Performing quantum computing experiments in the cloud

Simon J. Devitt
Center for Emergent Matter Science, RIKEN, Wakoshi, Saitama 315-0198, Japan

(Received 5 August 2016; published 29 September 2016)

Quantum computing technology has reached a second renaissance in the past five years. Increased interest from both the private and public sector combined with extraordinary theoretical and experimental progress has solidified this technology as a major advancement in the 21st century. As anticipated my many, some of the first realizations of quantum computing technology has occured over the cloud, with users logging onto dedicated hardware over the classical internet. Recently, IBM has released the Quantum Experience, which allows users to access a five-qubit quantum processor. In this paper we take advantage of this online availability of actual quantum hardware and present four quantum information experiments. We utilize the IBM chip to realize protocols in quantum error correction, quantum arithmetic, quantum graph theory, and fault-tolerant quantum computation by accessing the device remotely through the cloud. While the results are subject to significant noise, the correct results are returned from the chip. This demonstrates the power of experimental groups opening up their technology to a wider audience and will hopefully allow for the next stage of development in quantum information technology.

DOI: 10.1103/PhysRevA.94.032329

I. INTRODUCTION

The accelerated progress of quantum information technology in the past five years has resulted in a substantial increase in investment and development of active quantum technology. As expected, access to the first prototypes for small quantum computers has occurred over the cloud, with hardware groups opening up access to their hardware through the classical Internet. The first case was the Center for Quantum Photonics (CQP) at the university of Bristol that connected a two-qubit optical chip to the Internet and invited people to design and test their own experiments [1]. This was a remarkable achievement at the time but suffered from the disadvantage that there were very few experiments in quantum computing that could occur with their cloud-based hardware.

Recently, IBM did the same thing and allowed access to a five-qubit superconducting chip to the Internet community through an interactive platform called the Quantum Experience (QE) [2]. This approach is substantially more advanced as it allows access to a five-qubit, reprogrammable device and allowed for circuit design, simulation, testing, and actual execution of an algorithm on a physical device. The size of the IBM chip now allows for demonstration of quantum protocols out of the reach of people not associated with advanced experimental groups. It has already been used by researchers to violate a more general version of Bell's inequalities [3].

In this paper we present four separate quantum protocol experiments, designed and executed independently of the IBM team. We treat the QE website, interface, and chip as essentially a black box and run experiments related to four main areas of quantum information: error correction [4], quantum arithmetic [5], quantum graph theory [6], and fault-tolerant circuit design [7]. We detail the motivation and design of each experiment, restrict our analysis to the simple output coming from the IBM chip, and show that each individual protocol produces valid answers (at low fidelity). This work will hopefully motivate more people to get involved in cloud-based quantum interfaces and encourage experimental groups to open up their hardware for interaction with the general public to increase innovation and development of a quantum technology sector. Treating the interface as a black box, we designed, simulated, and implemented protocols over an array of five qubits. The primary results and explanations are provided in the main body of the paper, with raw data available in the Supplemental Material [8]. For each experiment we illustrate simulations provided by the QE interface and the resultant experimental data from the five-qubit chip. In each case we observe results consistent with the theory and the intended output of each protocol.

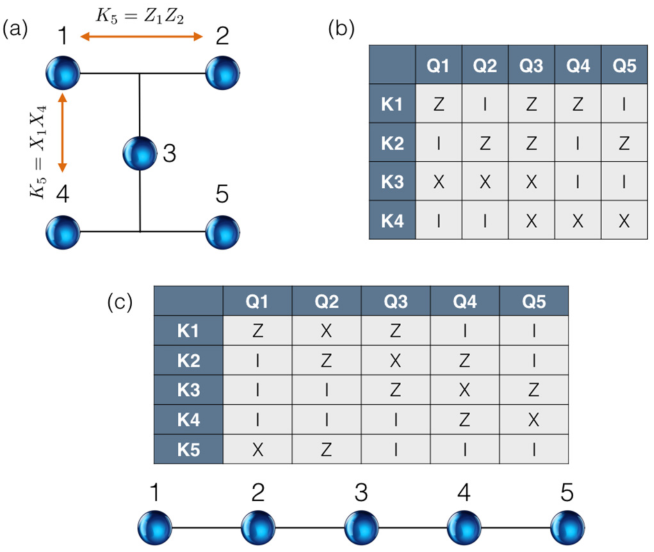
II. RESULTS

A. Error-corrected Rabi oscillations

The first experiment implemented on the IBM chip is basic Rabi oscillation spectra across a *logically* encoded qubit using a distance-two surface code. Surface-code quantum computing [4] is now the standard model used for large-scale quantum computing development [9–19], and results from superconducting and other technologies show extraordinary promise [20–24]. While the distance-three, five-qubit code [25] could be used with the IBM QE, we focus on the surface code due to its relevance to larger quantum architectures. The five-qubit surface code has already been investigated by the IBM team themselves [21], but in this experiment, the goal is not to artificially inject errors into the system, but rather to conduct a standard protocol used in the initial demonstration of a two-level controllable quantum system and see if an error-corrected version of the protocol shows some advantage.

A distance-two surface code consists of an array of five qubits, illustrated in Fig. 1(a) stabilized by the four operators in Fig. 1(b), with a fifth stabilizer $K_5 = Z_1Z_2$ ($K_5 = X_1X_4$) used to specify the logical Z (X) state of the encoded qubit. The Z stabilizers ($\{K_1,K_2\}$) are used to detect bit-flip errors, while the X stabilizers (K_3,K_4) are used to correct phase errors. Being distance two, this code can only detect errors (there is insufficient information to determine the location of any detected bit or phase error with certainty).

Initializing a distance-two surface code in the $|0\rangle_L$ state is equivalent to preparing a five-qubit linear cluster. The five-



(b)

	Q1	Q2	Q3	Q4	Q5
K1	Z	I	Z	Z	I
K2	I	Z	Z	I	Z
K3	X	X	X	I	I
K4	I	I	X	X	X

(c)

	Q1	Q2	Q3	Q4	Q5
K1	Z	X	Z	I	I
K2	I	Z	X	Z	I
K3	I	I	Z	X	Z
K4	I	I	I	Z	X
K5	X	Z	I	I	I

FIG. 1. Distance-two surface code. (a) illustrates the code structure, with two plaquette and two vertex stabilizers. Logical operators run left to right (Z_L) or top to bottom (X_L). (b) is the stabilizer set for the distance-two surface code. (c) is the structure and stabilizer set for a linear cluster state. Once a Hadamard is performed on qubits 1, 3, and 5, the state is equivalent to a distance-two surface code initialized in the $|0\rangle_L$ state.

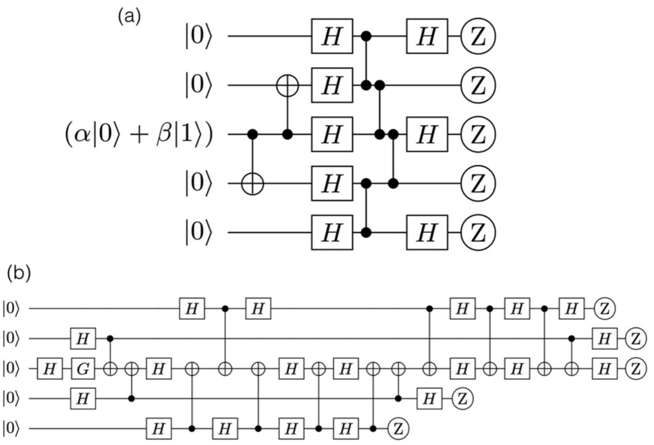


FIG. 2. Circuits to encode arbitrary states into the surface code. (a) is the general circuit, where the state $\alpha|++ +\rangle_{1,3,5}+\beta|---\rangle_{1,3,5}$ (where 1, 3, 5 are the central three qubits in the QE interface at input) is first prepared and then a linear cluster state is created afterwards. Following the last three Hadamard gates, the logic state $|0\rangle_L + |1\rangle_L$ is prepared. (b) is the circuit implementation in the QE interface. Only the third qubit in the IBM chip can be coupled to via a CNOT (and it always acts as the target qubit); hence various SWAP gates (decomposed into CNOT and Hadamard gates [26] are used). The gate G is the set of successive T rotations used to sequentially rotate the qubit into the state $e^{in(\pi/8)X_L}|0\rangle$ for $n \in \{0,\ldots,7\}$. After the state is prepared, it is immediately measured in the Z basis. After the circuit is run, qubit ordering is (4, 1, 2, 5, 3).

qubit linear cluster has the stabilizer set illustrated in Fig. 1(c). After a Hadamard is applied to qubits 1, 3, and 5 we can identify K_5' as the +1 eigenstate of the logical Z operator, and after multiplying stabilizers $K_2'K_5' = K_1$ and $K_2'K_4' = K_2$, we regain the stabilizer set for the distance-two surface code.

To perform an error correction version of a Rabi oscillation experiment, we instead require the ability to encode a rotated logical state of the form $e^{i\theta X_L}|0_L\rangle$, which cannot be directly prepared in the encoded space. Therefore we prepare a single qubit in the rotated state and encode. This is not a fault-tolerant operation, and the fidelity of this initial rotated state will bound the error on the final state (even if every other gate in the circuit were perfect). The direct encoding is shown in Fig. 2(a), where we prepare the state $\alpha|++ +\rangle_{1,3,5}+\beta|---\rangle_{1,3,5}$ and then create the linear cluster state from this initial state. This places the surface code into the state $\alpha|0\rangle_L + \beta|1\rangle_L$.

As with a single-qubit Rabi experiment, we measure the probability of measuring a $|0\rangle_L$ state after rotating the input with successively larger angles θ. In the QE interface, only Clifford $+T$ gates are available, and constructing rotations smaller than $R_x(\pi/4)$ would require decompositions into Clifford $+T$ sets [27,28]. But for this experiment we instead use successive $\pi/4$ rotations (T gates) around the Bloch sphere where an eight-gate cycle will oscillate the state from $|0\rangle_L \to |1\rangle_L$ and back to $|0\rangle_L$.

As the QE chip only allows for controlled-NOT (CNOT) gates to operate with the central qubit in their star geometry, additional SWAP operations are needed, which are also decomposed into CNOT and Hadamard gates. The full circuit implemented is illustrated in Fig. 2(b), where the gate G is each successive step in the rotation, $G \in \{T^0,\ldots,T^8\}$. The input ordering of qubits in the QE interface is (2, 1, 3, 5, 4), and at output the ordering is (4, 1, 2, 5, 3).

Once the state is prepared, we can measure in the Z basis on each of the five qubits and calculate the parity of K_5 to determine the logical state. Since, when measuring in the Z basis, results are modified by the presence of bit-flip errors, we can use the parity of the Z stabilizers to postselect from the 8192 runs allowed by the QE interface only instances where K_1 and K_2 return even-parity results. The raw data are included in the Supplemental Material.

Once the state is prepared, we first simulate the expected output using the master-equation solver included with the QE interface, and the simulation results are illustrated in Fig. 3(a). This plot has three curves. The first is the theoretical optimal, where the Rabi oscillations follow a $\cos^2(\theta)$ function. The second curve, with the lowest visibility, is when we run the circuit in Fig. 2 but *do not* postselect on trivial syndrome results for K_1 and K_2. The visibility in this case is clearly lower than the ideal case due to errors accumulating in the larger circuit. The third curve, which sits between the two, is where postselection occurs and we discard any results where a nontrivial syndrome is detected. In simulations, this gives a clearly better performance than the non-error-corrected case but is still far from ideal.

Figure 3(b) illustrates the actual experimental data returned from the QE interface. Each data point (representing successive rotations of the input state between $|0\rangle_L$ and $|0\rangle_L$; the rotation angle is 2θ, such that a rotation from $|0\rangle_L \to |0\rangle\rangle$ occurs for $\theta = \pi$) was taken using the maximum sampling instances of the QE chip (8192). For the purple line in Fig. 3(b),

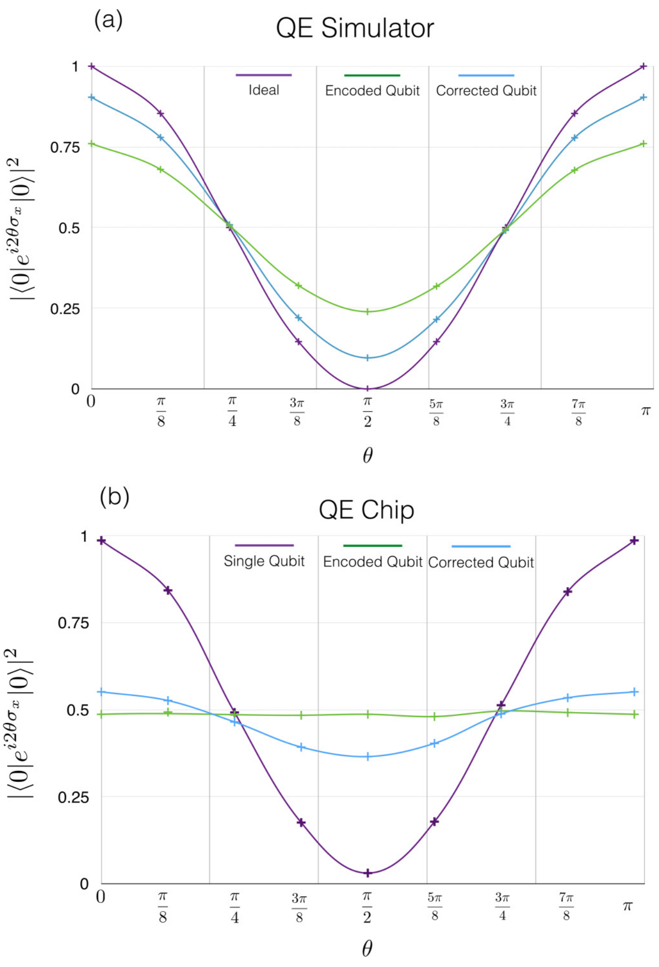


FIG. 3. Simulation and experimental results from the QE device. In (a) we have simulated plots for the ideal Rabi curve (purple), the encoded but *uncorrected* code (green), and the postselected, error-corrected results (blue). An improvement when correction is applied is clearly seen. In (b) we have the results directly from the QE chip. For the *corrected* state, a low-visibility oscillation is observed. The purple curve now represents successive rotations on an *individual* qubit and so has high visibility (but still is not perfect). Each data point was obtained using the maximum sampling in the QE interface of 8192 shots, and a sampling error for each point (omitted) can be calculated as $\sqrt{P_c(1-P_c)/8192}$, where P_c is the probability of each measurement.

we performed a Rabi experiment using a bare single qubit (via repeated rotations by T and then immediate measurement); this showed, unsurprisingly, the highest visibility. The data when we encoded into the surface code and did not perform the corrective operations are essentially random (showing a 50% probability of measuring $|0\rangle_L$ or $|1\rangle_L$). Once the data are postselected on trivial syndrome results, we again see a nonzero visibility, indicating that the error correction properties of the code are helping to restore the oscillation. The standard error (SE) can be calculated as $\sqrt{P_c(1-P_c)/8192}$, where P_c is the probability of a given measurement result. The SE is not illustrated in Fig. 3.

The complexity of the circuit to perform the encoding and the fact that the QE chip does not have sufficient control or accuracy to perform a fault-tolerant experiment mean that the error-corrected version *does not outperform* a non-error-corrected version of the same experiment. But active decoding of the information does result in better results when compared to the same circuit without active decoding. Additionally, the simulator bundled with the QE interface overestimates the performance of the chip, compared with the actual experimental data.

B. Fourier addition

Our second experiment is programmable quantum arithmetic [5]. We design a circuit that performs addition between two quantum registers, $|a\rangle$ and $|b\rangle$, by performing a Fourier transform on register $|a\rangle \otimes |b\rangle \rightarrow |a\rangle \otimes \psi(|b\rangle)$, controlled phase rotations taking $|a\rangle \otimes \psi(|b\rangle) \rightarrow |a\rangle \otimes \psi(|a\rangle + |b\rangle)$, and an inverse Fourier transform, giving $|a\rangle \otimes \psi(|a+b\rangle) \rightarrow |a\rangle \otimes |a+b\rangle$. The addition naturally occurs modulo 2 as the registers $|a\rangle$ and $|b\rangle$ are only two-qubit registers.

Even though the QE system has the ability to operate over five qubits, a quantum Fourier transform (QFT) over any more than two qubits requires $\sqrt{T}$ gates, which would need to be approximated via Clifford $+T$ sequences, for which there are insufficient resources. Two-qubit Fourier addition is possible as the smallest rotation needed for circuit decompositions is T gates, available in the QE interface.

In Fig. 4(a) we illustrate the addition circuit, the decomposition circuits used for controlled phase rotations [Fig. 4(b)], and the actual circuit implemented in the QE interface [Fig. 4(c)]. Once again, all CNOT gates need to involve the central qubit in the architecture (Q_3) as the target, in accordance with the geometry of the QE hardware. The gates $G_i, i \in \{1,\ldots,4\}$, are used to prepare the input states from the initial $|0\rangle$ states in the QE chip, and at the end of the circuit everything is measured in the Z basis.

As with the error-corrected Rabi oscillation, the final circuit is quite large, and hence errors will accumulate, and the probability of observing successful output will drop. We again ran these circuits in both the QE simulator and the actual QE chip for comparison. Figures 5 and 6 show the experimental and simulation results, respectively, for a selection of the 16 possible binary inputs of $|a\rangle$ and $|b\rangle$, along with various superposition and entangled inputs. The complete data set is contained in the Supplemental Material. The simulations and experimental results were averaged over 8192 runs, and the SE is omitted from the plots. The simulations and experiments show similar levels of noise in their outputs. The difference between the two is not as clear as the error-corrected Rabi oscillations shown earlier.

The size of the addition circuit is larger than the error-corrected Rabi experiment; hence noise in the QE chip is stronger. However, again, we see the *right* answer returned a plurality of the time, from the 8192 runs performed. The only exception is when we used an entangled input on the target register. Figure 5 (bottom right) illustrates when the transformation should be $(|0100\rangle + |0111\rangle)/\sqrt{2} \rightarrow (|0101\rangle + |0100\rangle)/\sqrt{2}$. In this case, the correct answer could not be inferred from the experimental output.

We expect that experimental refinement of the chip itself will lead to less noise accumulation such that desired results

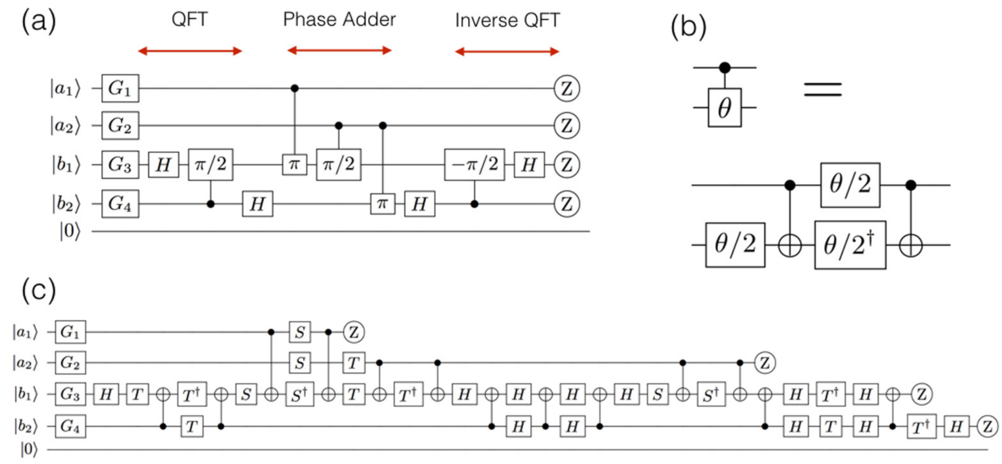


FIG. 4. Fourier addition circuits. In (a) we illustrate the Fourier addition circuit on two registers. First, a QFT is performed on the target register, after which controlled phase rotations are used to perform the addition. An inverse QFT then returns the target register to the computational basis. The gates $G_i, i \in \{1,\ldots,4\}$, are used to program in various inputs used in the simulations. (b) shows controlled rotation decompositions. For rotations smaller than $\pi/2$, the decomposition requires gates of the form $^n\sqrt{T}$, which are not available in the QE interface without performing Clifford $+T$ decompositions [27,28]. In (c) we illustrate the circuit implementation for the QE chip. Note that in (c) the target register $|b_1\rangle|b_2\rangle$ is flipped on output.

are returned with much higher probability. As experimental error rates go down, these experiments could be useful as a benchmark to ascertain if performance with large circuits increases over time.

C. Local complementarity in quantum graph states

The third experiment performed with the QE platform is a unique property of quantum graph states called local complementarity [6]. Local complimentary is where the structure of a quantum graph can be changed via *only* local operations. We demonstrate the creation of a five-qubit graph state (by measuring appropriate graph stabilizers and confirming measuring the even parity of each operator); we then permute this graph using local operations and confirm, through the measurement of the new stabilizers, that the underlying state has changed. A minimum of four qubits is required to perform any type of graph complementarity experiment (as all graphs containing two or three qubits have only a single local-equivalence class [6]), and five qubits give a large number of nonequivalent classes that we could test. The first example is creating a Greenberger-Horne-Zeilinger state (star graph) and, through local complementarity, perturb this to a completely connected graph and then perturb it back to another star graph with a different qubit acting as the star node. The second experiment is running several *orbital* steps on a second input graph, showing the creation of each new, locally equivalent graph.

A quantum graph state is easily defined from the classical graph that describes it. A classical (undirected, finite) graph, $G = (V,E)$, is a set of N nodes, $V \in \{1,\ldots,N\}$, connected by sets of edges, $E \subset V^2$, such that $E_{i,j} = 1$ for any two connected nodes $V_{i,j}$ and $E_{i,j} = 0$ if $V_{i,j}$ are not connected. To convert this classical graph state to a quantum graph state, we use the adjacency matrix of the classical graph to form a set of stabilizer operators that are used to specify the quantum graph. The conversion of a star graph to the stabilizer set is illustrated in Fig. 7.

Once a graph is written, an adjacency matrix can be formed that contains a 1 for each edge, $E_{i,j} = 1$, and 0 everywhere else. Converting an adjacency matrix into a list of stabilizers for the quantum graph state requires taking each row of the adjacency matrix and replacing the 0 on the diagonal with the Pauli X operator and any 1 entry with the Pauli Z operator and using the identity matrix everywhere else. For an N-qubit graph, there will be N operators corresponding to the N rows of the classical adjacency matrix. Creating a quantum state simultaneously stabilized by these N operators will produce the required quantum state.

Local complementarity operators are quite simple and consist of choosing one of the qubits of the quantum graph, denoted the *node*, and applying a $\sqrt{X} \equiv HSH$ operator to that qubit and a $\sqrt{Z} \equiv S$ operator to any qubit it is connected to (*leaf*). This has the effect of creating an edge directly between the leaves or deleting an edge if it originally existed. These local complementarity rules create what is known as an *orbital* and forms a closed set of locally equivalent quantum graphs. Different orbital groups are not locally equivalent to each other, and the number of different orbital groups grows as the total number of qubits increases. Our first experiment is illustrated graphically in Fig. 8, along with its required quantum circuit.

We first create a five-qubit star graph that is stabilized by the operators in Fig. 7, and then we either measure the stabilizer of the star graph through the local gates $G_i, i \in \{1,\ldots,5\}$, or first convert the graph and then measure the parity of the stabilizers associated with the new graph through local complementarity. The experimental results from the QE chip are illustrated in Figs. 9–11, and the simulated results are illustrated in Figs. 12–14. Again, each plot was generated using 8192 experimental runs, and the SE is omitted from each plot.

As expected, the calculated parities for each stabilizer match what is measured. Some parities flip during the graph

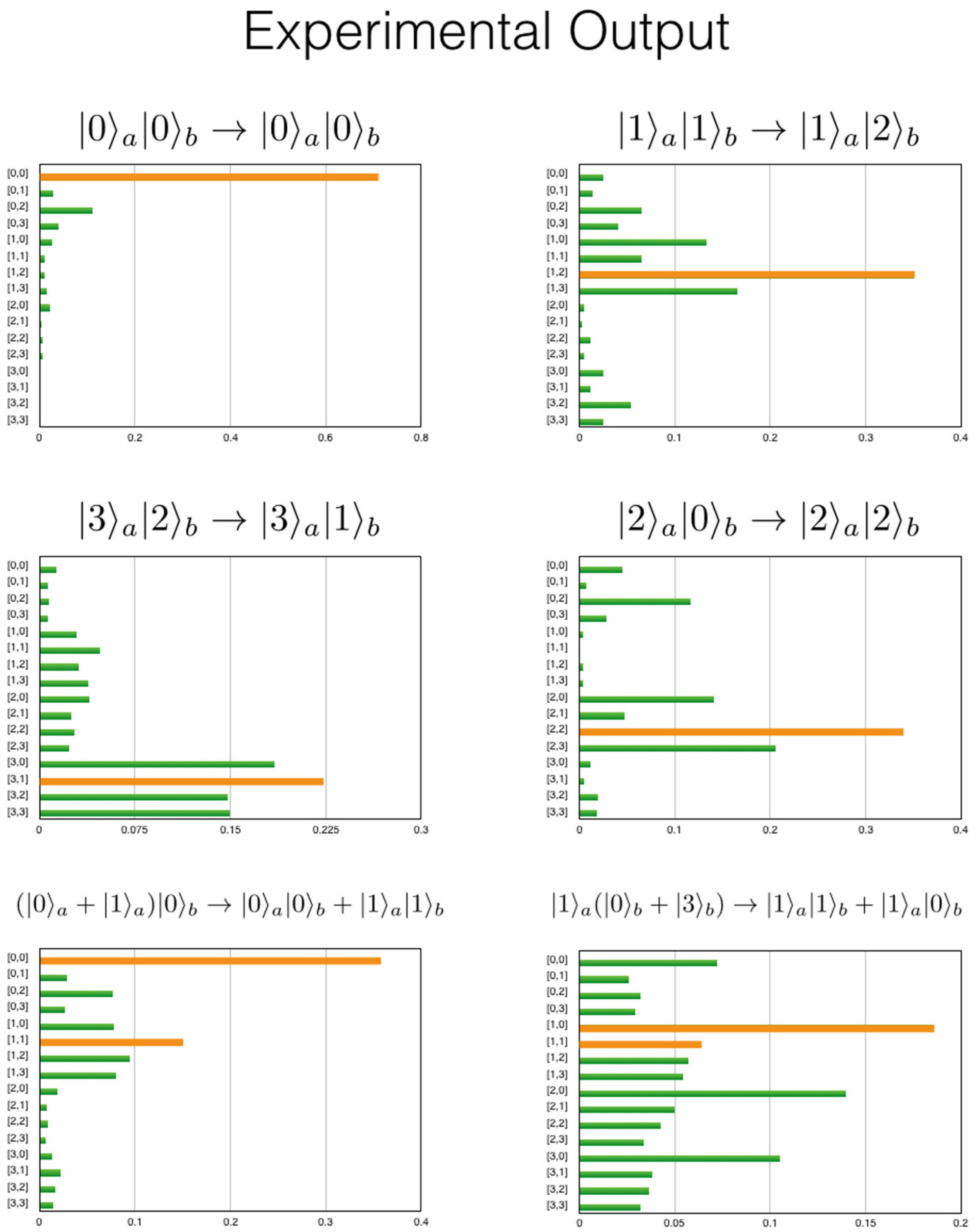


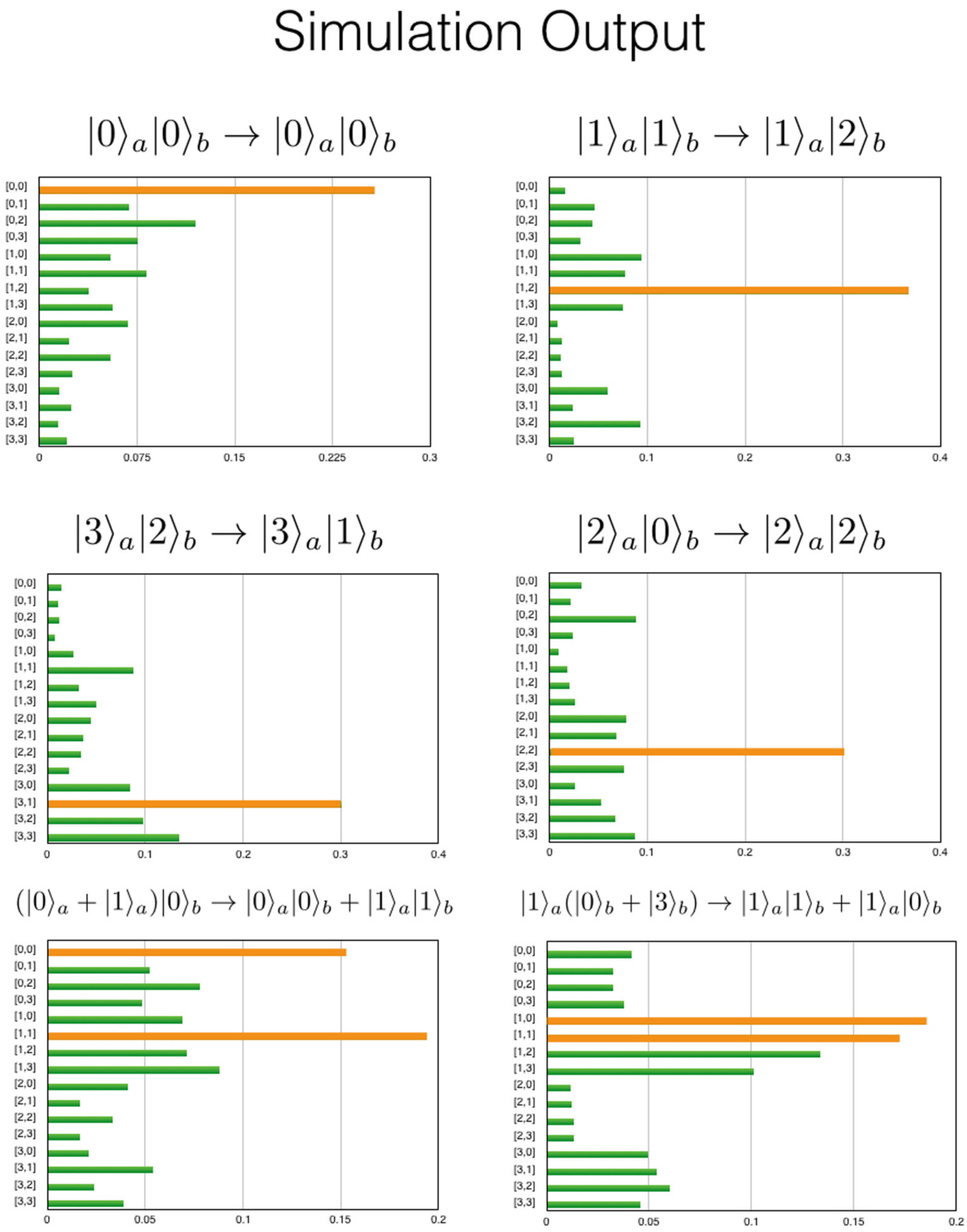
FIG. 5. Experimental Fourier addition. Each plot shows the 32 possible binary outputs. The input and output mapping for each plot is specified, and the *correct* binary outputs are illustrated in orange on each plot. For all experiments (except where a Bell state is created on the target register), the correct output is observed a plurality of the time.

complementarity operations, but these flips could be reversed by applying appropriate Pauli operations. As the circuit is quite small compared to other experiments, the degree to which we see nearly 100% probability of the correct parity is high. This experiment showed how to use local complementarity to permute the structure of a quantum graph. In this case we went from a star graph with qubit 3 acting as the node to a completely connected graph and back to a star graph with now qubit 1 acting as the node. Another example of five-qubit complimentary is shown in the Supplemental Material.

This type of experiment for all nonequivalent five-qubit graphs is fairly simple to perform in the QE interface. Preparing any five-qubit graph state does not require as many gates as either error-corrected Rabi oscillations or Fourier addition, and hence the QE chip should produce the correct state (with reasonably high fidelity). Investigating the structure of these locally equivalent graphs and how they can be utilized as communications links has the potential to open up an area of quantum network analysis that is not available in the classical world.

D. Deterministic T gates

Our final experiment is comparatively quite simple but highly relevant to the construction of Fault-tolerant quantum circuits [7]. When designing high-level quantum circuits, the universal gate library of choice is the Clifford $+T$ set. There has been significant work in both compiling and optimizing


Simulation Output

$|0\rangle_a|0\rangle_b \rightarrow |0\rangle_a|0\rangle_b$

$|1\rangle_a|1\rangle_b \rightarrow |1\rangle_a|2\rangle_b$

$|3\rangle_a|2\rangle_b \rightarrow |3\rangle_a|1\rangle_b$

$|2\rangle_a|0\rangle_b \rightarrow |2\rangle_a|2\rangle_b$

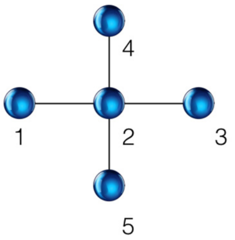
$(|0\rangle_a + |1\rangle_a)|0\rangle_b \rightarrow |0\rangle_a|0\rangle_b + |1\rangle_a|1\rangle_b$

$|1\rangle_a(|0\rangle_b + |3\rangle_b) \rightarrow |1\rangle_a|1\rangle_b + |1\rangle_a|0\rangle_b$

FIG. 6. Simulated Fourier addition. We use the QE interface to first simulate the addition circuit. Results are closer to experimental output than the error-corrected Rabi experiment and in some cases show more noise than what was output from the chip itself.

circuits of this form [29–34]. In fault-tolerant models, T gates are generally applied using teleportation circuits with an appropriate ancillary state that is prepared using complicated protocols such as state distillation [35]. These circuits are intrinsically probabilistic, and for the T gate, corrections need to involve the active application of a subsequent S gate. Hence there was the potential that high-level circuit construction would need to be dynamic, adjusting itself every time a T gate is applied in the hardware. However, recent work has shown how to construct *deterministic circuitry* for any high-level algorithm [7,36]. Subcircuits known as selective source and selective destination teleportation [36] are used to patch in correction operations for each T gate via choosing to measure certain qubits in either the X or Z basis. This deterministic T gate is illustrated in Fig. 15.

The operations on the first two qubits are the teleported T gate, which utilizes a magic-state ancilla, $|A\rangle = |0\rangle + e^{i\pi/8}|1\rangle$, a subsequent CNOT, and Z-basis measurement to enact the gate. The logical result of the Z measurement determines if a T gate or a $T^\dagger$ gate is applied (the T gate occurs when a $|0\rangle$ result is measured). Depending on this measurement result, a possible S-gate correction needs to be applied, which is a second teleportation circuit utilizing a second magic-state ancilla, $|Y\rangle = |0\rangle + e^{i\pi/4}|1\rangle$. The S-gate correction can also require a correction, but this correction is a Pauli Z gate and hence does not need to be actively applied.



	V1	V2	V3	V4	V5
	0	1	0	0	0
	1	0	1	1	1
	0	1	0	0	0
	0	1	0	0	0
	0	1	0	0	0

	Q1	Q2	Q3	Q4	Q5
K1	X	Z	0	0	0
K2	Z	X	Z	Z	Z
K3	0	Z	X	0	0
K4	0	Z	0	X	0
K5	0	Z	0	0	X

FIG. 7. The link between a classical graph and the corresponding quantum graph. The classical star graph can be represented as an adjacency matrix, with ones present to indicate the four bonds from the central node to each leaf. To convert this to the equivalent quantum graph state, we construct a stabilizer table by replacing each 1 with a Z operator, X operators on the diagonal, and I everywhere else. Each row of the adjacency matrix is now one of the five stabilizers of the quantum star graph.

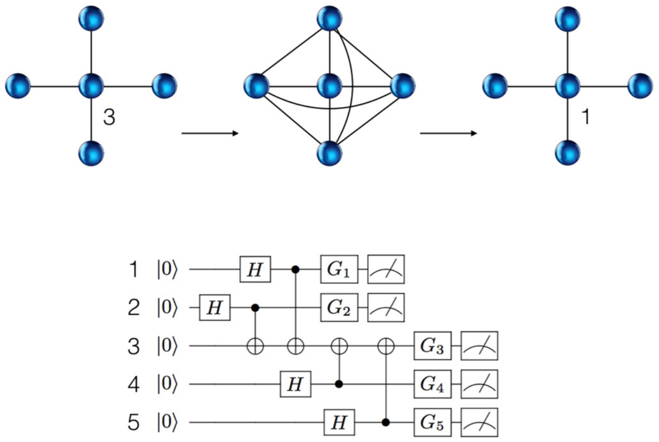


FIG. 8. Circuits to implement graph complementarity and stabilizer measurements. In the experiment we initially prepare a five-qubit star graph, use local complementarity to convert it to a completely connected graph, and then permute again to reconvert to a star graph with a different qubit acting as the central node. After each conversion we measure the five associated stabilizers. Illustrated is the quantum circuit necessary to do this in the QE interface. The initial part of the circuit creates the star graph. The gates $G_i, i \in \{1,\ldots,5\}$, allow us to perform the graph complementarity operations (with S and T gates) and/or to measure in either the X or Z basis.

The circuit in Fig. 15 uses two circuits known as selective source and selective destination teleportation [36] to put the T gate into a form called ICM [7]. An ICM form of a quantum circuit consists of a layer of qubit initializations, an array of CNOT operations, and a time-staggered series of X- and Z-basis measurements. The choices of X- and Z-basis measurements are determined by the initial Z-basis measurement in the T-gate teleportation circuit and can dynamically *patch* in the correction circuit (or not).

We can simulate this circuit in the QE interface, but since the original circuit requires six qubits, we instead emulate the application of the T or $T^\dagger$ gate directly and

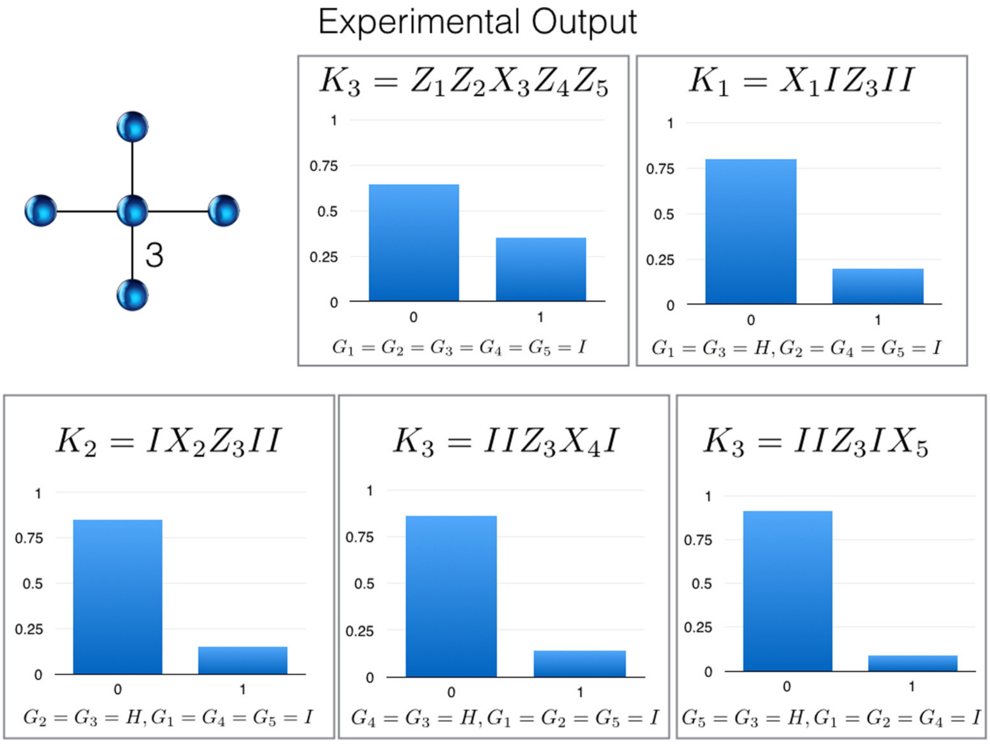


FIG. 9. Experimental results for the initial star graph. For each plot, we measure the parity of the associated stabilizer. The settings for each of the $G_{i\in 5}$ gates are illustrated for each plot.

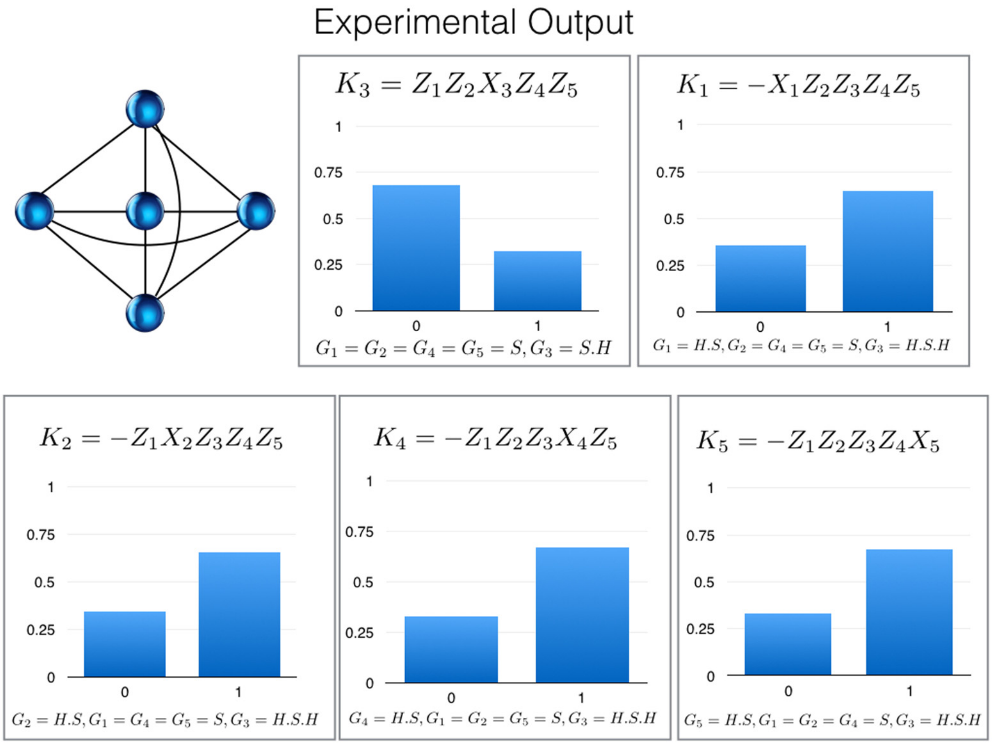


FIG. 10. Experimental results for the completely connected graph. The completely connected graph is constructed by performing a HSH gate on qubit 1 and an S gate on every other qubit. The permutation of the stabilizer set after this operation can cause some eigenvalues to flip between $+1$ and -1. For each plot, we measure the parity of the associated stabilizer. The settings for each of the $G_{i\in 5}$ gates are illustrated for each plot.

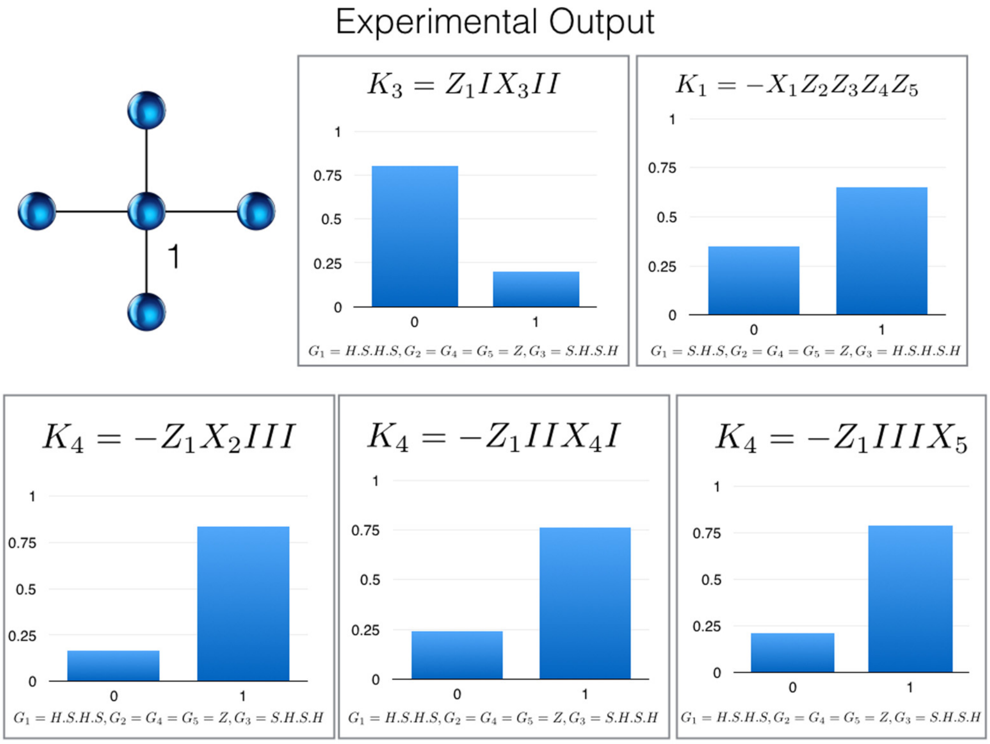


FIG. 11. Experimental results for the final star graph. We construct the final star graph (with qubit 1 acting as the node) by performing HSH to qubit 1 and S to every other qubit. This permutation does not change the eigenvalue of any of the stabilizers. For each plot, we measure the parity of the associated stabilizer. The settings for each of the $G_{i\in 5}$ gates are illustrated for each plot.

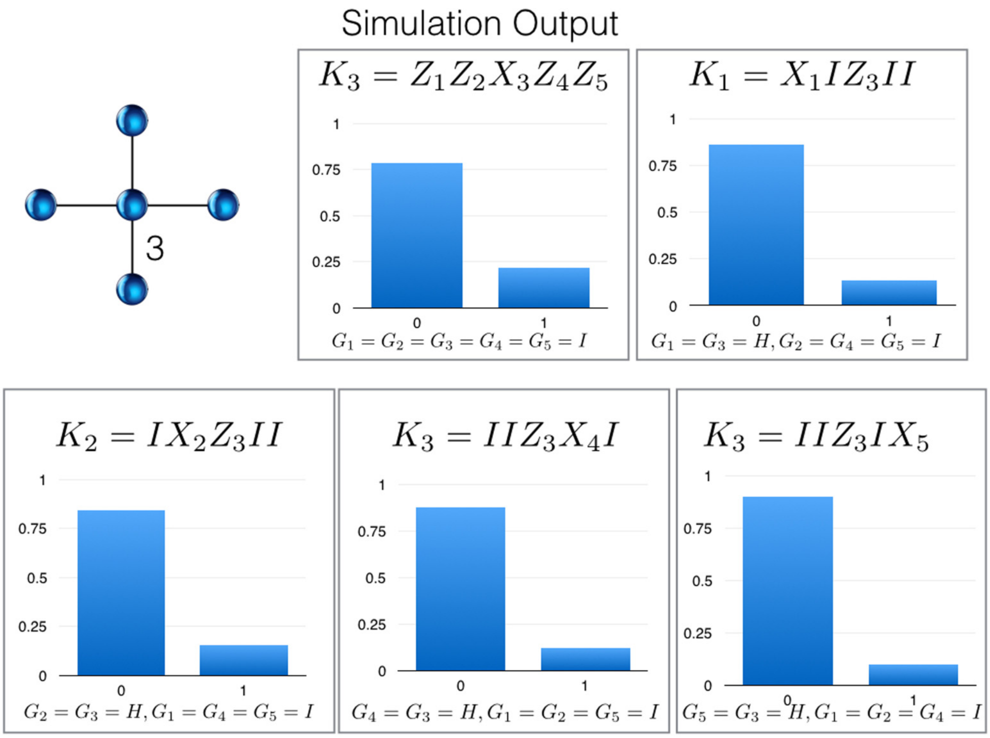


FIG. 12. Simulated results for Fig. 9.

then, based on which gate we choose, measure the four other qubits in the appropriate basis to teleport a T gate to the output regardless of whether we choose T or $T^\dagger$ at input. Unfortunately, while the QE interface does allow for qubit tomography to be performed, it does not allow both standard basis measurements and tomographic mapping on the same circuit run. Therefore, to confirm the application of the T gate, we simply reverse the initial circuit and confirm that $|0\rangle_{\text{in}} \rightarrow |0\rangle_{\text{out}}$. The circuit implemented in the QE chip is shown in Fig. 16.

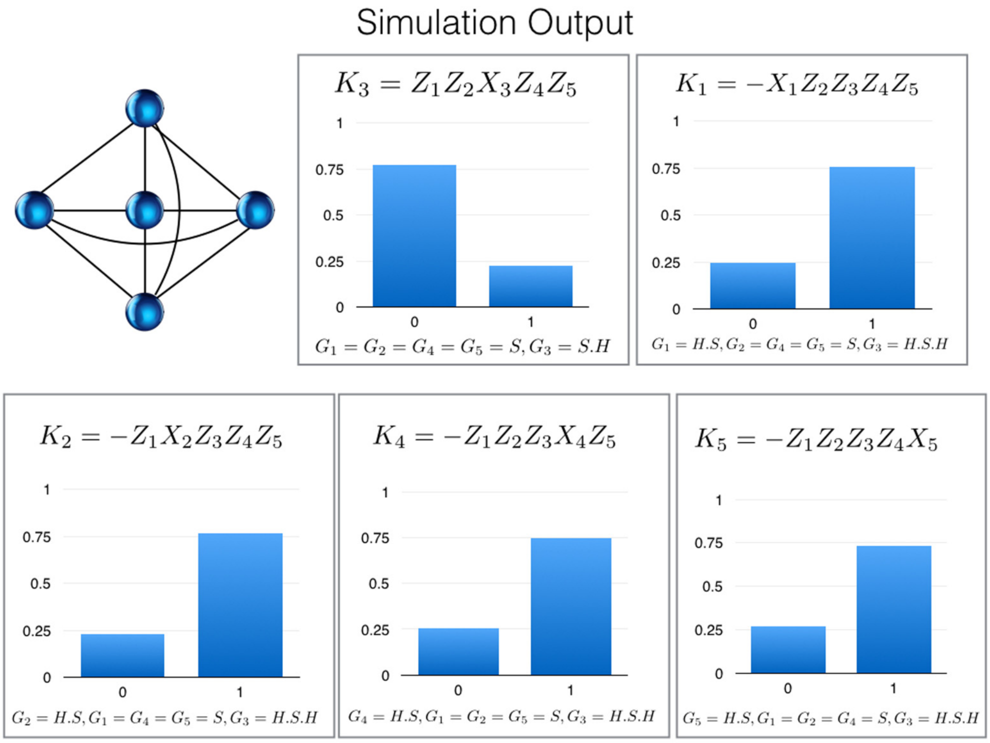


FIG. 13. Simulated results for Fig. 10.

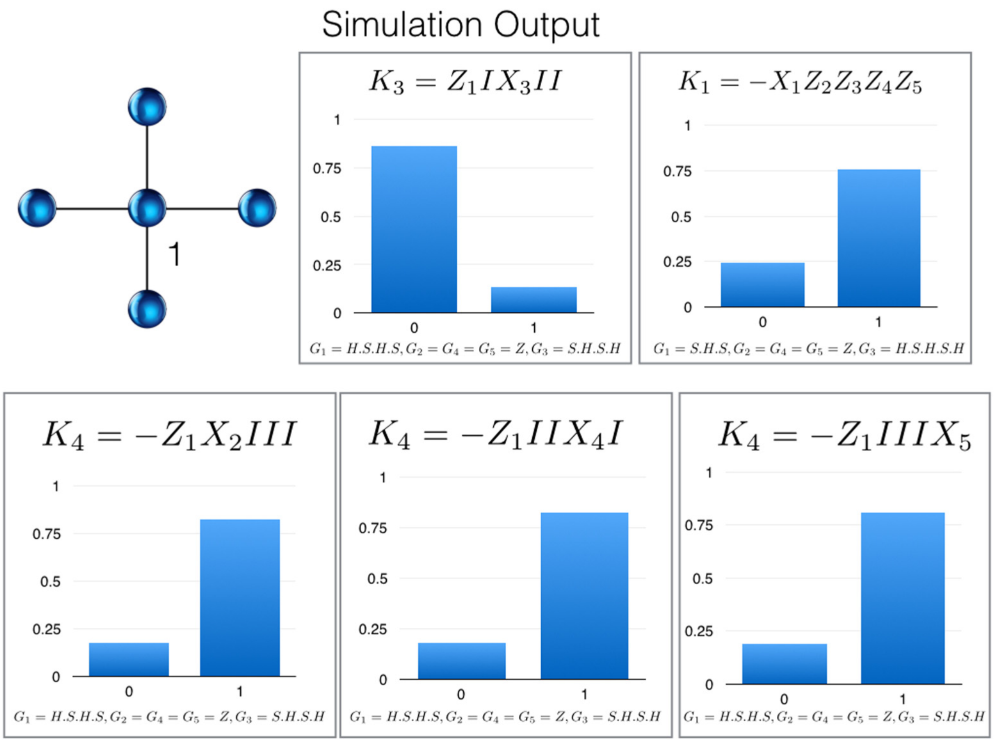


FIG. 14. Simulated results for Fig. 11.

Extra SWAP operations are needed in Fig. 16 because of the restrictions of the QE architecture. On the top qubit we apply either $G = T$ or $G = T^\dagger$, and depending on that choice, the four subsequent measurements are either in the $\{Z,Z,X,X\}$ basis or the $\{X,X,Z,Z\}$ basis. Once classical Pauli corrections based on measurements are taken into account (Table I, which we will discuss further below) and tracked, the inverse gate $T^\dagger$ (as the output should *deterministically* be $T|+\rangle$ regardless of the choice for G) is applied, and we measure the output qubit in the X basis. In the absence of circuit errors, the X-basis measurement should always return $|0\rangle$, indicating that the circuit dynamically applies the S correction and the output is always a T rotation. Simulations and experiments with the QE interface are illustrated in Fig. 17, again using 8192 instances with the SE omitted from the plot. The requirement for us to *undo* the initial gates to confirm the circuit introduces interesting behavior related to Pauli tracking [37] that is highly relevant for large-scale operations of a quantum computer.

Looking at both the simulation and experimental results, we observe the correct answer with a probability much less than 1. This is not only caused by experimental noise. In Fig. 18 we illustrate the results for an ideal application of the circuit (again performed within the QE interface). Even in ideal circumstances, the correct result is only observed (after Pauli correction) 75% of the time. The other 25% is when the wrong answer is reported *and* X corrections are required due to the teleportation operations. This is because the $T^\dagger$ gate used to invert the circuit for verification does not commute with this corrective bit-flip gate and the QE chip does not allow us to do *dynamic feedforward*, i.e., change circuits based upon the classical measurement result obtained earlier.

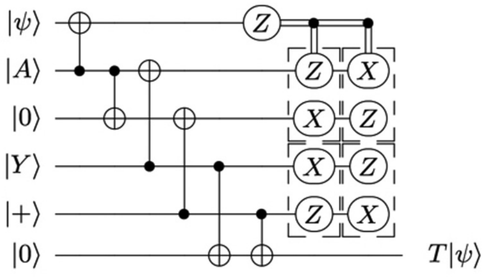


FIG. 15. Teleported T gate with deterministic circuitry. Built using selective source and destination circuits [36], this circuit will first apply the T gate, through a teleportation circuit with a magic-state ($|A\rangle = |0\rangle + e^{i\pi/4}|1\rangle$) ancilla. A possible S correction may be needed, requiring the second magic-state ancilla, $|Y\rangle = |0\rangle + i|1\rangle$. By selecting one of two choices of X- and Z-basis measurements (based on the top Z measurement), the S correction is applied or not. Hence the output is always the T-gate operating on the input state $|\psi\rangle$.

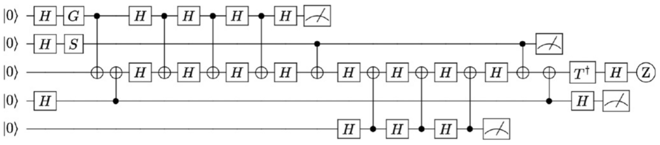


FIG. 16. Implementation of the deterministic T gate in the QE chip. The gate G can be chosen to be either T or $T^\dagger$, and the output from the circuit will be the same. The SWAP operations are again needed due to the QE architecture, at which point the four measurements are either in the basis $Z_1X_2X_4Z_5$ or $X_1Z_2Z_4X_5$, depending on the initial choice for G. The output qubit is rotated by $HT^\dagger$ and measured; the result should always be $|0\rangle$.

TABLE I. Pauli corrections for each of the four measurements in the deterministic T gate. For each of the 16 possible measurement results in Fig. 16, an appropriate Pauli correction to the output is needed. This correction needs to be *known* before subsequent gates are applied (as they may need to be altered).

Measurement Basis	Results	Corrections
$XZZX(ZXXZ)$	(0,0,0,0)	I (I)
	(1,0,0,0)	Z (X)
	(0,1,0,0)	X (Z)
	(1,1,0,0)	ZX (ZX)
	(0,0,0,1)	Z (X)
	(1,0,0,1)	I (I)
	(0,1,0,1)	ZX (ZX)
	(1,1,0,1)	X (Z)
	(0,0,1,0)	X (Z)
	(1,0,1,0)	XZ (XZ)
	(0,1,1,0)	I (I)
	(1,1,1,0)	Z (X)
	(0,0,1,1)	XZ (XZ)
	(1,0,1,1)	X (Z)
	(0,1,1,1)	Z (X)
	(1,1,1,1)	I

In a real operating quantum computer, the results of the four teleportation measurements (particularly the Z measurements that induce X corrections) need to be known *before* the application of the next gate, in this case the second $T^\dagger$ gate. If an X correction is present, the fact that $TX = XT^\dagger$ implies that we need to interchange a desired $T^\dagger$ gate with a T and vice versa if a bit-flip correction is present on the input. This bit-flip correction may come from circuits such as the deterministic T gate, but they can also come from other sources such as the error correction underlying the circuit [4] or corrections coming from the tracking of information from a topological quantum circuit [38].

In Fig. 18 we can reverse the distribution of probabilities by changing the gate that we use to invert the original T gate from $T^\dagger$ to T. In this case, if X corrections are not required, the inverse gate is wrong, and the output does not go back to $|0\rangle$. In Fig. 18 we illustrate all 32 possible outputs where the red bars are results that, after appropriate Pauli corrections and tracking, result in the output measuring $|0\rangle$. The basis states with approximately 3% probability of being observed have X corrections from the teleportation circuits that interfere with the function of the final $HT^\dagger$ gate needed to invert the circuit. In a circuit that would be *dynamically* changed depending on measurement results, this would not occur, and the output probability of $|0\rangle$ in the circuit would be 100%.

The results of this experiment directly demonstrate the notion of why T depth is an important concept [36]; the fact that certain results need to be known prevents us from building all circuits with a T depth of 1 (even through from a purely circuit perspective, this should look possible [7]). If dynamic circuit changes were possible within the QE interface, we could demonstrate a fully deterministic T gate.

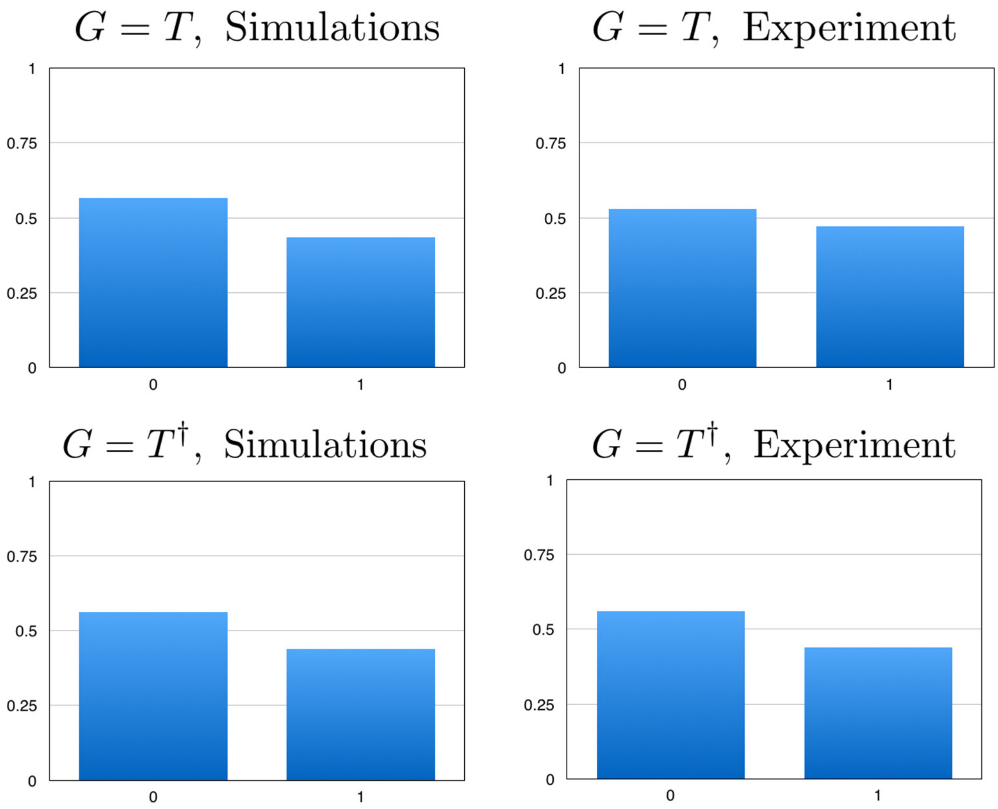


FIG. 17. Simulation and experimental results for $G = \{T, T^\dagger\}$. For both simulations and experiments there is only a slightly higher probability of measuring the expected result ($|0\rangle$); again, quantum errors for a large circuit in the QE chip are likely to blame. However, as explained in the text, this circuit only has a maximum probability of success of 0.75 in the ideal case.

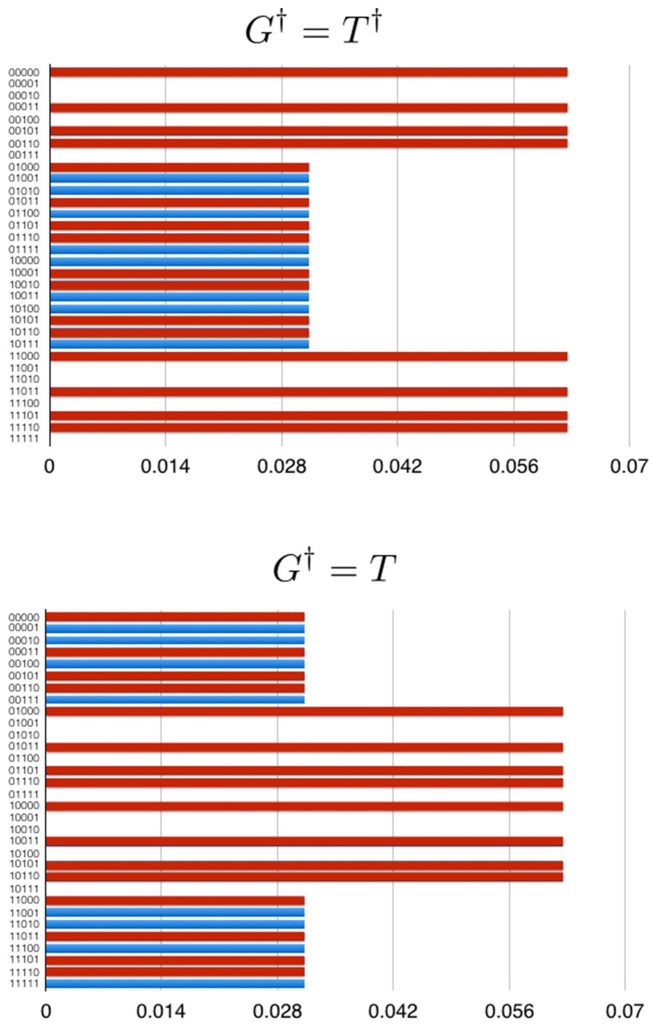


FIG. 18. Output distribution for an ideal application of the deterministic T gate. We show the output distribution for an ideal application of the deterministic T gate for $G = T$. Because Pauli X corrections arising from the teleportation circuits do not commute with the $T^\dagger$ gate used to invert the circuit and confirm output, the success probability has a maximum of 75%. The red bars show the binary outputs that, when corrected, are used to infer a $|0\rangle$ state output. If we change the $T^\dagger$ gate used at the end of the circuit to T, the probabilities invert since now the output will be wrong when *no* X correction from the teleportation circuit is needed.

III. DISCUSSION

In all four experiments we are simply looking at the output of the IBM chip rather than examining in detail the error behavior caused by noise in the chip; hence we have not included any significant analysis of the error properties of each qubit and/or gate (which is available from the QE interface for each run). Analyzing the detailed output from each experiment against the error data characterized for the QE chip would be interesting further work to try and refine simulation models to more accurately model how the actual device would behave given the input circuit and error characteristics of the chip.

Even without performing rigorous error analysis on the output, in all four experiments we do see the expected results, with some experiments being more susceptible to noise than others. The simulation feature in the QE interface also shows variability depending on the experiment we conducted. Out of the four experimental results, only the error correction Rabi experiment showed significant deviation between the simulations and experiment. Each of the four QE experiments allowed us to investigate the subtleties of implementing an actual quantum circuit on viable hardware and therefore required us to focus on details that were important to achieving the correct output that is often overlooked in more theoretical analysis. The results obtained for the deterministic T gate were particularly illustrative as they demonstrated the importance of having a real-time, up-to-date Pauli frame when programming a quantum device (something that is very often overlooked).

IV. CONCLUSIONS

In this work, we performed four separate experiments utilizing the IBM QE chip and interface, demonstrating protocols in error correction, quantum arithmetic, quantum graph theory, and fault-tolerant circuit design that have not been achievable so far with an active quantum processor. By utilizing the cloud interface, these experiments could be specified, tested (both in the ideal case and with simulated errors), and then run to output noisy (but expected) results. Each of these four protocols have direct relevance to quantum error correction, communications, algorithmic design, and fault-tolerant computation, and the ease of the QE interface makes subtle investigation into small quantum protocols straightforward.

Ideally, this work will help motivate others to make use of the online hardware produced by IBM and encourage other experimental groups to make their devices accessible to other researchers for testing and theoretical development. The five-qubit quantum processor has already shown significant flexibility to run tests on a large class of quantum information protocols, and increased fidelities and qubit numbers will hopefully spawn new and interesting protocols for small-scale quantum computing processors.

ACKNOWLEDGMENTS

We acknowledge support from the JSPS grant for challenging exploratory research and the JST ImPACT project. We are extremely grateful to the team at IBM and the IBM Quantum Experience project. This work does not reflect the views or opinions of IBM or any of its employees.

[1] Center for Quantum Photonics, Quantum in the Cloud, https://cnotmz.appspot.com/.

[2] IBM, The Quantum Experience, http://www.research.ibm.com/quantum/.

[3] D. Alsina and J. I. Latorre, Experimental test of Mermin inequalities on a five-qubit quantum computer, Phys. Rev. A **94**, 012314 (2016).

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

[5] T. Draper, Addition on a quantum computer, arXiv:quant-ph/0008033.

[6] M. Hein, J. Eisert, and H. J. Briegel, Multiparty entanglement in graph states, Phys. Rev. A **69**, 062311 (2004).

[7] A. Paler, I. Polian, K. Nemoto, and S. Devitt, A Compiler for Fault-Tolerant High Level Quantum Circuits, arXiv:1509.02004.

[8] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevA.94.032329 for additional experiments, raw data from the QE interface and the operating conditions of the device for each experimental run.

[9] S. J. Devitt, A. G. Fowler, A. M. Stephens, A. D. Greentree, L. C. L. Hollenberg, W. J. Munro, and K. Nemoto, Architectural design for a topological cluster state quantum computer, New. J. Phys. **11**, 083032 (2009).

[10] S. Devitt, W. Munro, and K. Nemoto, High performance quantum computing, Prog. Inf. **8**, 1 (2016) [arXiv:0810.2444].

[11] N. C. Jones, R. VanMeter, A. G. Fowler, P. L. McMahon, J. Kim, T. D. Ladd, and Y. Yamamoto, A Layered Architecture for Quantum Computing Using Quantum Dots, Phys. Rev. X **2**, 031007 (2012).

[12] N. Y. Yao, L. Jiang, A. V. Gorshkov, P. C. Maurer, G. Giedke, J. I. Cirac, and M. D. Lukin, Scalable Architecture for a Room Temperature Solid-State Quantum Information Processor, Nat. Commun. **3**, 800 (2012).

[13] K. Nemoto, M. Trupke, S. J. Devitt, A. M. Stephens, B. Scharfenberger, K. Buczak, T. Nöbauer, M. S. Everitt, J. Schmiedmayer, and W. J. Munro, Photonic architecture for scalable quantum information processing in diamond, Phys. Rev. X **4**, 031022 (2014).

[14] C. Monroe, R. Raussendorf, A. Ruthven, K. R. Brown, P. Maunz, L. M. Duan, and J. Kim, Large Scale Modular Quantum Computer Architecture with Atomic Memory and Photonic Interconnects, Phys. Rev. A **89**, 022317 (2014).

[15] M. Ahsan, R. V. Meter, and J. Kim, Designing a Million-Qubit Quantum Computer Using Resource Performance Simulator, arXiv:1512.00796.

[16] B. Lekitsch, S. Weidt, A. G. Fowler, K. Molmer, S. J. Devitt, C. Wunderlich, and W. K. Hensinger, Blueprint for a microwave ion trap quantum computer, arXiv:1508.00420.

[17] Y. Li, P. C. Humphreys, G. J. Mendoza, and S. C. Benjamin, Resource Costs for Fault-Tolerant Linear Optical Quantum Computing, Phys. Rev. X **5**, 041007 (2015).

[18] C. D. Hill, E. Peretz, S. J. Hile, M. G. House, M. Fuechsle, S. Rogge, M. Y. Simmons, and L. C. L. Hollenberg, A surface code quantum computer in silicon, Sci. Adv. **1**, e1500707 (2015).

[19] J. O'Gorman, N. Nickerson, P. Ross, J. Morton, and S. Benjamin, A silicon-based surface code quantum computer, npj Quantum Inf. **2**, 16014 (2016).

[20] R. Barends *et al.*, Logic gates at the surface code threshold: Superconducting qubits poised for fault-tolerant quantum computing, Nature (London) **508**, 500 (2014).

[21] A. D. Corcoles, E. Magesan, S. J. Srinivasan, A. W. Cross, M. Steffen, J. M. Gambetta, and J. M. Chow, Demonstration of a quantum error detection code using a square lattice of four superconducting qubits, Nat. Commun. **6**, 6979 (2015).

[22] K. Brown, J. Kim, and C. Munro, Co-designing a Scalable Quantum Computer with Trapped Atomic Ions, arXiv:1602.02840.

[23] F. Dolde *et al.*, Room-temperature entanglement between single defect spins in diamond, Nat. Phys. **9**, 139 (2013).

[24] J. P. Dehollain *et al.*, Bell's inequality violation with spins in silicon, Nat. Nanotehcnol. **11**, 242 (2016).

[25] R. Laflamme, C. Miquel, J. P. Paz, and W. H. Zurek, Perfect quantum error correcting code, Phys. Rev. Lett. **77**, 198 (1996).

[26] M. Nielsen and I. Chuang, *Quantum Computation and Information*, 2nd ed. (Cambridge University Press, Cambridge, 2000).

[27] N. Ross and P. Selinger, Optimal ancilla-free Clifford+T approximation of z-rotations, arXiv:1403.2975.

[28] D. Gosset, V. Kliuchnikov, M. Mosca, and V. Russo, An algorithm for the T-count, Quantum Info. Comput. **14**, 1261 (2014).

[29] S. Gay, Quantum Programming Languages, Math. Struct. Comput. Sci. **16**, 581 (2006).

[30] D. Wecker, and K. Svore, LIQUi| >: A Software Design Architecture and Domain-Specific Language for Quantum Computing, arXiv:1402.4467.

[31] A. Green, P. Lumsdaine, N. Ross, P. Selinger, and B. Valiron, Quipper: A Scalable Quantum Programming Language, ACM SIGPLAN Not. **48**, 333 (2013).

[32] D. Wecker, B. Bauer, B. K. Clark, M. B. Hastings, and M. Troyer, Gate-count estimates for performing quantum chemistry on small quantum computers, Phys. Rev. A **90**, 022305 (2014).

[33] S. Devitt, A. Stephens, W. Munro, and K. Nemoto, Requirements for fault-tolerant factoring on an atom-optics quantum computer, Nat. Commun. **4**, 2524 (2013).

[34] A. Paler, S. Devitt, and A. Fowler, Synthesis of Arbitrary Quantum Circuits to Topological Assembly, Sci. Rep. **6**, 30600 (2016).

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

[36] A. Fowler, Time Optimal Quantum Computation, arXiv:1210.4626.

[37] A. Paler, S. Devitt, K. Nemoto, and I. Polian, *Design, Automation and Test in Europe Conference and Exhibition (DATE)* (European design and automation association (EDAA), Dresden, 2014), pp. 1–4.

[38] A. Paler, S. J. Devitt, K. Nemoto, and I. Polian, Mapping of topological quantum circuits to physical hardware, Sci. Rep. **4**, 4657 (2014).