

State injection, lattice surgery, and dense packing of the deformation-based surface code

Shota Nagayama,^{1,*} Takahiko Satoh,² and Rodney Van Meter³

¹Graduate School of Media and Governance, Keio University, 5322 Endo, Fujisawa-shi, Kanagawa 252-0882, Japan

²Keio Research Institute at SFC, Keio University, 5322 Endo, Fujisawa-shi, Kanagawa 252-0882, Japan

³Faculty of Environment and Information Studies, Keio University, 5322 Endo, Fujisawa-shi, Kanagawa 252-0882, Japan

(Received 28 June 2016; revised manuscript received 14 October 2016; published 19 January 2017)

Resource consumption of the conventional surface code is expensive, in part due to the need to separate the defects that create the logical qubit far apart on the physical qubit lattice. We propose that instantiating the deformation-based surface code using superstabilizers will make it possible to detect short error chains connecting the superstabilizers, allowing us to place logical qubits close together. Additionally, we demonstrate the process of conversion from the defect-based surface code, which works as arbitrary state injection, and a lattice-surgery-like controlled NOT (CNOT) gate implementation that requires fewer physical qubits than the braiding CNOT gate. Finally, we propose a placement design for the deformation-based surface code and analyze its resource consumption; large-scale quantum computation requires $\frac{25d^2+170d+289}{4}$ physical qubits per logical qubit, where d is the code distance of the standard surface code, whereas the planar code requires $16d^2 - 16d + 4$ physical qubits per logical qubit, for a reduction of about 50%.

DOI: 10.1103/PhysRevA.95.012321

I. INTRODUCTION

The surface code is, to date, the most feasible proposal [1–5] to tolerate the inevitable imperfections in qubit states in a quantum computer [6–14]. The surface code has advantages for implementation over other quantum error-correcting codes; the surface code requires only a two-dimensional (2D) lattice of physical qubits with nearest-neighbor interactions, sustains scalability across a large range since the surface code can be enlarged by lengthening the columns and the rows of the 2D lattice, and has a higher error threshold than other codes. There are several proposals for producing a logical qubit on the surface-code lattice; the planar code produces a logical qubit on an independent, finite, regular lattice with two types of boundaries (X and Z) that give a degree of freedom [15,16]. The defect-based code produces a logical qubit by cutting two holes in a large regular lattice in which the holes are made from only an X or a Z boundary [4,17]. They achieve universality by providing arbitrary state injection and a basic set of one- and two-qubit fault-tolerant gates.

Bombin and Delgado introduced another way to produce a qubit on the surface code, the deformation-based surface code [18]. The deformation-based code produces a logical qubit by cutting a hole in a large regular lattice in which the boundary of the unused region is composed of both types of boundaries; hence it is like turning a planar code qubit inside out. They showed Clifford gates and initialization to $|0\rangle$ and $|+\rangle$. They demonstrated a controlled NOT (CNOT) gate by braiding, which can be executed between two logical qubits in the deformation-based code and even between the deformation-based code and the defect-based code. Since a SWAP gate can be implemented with three CNOT gates, arbitrary state injection to the deformation-based code can be achieved utilizing this heterogeneous CNOT gate. The first step is to use the standard state-injection method in the defect-based code, then swap into the deformation-based code. However,

this method is an indirect way to achieve state injection to the deformation-based code.

In this paper we show a conversion from the defect-based code to the deformation-based code that enables the deformation-based code to hold an arbitrary state and demonstrate that a crossed pair of an X superstabilizer and a Z superstabilizer produces a deformation-based qubit without sacrificing the advantages above. We employ the fault-tolerant stabilization utilizing a cat state generated by parallel ZZ stabilization. Additionally, we demonstrate a lattice-surgery-like CNOT gate for the deformation-based code [16]. Lattice surgery is a nontransversal, scalable means of executing a CNOT gate on the planar code that requires fewer resources than the braiding of the defect-based code. Our lattice-surgery-like CNOT gate for the deformation-based code requires fewer qubits than the conventional braiding. Nevertheless, the error suppression ability is similar to conventional surface code since the logical state is protected by normal stabilizers. Our proposals may reduce the resource requirements of the surface code in spatial accounting.

II. OVERVIEW OF THE DEFORMATION-BASED SURFACE CODE

Figure 1 shows a distance-3 deformation-based qubit that exists on the surface-code lattice. The surface code uses physical qubits placed on a 2D lattice. The black dots are data qubits, and the white dots are ancilla qubits. The lattice is separated into plaquettes, as shown by black lines in Fig. 1. A stabilizer U is an operator which does not change a state,

$$U|\psi\rangle = |\psi\rangle. \quad (1)$$

An ancilla qubit in the center of a plaquette is used to measure the eigenvalue of a Z stabilizer such as $Z_a Z_b Z_c Z_d$, where $a-d$ denote the surrounding four data qubits. An ancilla qubit on the vertex is used for an X stabilizer.

The number of logical qubits k on a state of n physical qubits is $k = n - s$, where s is the number of independent stabilizers. In Fig. 1, there are 48 data qubits, 19 Z stabilizers,

*kurosagi@sfc.wide.ad.jp

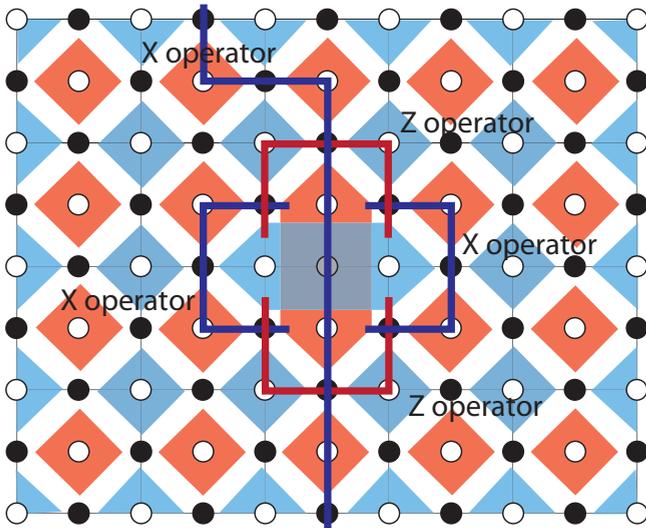


FIG. 1. The deformation-based qubit of distance 3. Black dots depict data qubits, and white dots are ancilla qubits. Each red diamond describes a Z stabilizer, and each blue diamond describes an X stabilizer. The gray dot in the center depicts the unused data qubit, and the two four-qubit Z stabilizers the unused data qubit originally belonged to are merged to form the six-qubit Z stabilizer shown. The two four-qubit X stabilizers the unused data qubit originally belonged to are also merged to form the six-qubit X stabilizer shown. The thick lines are logical operators of the superstabilizer qubit. Any blue or red path serves as a logical X operator or a logical Z operator, respectively.

and 28 independent X stabilizers since any of the X stabilizers is the product of all the others, leaving a single degree of freedom for one logical qubit.

Two Z_L operators of a deformation-based qubit are shown in Fig. 1, either of which acts on the logical qubit. Three X_L operators are shown in the figure, also working on the same logical qubit. Two of the X_L operators are the same shape as the described Z_L operators, while the third crosses the Z superstabilizer ends at the boundaries of the lattice. Those two Z_L and two X_L logical operators surrounding the superstabilizers correspond to the logical operators shown in Fig. 5(a) in [18], except that our deformation-based qubit employs superstabilizers. As with other surface-code qubits, the products of a logical operator and stabilizers produce the redundancy for measurements of logical operators.

Figure 1 shows another important characteristic of the deformation-based qubit, how to count its code distance. Each logical operator consists of operations on three physical qubits; therefore the code distance of this deformation-based qubit is 3. An example of a longer code distance is shown in Fig. 2, which depicts two deformation-based qubits of distance 5.

Figure 2 shows an advantage of deformation-based qubits compared to defect-based surface-code qubits. The deformation-based qubit exists at the junction of two superstabilizers, so that every data qubit alive in the lattice belongs to two X stabilizers and two Z stabilizers. The two Z superstabilizers find the X error on the marked qubit in Fig. 2; hence the deformation-based qubits can be placed close to each other without being susceptible to logical errors, although

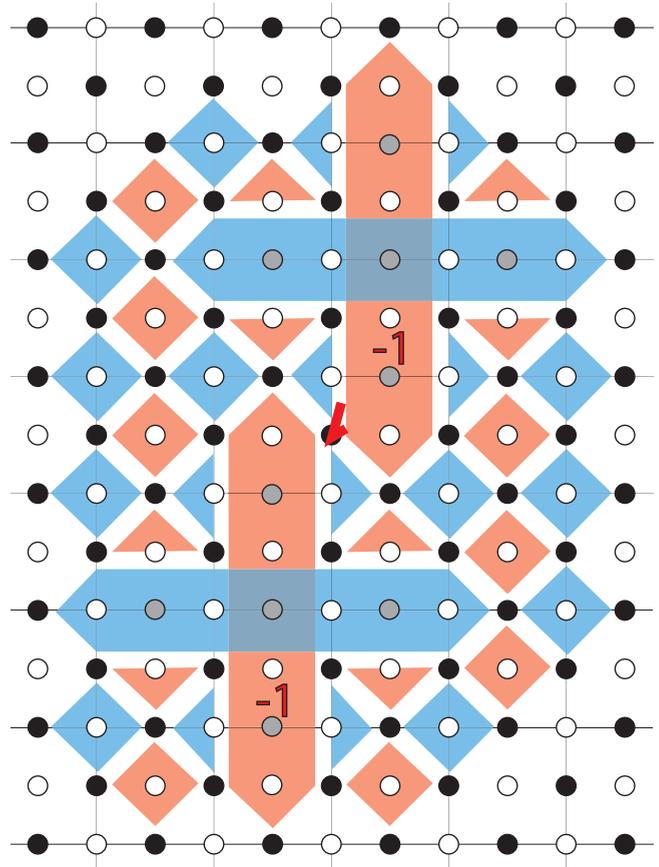


FIG. 2. Neighboring distance-5 deformation-based qubits. Placement code distance apart from the boundary of the lattice is assumed. An X error on the marked qubit results in -1 eigenvalues of the two red Z superstabilizers. The two-defect surface code cannot correct an X error on a data qubit which belongs to two defects, but the superstabilizers of the deformation-based code can.

other surface-code qubits must be placed code distance away.

III. TRANSFORMATION

We have shown the “four-fin”-style deformation-based qubits. Figure 3 shows two transformed deformation-based qubits of distance 5. The deformation-based qubit in Fig. 3(a) is extended in the horizontal direction and compressed in the vertical direction. The perimeter of the Z (X) superstabilizer can be considered to be separated by the X (Z) superstabilizer. The logical Z (X) operator exists at any path connecting the separated halves. The deformation-based qubit in Fig. 3(b) has a single, skewed Z superstabilizer. This transformation is achieved with more or less the defect-moving operations of the defect-based surface code [1]. The only difference is that the defect that does not have a stabilizer measurement is replaced with the superstabilizer here.

IV. CONVERSION FROM A TWO-DEFECT-BASED QUBIT

Direct conversion from a two-defect surface-code qubit to a deformation-based qubit can be achieved. This

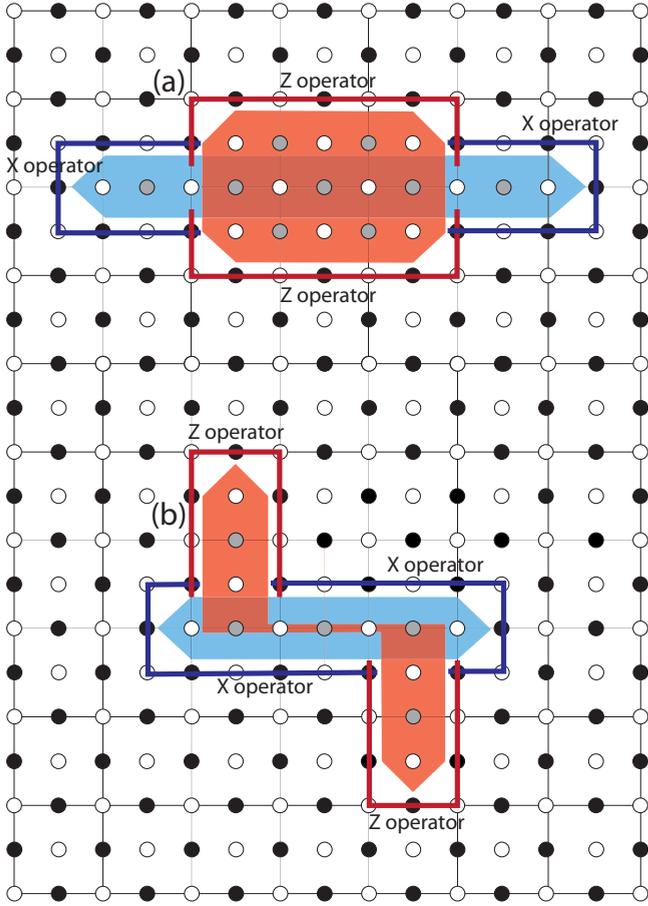


FIG. 3. (a) “Bar-form” deformation-based qubit, which has code distance 5. The Z (X) logical operators exist between halves of the X (Z) superstabilizer separated by the Z (X) superstabilizer. (b) A deformation-based qubit of code distance 5 that has “skew fin.” The Z (X) logical operators exist between halves of the X (Z) superstabilizer separated by the Z (X) superstabilizer.

conversion works as the state injection for the deformation-based qubit and, e.g., to support networking among multiple quantum computers that employ heterogeneous error-correcting codes [19]. To complete the universality of the deformation-based surface code, we demonstrate the arbitrary state injection in this section. We first inject an arbitrary qubit into a two-defect surface code following Fowler *et al.* [1], as depicted on a fragment of surface code in Fig. 4.

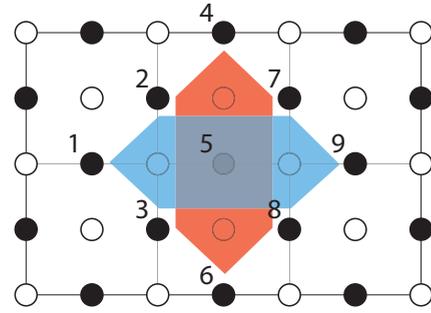


FIG. 4. Surface-code fragment to inject an arbitrary qubit. The lattice has only normal stabilizers at first. The shown superstabilizers are introduced in several steps, as described in Sec. IV.

The surface begins in normal operation, using qubit 5 and measuring all four-qubit stabilizers,

$$\begin{array}{c|ccccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline X & X & X & & & X & & & & \\ & & & & & & X & X & X & X, \\ & & Z & & Z & Z & & Z & & \\ & & & Z & & Z & Z & & Z & \end{array} \quad (2)$$

where each number corresponds to the same number in Fig. 4. First, we measure qubit 5 in the X basis, disentangling it from the larger state where M_a^b denotes a measured value, where a is the measurement basis and b is the qubit index.

$$(-1)^{M_X^5} \begin{array}{c|ccccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline X & X & X & X & & X & & & & \\ & & & & & & X & X & X & X, \\ & & Z & Z & Z & & Z & Z & Z & \end{array} \quad (3)$$

If the -1 eigenvalue is measured, apply either $Z_2Z_4Z_5Z_7$ or $Z_3Z_5Z_6Z_8$ to restore $X_1X_2X_3$ and $X_7X_8X_9$ to $+1$ eigenvalues,

$$\begin{array}{c|ccccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline X & X & X & X & & & & X & X & X, \\ & & & & & X & & & & \\ & & Z & Z & Z & & Z & Z & Z & \end{array} \quad (4)$$

Next, qubit 5 is rotated to the arbitrary desired state [20], $\alpha(Z) + \beta(-Z)$,

$$\alpha \left(+ \begin{array}{c|ccccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline X & X & X & & & & & X & X & X \\ & & & Z & Z & Z & & Z & Z & Z \end{array} \right) + \beta \left(- \begin{array}{c|ccccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline X & X & X & & & & & X & X & X \\ & & Z & Z & Z & & Z & Z & Z & \end{array} \right). \quad (5)$$

Then we measure $Z_2Z_4Z_5Z_7$ and $Z_3Z_5Z_6Z_8$,

$$\alpha \left(+ \begin{array}{c|ccccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline X & X & X & & & & X & X & X \\ & & Z & & Z & Z & & Z & & \\ & & & Z & & Z & Z & & Z & \end{array} \right) + \beta \left(- \begin{array}{c|ccccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline X & X & X & & & & X & X & X \\ & & Z & & Z & Z & & Z & & \\ & & & Z & & Z & Z & & Z & \end{array} \right). \quad (6)$$

If the -1 eigenvalue is measured, apply either $X_1X_2X_3$ or $X_7X_8X_9$ to give the desired state. The two defects exist at $X_1X_2X_3X_5$ and $X_5X_7X_8X_9$, a minimal logical qubit of distance 1,

$$\alpha \left(+ \left| \begin{array}{c|cccccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline X & X & X & & & & X & X & X \\ & & Z & & Z & & Z & & & \\ & & & Z & Z & & Z & & & \\ & & & & Z & Z & & Z & & \end{array} \right) + \beta \left(- \left| \begin{array}{c|cccccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline X & X & X & & & & X & X & X \\ & & Z & & Z & & Z & & & \\ & & & Z & Z & & Z & & & \\ & & & & Z & Z & & Z & & \end{array} \right) \right). \quad (7)$$

So far we have the logical qubit of the two-defect surface code. Next, we start to convert this logical qubit to the deformation-based surface code.

For pedagogical clarity, we omit writing the stabilizers that do not change over the course of this operation, depicted in white in the figures, and we write $Z_2Z_4Z_5Z_7 \otimes Z_3Z_5Z_6Z_8 = Z_2Z_3Z_4Z_6Z_7Z_8$, which is a product of two stabilizers and which can be measured as a stabilizer without breaking the logical state. We again measure qubit 5 in the X basis, merging the two minimal defects into one superstabilizer,

$$\alpha \left(\begin{array}{c} (-1)^{M_X^5} \\ + \\ + \end{array} \left| \begin{array}{c|cccccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline X & X & X & & & X & X & X \\ & & Z & & Z & & Z & & & \\ & & & Z & Z & & Z & & & \\ & & & Z & Z & Z & Z & Z & & \end{array} \right) + \beta \left(\begin{array}{c} (-1)^{M_X^5} \\ - \\ - \end{array} \left| \begin{array}{c|cccccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline X & X & X & & & X & X & X \\ & & Z & & Z & & Z & & & \\ & & & Z & Z & & Z & & & \\ & & & Z & Z & Z & Z & Z & & \end{array} \right) \right). \quad (8)$$

If the -1 eigenvalue is obtained, apply either $Z_2Z_4Z_5Z_7$ or $Z_3Z_5Z_6Z_8$ to preserve the parity of the logical X operator such as $X_1X_2X_3X_5$ into $X_1X_2X_3$, giving

$$\alpha \left(+ \left| \begin{array}{c|cccccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline X & X & X & & & X & X & X \\ & & Z & & Z & & Z & & & \\ & & & Z & Z & & Z & & & \\ & & & Z & Z & Z & Z & Z & & \end{array} \right) + \beta \left(- \left| \begin{array}{c|cccccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ \hline X & X & X & & & X & X & X \\ & & Z & & Z & & Z & & & \\ & & & Z & Z & & Z & & & \\ & & & Z & Z & Z & Z & Z & & \end{array} \right) \right). \quad (9)$$

Now $Z_2Z_4Z_7$ and $Z_3Z_6Z_8$ share the desired state. We can now begin measuring $Z_2Z_3Z_4Z_6Z_7Z_8$ as our superstabilizer. As is common with state injections, because the process begins with a raw qubit, state distillation on the logical qubit is required after this process.

V. CNOT GATE

A CNOT gate can be performed utilizing lattice surgery [16]. The basic concept of the CNOT gate by lattice surgery is as follows:

(1) Prepare a control (C) qubit in $\alpha|0_C\rangle + \beta|1_C\rangle$ and a target (T) qubit in $\alpha'|0_T\rangle + \beta'|1_T\rangle$.

(2) Prepare an intermediate (INT) qubit in $|+\rangle$. The initial state is

$$|\psi^{\text{init}}\rangle = (\alpha|0_C\rangle + \beta|1_C\rangle) \otimes |+\rangle \otimes (\alpha'|0_T\rangle + \beta'|1_T\rangle). \quad (10)$$

(3) Measure Z_CZ_I , and get

$$|\psi'\rangle = (\alpha|0_C0_I\rangle + \beta|1_C1_I\rangle) \otimes (\alpha'|0_T\rangle + \beta'|1_T\rangle) \quad (11)$$

by applying X_I if the -1 eigenvalue is observed.

(4) Measure $X_I X_T$, and get

$$\begin{aligned} |\psi''\rangle = & \alpha|0_C\rangle(\alpha'|0_I0_T\rangle + \beta'|0_I1_T\rangle + \beta'|1_I0_T\rangle + \alpha'|1_I1_T\rangle) \\ & + \beta|1_C\rangle(\beta'|0_I0_T\rangle + \alpha'|0_I1_T\rangle \\ & + \alpha'|1_I0_T\rangle + \beta'|1_I1_T\rangle) \end{aligned} \quad (12)$$

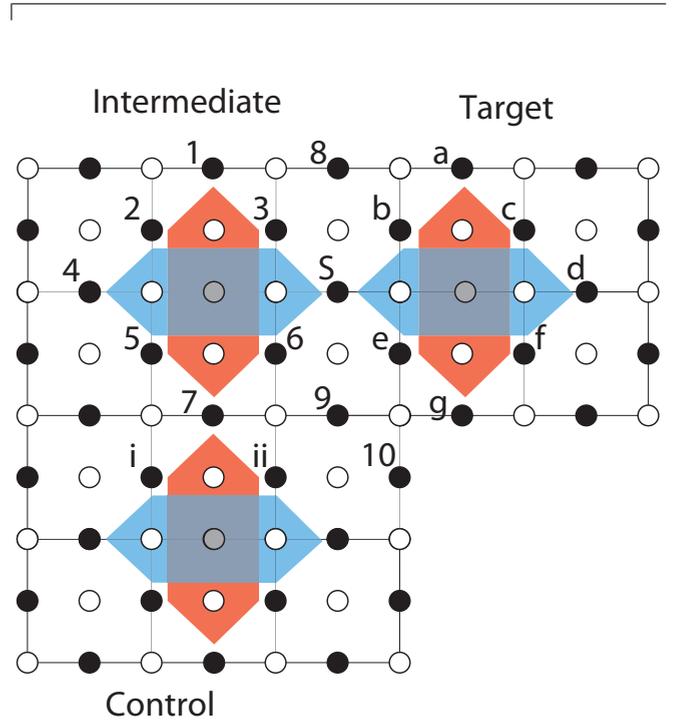


FIG. 5. Three deformation-based qubits to demonstrate the CNOT gate between the control qubit and the target qubit by lattice-surgery-like operations in Sec. V. The intermediate qubit is initialized in $|+\rangle$. The code distance for those logical qubits is still 3 during lattice surgery.

if the +1 eigenvalue is observed and get

$$\begin{aligned}
 |\psi'''\rangle = & \alpha|0_C\rangle(\alpha'|0_I0_T\rangle + \beta'|0_I1_T\rangle - \beta'|1_I0_T\rangle - \alpha'|1_I1_T\rangle) \\
 & + \beta|1_C\rangle(-\beta'|0_I0_T\rangle - \alpha'|0_I1_T\rangle \\
 & + \alpha'|1_I0_T\rangle + \beta'|1_I1_T\rangle)
 \end{aligned} \tag{13}$$

if the -1 eigenvalue is observed. Apply $Z_C Z_I$ and get Eq. (12) when -1 is observed. Next, as we merge the I and T qubits using lattice surgery, the resulting qubit is the exclusive OR (XOR) of the two Z operators, and finally, we get

$$\begin{aligned}
 |\psi^{\text{final}}\rangle = & \alpha|0_C\rangle(\alpha'|0_m\rangle + \beta'|1_m\rangle) \\
 & + \beta|1_C\rangle(\beta'|0_m\rangle + \alpha'|1_m\rangle),
 \end{aligned} \tag{14}$$

where m stands for *merged*, indicating the merged qubit of I and T .

Figure 5 depicts the logical CNOT gate of the deformation-based qubit by lattice surgery.

To measure $Z_C Z_I$, we measure $Z_5 Z_6 Z_i Z_{ii}$. This is achieved by swapping qubit 7 with a neighboring ancilla qubit and using the fault-tolerant stabilizer measurement described in Sec. VI. This measurement is repeated d times for majority voting to correct errors, where d is the code distance. If the -1 eigenvalue is observed from the $Z_C Z_I$ measurement, X_I is applied. During the measurement of $Z_C Z_I$, we cannot measure the Z superstabilizers of the intermediate qubit and the control qubit; meanwhile, normal Z stabilizers can be measured. Hence, error chains connecting the two Z superstabilizers, such as X_7 and $X_6 X_9 X_{ii}$, may be caused. (Fig. 5 shows distance 3 code; therefore we should not allow an error chain of length less than 3 to go undetected.) However, those error chains do not matter since they are stabilizers for $Z_5 Z_6 Z_i Z_{ii}$.

Next, we measure $X_I X_T$ and merge the intermediate qubit and the target qubit. Here we describe the merge operation of deformation-based qubits. The original state is

$$(\alpha|0_C0_I\rangle + \beta|1_C1_I\rangle) \otimes (\alpha'|0_T\rangle + \beta'|1_T\rangle) = \alpha\alpha'|0_C0_I0_T\rangle + \alpha\beta'|0_C0_I1_T\rangle + \beta\alpha'|1_C1_I0_T\rangle + \beta\beta'|1_C1_I1_T\rangle. \tag{15}$$

The first term of Eq. (15) is

$$\alpha\alpha'|0_C\rangle \left(\begin{array}{c|cccccccccccccccc}
 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & S & a & b & c & d & e & f & g \\
 \hline
 Z & Z & Z & & Z & Z & Z & & & & & & & & & & & \\
 & & X & X & X & X & X & & & & X & & & & & & & \\
 & & & & & & & & & & & Z & Z & Z & & Z & Z & Z \\
 & & & & Z & & & & & Z & X & X & X & X & X & X & & \\
 & & & & & & & & Z & Z & Z & Z & & & & & & \\
 + & & & & & Z & Z & Z & & & & & & & & Z & & \\
 + & & & & & & & & & & & & & & & Z & Z & Z
 \end{array} \right) \tag{16}$$

where the logical state of two qubits exists in $Z_1 Z_2 Z_3$ and $Z_a Z_b Z_c$. The two bottom lines are the logical operator states. Measure qubit S in the Z basis, giving

$$\alpha\alpha'|0_C\rangle \left(\begin{array}{c|cccccccccccccccc}
 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & S & a & b & c & d & e & f & g \\
 \hline
 Z & Z & Z & & Z & Z & Z & & & & & & & & & & & \\
 & & X & X & X & X & X & & & & & & X & X & X & X & X & \\
 & & & & & & & & & & & Z & Z & Z & & Z & Z & Z \\
 (-1)^{M_Z^S} & & & Z & & & & & Z & & & & Z & & & & & \\
 (-1)^{M_Z^S} & & & & & & Z & & Z & & & & & & & Z & & \\
 (-1)^{M_Z^S} & & & & & & & & & Z & & & & & & & & \\
 + & & & & & Z & Z & Z & & & & & & & & Z & Z & Z \\
 + & & & & & & & & & & & & & & & Z & Z & Z
 \end{array} \right). \tag{17}$$

If the -1 eigenvalue is obtained, apply either $X_2 X_3 X_4 X_5 X_6 X_S$ or $X_b X_c X_d X_e X_f X_S$, and get

$$\alpha\alpha'|0_C\rangle \left(\begin{array}{c|cccccccccccccccc}
 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & S & a & b & c & d & e & f & g \\
 \hline
 Z & Z & Z & & Z & Z & Z & & & & & & & & & & & \\
 & & X & X & X & X & X & & & & & & X & X & X & X & X & \\
 & & & & & & & & & & & Z & Z & Z & & Z & Z & Z \\
 & & & Z & & & & & Z & & & & Z & & & & & \\
 & & & & & & Z & & Z & & & & & & & Z & & \\
 + & & & & & Z & Z & Z & & & Z & & & & & Z & & \\
 + & & & & & & & & & & & & & & & Z & Z & Z
 \end{array} \right). \tag{18}$$

Next, we measure $X_3 X_b X_6 X_e$ for the third step of lattice surgery.

We can measure X_3 , X_b , X_6 , and X_e both to execute our merge and to measure $X_3X_bX_6X_e$. Measure qubit 3 in the X basis. If -1 is obtained, apply either $Z_3Z_8Z_b$ or $Z_1Z_2Z_3Z_5Z_6Z_7$.

$$\alpha\alpha'|0_C\rangle \left(\begin{array}{c|cccccccccccccccc} \hline & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & S & a & b & c & d & e & f & g \\ \hline & Z & Z & & & Z & Z & Z & Z & & & & & & Z & & & \\ & & X & & X & X & X & & & & & & & X & X & X & X & X \\ & & & & & & & & & & Z & Z & Z & & & Z & Z & Z \\ & & & & & & Z & & & Z & & & & & & Z & & \\ + & & & & & Z & Z & Z & & & & & & & & & & \\ + & & & & & & & & & & & & & & & Z & Z & Z \end{array} \right). \quad (19)$$

Measure qubit b in the X basis. If the -1 is obtained, apply either $Z_1Z_2Z_5Z_6Z_7Z_8Z_b$ or $Z_aZ_bZ_cZ_eZ_fZ_g$.

$$\alpha\alpha'|0_C\rangle \left(\begin{array}{c|cccccccccccccccc} \hline & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & S & a & b & c & d & e & f & g \\ \hline & Z & Z & & & Z & Z & Z & Z & & & Z & & & Z & Z & Z & Z \\ & & X & & X & X & X & & & & & & & X & X & X & X \\ & & & & & & Z & & & Z & & & & & & Z & & \\ + & & & & & Z & Z & Z & & & & & & & & & & \\ + & & & & & & & & & & & & & & & Z & Z & Z \end{array} \right). \quad (20)$$

Measure qubit 6 in the X basis, and apply $Z_1Z_2Z_5Z_6Z_7Z_8Z_aZ_cZ_eZ_fZ_g$ if the -1 eigenvalue is observed.

$$\alpha\alpha'|0_C\rangle \left(\begin{array}{c|cccccccccccccccc} \hline & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & S & a & b & c & d & e & f & g \\ \hline & Z & Z & & & Z & & Z & Z & Z & & Z & & & Z & & Z & Z \\ & & X & & X & X & & & & & & & & X & X & X & X \\ + & & & & & Z & & Z & & Z & & & & & & Z & & \\ + & & & & & & & & & & & & & & & Z & Z & Z \end{array} \right). \quad (21)$$

Measure qubit e in the X basis and apply both $Z_5Z_7Z_9$ as Z_I and $Z_iZ_7Z_{ii}$ as Z_C if the -1 eigenvalue is obtained.

Alternately, we can measure X_3 , X_b , X_6 , and X_e in parallel. After the parallel measurements, if an even number of -1 eigenvalues is observed, as in normal error correction, a physical Z operator chain connecting the remaining X stabilizers with -1 eigenvalues is executed. If an odd number of -1 eigenvalues is observed, we execute the physical Z operator chain, and there still remains an X stabilizer with a -1 eigenvalue. The X superstabilizer of the merged qubit actually has the -1 eigenvalue in this case; hence we connect the remaining X stabilizer and the intermediate qubit side of the X superstabilizer. This operation keeps the eigenvalues of the lattice $+1$ and works as Z_I , like $Z_5Z_7Z_9$ was used in the sequential form above. We execute $Z_iZ_7Z_{ii}$ as Z_C when an odd number of -1 eigenvalues is observed.

Those measurements work for connecting the superstabilizers. Therefore those measurements are allowed to be non-fault-tolerant since the remaining stabilizers confirm the correctness of the measurements; when qubit e is measured in the X basis, regardless of whether a measurement error occurs, if the remaining stabilizer $X_9X_gX_{10}$ outputs -1 repeatedly, we can conclude the correct measurement of qubit e is -1 .

Now we have code space for only one qubit, and the two qubits are merged into a qubit whose logical operator state is the product of the first two, shown in the bottom line,

$$\alpha\alpha'|0_C\rangle \left(\begin{array}{c|cccccccccccccccc} \hline & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & S & a & b & c & d & e & f & g \\ \hline & Z & Z & & & Z & & Z & Z & Z & & Z & & & Z & & Z & Z \\ & & X & & X & X & & & & & & & & X & X & & X \\ + & & & & & Z & & Z & & Z & & & & & & Z & Z \end{array} \right). \quad (22)$$

By similar operations, Eq. (15) is rewritten as

$$\begin{aligned} & \alpha\alpha'|0_C\rangle \left(\begin{array}{c|cccccccccccccccc} \hline & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & S & a & b & c & d & e & f & g \\ \hline & Z & Z & & & Z & & Z & Z & Z & & Z & & & Z & & Z & Z \\ & & X & & X & X & & & & & & & & X & X & & X \\ + & & & & & Z & & Z & & Z & & & & & & Z & Z \end{array} \right) \\ & + \alpha\beta'|0_C\rangle \left(\begin{array}{c|cccccccccccccccc} \hline & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & S & a & b & c & d & e & f & g \\ \hline & Z & Z & & & Z & & Z & Z & Z & & Z & & & Z & & Z & Z \\ & & X & & X & X & & & & & & & & X & X & & X \\ - & & & & & Z & & Z & & Z & & & & & & Z & Z \end{array} \right) \end{aligned}$$

$$\begin{aligned}
 & + \beta\alpha'|1_C\rangle \left(\begin{array}{c|cccccccccccccccc}
 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & S & a & b & c & d & e & f & g \\
 \hline
 - & Z & Z & & & Z & & Z & Z & Z & & Z & & Z & & & & \\
 & & X & & X & X & & & & & & & & X & X & & X & \\
 & & & & & Z & & Z & & Z & & & & & & & Z & Z
 \end{array} \right) \\
 & + \beta\beta'|1_C\rangle \left(\begin{array}{c|cccccccccccccccc}
 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & S & a & b & c & d & e & f & g \\
 \hline
 + & Z & Z & & & Z & & Z & Z & Z & & Z & & Z & & & & \\
 & & X & & X & X & & & & & & & & X & X & & X & \\
 & & & & & Z & & Z & & Z & & & & & & & Z & Z
 \end{array} \right). \tag{23}
 \end{aligned}$$

Using a new definition, we now have

$$|0_m\rangle = Z_5 Z_7 Z_9 Z_f Z_g, \tag{24}$$

$$|1_m\rangle = -Z_5 Z_7 Z_9 Z_f Z_g, \tag{25}$$

where m stands for merged. Equation (23) can be written as

$$\alpha|0\rangle(\alpha'|0_m\rangle + \beta'|1_m\rangle) + \beta|1\rangle(\beta'|0_m\rangle + \alpha'|1_m\rangle); \tag{26}$$

therefore now we have a complete CNOT gate. From this point in the operation, we start to measure the new superstabilizers.

VI. ARBITRARY-SIZE STABILIZER MEASUREMENT

We suggest using a cat state of an arbitrary length to measure superstabilizers. In this section, we first discuss fault-tolerant preparation and then generic use of cat states for constant-time stabilizer measurement, before addressing superstabilizers in our system. Finally, we return to the issue of errors.

A. Arbitrary-length cat-state preparation

The non-fault-tolerant circuit to prepare an arbitrary-length cat state in constant time is depicted in Fig. 6. In the circuit, many qubits in $|+\rangle$ are created and entangled by measuring ZZ of every pair of neighboring qubits. Here, we prepare two qubits in $|+_0+2\rangle$ and a third qubit in $|0_1\rangle$,

$$|\psi_{012}\rangle = |+_00_1+2\rangle, \tag{27}$$

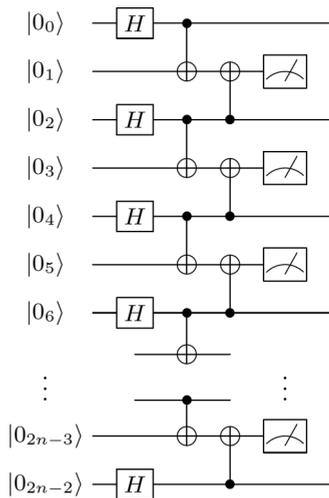


FIG. 6. Non-fault-tolerant circuit to make an n -size cat state in five steps.

with this order corresponding to the physical placement. Dispensing with normalization, as every term has the same amplitude, apply CNOT for Z_0Z_2 measurement:

$$\begin{aligned}
 |\psi'_{012}\rangle & = \text{CNOT}[2,1]\text{CNOT}[0,1]|+_00_1+2\rangle \\
 & = |0_00_10_2\rangle + |0_01_11_2\rangle + |1_01_10_2\rangle + |1_00_11_2\rangle, \tag{28}
 \end{aligned}$$

where $\text{CNOT}[a,b]$ denotes that qubit a is the control qubit and b is the target. Measure ancilla qubit 1 in the Z basis, and if the -1 eigenvalue is obtained, apply X_1 to get

$$|\psi''_{02}\rangle = |0_00_2\rangle + |1_01_2\rangle. \tag{29}$$

We can entangle another qubit in $|+\rangle$ to this state in the same way, and we can make a cat state of arbitrary length. However, this procedure is not fault tolerant, and there is a chance of getting a problematic state such as $|00001111\rangle + |11110000\rangle$. Using this state for a stabilizer measurement may produce a logical error because the logical operator of the deformation-based qubit is half of a superstabilizer. Therefore we need to confirm that we have a proper cat state. It is well known that measuring ZZ of every pair of qubits comprising the cat state is good enough for this proof [6]. Since measuring the ZZ stabilizer of every pair of qubits requires many SWAP gates and a lot of steps, we suggest repeating the ZZ measurement of every pair of neighboring qubits d times, where d is the code distance. This procedure guarantees that the probability of the state being in a problematic state is $O(p^{\lceil d/2 \rceil})$, where p is the physical error rate, giving the same error rate as that achieved by fault-tolerant quantum computation. (The state could still be an imperfect cat state such as $|00100000\rangle + |11011111\rangle$ due to individual physical errors, which is tolerable.)

B. Stabilizer measurement in constant time using a cat state

A three-qubit cat state can be rewritten as

$$\begin{aligned}
 |\psi_{\text{cat}}\rangle & = |000\rangle + |111\rangle \\
 & = (|+\rangle + |-\rangle)(|+\rangle + |-\rangle)(|+\rangle + |-\rangle) \\
 & \quad + (|+\rangle - |-\rangle)(|+\rangle - |-\rangle)(|+\rangle - |-\rangle) \tag{30}
 \end{aligned}$$

$$= |+++ \rangle + |+-+ \rangle + |-++ \rangle + |--- \rangle. \tag{31}$$

The $|000\rangle$ and $|111\rangle$ terms are rewritten in symmetric fashion except that the signs of factors involving an odd number of $|-\rangle$ differ, as shown in Eq. (30). From this fact and the binomial expansion, a cat state of any length involves an even number of $|-\rangle$. Applying a Z to any qubit in the cat state, the state in Eq. (31) is changed to

$$|\psi'_{\text{cat}}\rangle = |--+ \rangle + |--+ \rangle + |++- \rangle + |+-+ \rangle. \tag{32}$$

Applying a Z to any qubit again, this state returns to the state in Eq. (31). To observe whether we have the “even” cat state or the “odd” cat state, we need to measure all ancilla qubits in the X basis and calculate the product of the measured values.

Let us assume that we have as many ancillae for the cat state as we have data qubits to stabilize, and we can assign a qubit in the cat state to each data qubit, then apply a CNOT from each cat-state qubit to the corresponding data qubit. This set of CNOT gates is equivalent to the syndrome propagation for the $X_1 X_2 \cdots X_n$ stabilizer. The cat state starts from the “even” state, and if an odd number of flips is performed, the cat state results in the “odd” state. The CNOT gates can be applied simultaneously, and the measurement can be performed simultaneously; therefore this procedure requires three steps (CNOT gates, Hadamard, and measurement in the Z basis).

C. Superstabilizer implementation

To suppress the probability of having an improper cat state to $O(p^{\lceil \frac{d}{2} \rceil})$, a linear placement requires d cycles of ZZ stabilizers, but a circular placement requires only $\lceil \frac{d}{2} \rceil$ cycles. Let us assume that an example of the problematic states, $|0_0 0_1 0_2 0_3 1_4 1_5 1_6 1_7\rangle + |1_0 1_1 1_2 1_3 0_4 0_5 0_6 0_7\rangle$, has developed. In a linear arrangement, we have ZZ stabilizers only between neighboring qubits. After d cycles of ZZ stabilizers, the problematic state generation is caused by $p^{\lceil \frac{d}{2} \rceil}$ errors at the $Z_3 Z_4$ stabilizer. In a circular arrangement, we have another $Z_7 Z_0$ stabilizer. Hence, after d cycles of ZZ stabilizers, even though we have $p^{\lceil \frac{d}{2} \rceil}$ errors at the $Z_3 Z_4$ stabilizer, d cycles of the $Z_7 Z_0$ stabilizer tell us that we have an improper cat state. Therefore, for instance, $p^{\lceil \frac{d}{2} \rceil}$ errors at the $Z_3 Z_4$ stabilizer and $p^{\lceil \frac{d}{2} \rceil}$ errors at the $Z_7 Z_0$ stabilizer are required to generate a problematic cat state after d cycles of ZZ stabilizers, suppressing the improper cat-state generation probability to $O(p^d)$. Hence, to suppress the error probability to $O(p^{\lceil \frac{d}{2} \rceil})$, circular-fashion cat-state generation requires only $\lceil \frac{d}{2} \rceil$ cycles of ZZ stabilizers.

Figure 7 depicts the placement of two sets of ancilla qubits, each of which is prepared in a cat state for the X superstabilizer and for the Z superstabilizer. The dashed lines describe the cat-state qubits; red (blue) dots are qubits composing the cat state for the Z (X) superstabilizer, and gray dots are ancilla qubits to create and confirm the cat state (the ancilla’s ancilla). The qubits under both dashed lines are used for the Z and X ancilla qubits alternately. Therefore we need $\frac{d}{2} \times 2 = d$ cycles to measure both the Z superstabilizer and the X superstabilizer. The “thickness” of the deformation-based qubit in Fig. 7 is 2 to allow us to have the loop cat state. Greater thickness requires fewer cycles of the repeating ZZ stabilizer to confirm the cat state. We assume that the thickness is 2 throughout the rest of this paper to show the basic idea of our architecture.

The depth of the circuit to initialize a cat state is 5. A cycle of the following ZZ measurements for the proof requires four steps. The maximum number of CNOT gates to propagate error syndromes from data qubits to an ancilla qubit is 2, as shown in Fig. 7, at the corners of the superstabilizers. The total number of steps to measure a superstabilizer is the sum of $5 + 4(d - 1) = 4d + 1$ steps for cat-state creation and the proof, one step for a Hadamard gate for Z superstabilizer, two steps for syndrome propagation, one step for a Hadamard

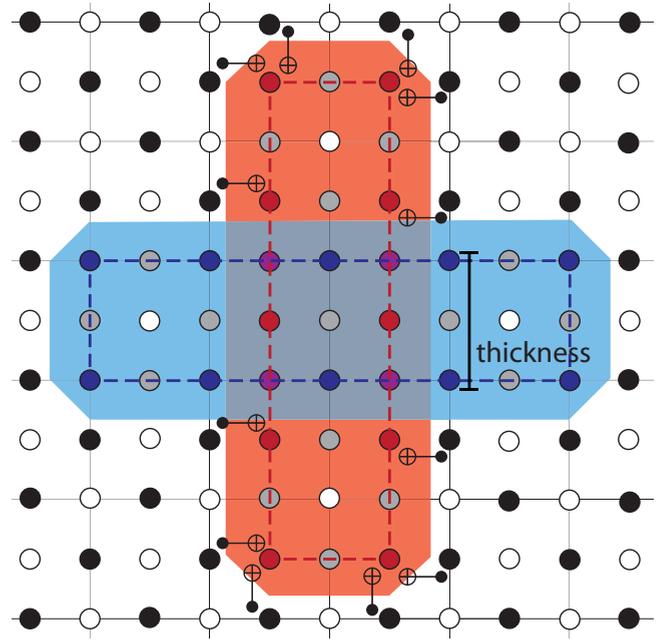


FIG. 7. Implementation of two cat states for a superstabilizer. The red dots are ancilla qubits prepared in a cat state for the Z superstabilizer. The red dashed loop describes the pairs for ZZ stabilizers to create and confirm the cat state. The ZZ stabilizer on each pair of neighboring red dots in this red dashed loop is executed. The gray qubits under the red dashed circle are qubits with odd indices in Fig. 6, used to measure ZZ stabilizers. The blue dots and the blue dashed circle are the same but for the X superstabilizer. The qubits (dots) where the dashed loops intersect are used for during the creation of both cat states in alternating fashion. The “thickness” of this deformation-based qubit is 2. The CNOT gates of the Z superstabilizer are shown. Each ancilla qubit on the corner of the loop handles two data qubits, and those along the sides handle one.

gate for the X superstabilizer, and one step for measurements, where d is the code distance. Therefore the number of steps to measure a superstabilizer is $4d + 5$.

However, in Fig. 7, two data qubits neighboring a corner of a loop cat state execute CNOT gates with the corner qubit so that an error on the corner qubit may propagate to the two data qubits, which may reduce the error suppression ability of the code. By judicious use of the unused qubits, we can recover the code distance lost, as shown in Fig. 8. In Fig. 7, for simplicity we show thickness $t = 2$ employing a cat state forming a complete loop, in which each corner cat-state qubit stabilizes two data qubits, resulting in reducing the effective code distance by 2. Figure 8 shows that, by utilizing unused physical qubits inside a superstabilizer, we can add more qubits to the cat state and can allow every cat state qubit to stabilize a data qubit. This improvement can be applied with code distance 8 or higher. This process is the same as the previous one, except that only one step is required for propagation. The first SWAP gates overlap with the measurements; then we add one step for the second SWAP gates, one step for syndrome propagation of ranged pairs, one step for a Hadamard gate for the X superstabilizer, and one step for measurements to replace a corner cat-state qubit with one made inside the

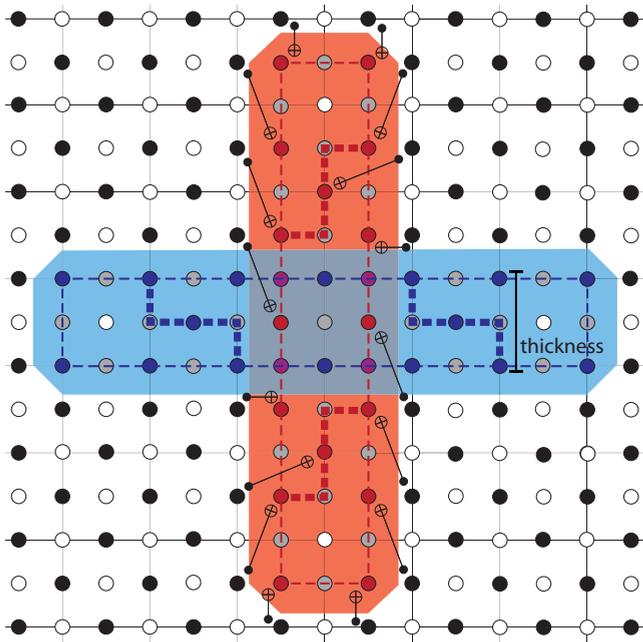


FIG. 8. Z superstabilizer in which a cat-state qubit stabilizes a data qubit. Qubits on the thick dashed lines are newly added to the cat-state qubits. It does not matter that the cat-state qubit on the cross of a thick dashed line and a thin dashed line is stabilized by three stabilizers to prove the correctness of the cat state since one cycle of stabilization for the proof takes four steps. Non-neighbor CNOT gates are executed after SWAP gates to neighbor the control and the target qubits.

superstabilizer, followed by error syndrome propagation and measurement. In total, $4d + 9$ steps are required.

VII. ERRORS

Although it might be thought that the deep circuit of the superstabilizer measurement results in a higher logical error rate than another surface code in which any stabilizer requires eight steps, we argue that the deformation-based surface code will exhibit a similar logical error rate with the conventional surface code. Figure 9 shows an example of two deformation-based qubits. Obviously, any single logical operator is protected by code distance 5, as shown in Fig. 2. Any single operator is protected by normal stabilizers every eight physical steps. Therefore conventional error analysis for surface code can be applied.

The pair of blue lines in Fig. 9 indicates the product of the two logical qubits' logical X operators. In order for a logical error to arise undetected, both error chains must occur. The short fragment of the operator product between (b) and (c) may occur easily and will be detected only by superstabilizer measurements, which are completed at every $4d + 5$ physical steps. The long fragment of the operator product between (a) and (d) should occur only rarely because the long fragment is protected by normal stabilizers and has a longer length than the code distance. Therefore the probability that this product operator happens to be executed by errors is strongly suppressed, although (b) and (c) are close and $4d + 5$ physical steps are required to measure superstabilizers.

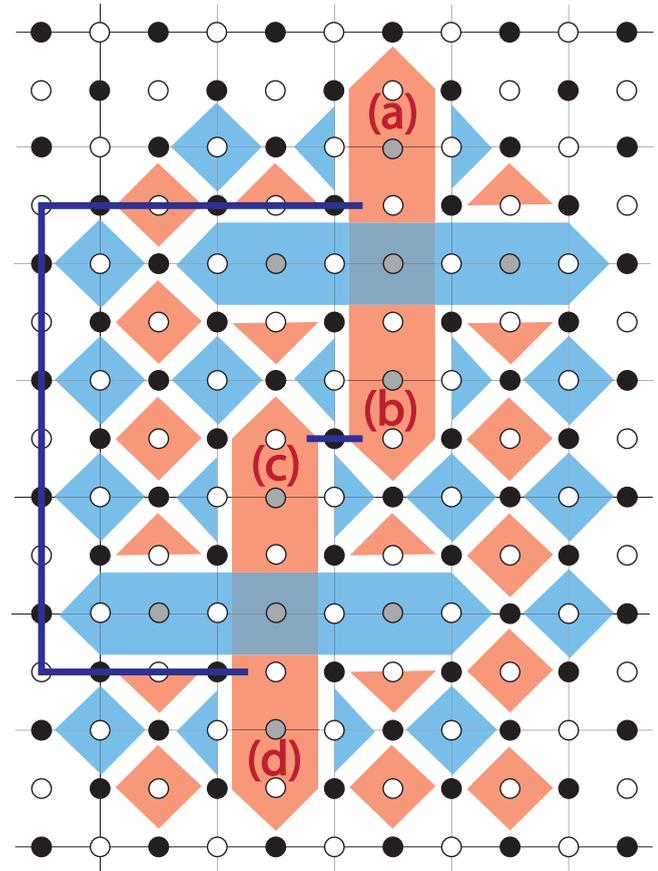


FIG. 9. Errors on deformation-based qubits due to the long execution time to measure superstabilizers. Both (a) and (b) are one half of a Z superstabilizer. A physical X error chain connecting those halves results in a logical X error for this deformation-based qubit. The same is true for (c) and (d). The set of blue lines describes the product of logical X operators of the two deformation-based qubits.

Figure 10 shows a problematic placement of deformation-based qubits. The code distance of each deformation-based qubit is 10. However, the product of the four logical X operators of those deformation-based qubits results in the combination of the four blue lines, each of which exists between two neighboring Z superstabilizers, consisting of only four physical qubits, reducing our minimum error chain to 4. Deformation-based qubits must be placed so that their superstabilizers do not form a loop.

In the next section, we present our scheme for dense packing that meets these constraints, then continue the discussion of errors.

VIII. DENSE PLACEMENT

Because of the restrictions described in Sec. VII, we locally set four deformation-based qubits as a box, as shown in Fig. 11, and globally place the boxes apart to maintain fault tolerance and to have free space available for routing intermediate qubits, as shown in Fig. 12. This local placement actually achieves dense packing; however, placing the logical qubits close together in this fashion results in error paths that shorten the effective distance.

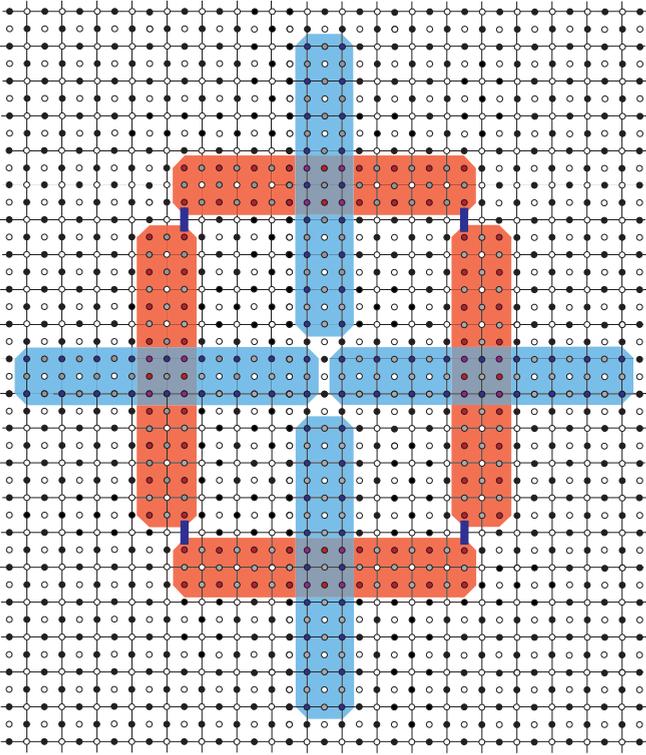


FIG. 10. Problematic placement of deformation-based qubits. Each deformation-based qubit has code distance 10. The shortest combined logical X operator for those four logical qubits is only 4, the combination of the blue lines.

To qualitatively analyze this effect, we define a hierarchy of distances: First, d_o is the original code distance of a single logical qubit, corresponding to the length around one of the arms, as in Fig. 1. Next, d_s is the shortened code distance, where the presence of neighboring superstabilizers may result in an error path of fewer hops. Finally, d_e is the effective code distance: the superstabilizers' longer cycle time results in higher vulnerability to errors, so we downgrade their ability to protect our data in this analysis by creating this artificially shortened distance. We want this final d_e to give us protection equivalent to or better than the protection of a planar code qubit of distance d , leading to the relation

$$d_o \geq d_s \geq d_e \geq d. \quad (33)$$

In the rest of this section, we explore this relationship in detail by comparing the number of error paths of the minimum length in several scenarios.

In Fig. 11, the path labeled d_s crosses two superstabilizers, one laterally and the other longitudinally. An error chain can cross a superstabilizer in a single hop. Thus, although the path d_s covers more ground than d_o , the number of errors in an undetected error chain is $d_s = d_o - t + 1$, where t is the thickness of the superstabilizer crossed laterally.

Worse, the deeper circuit of the superstabilizer increases the likelihood of error, so we choose to treat the superstabilizers as having no positive effect on error suppression. Removing them, our effective distance is $d_e = d_s - 2 = d_o - t - 1$.

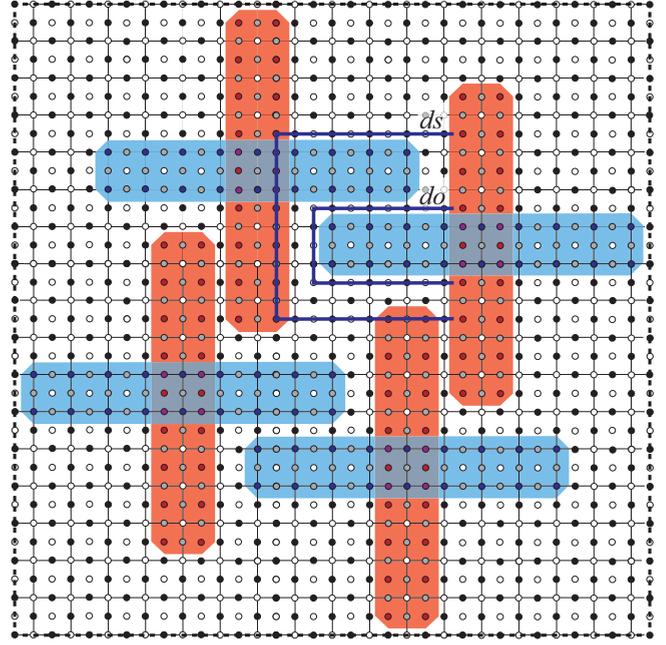


FIG. 11. Local placement of the deformation-based surface code. There are four logical qubits of distance 10 (d_o); however, since the thickness of superstabilizers shortens others' code distance by 1, the shortened code distance d_s is 9. This placement enables the four logical qubits to have lattice-surgery-like CNOT with other logical qubits. For thickness $t = 2$, each row and column has $3d_s + 8$ physical qubits, and $(3d_s + 8)^2 = 9d_s^2 + 48d_s + 64$ physical qubits are required for four logical qubits. The dashed box corresponds to the dashed box in Fig. 12. The blue lines describe minimal X error chains from the point of view of d_o and d_s , respectively, for the top right logical qubit. To downgrade the error tolerance in this analysis because of the two superstabilizers which have longer cycle time, we introduce the effective code distance d_e , where $d_e = d_s - 2$ and $(3d_s + 8)^2 = (3d_e + 14)^2 = 9d_e^2 + 84d_e + 196$ physical qubits are required for four logical qubits.

Finally, setting $t = 2$, this leads us to the following relationship for the dense packing of Fig. 11:

$$d_e = d, \quad (34)$$

$$d_s = d_e + 2, \quad (35)$$

$$d_o = d_s + 1 = d_e + 3 = d + 3. \quad (36)$$

With this layout, our four-fin logical qubits begin with a distance 3 longer than the defect-based qubits to achieve comparable logical error rates. As a result, $\frac{(5d_e+17)^2}{4} = \frac{25d_e^2+170d_e+289}{4}$ physical qubits are required for a logical qubit.

The global placement is shown in Fig. 12. The transformed qubit indicated by (A) is being routed. (A) is transformed during moving from one crossroads to another.

Since the surface code places data qubits and ancilla qubits alternately, $2d$ columns and rows are required to have code distance d . To avoid the situation shown in Fig. 10, (I) + (II) + (III) $\geq 2d$ must be satisfied to guarantee code distance d of (B) and (C). Since (III) is d , (I) + (II) needs to be d or

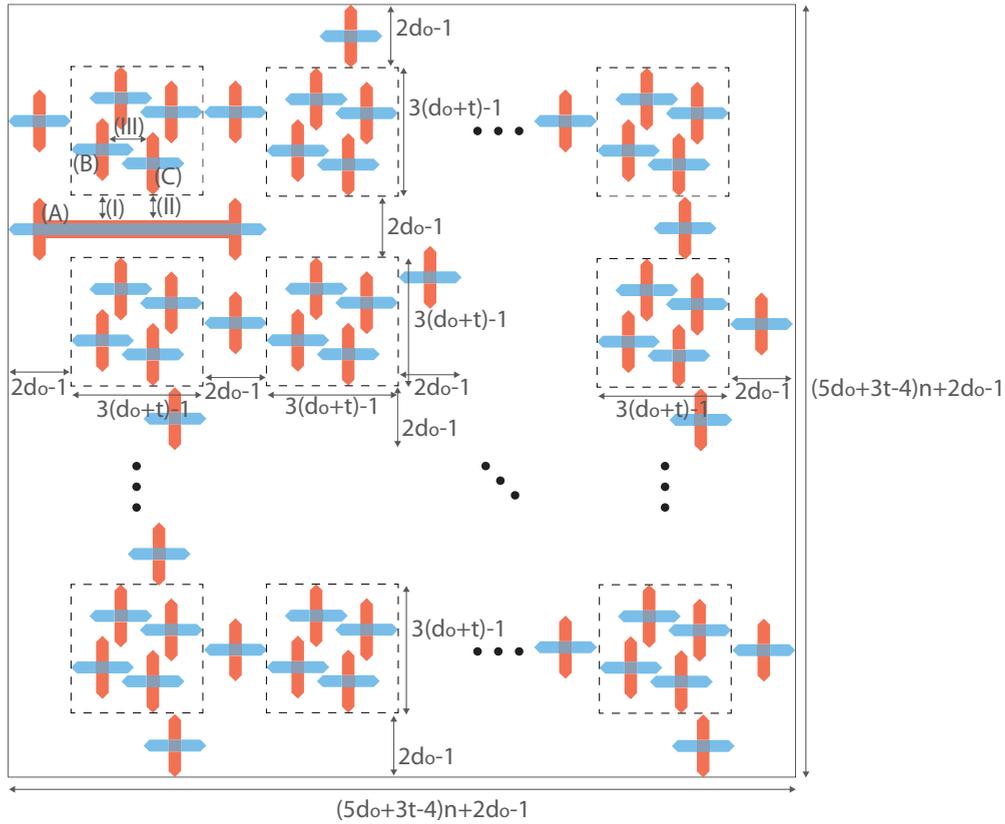


FIG. 12. Global placement of the deformation-based surface code. Each dashed box is the dashed box shown in Fig. 11. The spaces between the boxes are paths to move logical qubits and intermediate qubits. The deformation-based qubits outside of the dashed boxes are examples of intermediate qubits. There are $n \times n$ sets of the local placement. The lengths include both data qubits and ancilla qubits; hence $2d$ corresponds here to the code distance d . The stretched qubit indicated by (A) is being routed from location to location. To retain the fault tolerance of (B) and of (C), (I) + (II) needs to be $\frac{d}{2}$ or more; therefore (A) is transformed. The qubits on the boundary between a local placement set and a path are included in both $3(d_o + t) - 1$ and $2d_o - 1$; hence there are $(5d_o + 3t - 4)n + 2d_o - 1$ rows and $(5d_o + 3t - 4)n + 2d_o - 1$ columns. The total number of physical qubits is $[(5d_o + 3t - 4)n + 2d_o - 1]^2$ for $4n^2$ logical qubits, excluding intermediate qubits. This placement requires $(\frac{5d_o+3t-4}{2})^2 = \frac{25d_o^2+30d_o t+9t^2-40d_o-24t+16}{4}$ physical qubits per logical qubit for large enough n . This corresponds to $(\frac{5d_e+17}{2})^2 = \frac{25d_e^2+170d_e+289}{4}$ physical qubits being required for a logical qubit for $t = 2$.

more; hence (I) and (II) each must be $\frac{d}{2}$ or greater. Therefore (A) is transformed.

This placement design requires $(\frac{5d_o+3t-4}{2})^2 = \frac{25d_o^2+30d_o t+9t^2-40d_o-24t+16}{4}$ physical qubits per logical qubit for large enough n . Choosing $t = 2$, $(\frac{5d_e+7}{2})^2 = (\frac{5d_e+17}{2})^2 = \frac{25d_e^2+170d_e+289}{4}$ physical qubits are required for a logical qubit, including ancilla qubits.

In contrast, the planar code's placement for lattice-surgery-based operation, shown in Fig. 13, requires $(4d - 2)^2 = 16d^2 - 16d + 4$ physical qubits per logical qubit.

As a result, the deformation-based surface code requires 50% fewer physical qubits than the planar code. Horsman *et al.* showed that the number of required qubits for the defect-based surface code is similar to that of the planar code in large-scale quantum computation, so deformation-based surface code also requires fewer physical qubits than the defect-based surface code [16].

The complex interactions during syndrome extraction and the difficulties of the error-matching processing make direct calculation of residual logical error rates infeasible, but we can make a qualitative comparison by examining the number

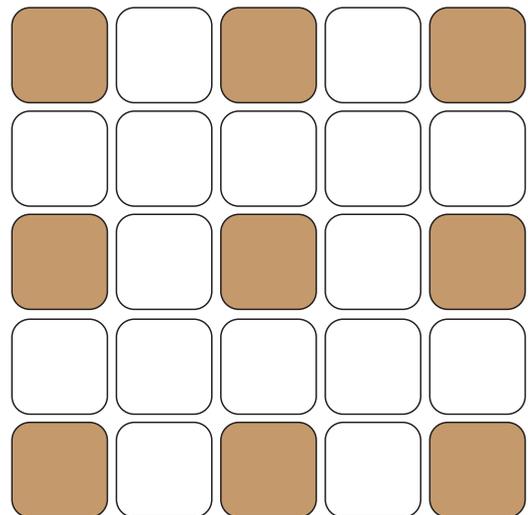


FIG. 13. Planar code placement for comparison, after Fig. 12 in [16]. Each shaded area holds a logical data qubit, and blank areas are available for intermediate qubits for CNOT gate by lattice surgery. Each area has $2d - 1$ by $2d - 1$ physical qubits, including ancillae.

of redundant logical operators, each of which may potentially be a logical error. For a code of distance d , logical operators of length d dominate. A planar code qubit has d redundant logical operators of length d for each type of logical operator. A defect-based code qubit has $\frac{d}{4}$ redundant logical operators of length d between the two defects.

In contrast, a deformation-based qubit which is distant from the lattice boundary and distant from other logical qubits has exactly two redundant logical operators of length d_e for each type.

Hence the isolated deformation-based code has fewer potential error operators. With the placement shown in Fig. 11, because of the presence of a superstabilizer of a neighboring logical qubit, the number of potential logical error operators increases to $\frac{d_e^3 - 11d_e^2 + 35d_e - 25}{8}$ redundant logical operators of length d_e .

This is because, for example, a horizontal error chain at the top left of the lattice and a horizontal error chain at the bottom right do not result in a logical error since they would be distant if the surface consisted only of normal stabilizers. However, if there is a vertical long superstabilizer in the middle of the lattice which makes the error chains close, those two error chains may cause a logical error.

For distances 9, 19, and 29, the deformation-based code may have 2, 23, and 70 times as many potential logical error operators of length d_e as the planar code. (Obviously, a deformation-based code of effective code distance d_e does not have error operators of geometric length d_e since the shortened code distance d_s is the geometric distance and $d_e = d_s - 2$; hence we counted error operators by the shortened code distance corresponding to the effective code distance we want.) The effective code distance d_e is defined stringently; therefore we may not need to be concerned about this overhead to compare the deformation-based qubit and other surface-code qubits. Otherwise, this overhead can be tolerated by employing one greater code distance.

A surface-code qubit utilizing the rotated lattice requires only $2d^2 - 1$ physical qubits to encode a logical qubit [16]. However, since the rotated planar code does not directly support lattice surgery because of its irregular boundaries, either a transversal CNOT gate or conversion to a standard planar code is required to achieve practical quantum computation. Implementing a transversal CNOT gate may kill the surface code's advantage on feasibility. Conversion to the standard planar code requires a memory area large enough for the standard planar code and requires the paths for logical qubit transfer wide enough to transfer standard planar code qubits, eventually killing the rotated lattice qubit's advantage on the resource requirement. Therefore we focus on the standard planar code for comparison rather than the rotated-lattice planar code.

We employed the thickness $t = 2$ in this example for simplicity. Using thickness $t = 3$ instead will shorten the columns and the rows of a deformation-based qubit. Because

an even code distance has the same error suppression capability as the odd distance just below it, a $t = 2$ logical qubit and a $t = 3$ logical qubit should have $2d + 1$ and $2d - 1$ columns and rows, respectively. This allows us to slightly narrow the interblock channels in Fig. 12.

As in the defect-based code, the Hadamard gate is executed by isolating the logical qubit from the rest of the surface, exchanging X and Z stabilizers, then reconnecting it to the surface. With the dense packing, there is no room around the qubit to disconnect it from the surface, so the qubit first should be moved out into the channel before performing the Hadamard.

IX. DISCUSSION

We have shown the acceptability of close placement of the deformation-based surface code by measuring superstabilizers which produce deformation-based qubits; direct conversion from the defect-based surface code to the deformation-based surface code, which can be used as state injection for the deformation-based surface code; and a lattice-surgery-like CNOT gate for the deformation-based qubits which requires fewer physical qubits than the braiding CNOT gate. The acceptability of close placement and the space-saving CNOT gate allow deformation-based qubits to be packed more tightly than planar code qubits and defect-based qubits.

We have shown theoretical basic concepts but have not calculated the error suppression ability since that of the surface code has been investigated well. The superstabilizers which compose deformation-based qubits require $4d_e + 9$ steps for stabilizer measurements where d_e is the effective code distance. Our placement design preserves logical qubits as any logical operator passes through a chain of normal stabilizers that compose the effective code distance d_e . Hence, by adding 3 to the original code distance, the long stabilizer measurement does not degrade the error suppression efficiency. The deformation-based surface code should have a residual error rate similar to the conventional surface code of code distance 3 shorter, and hence conventional error analysis for the surface code can be applied to the deformation-based surface code.

Our design requires $\frac{25d_e^2 + 170d_e + 289}{4}$ physical qubits for a logical qubit, compared to the $16d^2 - 16d + 4$ physical qubits required in the conventional design. Our design would halve the resources required to build a large-scale quantum computer.

Note added. Recently, a preprint appeared with a similar goal of reducing the resource consumption of the surface code but using very different methods [21].

ACKNOWLEDGMENT

This work is supported by JSPS KAKENHI Grant No. 25:4103 and Kiban B 16H02812.

[1] A. G. Fowler, A. M. Stephens, and P. Groszkowski, *Phys. Rev. A* **80**, 052312 (2009).

[2] A. Kitaev, *Ann. Phys. (NY)* **303**, 2 (2003).

[3] S. B. Bravyi and A. Y. Kitaev, [arXiv:quant-ph/9811052](https://arxiv.org/abs/quant-ph/9811052).

- [4] R. Raussendorf and J. Harrington, *Phys. Rev. Lett.* **98**, 190504 (2007).
- [5] R. Raussendorf, J. Harrington, and K. Goyal, *New J. Phys.* **9**, 199 (2007).
- [6] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, 2000).
- [7] R. Van Meter and C. Horsman, *Commun. ACM* **56**, 84 (2013).
- [8] J. Cirac *et al.*, *Nature (London)* **404**, 579 (2000).
- [9] N. Y. Yao, L. Jiang, A. V. Gorshkov, P. C. Maurer, G. Giedke, J. I. Cirac, and M. D. Lukin, *Nat. Commun.* **3**, 800 (2012).
- [10] J. Chiaverini, J. Britton, D. Leibfried, E. Knill, M. D. Barrett, R. B. Blakestad, W. M. Itano, J. D. Jost, C. Langer, R. Ozeri, T. Schaetz, and D. J. Wineland, *Science* **308**, 997 (2005).
- [11] A. G. Fowler, W. F. Thompson, Z. Yan, A. M. Stephens, B. L. T. Plourde, and F. K. Wilhelm, *Phys. Rev. B* **76**, 174507 (2007).
- [12] T. M. Stace, S. D. Barrett, and A. C. Doherty, *Phys. Rev. Lett.* **102**, 200501 (2009).
- [13] S. D. Barrett and T. M. Stace, *Phys. Rev. Lett.* **105**, 200502 (2010).
- [14] T. Ladd, F. Jelezko, R. Laflamme, Y. Nakamura, C. Monroe, and J. O'Brien, *Nature (London)* **464**, 45 (2010).
- [15] E. Dennis, A. Kitaev, A. Landahl, and J. Preskill, *J. Math. Phys.* **43**, 4452 (2002).
- [16] C. Horsman, A. G. Fowler, S. Devitt, and R. Van Meter, *New J. Phys.* **14**, 123011 (2012).
- [17] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, *Phys. Rev. A* **86**, 032324 (2012).
- [18] H. Bombin and M. A. Martin-Delgado, *J. Phys. A: Math. Theor.* **42**, 095302 (2009).
- [19] S. Nagayama, B.-S. Choi, S. Devitt, S. Suzuki, and R. Van Meter, *Phys. Rev. A* **93**, 042338 (2016).
- [20] Note that Eqs. (5)–(9) and (14)–(21) describe *states* that are stabilized by the corresponding terms but do not correspond directly to stabilizer measurements conducted for error correction purposes. In particular, the last line in Eqs. (8) and (9) represents the newly introduced superstabilizer itself, while the two stabilizers just above illustrate the degree of freedom representing our logical qubit. The stabilizers marking the degree of freedom are labeled in the leftmost column inside the parentheses with + or – as appropriate.
- [21] N. Delfosse, P. Iyer, and D. Poulin, [arXiv:1606.07116](https://arxiv.org/abs/1606.07116).