative code study for quantum fault-tolerance* (2009), arXiv:0711.1556 [quant-ph] .

[72] D. Amaro, *StabGraph*, `https://github.com/davamaro/stabgraph` (2019).

[73] D. Amaro, M. Müller, and A. K. Pal, *Scalable characterization of localizable entanglement in noisy topological quantum codes*, New Journal of Physics **22**, 053038 (2020).

[74] R. Zen, et al., *Quantum circuit discovery for fault-tolerant logical state preparation with reinforcement learning* (2024), arXiv:2402.17761 [quant-ph] .

[75] R. Zen, *Reinforcement learning for fault-tolerant quantum circuit discovery*, `https://github.com/remmyzen/rlftqc` (2024).

[76] T. Peham, L. Schmid, L. Berent, M. Müller, and R. Wille, *Automated synthesis of fault-tolerant state preparation circuits for quantum error-correction codes*, PRX Quantum **6**, 020330 (2025).

[77] F. Kreppel, et al., *Quantum circuit compiler for a shuttling-based trapped-ion quantum computer*, Quantum **7**, 1176 (2023).

[78] D. Schoenberger, S. Hillmich, M. Brandl, and R. Wille, *Using Boolean satisfiability for exact shuttling in trapped-ion quantum computers*, 29th Asia and South Pacific Design Automation Conference , 127 (2024).

[79] S. Aaronson and D. Gottesman, *Improved simulation of stabilizer circuits*, Physical Review A **70**, 052328 (2004).

[80] C. Ryan-Anderson, *Quantum algorithms, architecture, and error correction*, Ph.D. thesis, The University of New Mexico (2018).

[81] C. Ryan-Anderson, *PECOS: Performance estimator of codes on surfaces*, `https://github.com/PECOS-packages/PECOS` (2019).

[82] S. Heußen, D. Winter, M. Rispler, and M. Müller, *Dynamical subset sampling of quantum error-correcting protocols*, Physical Review Research **6**, 013177 (2024).

[83] D. Winter and S. Heußen, *qsample*, `https://github.com/dpwinter/qsample` (2023).

[84] A. Kubica, *The ABCs of the color code: A study of topological quantum codes as toy models for fault-tolerant quantum computation and quantum phases of matter*, Ph.D. thesis, California Institute of Technology (2018).

[85] C. Chamberland, A. Kubica, T. J. Yoder, and G. Zhu, *Triangular color codes on trivalent graphs with flag qubits*, New Journal of Physics **22**, 023019 (2020).

[86] Y. Takada and K. Fujii, *Improving threshold for fault-tolerant color-code quantum computing by flagged weight optimization*, PRX Quantum **5**, 030352 (2024).

[87] C. Chamberland, T. Jochym-O'Connor, and R. Laflamme, *Overhead analysis of universal concatenated quantum codes*, Physical Review A **95**, 022313 (2017).

[88] M. Vasmer and A. Kubica, *Morphing quantum codes*, PRX Quantum **3**, 030319 (2022).

[89] B. W. Reichardt, *Quantum universality from magic states distillation applied to CSS codes*, Quantum Information Processing **4**, 251 (2005).

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

[91] R. S. Gupta, et al., *Encoding a magic state with beyond break-even fidelity*, Nature **625**, 259–263 (2024).

[92] C. Chamberland and K. Noh, *Very low overhead fault-tolerant magic state preparation using redundant ancilla encoding and flag qubits*, npj Quantum Information **6**, 91 (2020).

[93] B. D. Jones, N. Linden, and P. Skrzypczyk, *The Hadamard gate cannot be replaced by a resource state in universal quantum computation*, Quantum **8**, 1470 (2024).

[94] T. Jochym-O'Connor and R. Laflamme, *Using concatenated quantum codes for universal fault-tolerant quantum gates*, Physical Review Letters **112**, 010505 (2014).

[95] T. J. Yoder, R. Takagi, and I. L. Chuang, *Universal fault-tolerant gates on concatenated stabilizer codes*, Physical Review X **6**, 031039 (2016).

[96] S. D. Buchbinder, C. L. Huang, and Y. S. Weinstein, *Encoding an arbitrary state in a [7, 1, 3] quantum error correction code*, Quantum information processing **12**, 699 (2013).

[97] P. Aliferis, D. Gottesman, and J. Preskill, *Quantum accuracy threshold for concatenated distance-3 codes*, Quantum Information and Computation **6**, 97 (2006).

[98] S. Yoshida, S. Tamiya, and H. Yamasaki, *Concatenate codes, save qubits*, npj Quantum Information **11**, 88 (2025).

[99] H. Goto, *High-performance fault-tolerant quantum computing with many-hypercube codes*, Science Advances **10**, eadp6388 (2024).

[100] H. Yamasaki and M. Koashi, *Time-efficient constant-space-overhead fault-tolerant quantum computation*, Nature Physics , 1 (2024).

[101] A. Katabarwa, K. Gratsea, A. Caesura, and P. D. Johnson, *Early fault-tolerant quantum computing*, PRX Quantum **5**, 020101 (2024).

[102] M. DeCross, et al., *Computational power of random quantum circuits in abitrary geometries*, Physical Review X **15**, 021052 (2025).

[103] S. Bravyi and J. Haah, *Magic-state distillation with low overhead*, Physical Review A **86**, 052329 (2012).

[104] H. Bombin, *Transversal gates and error propagation in 3D topological codes* (2018), arXiv:1810.09575 [quant-ph] .

[105] W. Hensinger, et al., *T-junction ion trap array for two-dimensional ion shuttling, storage, and manipulation*, Applied Physics Letters **88**, 034101 (2006).

[106] R. B. Blakestad, et al., *High-fidelity transport of trapped-ion qubits through an X-junction trap array*, Physical Review Letters **102**, 153002 (2009).

[107] K. Wright, et al., *Reliable transport through a microfabricated X-junction surface-electrode ion trap*, New Journal of Physics **15**, 033004 (2013).

[108] G. Shu, et al., *Heating rates and ion-motion control in a Y-junction surface-electrode trap*, Physical Review A **89**, 062308 (2014).

[109] A. Bautista-Salvador, et al., *Multilayer ion trap technology for scalable quantum computing and quantum simulation*, New Journal of Physics **21**, 043011 (2019).

[110] V. Kaushal, et al., *Shuttling-based trapped-ion quantum information processing*, AVS Quantum Science **2**, 014101 (2020).

[111] M. Malinowski, D. Allcock, and C. Ballance, *How to wire a* 1000*-qubit trapped-ion quantum computer*, PRX Quantum **4**, 040313 (2023).

[112] Z. Cai, A. Siegel, and S. Benjamin, *Looped pipelines enabling effective 3D qubit lattices in a strictly 2D device*, PRX Quantum **4**, 020345 (2023).

[113] M. Cain, et al., *Correlated decoding of logical algorithms with transversal gates* (2024), arXiv:2403.03272 [quant-ph] .

[114] K. H. Wan, M. Webber, A. G. Fowler, and W. K. Hensinger, *An iterative transversal CNOT decoder* (2024), arXiv:2407.20976 [quant-ph] .

[115] K. Sahay, Y. Lin, S. Huang, K. R. Brown, and S. Puri, *Error correction of transversal controlled-NOT gates for scalable surface code computation* (2024), arXiv:2408.01393 [quant-ph] .

[116] S. Veroni, M. Müller, and G. Giudice, *Optimized measurement-free and fault-tolerant quantum error correction for neutral atoms*, Physical Review Research **6**, 043253 (2024).

[117] S. P. Jain and V. V. Albert, *Transversal Clifford and T-Gate codes of short length and high distance*, IEEE Journal on Selected Areas in Information Theory **6**, 127 (2025).

[118] G. Zhu, S. Sikander, E. Portnoy, A. W. Cross, and B. J. Brown, *Non-Clifford and parallelizable fault-tolerant logical gates on constant and almost-constant rate homological quantum LDPC codes via higher symmetries* (2023), arXiv:2310.16982 [quant-ph] .

[119] J. Old, M. Rispler, and M. Müller, *Lift-connected surface codes*, Quantum Science and Technology **9** (2024).

[120] S. Huang, T. Jochym-O'Connor, and T. J. Yoder, *Homomorphic logical measurements* (2022), arXiv:2211.03625 [quant-ph] .

[121] A. Cross, Z. He, P. Rall, and T. Yoder, *Linear-size ancilla systems for logical measurements in qLDPC codes* (2024), arXiv:2407.18393 [quant-ph] .

[122] Q. Xu, et al., *Fast and parallelizable logical computation with homological product codes* (2024), arXiv:2407.18490 [quant-ph] .

[123] A. Paetznick and B. W. Reichardt, *Fault-tolerant ancilla preparation and noise threshold lower bounds for the 23-qubit Golay code* (2013), arXiv:1106.2190 [quant-ph] .

[124] D. Kielpinski, C. Monroe, and D. J. Wineland, *Architecture for a large-scale ion-trap quantum computer*, Nature **417**, 709 (2002).

[125] J. Durandau, et al., *Automated generation of shuttling sequences for a linear segmented ion trap quantum computer*, Quantum **7**, 1175 (2023).

[126] D. Schoenberger, S. Hillmich, M. Brandl, and R. Wille, *Shuttling for scalable trapped-ion quantum computers* (2024), arXiv:2402.14065 [quant-ph] .

[127] H. Kaufmann, et al., *Fast ion swapping for quantum-information processing*, Physical Review A **95**, 052319 (2017).

[128] M. Palmero, S. Martínez-Garaot, U. G. Poschinger, A. Ruschhaupt, and J. G. Muga, *Fast separation of two trapped ions*, New Journal of Physics **17**, 093031 (2015).

[129] T. Ruster, et al., *Experimental realization of fast ion separation in segmented Paul traps*, Physical Review A **90**, 033410 (2014).

[130] A. Walther, et al., *Controlling fast transport of cold trapped ions*, Physical Review Letters **109**, 080501 (2012).

[131] G. A. Paz-Silva, G. K. Brennen, and J. Twamley, *Fault tolerance with noisy and slow measurements and preparation*, Physical Review Letters **105**, 100501 (2010).

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

[133] S. Veroni, A. Paler, and G. Giudice, *Universal quantum computation via scalable measurement-free error correction* (2025), arXiv:2412.15187 [quant-ph] .

[134] F. Butt, D. F. Locher, K. Brechtelsbauer, H. P. Büchler, and M. Müller, *Measurement-free, scalable, and fault-tolerant universal quantum computing*, Science Advances **11**, eadv2590 (2025).

[135] F. Butt, et al., *Demonstration of measurement-free universal fault-tolerant quantum computation* (2025), arXiv:2506.22600 [quant-ph] .

[136] K. Brechtelsbauer, et al., *Measurement-free quantum error correction optimized for biased noise* (2025), arXiv:2505.15669 [quant-ph] .

A Physical quantum circuits

The subcircuits of fault-tolerant transversal code switching used for logical auxiliary state preparation and implementation of the one-way logical CNOT gate are shown in Figs. 11-17.

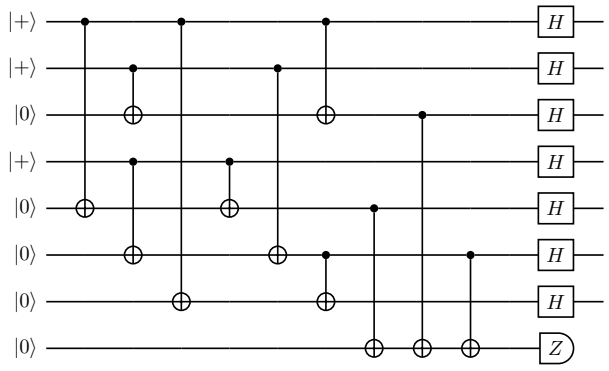

Figure 11: Flag-FT circuit to prepare the logical state $|+\rangle_S$ of the Steane code [12]. After the first eight CNOT gates, the state $|0\rangle_S$ is prepared non-fault-tolerantly on the upper seven qubits. The last three CNOTs are used to measure the eigenvalue of the logical Z-operator $Z_2Z_4Z_5$, with the help of a physical auxiliary qubit, which acts as a flag. If this last qubit is measured as $+1$, the state $|0\rangle_S$ is prepared fault-tolerantly. Otherwise it is discarded and the circuit is repeated. The last layer of physical H-gates implements the logical H-gate in the Steane code. We assume that these eight qubits can be fit into a single ion trap segment and that entangling gates can be performed between any pair of qubits.

The parity check matrix to construct the $[[17, 1, 5]]$ code is

$$H_{17} = \begin{bmatrix} 1&1&0&0&0&1&1&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&1&1&0&0&1&1&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&0&0&1&1&0&0&1&0&1&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&1&1&1&1 \\ 0&1&1&0&0&0&1&1&0&0&1&1&0&1&1&0&0 \\ 0&0&1&1&0&0&0&1&1&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&1&1&0&0&1&1&0&0&0&0 \\ 0&0&0&1&1&0&0&0&1&0&0&0&1&0&0&0&0 \end{bmatrix} \tag{9}$$

and the parity check matrix to construct the $[[19, 1, 5]]$ code is

$$H_{19} = \begin{bmatrix} 1&1&1&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&1&1&1&1&0&1&1&0&0&0&0&0&0&0&0&0&0 \\ 0&1&0&1&0&1&1&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&1&1&0&1&1&0&0&1&1&0&0&0&0&0 \\ 0&0&0&0&1&0&0&1&0&0&1&1&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&0&1&1&0&0&1&1&0&0&1&1&0&0 \\ 0&0&0&0&0&0&0&0&0&0&1&1&0&0&1&1&0&0&0 \\ 0&0&0&0&0&0&0&0&0&0&0&0&1&1&0&0&1&1&0 \\ 0&0&0&0&0&0&0&0&0&1&0&0&0&1&0&0&0&1&1 \end{bmatrix} \tag{10}$$

Due to their large size, we provide the stabilizers and encoding circuits of the 3D distance-5 codes in a more compact list format below in order to ease reproducing our results.

A.1 $[[49, 1, 5]]$ code

The 13 X-stabilizer generators have their non-trivial support on qubits `[[0, 1, 5, 6, 17, 18, 22, 23], [5, 6, 9, 10, 22, 23, 26, 27], [9, 10, 13, 15, 26, 27, 30, 32], [13, 14, 15, 16, 30, 31, 32, 33], [1, 2, 6, 7, 10, 11, 13, 14, 18, 19, 23, 24, 27, 28, 30, 31], [2, 3, 7, 8, 19, 20, 24, 25], [7, 8, 11, 12, 24, 25, 28, 29], [3, 4, 8, 12, 20, 21, 25, 29], [17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48], [34, 36, 38, 40, 42, 44, 46, 48], [35, 36, 39, 40, 43, 44, 47, 48], [37, 38, 39, 40, 45, 46, 47, 48], [41, 42, 43, 44, 45, 46, 47, 48]]` and the 35 $Z$-stabilizer generators are indexed by `[[0, 1, 5, 6, 17, 18, 22, 23], [5, 6, 9, 10, 22, 23, 26, 27], [9, 10, 13, 15, 26, 27, 30, 32], [13, 14, 15, 16, 30, 31, 32, 33], [1, 2, 6, 7, 10, 11, 13, 14, 18, 19, 23, 24, 27, 28, 30, 31], [2, 3, 7, 8, 19, 20, 24, 25], [7, 8, 11, 12, 24, 25, 28, 29], [3, 4, 8, 12, 20, 21, 25, 29], [17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48], [34, 36, 38, 40, 42, 44, 46, 48], [35, 36, 39, 40, 43, 44, 47, 48], [37, 38, 39, 40, 45, 46, 47, 48], [41, 42, 43, 44, 45, 46, 47, 48], [0, 1, 5, 6], [5, 6, 9, 10], [9, 10, 13, 15], [13, 14, 15, 16], [1, 2, 6, 7, 10, 11, 13, 14], [2, 3, 7, 8], [7, 8, 11, 12], [3, 4, 8, 12], [5, 6, 22, 23], [1, 6, 18, 23], [6, 10, 23, 27], [10, 13, 27, 30], [13, 14, 30, 31], [2, 7, 19, 24], [7, 11, 24, 28], [7, 8, 24, 25], [36, 40, 44, 48], [38, 40, 46, 48], [42, 44, 46, 48], [39, 40, 47, 48], [43, 44, 47, 48], [45, 46, 47, 48]]`.

The logical $|+\rangle$ state of the $[[49, 1, 5]]$ code can be prepared, starting from the state $|0\rangle^{\otimes 49}$, with the gate sequence `'H': [1, 3, 4, 6, 39, 40, 41, 43, 47, 16, 20, 22, 25, 27], 'CNOT': [(1, 0), (3, 0), (20, 0), (6, 0), (22, 0), (3, 2), (4, 2), (6, 2), (39, 2), (22, 2), (41, 2), (47, 2), (3, 5), (20, 5), (22, 5), (3, 7), (4, 7), (6, 7), (39, 7), (41, 7), (47, 7), (20, 7), (22, 7), (25, 7), (25, 8), (3, 8), (20, 8), (27, 9), (6, 9), (22, 9), (3, 10), (27, 10), (20, 10), (20, 11), (6, 11), (39, 11), (22, 11), (41, 11), (25, 11), (47, 11), (25, 12), (4, 12), (20, 12), (16, 13), (1, 13), (6, 13), (39, 13), (41, 13), (27, 13), (47, 13), (16, 14), (3, 14), (20, 14),`

(6, 14), (22, 14), (1, 15), (3, 15), (39, 15), (41, 15), (47, 15), (16, 15), (20, 15), (22, 15), (27, 15), (1, 17), (6, 17), (22, 17), (1, 18), (3, 18), (20, 18), (4, 19), (20, 19), (6, 19), (39, 19), (22, 19), (41, 19), (47, 19), (3, 21), (4, 21), (20, 21), (3, 23), (20, 23), (6, 23), (4, 24), (6, 24), (39, 24), (22, 24), (41, 24), (25, 24), (47, 24), (3, 26), (20, 26), (6, 26), (22, 26), (27, 26), (3, 28), (6, 28), (39, 28), (22, 28), (41, 28), (25, 28), (47, 28), (25, 29), (3, 29), (4, 29), (1, 30), (3, 30), (6, 30), (39, 30), (41, 30), (47, 30), (16, 30), (20, 30), (27, 30), (16, 31), (6, 31), (22, 31), (16, 32), (1, 32), (22, 32), (39, 32), (41, 32), (27, 32), (47, 32), (16, 33), (3, 33), (20, 33), (40, 34), (41, 34), (47, 34), (43, 35), (39, 35), (47, 35), (40, 36), (43, 36), (47, 36), (41, 37), (43, 37), (39, 37), (40, 38), (41, 38), (43, 38), (40, 42), (41, 42), (39, 42), (40, 44), (43, 44), (39, 44), (41, 45), (43, 45), (47, 45), (39, 46), (40, 46), (41, 46), (43, 46), (47, 46), (40, 48), (39, 48), (47, 48)]].

The one-way transversal CNOT gate connects qubits 0 to 16 of both codes.

A.2 $[[65, 1, 5]]$ code

The 16 X-stabilizer generators have their non-trivial support on qubits [[0, 1, 2, 3, 19, 20, 33, 50], [22, 24, 27, 28, 35, 37, 54, 64], [4, 7, 10, 11, 21, 23, 25, 26, 52, 57, 58, 59], [5, 6, 8, 9, 12, 13, 34, 36, 38, 39, 51, 53, 55, 56, 60, 61, 62, 63], [1, 3, 5, 6, 33, 34, 50, 51], [9, 13, 17, 18, 39, 42, 49, 63], [35, 36, 37, 38, 40, 41, 43, 44, 54, 55, 60, 64], [7, 8, 11, 12, 15, 16, 45, 46, 47, 48, 52, 54, 56, 57, 58, 59, 61, 62], [27, 28, 31, 32, 37, 40, 43, 64], [25, 26, 29, 30, 45, 47, 58, 59], [38, 39, 41, 42, 44, 46, 48, 49, 60, 61, 62, 63], [19, 20, 21, 22, 23, 24, 33, 34, 35, 36, 50, 51, 52, 53, 54, 55, 56, 57], [2, 3, 4, 5, 7, 8, 19, 21, 50, 51, 52, 53], [10, 11, 14, 15, 25, 29, 47, 59], [12, 13, 16, 17, 48, 49, 62, 63], [23, 24, 26, 27, 30, 31, 43, 44, 45, 46, 54, 55, 56, 57, 58, 60, 61, 64]] and the 48 Z-stabilizer generators are indexed by [[0, 18, 49, 50, 51, 52, 53, 56, 57, 62], [1, 18, 49, 50, 52, 56, 57, 62], [2, 18, 42, 50, 51, 53, 56, 61], [3, 18, 42, 50, 56, 61], [4, 18, 42, 53, 57, 61], [5, 18, 42, 51, 56, 61], [6, 18, 49, 51, 52, 56, 57, 62], [7, 18, 42, 52, 56, 61], [8, 18, 42, 52, 57, 61], [9, 18, 49, 63], [10, 18, 42, 52, 53, 58, 59, 61], [11, 18, 42, 56, 57, 58, 59, 61], [12, 18, 42, 62], [13, 18, 42, 63], [14, 18, 42, 49, 52, 53, 56, 57, 58, 59, 61, 63], [15, 18, 42, 49, 58, 59, 61, 63], [16, 18, 42, 49, 62, 63], [17, 18, 42, 49], [19, 50, 51, 53], [20, 42, 49, 50, 51, 52, 53, 57, 61, 62], [21, 53, 56, 57], [22, 42, 49, 54, 60, 62], [23, 52, 53, 56], [24, 54, 60, 61], [25, 52, 53, 56, 57, 59], [26, 52, 53, 56, 57, 58], [27, 52, 53, 56, 57, 60, 61, 64], [28, 42, 49, 52, 53, 56, 57, 60, 62, 64], [29, 49, 52, 53, 59, 63], [30, 49, 52, 53, 58, 63], [31, 49, 54, 56, 63, 64], [32, 42, 54, 56, 61, 62, 63, 64], [33, 42, 49, 50, 52, 57, 61, 62], [34, 42, 49, 51, 52, 57, 61, 62], [35, 42, 49, 52, 53, 54, 56, 57, 61, 62], [36, 42, 49, 56, 60, 62], [37, 42, 49, 61, 62, 64], [38, 42, 49, 60, 61, 62], [39, 42, 49, 63], [40, 42, 52, 53, 54, 57, 60, 62, 63, 64], [41, 42, 60, 61, 62, 63], [43, 49, 52, 53, 54, 57, 60, 61, 63, 64], [44, 49, 60, 63], [45, 49, 56, 57, 58, 63], [46, 49, 61, 63], [47, 49, 56, 57, 59, 63], [48, 49, 62, 63], [55, 56, 60, 61]].

The logical $|+\rangle$ state of the $[[65, 1, 5]]$ code can be prepared, starting from the state $|0\rangle^{\otimes 65}$, with the gate sequence 'H': [2, 4, 7, 10, 12, 14, 16, 30, 32, 34, 41, 46, 49, 50, 54, 57, 62], 'CNOT': [(16, 0), (34, 0), (50, 0), (4, 0), (10, 0), (12, 0), (14, 0), (16, 1), (2, 1), (34, 1), (4, 1), (62, 1), (57, 1), (46, 1), (50, 3), (14, 3), (62, 3), (57, 3), (10, 3), (12, 3), (46, 3), (16, 5), (2, 5), (50, 5), (4, 5), (62, 5), (57, 5), (46, 5), (34, 6), (14, 6), (62, 6), (57, 6), (10, 6), (12, 6), (46, 6), (16, 8), (7, 8), (10, 8), (12, 8), (14, 8), (2, 9), (34, 9), (4, 9), (7, 9), (12, 9), (16, 9), (49, 9), (50, 9), (62, 9), (10, 11), (4, 11), (7, 11), (16, 13), (49, 13), (62, 13), (4, 15), (14, 15), (7, 15), (49, 17), (12, 17), (62, 17), (49, 18), (2, 18), (34, 18), (4, 18), (50, 18), (7, 18), (62, 18), (2, 19), (14, 19), (62, 19), (57, 19), (10, 19), (12, 19), (46, 19), (16, 20), (34, 20), (50, 20), (4, 20), (62, 20), (57, 20), (46, 20), (4, 21), (14, 21), (62, 21), (57, 21), (10, 21), (12, 21), (46, 21), (46, 22), (41, 22), (54, 22), (57, 23), (4, 23), (7, 23), (2, 24), (34, 24), (4, 24), (7, 24), (41, 24), (16, 24), (50, 24), (54, 24), (62, 24), (4, 25), (30, 25), (7, 25), (62, 25), (10, 25), (12, 25), (46, 25), (10, 26), (14, 26), (30, 26), (32, 27), (4, 27), (7, 27), (41, 27), (10, 27), (14, 27), (46, 27), (54, 27), (57, 27), (32, 28), (2, 28), (34, 28), (41, 28), (10, 28), (14, 28), (16, 28), (50, 28), (54, 28), (57, 28), (62, 28), (4, 29), (14, 29), (30, 29), (7, 29), (62, 29), (12, 29), (46, 29), (32, 31), (2, 31), (34, 31), (4, 31), (7, 31), (46, 31), (16, 31),

(50, 31), (62, 31), (16, 33), (2, 33), (34, 33), (4, 33), (10, 33), (12, 33), (14, 33), (16, 35), (2, 35), (34, 35), (50, 35), (54, 35), (62, 35), (46, 35), (16, 36), (14, 36), (41, 36), (10, 36), (12, 36), (46, 36), (57, 36), (32, 37), (54, 37), (57, 37), (10, 37), (14, 37), (16, 38), (41, 38), (12, 38), (16, 39), (49, 39), (2, 39), (34, 39), (4, 39), (50, 39), (7, 39), (32, 40), (16, 40), (2, 40), (34, 40), (50, 40), (41, 40), (62, 40), (49, 42), (2, 42), (34, 42), (4, 42), (50, 42), (7, 42), (12, 42), (32, 43), (4, 43), (7, 43), (41, 43), (46, 43), (2, 44), (34, 44), (4, 44), (7, 44), (41, 44), (46, 44), (16, 44), (50, 44), (62, 44), (4, 45), (30, 45), (7, 45), (30, 47), (14, 47), (62, 47), (12, 47), (46, 47), (16, 48), (12, 48), (62, 48), (16, 51), (2, 51), (50, 51), (4, 51), (10, 51), (12, 51), (14, 51), (62, 52), (7, 52), (46, 52), (57, 52), (10, 52), (12, 52), (14, 52), (16, 53), (4, 53), (62, 53), (57, 53), (46, 53), (2, 55), (34, 55), (4, 55), (7, 55), (41, 55), (10, 55), (12, 55), (14, 55), (50, 55), (57, 55), (62, 55), (16, 56), (57, 56), (10, 56), (12, 56), (14, 56), (4, 58), (30, 58), (7, 58), (10, 58), (14, 58), (30, 59), (62, 59), (10, 59), (12, 59), (46, 59), (2, 60), (34, 60), (4, 60), (7, 60), (41, 60), (12, 60), (46, 60), (50, 60), (62, 60), (16, 61), (12, 61), (46, 61), (16, 63), (49, 63), (12, 63), (32, 64), (2, 64), (34, 64), (4, 64), (7, 64), (10, 64), (14, 64), (46, 64), (16, 64), (50, 64), (54, 64), (57, 64), (62, 64)]].

The one-way transversal CNOT gate connects qubits 0 to 18 of both codes.

Note that these definitions correspond to the original definition of the 3D color code with X-type cells and Z-type faces, which achieves a higher protection against X-errors than against Z-errors, as opposed to the $d = 3$-example discussed in the main text. In simulations for the $d = 5$-case, we use this convention so that the state $|+\rangle_S$ as input is analogous to an input state $|0\rangle_S$ with the triorthogonal code definition where X- and Z-type stabilizers are interchanged, such that more Z- than X-errors can be corrected. This interchange does not affect logical failure rate estimations of the $|+i\rangle_S$ state. All circuits can be accessed under https://doi.org/10.5281/zenodo.15674834.

B Large code state preparation in segmented ion trap

Entangling gates in ion traps are usually based on coupling of internal electronic degrees of freedom of ions and motional modes of the ion crystal during the gate operation [67–69]. The number of motional modes in an N-ion crystal is given by $3N$. Most gates require cooling all gate and spectator modes at least close to the motional ground state to reach high fidelities, which becomes more challenging for larger ion crystals. Another critical part arising from too large ion crystals can be an increased amount of crosstalk due to the fact that ions reside closer to each other. In this case, one needs to achieve a more tightly focused laser beam and higher beam pointing stability, which can pose considerable technical challenges. A potential solution for this scaling problem is realized in the QCCD architecture [56, 124] – especially in the combination of 1) addressing a suitably-sized ion crystal with 2) performing ion-shuttling operations in a segmented ion trap.

Let us now discuss the initialization of the logical zero state of the 15-qubit Tetrahedral code in a segmented ion trap architecture where we allow, as an example, up to 8 ions trapped in a single linear crystal. Shuttling of individual ions is employed to perform crystal reconfigurations that restore effective all-to-all connectivity across individual segments. The ion configurations in Fig. 18 match the coloring of CNOT gates in the encoding circuit of Fig. 12. It indicates a possible parallelized gate sequence to initialize $|0\rangle_T$ in two ion crystals labelled A and B with three sequential ion configurations.

The five subsequent flag measurements that verify the state preparation, shown in Fig. 13, can be conducted largely independently in separated zones. Only the shared flag qubits need to be shuttled when their respective entangling gates are scheduled for execution. Since the two Z-flags and two of the three X-flags act on an entirely disjoint set of data qubits it is sensible to use distinct ions for their measurement in order to parallelize the respective gate sequences.

Nowadays, most shuttling compilers focus on reducing the overall amount of ion crystal reconfiguration operations, while some already include consideration of different cost for different types of shuttling operations [125]. It shall be noted that an "optimal" shuttling schedule is not necessarily identical with this shortest-sequence approach. For example, moving ions between trap segments can be realized in different ways in a given architecture: One can either use physical SWAP operations of ions directly, where ions are moved out of the otherwise one-dimensional chain to interchange their positions (see Fig. 5). Another possibility, which retains all ion movement quasi-1D, is to first move some ions out of the way with the help of junctions and then directly transport ions without physical SWAPs [126] (see Fig. 10). From the typical reconfiguration operations, usually coined SWAP [127], MERGE, SPLIT [128, 129] and MOVE [130], some might be preferred over others, depending on their impact on the ions such as the resulting error rate or induced time overhead for recooling [110]. Compilers could include this behaviour, which can de-

pend very much on the concrete experimental setup, by utilizing customized cost functions for ion reconfiguration instructions.

C Measurement-free code switching

Owing to current experimental limitations of quantum computing hardware, there is a growing interest in running FT quantum circuits without measurements of physical qubits [116, 131–136].

Note that measurement-free code switching basically amounts to performing a SWAP gate between two different QEC codes and discarding the original qubit. A SWAP gate can be decomposed into three CNOT gates, two of which we can perform in the one-way fashion described in the main text. As of now, we are not aware of a unitary logical CNOT gate between the 2D and 3D color code in the other direction. To obtain this "reversed" CNOT gate, we note that one can concatenate each code with the respective other code, e.g., apply encoding circuits of the Tetrahedral code to the Steane code and apply encoding circuits of the Steane code to the Tetrahedral code, which yields a 105-qubit code for both logical qubits. Then, the CNOT gate could be performed transversally between the two logical qubits and decoding circuits of the respective codes might finally be applied to each logical qubit to re-obtain the original code blocks. We conjecture that this could be possible without any additional FT overhead.

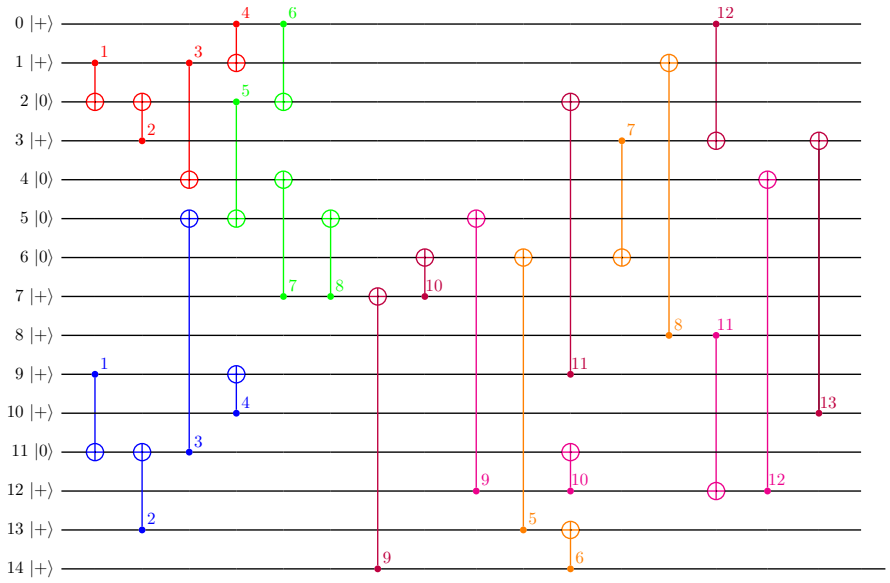

Figure 12: Circuit to non-fault-tolerantly encode the logical zero state of the Tetrahedral code defined by the stabilizer generators in Eqs. (2) and (3) with 25 CNOT gates based on Ref. [41]. Coloring of the CNOT gates indicates parallelizability in two linear ion traps with all-to-all connectivity within each ion crystal. Red and blue gates can be executed in parallel, so can green and orange gates as well as purple and magenta gates (see Fig. 18). Ion reconfiguration needs to be performed in between. Colored numbers indicate the respective order of CNOT gates in each trap.

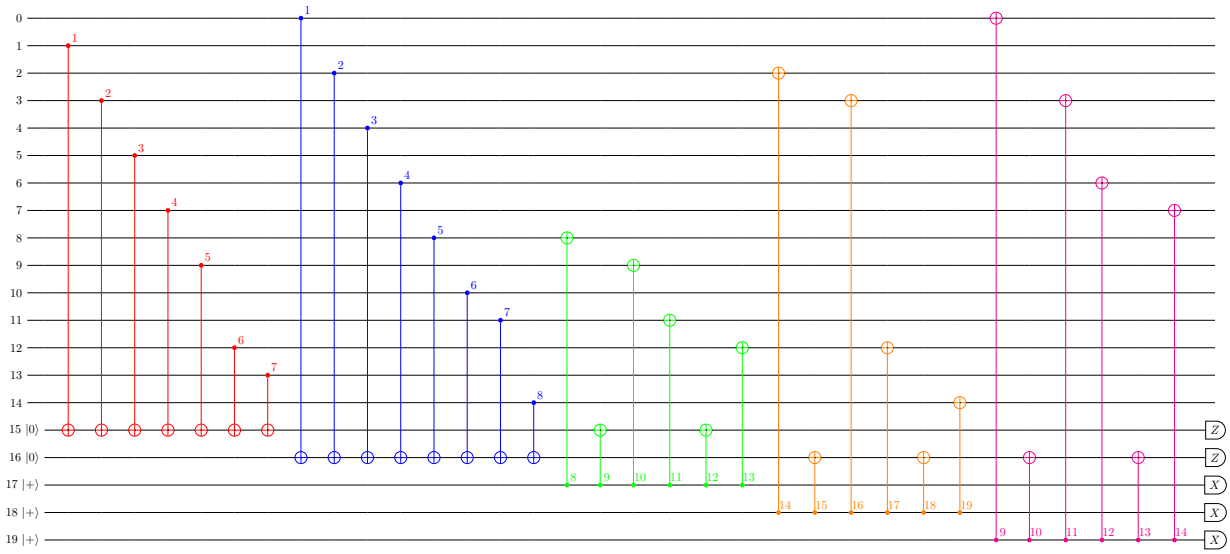

Figure 13: Flag verification circuit consisting of 33 physical CNOT gates that renders the $|0\rangle_{\mathrm{T}}$ state preparation subroutine suitable for fault-tolerant code switching with five auxiliary qubits. This circuit is applied after the one in Fig. 12. A prepared state is accepted if all auxiliary qubits are measured as $+1$ and discarded otherwise. Note that the Z-type operator measurements do not require additional flag qubits because only Z-errors could propagate from the auxiliary qubit to the data qubits. The X-type operator measurements need an additional flag mechanism to sort out propagating high-weight X-errors. Colors again indicate parallel execution in two separate ion traps. Red and blue gates can be executed in parallel as well as green and magenta gates. Ion crystals need to be reconfigured in between. Colored numbers indicate the respective order of CNOT gates in each trap. To correctly flag all dangerous errors for the ion mapping given in Figs. 12 and 18, the last (magenta) flag that measures the operator $X_0X_3X_6X_7$ needs to be replaced to instead measure $X_2X_3X_5X_6$.

p	10^{-4}	3×10^{-4}	10^{-3}	3×10^{-3}
$[[17, 1, 5]], \|0\rangle$	2.2	2.5	4.0	15.9
$[[49, 1, 5]], \|+\rangle$	1.3	2.2	12.8	2098.2
$[[19, 1, 5]], \|0\rangle$	2.2	2.5	4.5	16.8
$[[65, 1, 5]], \|+\rangle$	1.5	3.5	64.9	2022.2

Table 3: Average number of repetitions until the respective Pauli states are accepted with Steane-type state preparation in the single-parameter noise model for the distance-5 codes discussed in Sec. 5. The required number of repetitions quickly surges for the 3D codes when the depolarizing error rate p is increased. A more efficient FT state preparation subroutine should be employed for a practical application.

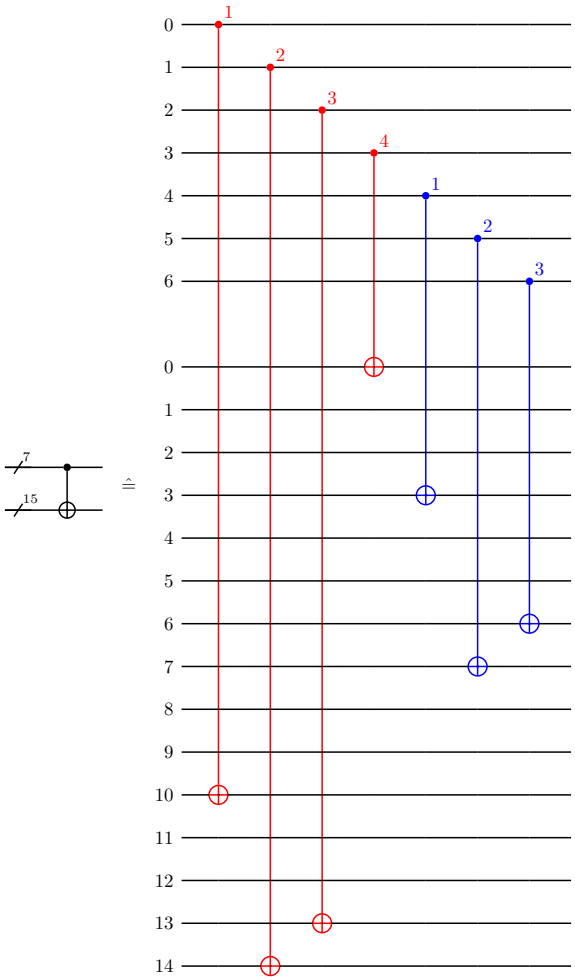

Figure 14: Physical CNOT gates to implement the one-way transversal logical CNOT gate given in Sec. 2.2. One possible arrangement of physical qubits into two different ion trap segments is indicated by the coloring. Gate sequences as marked by colored numbers can be applied simultaneously within each zone.

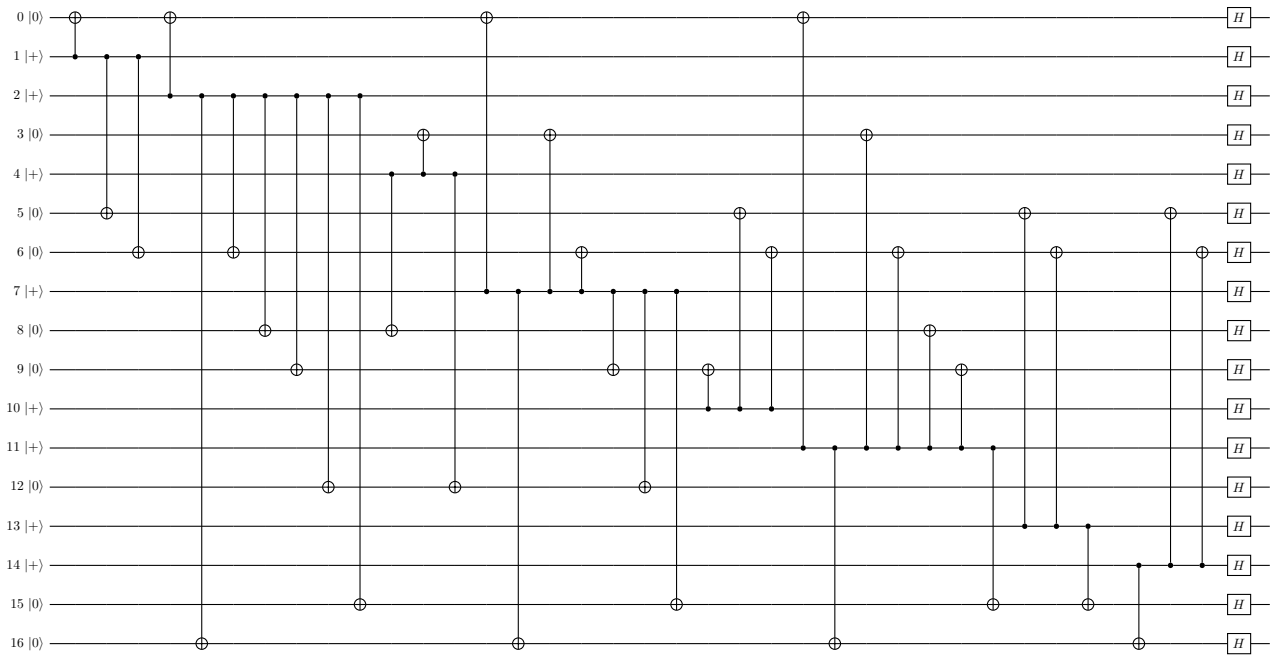

Figure 15: Circuit to non-fault-tolerantly encode the logical plus state of the $[[17, 1, 5]]$ code, an instance of the 4.8.8 2D color code, with 36 CNOT gates.

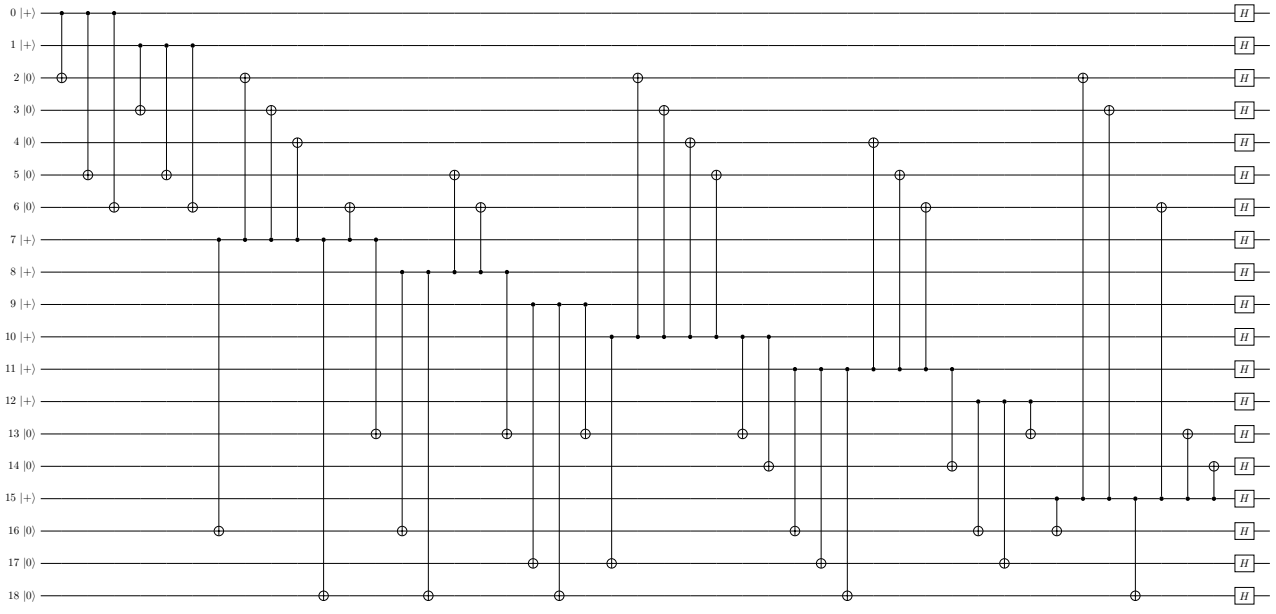

Figure 16: Circuit to non-fault-tolerantly encode the logical plus state of the $[[19, 1, 5]]$ code, an instance of the 6.6.6 2D color code, with 45 CNOT gates.

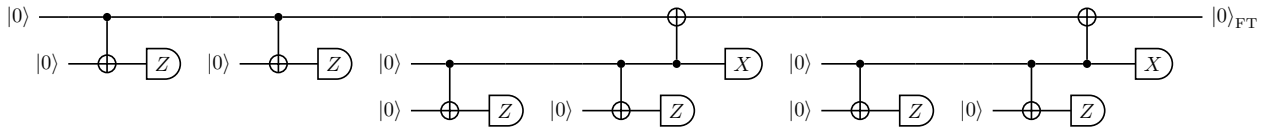

Figure 17: Circuit on the logical level for Steane-type FT state preparation with a distance-5 CSS code that can correct $t = 2$ errors. $|0\rangle$ indicates non-FT preparation of the logical zero state. Logical operators of each type X or Z after the first non-FT state preparation propagate to the logical auxiliary qubits. Such errors in the non-FT logical auxiliary qubit state preparations are heralded when propagating to the upper wire by the measurements. Any individual CNOT + measurement is therefore repeated t times. The state is accepted if all Z-measurements indicate a trivial Z-syndrome and the correct logical state and all X-measurements indicate a trivial X-syndrome. The state preparation subroutine is aborted and repeated as soon as a single measurement heralds a potential error. Due to transversality of the CNOT gates and measurements, Steane-type state preparation is strictly FT. The corresponding circuit for FT preparation of the logical plus state is obtained by exchanging X- and Z-type operations and reversing the directions of the CNOT gates. An attempt to optimize the scheme was given, for instance, in Ref. [123].

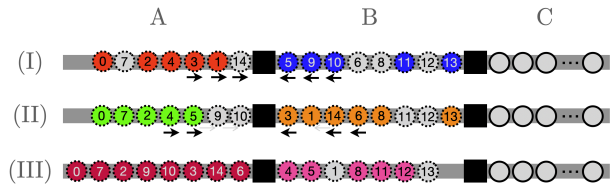

Figure 18: Simplified sketch of a segmented ion trap with three zones (grey bars), which each hold a linear ion crystal with all-to-all connectivity (cf. Fig. 5). Ions can be shuttled from one zone to another (separation indicated as black square). The segments A and B are used to prepare the logical state $|0\rangle_{\mathrm{T}}$ on 15 ions marked by the dotted line circles. Zone C holds a Steane code state on seven ions marked as solid line circles. Application of entangling gates is indicated by color in accordance with the coloring in Fig. 12. No entangling gates are applied to ions colored grey. Three ion configurations, labelled (I)-(III), are sufficient for non-FT state preparation. Arrows indicate the direction of ion movement between configurations – black arrows for shuttling through the junction, grey arrows for moving past another ion within the zone.