



OPEN

# Step-by-step magic state encoding for efficient fault-tolerant quantum computation

SUBJECT AREAS:  
COMPUTER SCIENCE  
QUANTUM INFORMATION

Hayato Goto

Received  
25 April 2014Accepted  
27 November 2014Published  
16 December 2014Correspondence and  
requests for materials  
should be addressed to  
H.G. (hayato1.goto@  
toshiba.co.jp)

Frontier Research Laboratory, Corporate Research &amp; Development Center, Toshiba Corporation, 1, Komukai Toshiba-cho, Saiwai-ku, Kawasaki-shi, 212-8582, Japan.

Quantum error correction allows one to make quantum computers fault-tolerant against unavoidable errors due to decoherence and imperfect physical gate operations. However, the fault-tolerant quantum computation requires impractically large computational resources for useful applications. This is a current major obstacle to the realization of a quantum computer. In particular, magic state distillation, which is a standard approach to universality, consumes the most resources in fault-tolerant quantum computation. For the resource problem, here we propose *step-by-step magic state encoding* for concatenated quantum codes, where magic states are encoded step by step from the physical level to the logical one. To manage errors during the encoding, we carefully use error detection. Since the sizes of intermediate codes are small, it is expected that the resource overheads will become lower than previous approaches based on the distillation at the logical level. Our simulation results suggest that the resource requirements for a logical magic state will become comparable to those for a *single* logical controlled-NOT gate. Thus, the present method opens a new possibility for efficient fault-tolerant quantum computation.

An arbitrarily long quantum computation can, in principle, be performed reliably even with faulty quantum hardware with high physical error probabilities up to about 1%<sup>1-3</sup>. The fault-tolerant quantum computation is based on the use of quantum error correction<sup>4</sup>. However, it is known that the resource overheads for fault-tolerant quantum computation become enormous for practical applications<sup>5-7</sup>, such as factoring large integers by Shor's algorithm<sup>8</sup>.

The most resource-intensive part in fault-tolerant quantum computation is *magic state distillation*<sup>5,6,9-12</sup>, the role of which is as follows. For many quantum error-correcting codes, the basic operations called the Clifford operations (or stabilizer operations) can be performed reliably with relatively low overheads. However, it is well known that the Clifford operations are not only insufficient for universal quantum computation, but also can be simulated efficiently by classical computers<sup>4,13</sup>. Thus, useful applications require at least one non-Clifford operation. Although universality can be achieved by adding a non-Clifford gate to the Clifford operations<sup>4</sup>, most non-Clifford gates are difficult to perform reliably. Magic state distillation solves this problem as follows. Some non-stabilizer states called *magic states* have a good property that they can be distilled using perfect Clifford operations alone, where state distillation is a technique to produce high-fidelity states from more low-fidelity states. (Non-stabilizer states are defined as states that cannot be prepared using Clifford operations alone.) The assumption of perfect Clifford operations is valid if we use logical qubits and gates encoded with a sufficiently large code. On the other hand, the low-fidelity magic states can be prepared by the technique called state injection<sup>1,6</sup>. Thus, we can produce reliable magic states by combining magic state distillation with state injection, which enable reliable non-Clifford gates<sup>9</sup>.

The magic state distillation is a standard approach to universality in fault-tolerant quantum computation. This was first proposed by Bravyi and Kitaev<sup>9</sup>. (A similar method with equivalent performance was also proposed by Knill independently<sup>1,14</sup>.) Since the magic state distillation assumes perfect Clifford operations, this requires many logical qubits and gates encoded with a large code, as mentioned above. For instance, the first 15-to-1 distillation protocol requires 15 logical ancillary magic states (ancillas) and 34 logical controlled-NOT (CNOT) gates to obtain one high-fidelity magic state<sup>5,9</sup>. Thus, the resource overheads become enormous.

Recently, several improvements for magic state distillation have been proposed<sup>10-12</sup>. However, a recent detailed study<sup>15</sup> with a specific code called the surface code has shown that a state-of-the-art method<sup>12</sup> is comparable to the original one<sup>9</sup>. Hence, substantial reduction of resource overheads for magic state distillation has still been desired.



(Very recently, new approaches to universality without magic state distillation have been proposed<sup>16,17</sup>. However, they are based on the use of inefficient quantum codes for computation, and consequently they may not reduce resource overheads substantially.)

The previous methods<sup>9–12</sup> of magic state distillation assume perfect Clifford operations and therefore require many logical qubits and gates encoded with large codes in order to utilize these methods. (For instance, in refs 5 and 6, errors in logical Clifford operations used for the 15-to-1 distillation<sup>9</sup> are neglected by using sufficiently large surface codes.) This may be the reason why the resource overheads have not been reduced substantially so far. Towards the substantial reduction of the resource overheads, here we carefully treat errors in encoded Clifford operations. As a result, we show that magic states can be encoded step by step from the physical level to the logical one with a concatenated Calderbank-Shor-Steane (CSS) code<sup>4</sup>. We call this method *step-by-step magic state encoding*, where we carefully use error detection. For the error detection, we introduce two new techniques: magic state distillation with a *Bell-state ancilla* and *level-raising teleportation*. Since the sizes of intermediate codes are small, it is expected that the resource overheads will become low compared to previous approaches based on state injection and distillation at the logical level. Our simulation results suggest that the resource requirements for a logical magic state will become comparable to those for a *single* logical controlled-NOT gate.

## Results

**Error model and error probabilities.** Before describing the present method, we start with the error model assumed here and the basics of quantum error correction.

All the physical operations required for quantum computation are as follows: CNOT gates; single-qubit gates, especially Hadamard gates (denoted by  $H$ ); qubit measurements in the bases  $\{|0\rangle, |1\rangle\}$  and  $\{|+\rangle, |-\rangle\}$  (denoted by  $M_{0/1}$  and  $M_{+/-}$ , respectively), where  $|\pm\rangle \equiv \frac{|0\rangle \pm |1\rangle}{\sqrt{2}}$ ; and state preparations of  $|0\rangle$  and  $|+\rangle$ . The output,  $m$ , of  $M_{+/-}$  is defined as  $m = 0$  and  $m = 1$  for  $|+\rangle$  and  $|-\rangle$ , respectively.

The error model assumed here is very simple: only physical CNOT gates are imperfect among the above physical operations<sup>7</sup>. This assumption is valid and useful for the following reasons: (1) CNOT gates usually have the highest error probability among the above operations; (2) in most cases CNOT gates are used most frequently among the above operations; and (3) the other operations are almost always preceded or followed by CNOT gates and therefore we can assume that errors of the other operations are included in those of CNOT gates. The model for an imperfect CNOT gate is the standard one<sup>17</sup>: one of the 15 two-qubit Pauli errors occurs on the control and target qubits with probability  $p_c/15$  after a perfect CNOT gate operation. More explicitly, the density operator  $\rho$  for the total system is converted by an imperfect physical CNOT gate as follows:  $\rho \rightarrow$

$$(1-p_c)U_{\text{CNOT}}\rho U_{\text{CNOT}}^\dagger + \frac{p_c}{15} \sum_{n=1}^{15} P_n U_{\text{CNOT}} \rho U_{\text{CNOT}}^\dagger P_n, \text{ where } U_{\text{CNOT}}$$

denotes the unitary operator for the perfect CNOT gate operation and  $P_n$  denotes one of the 15 two-qubit Pauli operators ( $I \otimes X, I \otimes Y, I \otimes Z, X \otimes I, X \otimes X, X \otimes Y, X \otimes Z, Y \otimes I, Y \otimes X, Y \otimes Y, Y \otimes Z, Z \otimes I, Z \otimes X, Z \otimes Y, Z \otimes Z$ , where the first and second Pauli operators correspond to the control and target qubits for the physical CNOT gate, respectively). (The Pauli operators and the identity operator are denoted by  $X, Y, Z$ , and  $I$ , respectively.)

Next, we present the basics of quantum error correction. Quantum error correction is achieved by encoding quantum information into physical qubits with a quantum error-correcting code. Most useful quantum error-correcting codes are stabilizer codes<sup>4</sup>, the code space of which is defined as a simultaneous eigenspace of a set of Pauli products called stabilizer generators (or check operators). The eigenvalues of the stabilizer generators are called error syndromes, which

are usually set to  $+1$  for the code space. If some syndromes are  $-1$ , there exists at least an error.

The stabilizer codes each of whose stabilizer generators consists of only  $X$  or only  $Z$  are called CSS codes<sup>4</sup>. For CSS codes, the encoded  $X$  and  $Z$  operators also consist of only physical  $X$  and  $Z$  operators, respectively. In this paper, we focus only on CSS codes.

An important quantity characterizing error-correcting codes is a *code distance*. The distance of a CSS code is defined as the minimum number of non-identity physical Pauli operators in the encoded Pauli operators among all the possible representations. (Note that the product of an encoded operator and a stabilizer generator gives another representation of the encoded operator.) This is equal to the minimum number of physical Pauli errors inducing encoded Pauli errors.

In general, distance- $d$  CSS codes can detect physical single-qubit errors less than  $d$ , because such errors take a state out of the code space and therefore are detectable by error syndromes. On the other hand, distance- $d$  CSS codes can also correct, in principle, physical single-qubit errors less than  $\lfloor (d+1)/2 \rfloor$ , where  $\lfloor x \rfloor$  denotes the floor function which returns the greatest integer equal to or less than  $x$ . (See ref. 4 for details.)

The process for estimating the results at encoded qubit measurements or extracting the error information from syndrome measurements is called *decoding*. The decoding for the above error detection is simple: unless all the syndromes are  $+1$ , the decoder indicates that errors exist. We refer to errors undetectable by this decoding as *undetectable errors*. (The others are detectable errors.) Since the undetectable errors convert a state into another within the code space, they are equivalent to encoded operations for the code. In this sense, we also refer to the undetectable errors as *encoded errors*. (Encoded errors for logical qubits are particularly referred to as *logical errors*.) Note that encoded Pauli errors are typical such errors. The decoding for the above error correction is much more difficult. It is known that such an optimal decoding for error correction is computationally hard for general codes. However, the optimal decoding can be performed efficiently for *concatenated quantum codes* composed of small codes<sup>7,18,19</sup>, which are the very ones considered in the present work. We refer to errors uncorrectable by this optimal decoding as *uncorrectable errors*. (The others are correctable errors.) Here it should be noted that “undetectable” and “uncorrectable” depend on the code used. Therefore we should clarify the code used (if it is not obvious) in mentioning the two terms.

Finally, we examine decoding error probabilities. If there exist only independent physical single-qubit errors with probability  $p$ , the undetectable and uncorrectable error probabilities for distance- $d$  codes are  $O(p^d)$  and  $O(p^{\lfloor (d+1)/2 \rfloor})$ , respectively, where  $O(f(p))$  is defined such that there exists a constant  $C$  satisfying  $|O(f(p))| < C|f(p)|$  in the limit  $p \rightarrow 0$ . (Note that larger exponents mean lower error probabilities assuming that  $p$  is sufficiently small.) It is noticeable that we can also choose decoding algorithms achieving intermediate performance<sup>1,7,18,19</sup> such that the undetectable error probability is  $O(p^x)$  where  $\lfloor (d+1)/2 \rfloor < x < d$ . To distinguish such intermediate error detection from the above error detection, we refer to the above error detection as *full* error detection. The intermediate error detection plays an important role to reduce the resource overheads in the step-by-step magic state encoding.

If physical single-qubit errors in a code block are correlated, the decoding error probabilities become higher. Therefore, it is important to retain the independence of physical single-qubit errors. Encoded operations implemented transversally<sup>4</sup> (by bitwise physical operations) fulfill this condition automatically. For CSS codes, an encoded CNOT gate, encoded Pauli gates, and encoded qubit measurements ( $M_{0/1}$  and  $M_{+/-}$ ) can be implemented transversally.

**Step-by-step magic state encoding.** Here we describe the step-by-step encoding of a level-3 magic state as a logical qubit for



computation, which is depicted in Fig. 1. In Fig. 1 and throughout this paper, we suppose that the concatenated code used is composed of distance-2 CSS codes. (The generalization to distances larger than 2 is an open question.) In this case, the code distance is  $2^L$ , where  $L$  denotes the concatenation level.

In Fig. 1, the following two points should be noted. (1) The uncorrectable error probability of the level-3 magic state is sufficiently lowered to  $O(p_c^4)$ . (Since the distance of the level-3 code is 8, we cannot achieve, in principle, the uncorrectable error probability of a higher order of  $p_c$ .) (2) Ancillas used for the encoding of the level-3 magic state are only level-1 and level-2 ones. Hence, it is expected that the necessary resources will become much less than those for previous approaches based on state injection and distillation at the logical level. (For instance, if we use the 15-to-1 protocol at the logical level, we need 15 level-3 ancillas together with 34 level-3 encoded CNOT gates.)

Note that such an encoding will be possible if we use full error detection after all the encoded operations because the undetectable error probabilities for level- $L$  encoded qubits are of the same order of  $p_c$ ,  $O(p_c^{2^L})$ , as the uncorrectable error probabilities for level- $(L+1)$  ones. (Here the encoded operations are assumed to be implemented transversally.) However, this naïve application of error detection will lead to the substantial increase of overheads due to postselection based on the error detection. To reduce the overheads, we use error detection more carefully. For the error detection, we introduce the two techniques: magic state distillation with a Bell-state ancilla and level-raising teleportation, where the former is used for the distillation steps (Steps 2 and 4 in Fig. 1) and the latter is used for the encoding steps (Steps 3 and 5 in Fig. 1). The magic state distillation with a Bell-state ancilla allows us to use intermediate error detection, not full one, for the distillation. The full error detection is used only for the level-raising teleportation, the increase of overheads for which is not very large for small  $p_c$  (see Supplementary Information). In the following, we explain our method in detail.

First of all, we suppose that the magic state distilled is the  $+1$  eigenstate of the Hadamard operator,  $|H\rangle = \cos(\pi/8)|0\rangle + \sin(\pi/8)|1\rangle$ , and the distillation protocol is the 14-to-2 one with a  $H_6$  code<sup>12</sup> (see below for the details). The magic state  $|H\rangle$  enables to perform a non-Clifford gate  $R_Y(\pi/4)$ , as shown in Fig. 2a, where  $R_Y(\theta)$  is defined as  $R_Y(\theta) = \begin{pmatrix} \cos(\theta/2) & -\sin(\theta/2) \\ \sin(\theta/2) & \cos(\theta/2) \end{pmatrix}$  in the computational basis<sup>4</sup>. Note that  $|H\rangle$  is expressed as  $|H\rangle = R_Y(\pi/4)|0\rangle$ .

The first step (Step 1 in Fig. 1) is the encoding of a magic state into physical qubits with a code. This can be done easily by an encoder of the code. Since physical CNOT gates in the encoder induce corre-

lated errors in the code block, the undetectable error probability is  $O(p_c)$ , not  $O(p_c^2)$ , as shown in Fig. 1.

The second step (Step 2 in Fig. 1) is the distillation, the role of which is the reduction of the undetectable error probability, as shown in Fig. 1. Since this step is very similar to Step 4 in Fig. 1, here we explain both together.

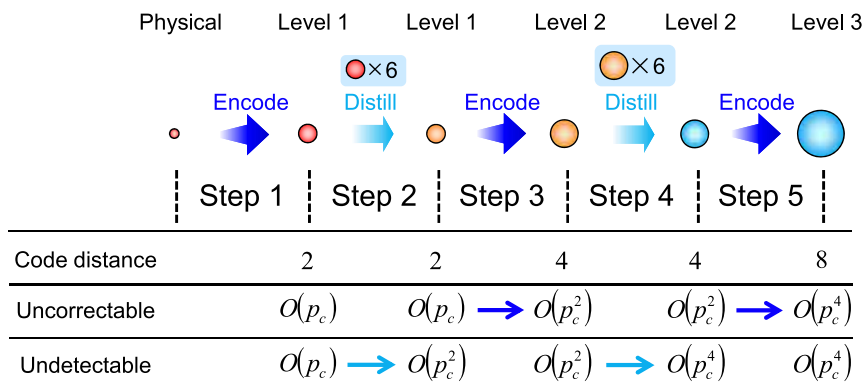
As mentioned above, we use the 14-to-2 distillation protocol with a  $H_6$  code. The  $H_6$  code encodes two qubits into six qubits and has distance 2. This also has a good property that the encoded Hadamard gate can be implemented transversally. The  $H_6$  code used here is defined as follows: the stabilizer generators are XXXXII, IXXXXX, ZZZZII, and IIZZZZ; the encoded Pauli operators are XIXIXI and ZIZIZI for the first encoded qubit, and IXIXIX and IZIZIZ for the second encoded qubit.

The quantum circuit for the 14-to-2 protocol is shown in Fig. 2c, where the controlled-Hadamard gates are implemented as shown in Fig. 2b and each  $R_Y(\pi/4)$  in Fig. 2b is performed with an imperfect  $|H\rangle$ , as shown in Fig. 2a. Importantly, we use a Bell state  $\frac{|0\rangle|0\rangle + |1\rangle|1\rangle}{\sqrt{2}}$  as an ancilla, instead of  $|+\rangle$ , unlike ref. 12. This change is essential for the successful distillation with imperfect Clifford operations, as discussed later.

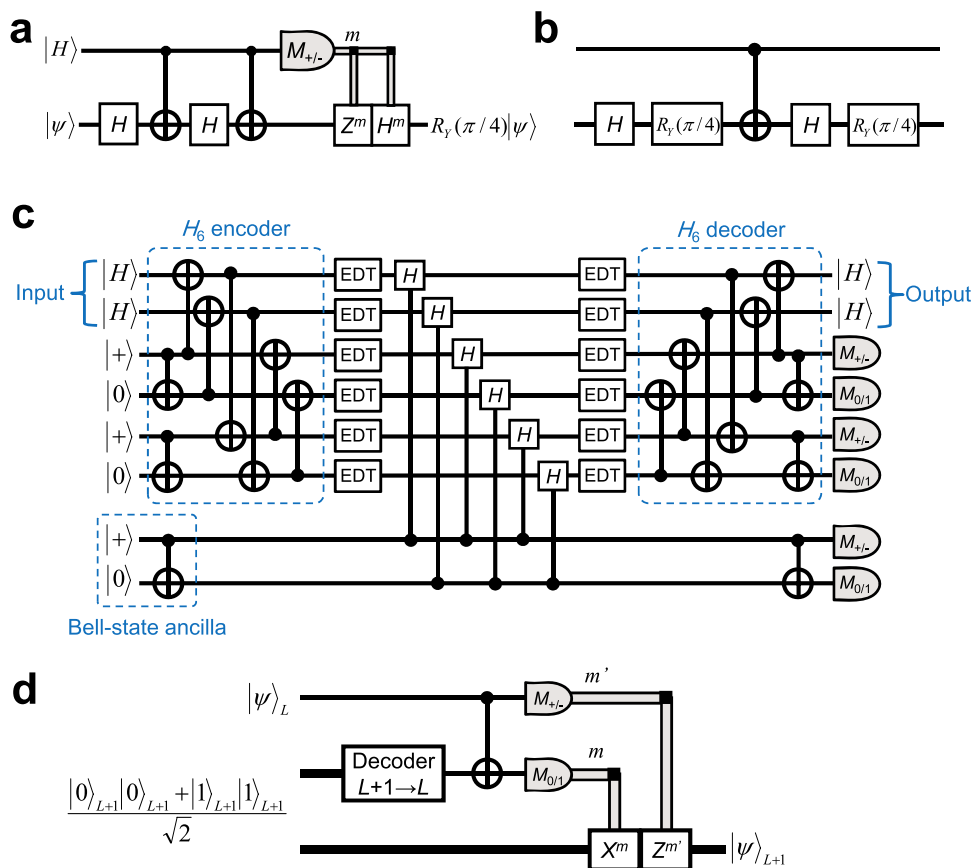
The encoded operations in Figs. 2a–2c are implemented as follows. In Figs. 2a–2c, all the encoded CNOT and Hadamard gates are implemented by transversal physical ones. (The transversality of the Hadamard gate is not necessary but assumed for simplicity.) As shown in Fig. 2c, the  $H_6$  encoder and the controlled-Hadamard gates are followed by blockwise error-detecting teleportations (EDTs)<sup>1,7</sup>, which convert independent physical errors due to transversal gates into encoded Pauli errors. The role of the EDTs is not only to reduce error probabilities but also to simplify the error analysis of the distillation, as discussed later. At the encoded qubit measurements, we also perform error detection. The distillation is repeated until no error is detected and all the measurement results in Fig. 2c become zero.

The decoding algorithms used for the EDTs and the encoded qubit measurements are such that the undetectable error probabilities for independent physical errors due to transversal gates are  $O(p_c^2)$  and  $O(p_c^3)$  in Steps 2 and 4, respectively. In Step 4, we use not full error detection but intermediate one to reduce the overheads due to the repetition, as mentioned above.

Here we explain how the distillation achieves the desired reduction of the undetectable error probabilities shown in Fig. 1. Since the EDTs and the encoded qubit measurements convert independent physical errors due to transversal gates into encoded errors, the



**Figure 1 | Step-by-step magic state encoding.** This illustrates the step-by-step encoding of a level-3 magic state as a logical qubit for computation. The circles represent physical or encoded qubits, the larger sizes of which correspond to the higher levels of concatenation. The rows labeled “Uncorrectable” and “Undetectable” show the uncorrectable and undetectable error probabilities, respectively, for the concatenated code at each position, where  $p_c$  denotes the error probability per physical CNOT gate. Distillation and encoding are performed by magic state distillation with a Bell-state ancilla (Fig. 2c) and level-raising teleportation (Fig. 2d), respectively.



**Figure 2 | Magic state distillation with a Bell-state ancilla and level-raising teleportation.** (a) Non-Clifford gate  $R_Y(\pi/4)$  with a magic state  $|H\rangle = R_Y(\pi/4)|0\rangle$ .  $m$  denotes the measurement result. (b) Implementation of the controlled-Hadamard gate, where the upper and lower lines represent the control and target qubits, respectively. (c) Quantum circuit for magic state distillation with a Bell-state ancilla, which is used in the distillation steps in the step-by-step magic state encoding (Steps 2 and 4 in Fig. 1). “EDT” represents error-detecting teleportation. (d) Level-raising teleportation. This is used in the encoding steps in the step-by-step magic state encoding (Steps 3 and 5 in Fig. 1).

following reduced error model based on encoded errors is valid for the present error analysis: encoded errors occur on the input states ( $|H\rangle$ ,  $|0\rangle$ , and  $|+\rangle$ ) with probabilities  $O(p_c)$  and  $O(p_c^2)$  in Steps 2 and 4, respectively; encoded two-qubit errors occur on the control and target qubits of encoded CNOT gates with probabilities  $O(p_c^2)$  and  $O(p_c^3)$  in Steps 2 and 4, respectively, with the exception that in Step 4, encoded errors on the outputs of the distillation induced by the encoded CNOT gates in the  $H_6$  decoder occur with probability  $O(p_c^4)$ , not  $O(p_c^3)$ , because of the level-raising teleportation with full error detection (see below). Since the probabilities that two of the above encoded errors occur are  $O(p_c^2)$  and  $O(p_c^4)$  in Steps 2 and 4, respectively, it is sufficient for the desired result to show that if only one of the above encoded errors occurs, this can be detected. This is shown as follows.

First, encoded errors on the first set of the six EDTs, which are induced by the input states and the encoded CNOT gates for the  $H_6$  encoder, are divided into undetectable and detectable errors for the  $H_6$  code. The undetectable errors for the  $H_6$  code induce a phase flip in the Bell-state ancilla via the controlled-Hadamard gates<sup>12</sup>. This can be detected by the ancilla measurements. On the other hand, any detectable error for the  $H_6$  code can be detected by the measurements after the  $H_6$  decoder. (These correspond to the syndrome measurements of the  $H_6$  code).

Second, encoded errors on the Bell-state ancilla can be detected by the ancilla measurements. Note that both encoded  $X$  and  $Z$  errors are detectable, unlike the case of the  $|+\rangle$  ancilla used in ref. 12. This is important because the encoded  $X$  errors on the ancilla induce an undetectable error for the  $H_6$  code via the controlled-Hadamard

gates and consequently lead to the failure of the distillation. This is the reason why we use a Bell state as the ancilla.

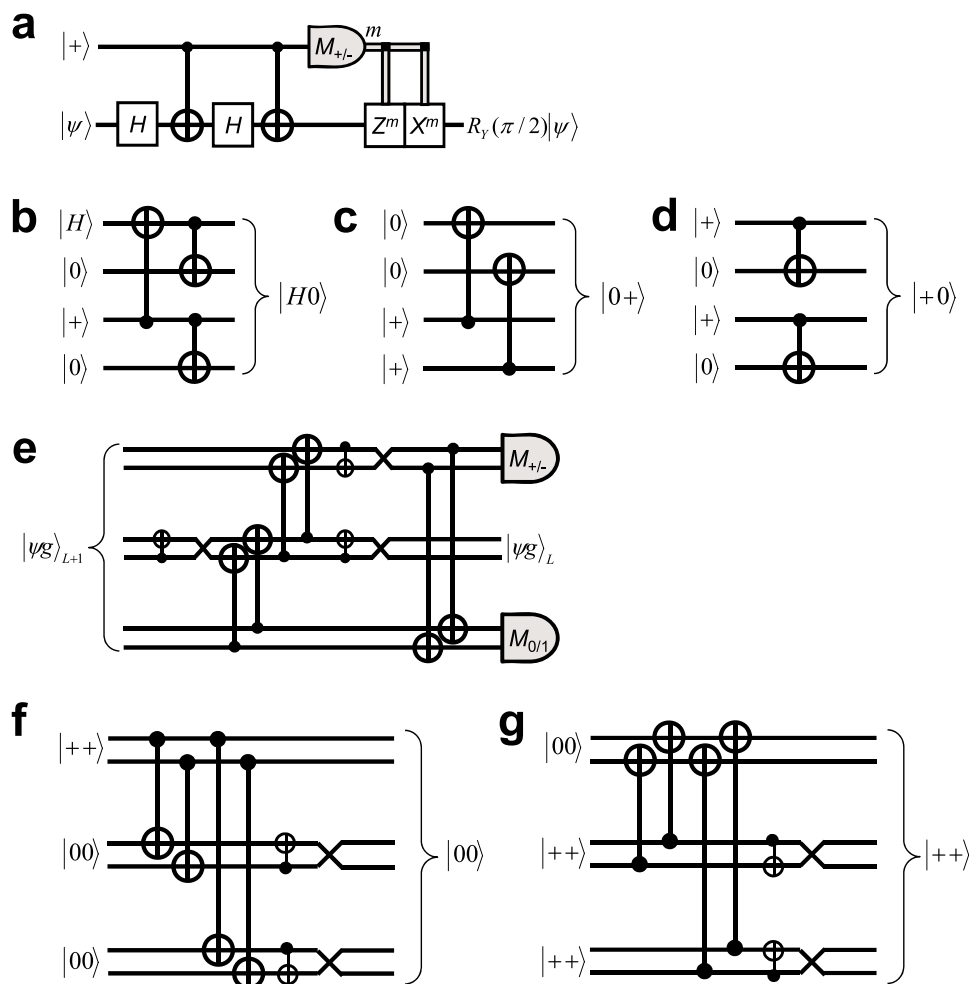
Third, encoded errors on the second set of the six EDTs, which are induced by the controlled-Hadamard gates, can be detected by the syndrome measurements of the  $H_6$  code (the measurements after the  $H_6$  decoder). (Here we can assume that the encoded errors due to the controlled-Hadamard gates are independent because their correlation comes only from the ancilla and the encoded errors on the ancilla can be detected and removed as explained above.)

Fourth and finally, encoded errors due to the encoded CNOT gates in the  $H_6$  decoder, which induce flips of the results of the measurements after the  $H_6$  decoder, can be detected because we know that the measurement results should be zero. (It is unnecessary to consider errors on the outputs because of the level-raising teleportation with full error detection.)

Thus, we obtain the desired result: the distillation with a Bell-state ancilla can achieve the reductions of the undetectable error probabilities shown in Fig. 1 even with imperfect encoded Clifford operations.

Finally, we describe Steps 3 and 5 in Fig. 1, the role of which is to reduce the uncorrectable error probabilities by raising the concatenation levels. Note that for this purpose, we cannot use direct encoders because such encoders convert uncorrectable errors for level  $L$  into uncorrectable ones for level  $(L+1)$ . To achieve the reductions of the uncorrectable error probabilities, we introduce the level-raising teleportation, which is depicted in Fig. 2d.

For the conversion of  $|\psi\rangle_L$  into  $|\psi\rangle_{L+1}$  (the subscripts represent the concatenation levels), we first prepare a level- $(L+1)$  Bell state. Since the Bell state is a stabilizer state, this can be prepared using Clifford operations alone<sup>1</sup>. Next, the first qubit of the Bell state is converted



**Figure 3** | Quantum circuits for the magic state emulation with the  $C_4/C_6$  code. (a) Clifford gate  $R_Y(\pi/2)$  with  $|+\rangle = R_Y(\pi/2)|0\rangle$ . (b) Encoder for  $C_4$ , which is used in Step 1 in Fig. 1. (c) and (d) are the simplified encoders for level-1  $|0\rangle$  and  $|+\rangle$ , respectively, which are used in Step 2 in Fig. 1. (e) Decoder for  $C_6$ , which is used for the level-raising teleportation (Steps 3 and 5 in Fig. 1). (f) and (g) are the simplified encoders for level-2  $|0\rangle$  and  $|+\rangle$ , respectively, which are used in Step 4 in Fig. 1.

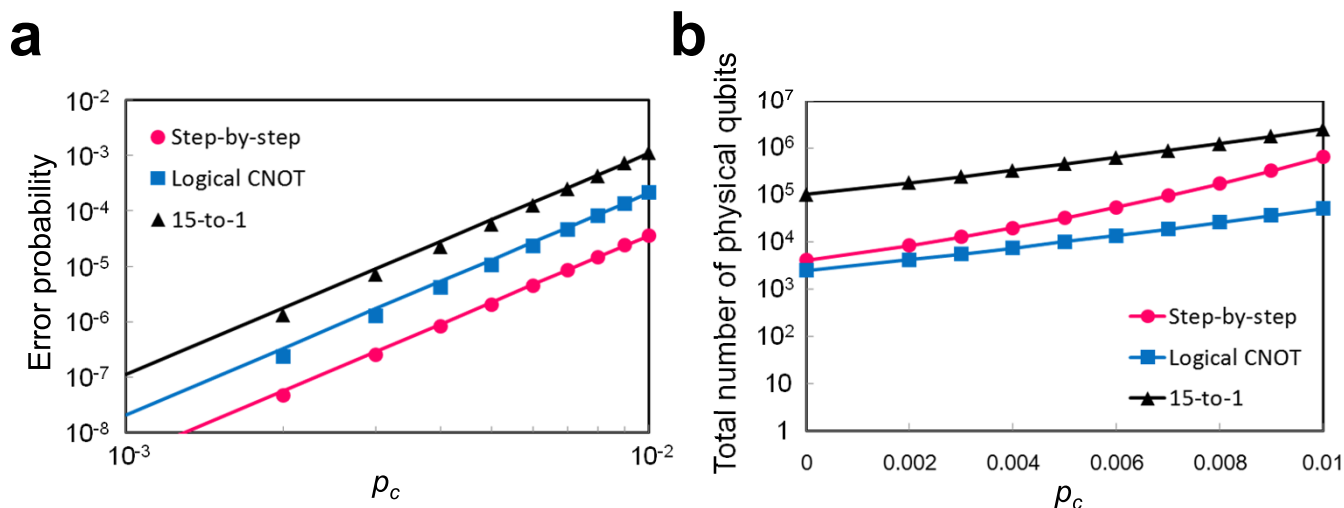
into the level- $L$  one, where this conversion can also be performed using Clifford operations alone<sup>1</sup>. (This conversion is the inversion of the encoding, and therefore also called “decoding.”) We obtain  $|\psi\rangle_{L+1}$  at the second qubit of the Bell state by quantum teleportation with the Bell state, as shown in Fig. 2d, where the CNOT gate and the qubit measurements for the Bell measurement are implemented by transversal physical ones. At the encoded qubit measurement, we perform full error detection. The undetectable error probabilities are  $O(p_c^2)$  and  $O(p_c^4)$  for  $L = 1$  and  $L = 2$ , respectively. Since the second qubit of the Bell state is a higher-level encoded qubit by one than the input qubit, its initial uncorrectable error probability is sufficiently low ( $O(p_c^2)$  and  $O(p_c^4)$  for level 2 and level 3, respectively). Thus, the desired result has been shown. We confirmed the validity of this discussion by numerical simulations. The simulation results also show that the increase of overheads due to the full error detection is not large for small  $p_c$ . See Supplementary Information.

**Magic state emulation.** To investigate the performance of the step-by-step magic state encoding more quantitatively, we performed numerical simulations with the level-3  $C_4/C_6$  code<sup>1,7</sup> (also see Methods and Supplementary Information). Since the exact simulation of magic state encoding requires full quantum simulation, large-scale simulation is computationally hard. To overcome the difficulty, here we use  $|+\rangle = R_Y(\pi/2)|0\rangle$  instead of  $|H\rangle = R_Y(\pi/4)|0\rangle$ . The stabilizer state  $|+\rangle$  can be used for a Clifford gate  $R_Y(\pi/2)$ , as

shown in Fig. 3a. The step-by-step encoding of  $|+\rangle$  can be performed in a similar manner to that of  $|H\rangle$  by replacing  $|H\rangle$  with  $|+\rangle$  and using  $X^m$  for  $R_Y(\pi/2)$  instead of  $H^m$  for  $R_Y(\pi/4)$  (compare Fig. 3a with Fig. 2a). The simulation of the step-by-step encoding of  $|+\rangle$  can be performed efficiently.

Although this *magic state emulation* with  $|+\rangle$  is not the exact simulation of the encoding of  $|H\rangle$ , this is useful as an approximation. First, as to the error probability, the above detailed discussion of the step-by-step encoding using Fig. 1 guarantees that the final uncorrectable error probability becomes  $O(p_c^4)$ . Therefore, the difference from the exact one is only in the proportionality factor for this. This factor will be estimated approximately by the magic state emulation. Second, as to the resource requirements, the resources for the encoding of  $|H\rangle$  can be evaluated exactly by the magic state emulation in the limit  $p_c \rightarrow 0$ . Thus, the difference from the exact one is only in the increasing factor for the overheads due to the error detection. This factor will also be estimated approximately by the magic state emulation. These approximations are expected to be good because the positions and number of physical CNOT gates, which are the only error sources in our error model, for the encoding of  $|+\rangle$  are exactly the same as those for  $|H\rangle$ .

The quantum circuits used for the simulation are shown in Figs 3b–3g. Figure 3b is the encoder of  $C_4$ , which is used at Step 1 in Fig. 1. Figures 3c and 3d are the encoders for  $|0\rangle$  and  $|+\rangle$ , respectively, at level 1, which are used at Step 2 in Fig. 1. Figure 3e is the



**Figure 4 | Results for the magic state emulation with the  $C_4/C_6$  code.** (a) Log-log plot of the simulation results for error probabilities. (b) Semilog plot of the simulation results for resource requirements (the total number of physical qubits). The circles (“Step-by-step”) are the results for a logical  $|+\rangle$  prepared by step-by-step encoding. The squares (“Logical CNOT”) are the results for a logical CNOT gate. The triangles (“15-to-1”) are the results for a logical  $|+\rangle$  prepared by the state injection with a logical Bell state<sup>1</sup> and the 15-to-1 distillation protocol<sup>5,9</sup>. The straight lines in (a) are obtained by fitting  $\alpha p_c^4$  to the plots, where  $\alpha$  is a fitting parameter. The lines in (b) are just eye guides.

decoder of  $C_6$ , which is used for the level-raising teleportation (Steps 3 and 5 in Fig. 1). Figures 3f and 3g are the encoders for  $|0\rangle$  and  $|+\rangle$ , respectively, at level 2, which are used at Step 4 in Fig. 1.

The simulation results are shown in Figs 4a and 4b. The error probability (4a) and the resource requirements (4b) for a logical  $|+\rangle$  prepared by step-by-step encoding (circles) are compared with those for a logical CNOT gate (squares) and those for a logical  $|+\rangle$  prepared by state injection and distillation at the logical level (triangles). The state injection used here is the one using a logical Bell state proposed in ref. 1 and the distillation protocol is the 15-to-1 one<sup>5,9</sup> (See Methods and Supplementary Information for the details). The resource requirements are evaluated by the total number of physical qubits including the effect of the repetition due to error detection.

## Discussion

Here we discuss the results shown in Figs. 4a and 4b.

First, from Fig. 4a, the error probability of  $|+\rangle$  prepared by step-by-step encoding is lower than not only that of  $|+\rangle$  prepared by state injection and distillation, but also that of a logical CNOT gate. This is remarkable because no approach based on state injection and distillation at the logical level can achieve the error probability less than that of a logical CNOT gate.

Next, all the plots in Fig. 4a are well fitted by  $\alpha p_c^4$ , where  $\alpha$  is a fitting parameter. This means that the error probabilities are  $O(p_c^4)$ , as expected.

Finally, we discuss the resource requirements shown in Fig. 4b. The resource required for the step-by-step encoding of  $|+\rangle$  is not only much less than that for  $|+\rangle$  prepared by state injection and distillation, but also comparable to that for a single logical CNOT gate. This is also remarkable because no approach based on state injection and distillation at the logical level can achieve the resource requirements comparable to a single logical CNOT gate.

All the resources increase as  $p_c$  becomes larger. This increase is caused by the repetition due to error detection. Although the increase for the step-by-step encoding is faster than the other two, this difference is almost negligible as long as  $p_c$  is sufficiently small, e.g., smaller than 0.002.

This increasing factor will, of course, become larger rapidly as the code size becomes larger. Therefore, the present method can be applied only to concatenated codes of moderate size, such as the level-3  $C_4/C_6$  code. Nevertheless, this will be useful for first applica-

tions of quantum computers to some problems of small size. (This “small size” is not very small in the sense that from Fig. 4a,  $10^8$  logical gates can be used when  $p_c$  is as small as 0.001.)

In summary, we have proposed step-by-step magic state encoding for concatenated CSS codes. We have used error detection carefully and have introduced two new techniques: magic state distillation with a Bell-state ancilla and level-raising teleportation. The error probability of a logical magic state is sufficiently lowered only with ancillas of lower concatenation levels. Thus, the resource requirements will be substantially reduced. We have performed numerical simulations with a stabilizer state instead of a magic state, which we call magic state emulation. The simulation results strongly support our conclusion. The present results suggest that it may be possible to perform reliable non-Clifford gates with resources comparable to Clifford gates in the case of concatenated codes of moderate size. Thus, the step-by-step magic state encoding opens a new possibility for efficient fault-tolerant quantum computation.

## Methods

**Modified  $C_4/C_6$  code.** The  $C_4/C_6$  code proposed in ref. 1 is a CSS code concatenating two distance-2 codes,  $C_4$  and  $C_6$  (also see ref. 7). In the present work, we modified the definition of  $C_6$  such that encoded Hadamard and phase gates<sup>4</sup> can be implemented transversally. Thus, all the Clifford gates can be implemented transversally for the modified  $C_6$ , like the Steane code<sup>4</sup>. The stabilizer generators for this  $C_6$  are the same as the original ones, but the encoded Pauli operators are redefined by XIXIXI and ZIZIZI for the first qubit and by IXIXIX and IZIZIZ for the second qubit. See Supplementary Information for the detailed difference between the original and modified ones. (This modification is not for improvement of performance but for simplification of code properties.)

**Simulation methods.** The stabilizer simulator and the error-vector simulator used in this work are the same as ref. 7. Since all the simulations are Monte-Carlo ones, the quantities to be estimated are obtained by averaging the results of many trials. In the following, each trial in the simulations for Figs. 4a and 4b is briefly described. (See Supplementary Information for the detailed description.)

The procedure for the step-by-step encoding of  $|+\rangle$  is almost the same as that for  $|H\rangle$ , which has been described in detail. To check whether logical errors have occurred on the output or not, the output is finally measured, where optimal decoding for error correction<sup>7</sup> is used.

In the simulation for the logical CNOT gate, the gate is performed repeatedly ten times on the first logical qubits of two error-free logical Bell states. (The reason for the ten times repetition is to estimate the error probability during computation. See ref. 1.) After that, the Bell states are disentangled by error-free operations. Finally, the logical qubits are measured with optimal decoding for error correction<sup>7</sup> to check whether logical errors have occurred or not. See Supplementary Information for the details on the simulation and the implementation of the logical CNOT gate.



In the simulation for  $|+\rangle$  prepared by state injection and distillation at the logical level, we first estimated the error probability and the resource requirements for a noisy logical  $|+\rangle$  prepared by the state injection with a logical Bell state<sup>1</sup> (see Supplementary Information for details). Using this result together with that of the logical CNOT gate, we simulated the situation where 15 physical qubits in  $|+\rangle$  with the same error probability as the noisy logical  $|+\rangle$  are distilled by the 15-to-1 distillation protocol<sup>5,9</sup> using 34 physical CNOT gates with the same error probability as the logical CNOT gate. The results shown in Figs. 4a and 4b are estimated from this simulation. See Supplementary Information for the details.

- Knill, E. Quantum computing with realistically noisy devices. *Nature* **434**, 39–44 (2005).
- Raussendorf, R. & Harrington, J. Fault-tolerant quantum computation with high threshold in two dimensions. *Phys. Rev. Lett.* **98**, 190504 (2007).
- Wang, D. S., Fowler, A. G. & Hollenberg, L. C. L. Surface code quantum computing with error rates over 1%. *Phys. Rev. A* **83**, 020302(R) (2011).
- Nielsen, M. A. & Chuang, I. L. *Quantum Computation and Quantum Information* (Cambridge Univ. Press, Cambridge, 2000).
- Jones, N. C. *et al.* Layered architecture for quantum computing. *Phys. Rev. X* **2**, 031007 (2012).
- Fowler, A. G., Mariantoni, M., Martinis, J. M. & Cleland, A. N. Surface codes: towards practical large-scale quantum computation. *Phys. Rev. A* **86**, 032324 (2012).
- Goto, H. & Uchikawa, H. Fault-tolerant quantum computation with a soft-decision decoder for error correction and detection by teleportation. *Sci. Rep.* **3**, 2044 (2013).
- Shor, P. W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput.* **26**, 1484–1509 (1997).
- Bravyi, S. & Kitaev, A. Universal quantum computation with ideal Clifford gates and noisy ancillas. *Phys. Rev. A* **71**, 022316 (2005).
- Meier, A. M., Eastin, B. & Knill, E. Magic-state distillation with the four-qubit code. *arXiv*, 1204.4221 (2012).
- Bravyi, S. & Haah, J. Magic-state distillation with low overhead. *Phys. Rev. A* **86**, 052329 (2012).
- Jones, C. Multilevel distillation of magic states for quantum computing. *Phys. Rev. A* **87**, 042305 (2013).
- Aaronson, S. & Gottesman, D. Improved simulation of stabilizer circuits. *Phys. Rev. A* **70**, 052328 (2004).
- Reichardt, B. W. Quantum universality from magic states distillation applied to CSS codes. *Quant. Info. Proc.* **4**, 251–264 (2005).
- Fowler, A. G., Devitt, S. J. & Jones, C. Surface code implementation of block code state distillation. *Sci. Rep.* **3**, 1939 (2013).
- Paetznick, A. & Reichardt, B. W. Universal fault-tolerant quantum computation with only transversal gates and error correction. *Phys. Rev. Lett.* **111**, 090505 (2013).
- Jochym-O'Connor, T. & Laflamme, R. Using concatenated quantum codes for universal fault-tolerant quantum gates. *Phys. Rev. Lett.* **112**, 010505 (2014).
- Poulin, D. Optimal and efficient decoding of concatenated quantum block codes. *Phys. Rev. A* **74**, 052333 (2006).
- Goto, H. & Uchikawa, H. Soft-decision decoder for quantum erasure and probabilistic-gate error models. *Phys. Rev. A* **89**, 022322 (2014).

## Acknowledgments

H.G. thanks Dr. K. Ichimura for his useful comment on the manuscript.

## Author contributions

H.G. proposed and devised all the present methods and performed all the simulations.

## Additional information

Supplementary information accompanies this paper at <http://www.nature.com/scientificreports>

**Competing financial interests:** The authors declare no competing financial interests.

**How to cite this article:** Goto, H. Step-by-step magic state encoding for efficient fault-tolerant quantum computation. *Sci. Rep.* **4**, 7501; DOI:10.1038/srep07501 (2014).



This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivs 4.0 International License. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in the credit line; if the material is not included under the Creative Commons license, users will need to obtain permission from the license holder in order to reproduce the material. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>