6-9-2020

Designing Gates and Architectures for Superconducting Quantum Systems

Sahar Daraeizadeh
Portland State University

Designing Gates and Architectures for Superconducting Quantum Systems

by

Sahar Daraeizadeh

A dissertation submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy
in
Electrical and Computer Engineering

Dissertation Committee:
Xiaoyu Song, Chair
Marek Perkowski, Co-chair
Douglas Hall
Steven Bleiler
Anne Y. Matsuura

Portland State University
2020

Abstract

Large-scale quantum computers can solve certain problems that are not tractable by currently available classical computational resources. The building blocks of quantum computers are qubits. Among many different physical realizations for qubits, superconducting qubits are one of the promising candidates to realize gate model quantum computers. In this dissertation, we present new multi-qubit gates for nearest-neighbor superconducting quantum systems. In the absence of a physical hardware, we simulate the dynamics of the quantum system and use the simulated environment as a framework for test, design, and optimization of quantum gates and architectures. We explore three different simulation-based gate design methodologies: analytical approach, heuristic method, and machine learning techniques. Furthermore, we propose novel quantum error correction architectures utilizing our new gates, which have reduced computational overhead with better performance and reliability.

Dedication

To my loving parents Shahin Moradi and Reza Daraeizadeh
with deepest gratitude for their love, support, enlightenment, and encouragement.

To my beloved husband and best friend Mike Tavakoli
who has been supporting me with love and patience throughout the doctorate program.

To my dear brothers Ali and Saman, wonderful sister-in-laws Afsaneh and Sahar, and my precious nieces and nephew: Diana, Elena and Ryan,
who make my life full of joy and happiness.

To my friends and family
with special appreciation to Maryam Khakbaz, Nader Tavakoli, and Houman Tavakoli who I am truly blessed to have them in my life.

To Leo for his companionship and unconditional love.

Acknowledgments

I would like to express my sincerest gratitude to my advisory committee.
I would like to thank my advisor Dr. Song for his motivation and guidance, and for his continuous support throughout my doctoral program.
A special thank to Dr. Douglas Hall for his encouragement, invaluable life advice, and his support during all these years.
I would like to thank Dr. Marek Perkowski who was so generous with his expertise and precious time. I truly appreciate his immense knowledge and invaluable advice throughout my PhD.
I would like to acknowledge and thank Dr. Anne Matsuura for her support and advice as well as internship opportunity at Quantum Computing group at Intel.
I would like to thank Dr. Steven Bleiler for allowing me to attend his Quantum Game Theory classes and informative Topological Quantum Computing seminars.
I would like to acknowledge and thank Dr. Malgorzata Chrzanowska-Jeske for her CAD physical design classes that provided me with fundamental knowledge as well as the teaching assistant position at her Electronics Lab.
Last but not least, I would like to express my deepest appreciation to Dr. Preethika Kumar for her invaluable advice and continuous support, and for the research assistant position in Quantum Computing lab at Wichita State University.
This work was made possible in part thanks to Portland Institute for Computational Science and its resources acquired using NSF Grant # DMS 1624776 and ARO Grant #W911NF-16-1-0307.

Table of Contents

List of Tables

List of Figures

1 Introduction

Building quantum computers is one of the most active research areas in the century. Quantum computers exceed the computational power of today's best classical supercomputers. Large-scale quantum computers have the potential to exceed classical computational resources in solving certain problems. The academia and industry leaders across the world are collaborating to reach quantum supremacy, where the quantum computer would perform a computation task that is impossible for the classical computational resources. Quantum supremacy has been shown by Google for an algorithm designed specifically for this purpose, i.e., not from a useful application area.
Building a large-scale quantum computer is an interdisciplinary task and requires the close collaboration of physicists, mathematicians, computer scientists, and engineers. The electrical and computer engineers can extensively contribute to building quantum computers by designing control electronics, quantum architectures, low level quantum gate, and creating new design automation tools as well as development of physical computer aided design tools.

The building blocks of quantum computers are qubits. There exist different physical realizations of qubits such as Nuclear Magnetic Resonance (NMR), liquid state NMR, Ion Trap, Optical Cavity, Photo Optics, Harmonic Oscillator, Superconducting devices, etc. Among all, quantum systems based on superconducting devices seem one of the most promising realizations in terms of scalability and error rates. The number of startup companies and industry lead-

ers investing in superconducting quantum computing grows every day, some of them are as follows: D-Wave, Google QuAIL (NASA), Intel (TU Delft), IBM (MIT), Microsoft Research Station Q, Q-Ctrl, Quantum Circuits, Inc. (Yale), RIKEN (Tokyo University of Science), Delft Circuits (QuTech), Rigetti Computing, and Raytheon/BBN (MIT), etc. Many quantum systems are based on Nearest-Neighbor (NN) layouts where qubits have direct interactions with only nearest neighbor qubits. In some architectures the interaction or coupling between qubits can be shut-off during the course of computation while in others the interactions are always on.

In this dissertation, we design new quantum gates and architectures for NN superconducting quantum systems using the simulated environments. The gate design methodologies presented in this work are applicable to many physical realizations. We design new multi-qubit gates using three different methodologies: heuristics, machine learning/optimization methods, and analytical approach. The multi-qubit gates mimic the behavior of several gates at once, therefore, they can increase the efficiency of quantum circuits and algorithms implementations by reducing control circuitry, improving performance, and achieving smaller error rates. In each chapter of this dissertation we describe one of the mentioned methodologies to design new gates, and then provide an application for the new gates by utilizing them in quantum circuits and new quantum error correction architectures.

We contribute to the quantum computing research in the following main ways. First, we explain given the Hamiltonian (the operator representing the system energy) of an arbitrary quantum system, one can design new multi-qubit gates analytically and confirm the results in simulation. Second, by focusing on nearest-neighbor superconducting systems based on transmons in circuit

Quantum Electrodynamics (cQED) regime, we show how knowing the physics underlying the quantum system can help us to design new quantum gates. Third, for more complex quantum systems, we show how to model the quantum gate design problem as a control/optimization problem and use the computational intelligence (artificial intelligence/machine learning) approaches and high-performance computing resources to realize a desired quantum operation. Forth, we design new efficient quantum error correction architectures utilizing our new multi-qubit gates. Fifth, we describe the procedure of mapping a quantum circuit to a physical system and discuss the importance of development of Computer-Aided Design (CAD) tools for quantum systems.

In this chapter, first we provide some background on gate model quantum information processing by introducing the quantum gates and quantum error correction concepts using some examples. Finally, we put the next chapters in simulation-based gate design context.

1.1 Quantum gates

A quantum register with two distinct energy levels is called a qubit. In quantum computing literature, the ground state and excited state of the qubits are respectively represented by state vectors $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$ and $\begin{bmatrix} 0 \\ 1 \end{bmatrix}$, or in Dirac notation as $|0\rangle$ and $|1\rangle$, respectively. The state of a qubit can be in the superposition of its ground and excited states which is a linear combination of $|0\rangle$ and $|1\rangle$:

$$\left|\psi(t)\right\rangle = \alpha|0\rangle + \beta|1\rangle \ , \ |\alpha|^2 + \left|\beta\right|^2 = 1 \tag{1.1}$$

where α and β are complex numbers representing the probability amplitudes of being in states $|0\rangle$ and $|1\rangle$, respectively. And $\psi(t)$ is the state of the qubit at time

t, ψ is a vector in a complex vector space with the orthonormal basis of $|0\rangle$ and $|1\rangle$ [1]. We could use another orthonormal basis for instance $|+\rangle = (\frac{1}{\sqrt{2}})(|0\rangle + |1\rangle)$ and $|-\rangle = (\frac{1}{\sqrt{2}})(|0\rangle - |1\rangle)$ to represent the quantum state.

The state of a quantum system consisting of multiple qubits spans a Hilbert space of size 2^n, where n is the number of qubits in the system. Hilbert space is a complex inner product space that is a complete metric space with respect to the distance function induced by inner product. The state of the n-qubit system can be represented by tensor product (denoted by $\otimes$) of the state vectors of all qubits. For example, the vector representation of the state of a two-qubit system with states $\left|\psi_1\right\rangle$ and $\left|\psi_2\right\rangle$ is:

$$\left|\ \ \right\rangle = \left|\psi_1\right\rangle \otimes \left|\psi_2\right\rangle = \alpha\,|00\rangle + \beta\,|01\rangle + \gamma\,|10\rangle + \lambda\,|11\rangle\ ,\ |\alpha|^2 + \left|\beta\right|^2 + \left|\gamma\right|^2 + |\lambda|^2 = 1 \quad (1.2)$$

where the time t is omitted for simplicity. Here α, β, γ, and λ are the probability amplitudes of the two-qubit system being respectively in states $|00\rangle$, $|01\rangle$, $|10\rangle$, and $|11\rangle$.

In gate model quantum computation, information is processed through quantum circuits which consist of a sequence of quantum gates. Quantum gates are reversible unitary transformations on quantum states. Therefore quantum circuits are reversible in time [2]. This means that there is a one to one relation between input and output of a quantum circuit.

1.1.1 Single-qubit gates

An n-qubit gate is represented by unitary matrices of size $2^n \times 2^n$. Therefore, single-qubit gates can be represented by 2×2 matrices. Some of the single qubit gates are X, Y, and Z gates. They can be described using Pauli spin matrices as

follows:

$$X = \sigma_x = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \tag{1.3}$$

$$Y = \sigma_y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \tag{1.4}$$

$$Z = \sigma_z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \tag{1.5}$$

Other important single qubit gates are Hadamard, S, and T gates. The unitary transformation matrices for these gates are as follows:

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{1.6}$$

$$S = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix} \tag{1.7}$$

$$T = \begin{bmatrix} 1 & 0 \\ 0 & e^{\frac{i\pi}{4}} \end{bmatrix} \tag{1.8}$$

1.1.2 Multi-qubit gates

The multi-qubit gates or controlled-unitary gates can be applied on multiple qubits at the same time. In most of the controlled-unitary gates, some qubits have a target role and some qubits have the control role. Depending on the states of the control qubits, a desired gate operation is performed/not performed on the target qubits. One of the most famous two-qubit gates is the CNOT gate which performs a controlled-X (controlled-bit-flip) operation, where the state of the target qubit flips if and only if the state of the control qubit is $|1\rangle$.

$$\text{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{1.9}$$

A cPhase (CZ) gate is another example of a two-qubit gate which performs a controlled-Z operation (controlled-phase-flip), only if both qubits are in state $|1\rangle$. Although cPhase is a controlled-unitary operation, there is no control and target associated with it. We can realize a CNOT gate by surrounding cPhase gate between two Hadamard gates on any of the qubits.

$$\text{cPhase} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix} \tag{1.10}$$

Another important two-qubit gate is the SWAP gate which exchanges the state of two qubits.

$$\text{SWAP} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{1.11}$$

An example of a three-qubit gate is a Toffoli gate which is a controlled-controlled-X operation, where the state of the target qubit flips if and only if both control

qubits are in state $|1\rangle$.

$$\text{Toffoli} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \tag{1.12}$$

Moreover, the Fredkin gate is also a three-qubit gate which performs a controlled-SWAP operation, where the states of two target qubits exchange if and only if the control qubit is in state $|1\rangle$.

$$\text{Fredkin} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{1.13}$$

The cost of implementing the quantum gates depends on the technology of the quantum system. In some technologies it is more natural to the system to realize a cPhase gate rather than a CNOT gate. In some physical systems, a direct SWAP gate implementation is possible, while in others a SWAP gate is decomposed to three CNOT gates. The important factor in a quantum computer is that a universal set of quantum gates be implemented that are capable to

realize any operation in a quantum circuit. All other gates that are not directly implemented in the hardware are decomposed to a finite sequence of available gates from the universal set. For example, the set of (H, T, CNOT) gates and the set of (H, Toffoli) form two different universal sets for quantum computing. In general at least one multi-qubit gate is required in an universal set for quantum computing.

In many technologies, the multi-qubit gates with three or more qubits are realized by a decomposition to a sequence of single-qubit and two-qubit gates. A theorem from [3] states that any controlled-controlled-Unitary gate can be implemented utilizing CNOT, controlled-V, and controlled-$V^\dagger$ gates, where $V^2 = U$. This decomposition is shown in Fig. 1.1 where an arbitrary three-qubit gate U_0 is implemented by a quantum circuit consisting of a sequence of 5 two-qubit gates U_1, U_2, U_3, U_4, and U_5. Each wire in a quantum circuit denotes a qubit. The circuit reads left to right to account for the time that gates are being applied. In the notation of quantum circuits, single-qubit gates are shown using a box on the qubit with a label showing the operation. The CNOT gates are denoted by a line connecting the control qubit with a full dot notation to the target qubit with an XOR notation. In Fig. 1.1, controlled-V/controlled-$V^\dagger$ gates operate such that when the control qubit is in the high state, a unitary operation $V/V^\dagger$ is applied on the target qubit.

Consider we would like to realize the Toffoli gate which has broad applications in many quantum circuits. Using the theorem from [3], the Toffoli gate can be realized using five two-qubit gates where non-standard two-qubit gates such as controlled-V and controlled-$V^\dagger$ gates are required. Here according to theorem from [3], $V^2 = X$, and since V must be a unitary operator $VV\dagger = I$. In other words, V, and $V\dagger$ gates can be represented by c $\sqrt{\text{NOT}}$, and c $\sqrt{\text{NOT}}^\dagger$,

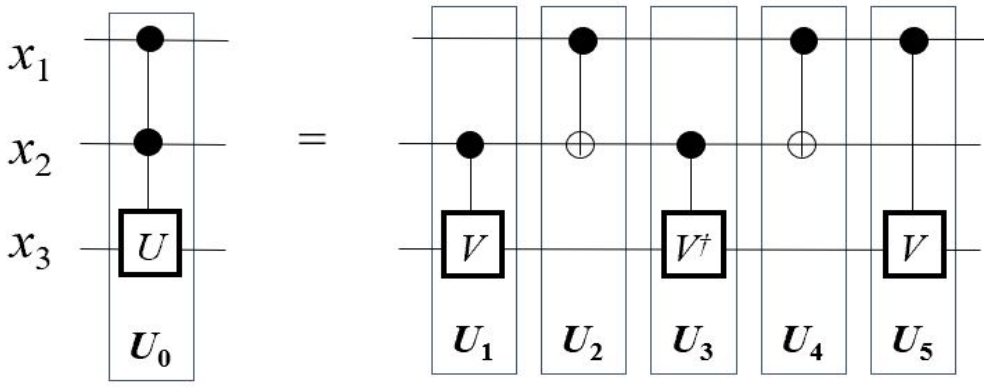


Figure 1.1 A quantum circuit realizing a controlled-controlled-Unitary gate based on two-qubit gates.

respectively. In chapter 3, we realize the c $\sqrt{\text{NOT}}$, and c $\sqrt{\text{NOT}}^\dagger$ gates using controlled-rotation flux-tunable gates in transmons in cQED systems.

In nearest-neighbor architectures, in order to perform quantum gates between two non-neighbor qubits, we need to bring them adjacent to each other by applying SWAP gate operations. In Fig. 1.1, the last controlled-V gate (shown as U_5) is between x_1 and x_3, where x_1 has the control role and x_3 has the target role. Consider a nearest-neighbor architecture where there is no direct interaction between x_1 and x_3, but they are both neighbors to x_2. In such system, we require to perform two SWAP gates between x_1 and x_2 before and after a controlled-V gate between x_2 and x_3, this way we bring the control from the non-neighbor qubit x_1 to the neighbor qubit x_2 which is adjacent to the target qubit x_3.

The best-known decomposition of the Toffoli (controlled-controlled-NOT) gate using standard single- and two-qubit gates [4] requires multiple single-qubit gates (H, T, and $\text{T}^\dagger$) and 6 CNOT gates as shown in Fig. 1.2.

In this decomposition, at least two of the CNOT gates are applied to non-neighbor qubits. Therefore, the circuit shown in Fig. 1.2 requires the addition

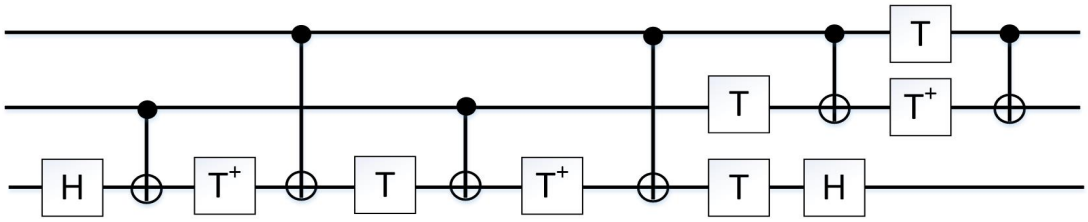


Figure 1.2 A quantum circuit realizing a Toffoli gate based on standard single-qubit and two-qubit gates.

of four SWAP gates in a nearest-neighbor architecture. The decomposition of the Toffoli gate based on single- and two-qubit gates is costly in most of the quantum systems. In chapter 4, we design a Toffoli gate based on a three-qubit controlled-controlled-Phase (ccPhase) gate and two single-qubit gates (Hadamard or single-qubit rotation gates).

1.2 Background on Quantum Error Correction

Due to the nature of the quantum systems, the quantum information is subject to decoherence where the state of the quantum system is affected by the environmentally-induced errors and some information is lost. The loss of information can be due to the loss of energy from the system (amplitude damping) or due to the the loss of the relative phase between the energy levels (phase damping). There are other sources of errors in quantum systems such as when a qubit which is considered as a two-level (ground and excited states) system leaks into non-computational states, or when noise is introduced to the system through environment, or error happens during qubit initialization, or because of erroneous gates or measurements. In general, quantum errors and instability of quantum states are considered fundamental obstacles to achieve large-scale quantum computers. To protect quantum information during storage and computation we can employ error mitigation methods or use Quantum Error Cor-

rection (QEC) schemes [5]. Currently, designing quantum architectures with the least latency and efficient QEC schemes is an important area of research. The chapters 2 and 4 investigate applying new multi-qubit gates to realize new error correction circuits and architectures for nearest-neighbor quantum systems. Here we work through a few examples to provide background about QEC codes.

1.2.1 Repetition Code

The QEC codes are fundamentally different than the classical error correction codes and are facing the following challenges:

1. It is not possible to copy an unknown quantum state using unitary evolution.

2. Quantum errors can be decomposed to multiple types (e. g. bit-flip and phase-flip).

3. Direct measurement destroys quantum superposition.

4. Quantum errors are continuous rather than discrete.

Here we show how a simple quantum error correction scheme works despite above mentioned challenges. In QEC codes, we create redundancy by encoding the unknown state of the qubit to the state of several physical qubits to form a logical qubit:

$$\left|\ \ \right\rangle = \alpha\left|0\right\rangle + \beta\left|1\right\rangle \ -> \ \left|\psi_L\right\rangle = \alpha\left|0_L\right\rangle + \beta\left|1_L\right\rangle \ , \tag{1.14}$$

where $\left|\psi_L\right\rangle$ is an entangled state where $|0_L\rangle = |000\rangle$, and $|1_L\rangle = |111\rangle$ [1]. We perform the measurements on the ancillary qubits so the state of the logical

qubit is not disturbed. Consider we have a logical qubit consisting of two data qubits and an ancillary qubit. We can realize a $\hat{Z}\hat{Z}$ operator by applying two CNOT gates where the ancillary qubit is the target qubit and two data qubits are the control qubits as depicted in Fig. 1.3. The ancillary target qubit is a measurement qubit called measure-Z qubit and it can be repetitively measured to detect any bit-flip between its neighbor data qubits.

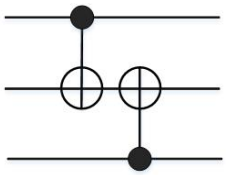


Figure 1.3 $\hat{Z}\hat{Z}$ operator built from two CNOT gates can detect any bit-flip error of the first and third qubits.

Similarly, the measure-X qubit can be used to detect phase-flip errors. An $\hat{X}\hat{X}$ operator can be realized using two CNOT gates applied to three qubits where the middle qubit is the measure-X ancillary qubit and has the control role while the two adjacent data qubits are the target data qubits as depicted in Fig. 1.4.

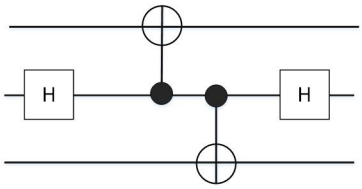


Figure 1.4 Two CNOT gates in combination with Hadamard gates on the second qubit can detect any phase-flip error of the first and third qubits.

In Fig. 1.4, by applying Hadamard gates before and after the ancillary measure-X qubit; we can change the state of measure-X qubit from $|0\rangle$ to $|+\rangle$ and revert it back after applying $\hat{X}\hat{X}$ operator. Since the phase errors propagate to the control qubit, if adjacent data qubits contain any phase-flip error, it will change the

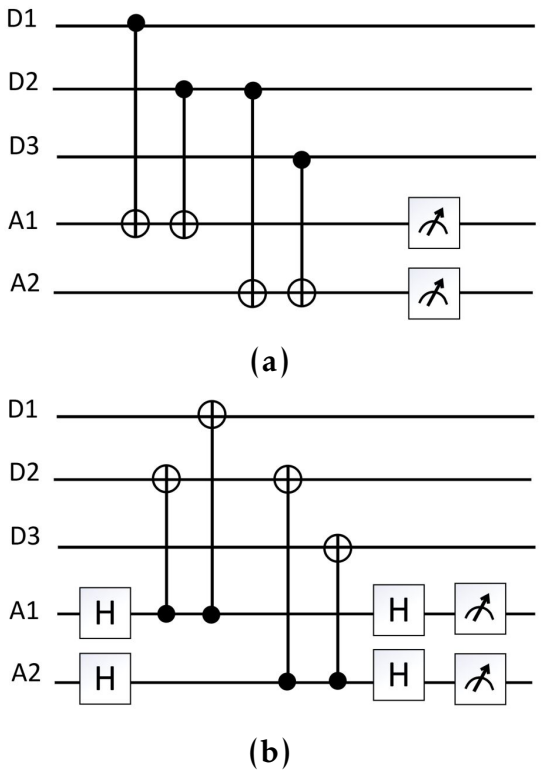


Figure 1.5 Error syndromes are detected by performing measurements on ancillary qubits. (a) Bit-flip repetition code (b)Phase-flip repetition code

state of measure-X qubit from $|+\rangle$ to $|-\rangle$, then the last Hadamard gate operation converts it to $|1\rangle$. Therefore, the $\hat{X}\hat{X}$ operator can be used to detect any phase-flip of the adjacent data qubits.

Applying $\hat{Z}\hat{Z}$ operator to $|00\rangle$ and $|11\rangle$ results in $+|00\rangle$ and $+|11\rangle$, respectively. Applying $\hat{Z}\hat{Z}$ operator to $|10\rangle$ and $|01\rangle$ results in $-|10\rangle$ and $-|01\rangle$, respectively. Therefore, states $|00\rangle$, $|01\rangle$, $|10\rangle$, and $|11\rangle$ are eigenstates of $\hat{Z}\hat{Z}$ operator with corresponding eigenvalues +1, -1, -1, and +1. The $\hat{Z}\hat{Z}$ operator then can be used to detect the changes in parity of data qubits.

The circuit depicted in Fig. 1.5 (a) realizes the repetition code on the logical qubit by introducing two ancillary qubits. A measurement performed on the first ancillary qubit detects if the first pair of qubits have odd parity. And a

measurement on the second ancillary qubit detects if the second pair of qubits have odd parity. In Fig. 1.5 (a), a correctly encoded state (codeword) is a +1 eigenvector of $\hat{Z}\otimes\hat{Z}\otimes\hat{I}$ operator: $(\hat{Z}\otimes\hat{Z}\otimes\hat{I})\left|\ \right\rangle = +1\left|\ \right\rangle$. A state with an odd parity between the first and the second qubit is a -1 eigenvector of $\hat{Z}\otimes\hat{Z}\otimes\hat{I}$ operator: $(\hat{Z}\otimes\hat{Z}\otimes\hat{I})\left|\ \right\rangle = -1\left|\ \right\rangle$. Similarly an odd parity between the second and the third qubit results in a -1 eigenvector of $\hat{I}\otimes\hat{Z}\otimes\hat{Z}$ operator. Error syndrome is formed by measuring enough operators to determine the location of error. The group of all operators detecting the error syndrome are called stabilizer of the code.

The circuit shown in Fig. 1.5 (a) can locate a single bit-flip error on any of the three physical qubits forming the logical qubit. If we change the basis to $|+\rangle$ and $|-\rangle$, we can detect the phase-flip errors.

$$\left|\ \right\rangle = \alpha|+\rangle + \beta|-\rangle \ -> \ \left|\psi_L\right\rangle = \alpha|+_L\rangle + \beta|-_L\rangle\ , \tag{1.15}$$

The circuit representing the phase-flip repetition code is shown in Fig. 1.5 (b). Here the measurement operators (group of stabilizers) are represented by $\hat{X}\otimes\hat{X}\otimes\hat{I}$ and $\hat{I}\otimes\hat{X}\otimes\hat{X}$.

1.2.2 Shor Code

The Shor's nine-qubit code [6] combines the bit-flip and phase-flip repetition codes to detect a single error of any type in the logical qubit consisting of nine physical qubits.

$$\left|\ \right\rangle = \alpha|0\rangle + \beta|1\rangle \ -> \ \left|\psi_L\right\rangle = \alpha(|000\rangle + |111\rangle^{\otimes 3}) + \beta(|000\rangle - |111\rangle^{\otimes 3}) \tag{1.16}$$

As depicted in Fig. 1.6, the information is encoded to a logical qubit consisting of nine physical qubits. Then the information is sent through an erroneous

communication channel E, then in destination the information is decoded and if there is a single error, it will be automatically fixed by the Toffoli gates. However, the encoding and decoding circuit elements are also subject to noise and decoherence.

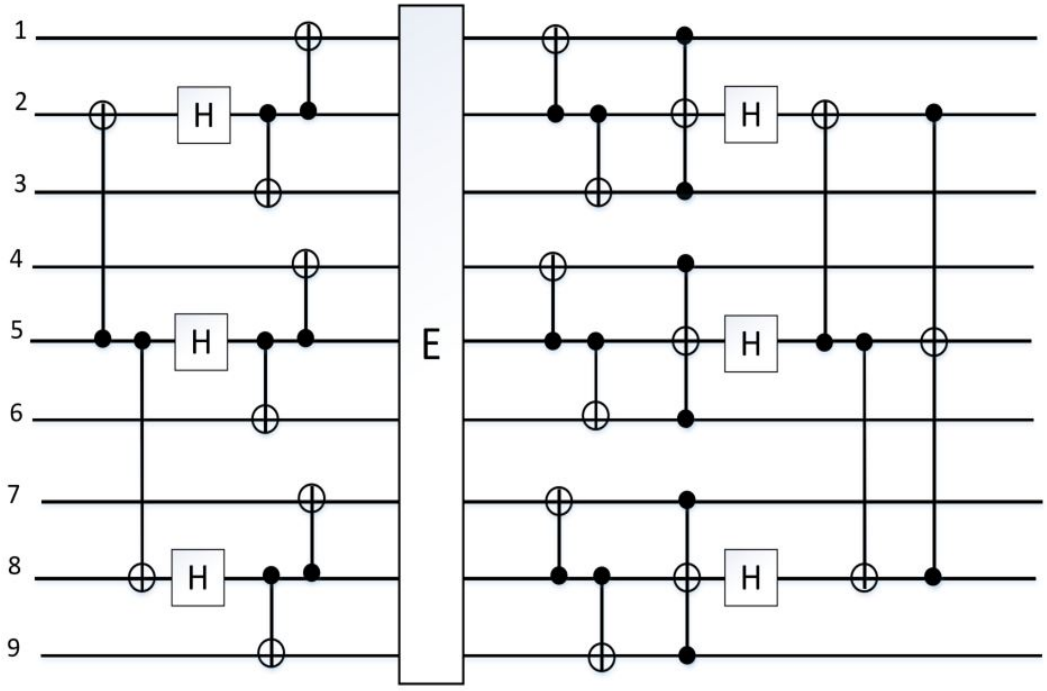


Figure 1.6 Circuit realizing Shor's nine-qubit code

Quantum errors on single qubits will propagate in quantum circuits through multi-qubit quantum gates. A bit-flip error in a control qubit propagates to the target qubit and a phase-flip error in a target qubit propagates to the control qubit. Additionally, erroneous gates can introduce errors to their coupled qubits. Furthermore, an error in measurement can introduce errors to the result of calculation. Therefore, we need an error correction scheme to be fault tolerant as well. A fault tolerant quantum computer works reliable despite the noise on stored quantum information, faulty quantum gates, faulty quantum preparation, and faulty measurements, as long as the error probability per op-

eration be below a certain threshold. In fault tolerant quantum computing, the computation is performed on the encoded qubits.

There are many quantum error correcting codes [5]. Some of the most recognized ones are Shor's 9 qubits code [6], Steane 7 qubits code [7], Calderbank-Shor-Steane (CSS) code [8], Stabilizer code [9], and the Bacon-Shore code, Repetition Code, and Surface Code [10-13] . However, not all error correction codes are fault tolerant. One of the most promising fault tolerant quantum computing schemes that follows the fault tolerant metrology proposed by Martinis [14] is the Surface Code [13].

1.2.3 Fault tolerance threshold

Martinis [14] proposed a metrology for fault-tolerant error correction for scalable quantum computers by measuring qubit parities which detect bit-flip and phase-flip errors in pairs of qubits. Based on his metrology, in a parity operation which consists of one-qubit, two-qubits and measurement components, we need to keep the error probability of each component less than a defined threshold to reach an error suppression factor $\wedge$ of higher than 1. Higher order error detection leads to lower logical error probability $P_l \simeq \wedge^{-(n+1)}$ [14], where n is the order of error, and $\wedge = \epsilon_t/\epsilon$ is the error suppression factor, here ϵ is the probability of physical error, and ϵ_t is the error threshold. According to Martinis, "$\wedge$ is the key metrological figure of merit that quantifies how much the decoding error drops as the order n increases by one" [14] . Here $\wedge > 1$ means that the physical error ϵ is lower than the threshold ϵ_t, and by making the error correction code larger the decoding error is decreased exponentially with n [14] .

Assuming after Martinis that a typical quantum algorithm implementation

uses 1018 operations, and that we need to achieve the overall logical error probability of less than $P_l = 10^{-18}$ meaning a suppression factor of $\wedge = 10$, this leads to order error n = 17. To achieve error correction of n order in a Surface Code architecture [14] , we need (4n+1)×(4n+1) array of qubits. This requires as many as 4761 qubits for n=17. Although the cited above number seems to be large, Surface Code architecture still is the best practical error correction method for fault tolerant quantum computing because of high tolerance to the errors which allows error rate of 1% per operation. Moreover, its two-dimensional physical layout with nearest neighbor couplings makes it a scalable and practical approach in solid-state quantum computers [13, 15-16]. Furthermore, because of simple projective measurements, and tracking of the detected errors in software, there is no need for applying physical correction gates, therefore introducing less noise and perturbation to the physical system. In Surface Code error correction, it is of high interest to be able to perform error correction cycles on many qubits simultaneously which is where the multi-qubit gates can be applied as explained in chapter 2.

1.2.4 Surface Code

The surface code architecture is based on the stabilizer formalism and consists of Z and X stabilizers [11]. Surface Code introduces ancillary qubits dedicated to these stabilizers and repetitively performs projective quantum nondemolition (QND) parity measurements on these ancillary qubits to measure the bit-flip and phase-flip errors of the data qubits [13] . The number of ancillary qubits in these measurements is approximately equal to the number of data qubits. Although it has been shown that this approach results in storing information with a lower error rate, the Surface Code methodology has a high

computational and resource overhead to realize the logical states and process information.

In surface code quantum computing, multiple physical qubits form a logical qubit in a 2-dimensional array with interleaving data qubits and measurement qubits called measure-Z and measure-X ancillary qubits, and a protocol is presented to protect the architecture from both bit-flip and phase-flip errors at the same time. In this scheme, the quantum information is distributed over many physical qubits that consist of data qubits and measurement qubits. Where the measure-X and measure-Z qubits detect phase-flip and bit-flip parities, respectively.

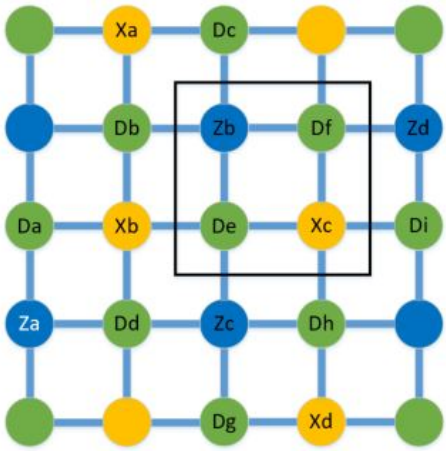


Figure 1.7 A 2D array of NN qubits forming a Surface Code. The labeled qubits form a logical qubit containing 17 physical qubits, 9 of which are data qubits and 8 of which are measurement qubits. The box shows two data qubits De and Df in green, one measure-Z qubit Zb in blue and one measure-X qubit Xc in orange.

At the start, all measurement qubits are initialized to zero. At each error correction cycle, we perform measurements only on the measurement qubits to detect the error syndromes (bit-flip, phase-flip, measurement error). A software maps these detected error syndromes to a graph model which keeps track of errors and fixes the errors [5, 12-13].

As we know $|+\rangle = (\frac{1}{\sqrt{2}})(|0\rangle + |1\rangle)$ and $|-\rangle = (\frac{1}{\sqrt{2}})(|0\rangle - |1\rangle)$ are eigenstates of $\hat{X}$ operator, with eigenvalues +1 and -1, respectively. Considering a two-qubit system, applying $\hat{X}\hat{X}$ operator to the Bell states $(\frac{1}{\sqrt{2}})(|00\rangle + |11\rangle)$, $(\frac{1}{\sqrt{2}})(|01\rangle + |10\rangle)$, $(\frac{1}{\sqrt{2}})(|00\rangle - |11\rangle)$, and $(\frac{1}{\sqrt{2}})(|01\rangle - |10\rangle)$ results in eigenvalues +1, +1, -1, and -1, respectively. While applying $\hat{Z}\hat{Z}$ operator to the same set of states results in eigenvalues +1, -1, +1, and -1, respectively. Knowing the fact that all $\hat{X}$ and $\hat{Z}$ operators on different qubits commute with one another, it is possible to measure the phase and amplitude of two data qubits simultaneously without perturbing the state of the two-qubit system.

Table 1.1 shows the set of Bell states as eigenstates of $\hat{X}_{Df}\hat{X}_{De}$ and $\hat{Z}_{Df}\hat{Z}_{De}$ operators with their corresponding eigenvalues (Please refer to the black box in Fig. 1.7). The eigenvalues for $\hat{X}_{Df}\hat{X}_{De}$ and $\hat{Z}_{Df}\hat{Z}_{De}$ are recorded by measuring measure-X and measure-Z qubits labeled Xc and Zb, respectively. Suppose we have initialized the system in the $(\frac{1}{\sqrt{2}})(|00\rangle + |11\rangle)$ state. If an X (bit-flip) error happens on Df data qubit, the new state of the system becomes $(\frac{1}{\sqrt{2}})(|01\rangle + |10\rangle)$. As such, the result of applying $\hat{X}_{Df}\hat{X}_{De}$ and $\hat{Z}_{Df}\hat{Z}_{De}$ operators will be changed from the pair of eigenvalues (+1, +1) to (+1, -1). Note that, if an X error happens on De, we will get the same result. Consequently, we cannot distinguish on which data qubit the error occurred because they both will have the same measurement result. This is true for Z (phase-flip) errors too. Therefore, to uniquely identify errors on specific data qubits, we need to consider a more complex mechanism such as Surface Code [13] .

In Surface Code, each data qubit is surrounded with 4 measurements qubits while each measurement qubit is surrounded with 4 data qubits as shown in Fig. 1.7. The measure-Z qubit stabilizes the product of $\hat{Z}$ operators on the surrounding qubits. For example, in Fig. 1.7, the qubit Zb forces the data qubits

Table 1.1 The set of eigenstates and corresponding eigenvalues for two-qubit stabilizers $\hat{X}_{Df}\hat{X}_{De}$and$\hat{Z}_{\mathrm{Df}}\hat{Z}_{\mathrm{De}}$

$\hat{X}_{Df}\hat{X}_{De}$	$\hat{Z}_{Df}\hat{Z}_{De}$	$\|\psi\rangle$
+1	+1	$(\frac{1}{\sqrt{2}})(\|00\rangle+\|11\rangle)$
+1	-1	$(\frac{1}{\sqrt{2}})(\|01\rangle+\|10\rangle)$
-1	+1	$(\frac{1}{\sqrt{2}})(\|00\rangle-\|11\rangle)$
-1	-1	$(\frac{1}{\sqrt{2}})(\|01\rangle-\|10\rangle)$

Df, De, Dc, and Db to an eigenstate of operator outer product $\hat{Z}_{Df}\hat{Z}_{De}\hat{Z}_{Dc}\hat{Z}_{Db}$. The measure-X qubit stabilizes the product of $\hat{X}$ operators on the surrounding qubits. In Fig. 1.7, the qubit Xc forces the data qubits Di, Df, Dh, and De to an eigenstate of operator outer product $\hat{X}_{Di}\hat{X}_{Df}\hat{X}_{Dh}\hat{X}_{De}$. Note the chosen stabilizers $\hat{X}_{Di}\hat{X}_{Df}\hat{X}_{Dh}\hat{X}_{De}$ and $\hat{Z}_{Df}\hat{Z}_{De}\hat{Z}_{Dc}\hat{Z}_{Db}$ must commute with one another to force the projective measurement outcome of the system into a unique eigenstate of all the stabilizers. Table 4 shows the eigenstates of $\hat{X}_{Di}\hat{X}_{Df}\hat{X}_{Dh}\hat{X}_{De}$ and $\hat{Z}_{Df}\hat{Z}_{De}\hat{Z}_{Dc}\hat{Z}_{Db}$ operators with their corresponding eigenvalues. Suppose we have initialized the system in $|0000\rangle$ state. If an X error happens on Df data qubit, measuring the system using $\hat{Z}_{Df}\hat{Z}_{De}\hat{Z}_{Dc}\hat{Z}_{Db}$ stabilizer reports a change from eigenstate $|0000\rangle$ to $|1000\rangle$. Now if an X error happens on De, measuring the system using $\hat{Z}_{Df}\hat{Z}_{De}\hat{Z}_{Dc}\hat{Z}_{Db}$ stabilizer reports a change from $|0000\rangle$ to $|0100\rangle$. As it can be seen in Table 1.2, a single error in any data qubit can be uniquely specified as the measurement result lands on a specific eigenstate with eigenvalue -1.

The Surface Code error correction methodology on 2-dimensional (2D) array of nearest-neighbor (NN) qubits preserves the logical states of qubits. In the beginning of an error correction cycle, all the measurement qubits are initialized

Table 1.2 The set of eigenstates and corresponding eigenvalues for four-qubit stabilizers $\hat{X}_{Di}\hat{X}_{Df}\hat{X}_{Dh}\hat{X}_{De}$ *and* $\hat{Z}_{Df}\hat{Z}_{De}\hat{Z}_{Dc}\hat{Z}_{Db}$. *Note* $\hat{X}$*and* $\hat{Z}$*are Pauli operators acting on the data qubits as shown in Fig. 12 where the measure-X qubit Xc stabilizes* $\hat{X}_{Di}\hat{X}_{Df}\hat{X}_{Dh}\hat{X}_{De}$ *operator and the measure-Z qubit Zb stabilizes* $\hat{Z}_{Df}\hat{Z}_{De}\hat{Z}_{Dc}\hat{Z}_{Db}$ *operator.*

Eigenvalue	$\hat{X}_{Di}\hat{X}_{Df}\hat{X}_{Dh}\hat{X}_{De}$	$\hat{Z}_{Df}\hat{Z}_{De}\hat{Z}_{Dc}\hat{Z}_{Db}$
+1	$\|++++\rangle$	$\|0000\rangle$
+1	$\|++--\rangle$	$\|0011\rangle$
+1	$\|+--+\rangle$	$\|0110\rangle$
+1	$\|--++\rangle$	$\|1100\rangle$
+1	$\|-++-\rangle$	$\|1001\rangle$
+1	$\|+-+-\rangle$	$\|0101\rangle$
+1	$\|-+-+\rangle$	$\|1010\rangle$
+1	$\|----\rangle$	$\|1111\rangle$
-1	$\|+++-\rangle$	$\|0001\rangle$
-1	$\|++-+\rangle$	$\|0010\rangle$
-1	$\|+-++\rangle$	$\|0100\rangle$
-1	$\|-+++\rangle$	$\|1000\rangle$
-1	$\|+---\rangle$	$\|0111\rangle$
-1	$\|-+--\rangle$	$\|1011\rangle$
-1	$\|--+-\rangle$	$\|1101\rangle$
-1	$\|---+\rangle$	$\|1110\rangle$

to zero. When qubits are idle, the $\hat{Z}$ and $\hat{X}$ operators are repeatedly applied to measure all the measure-Z and measure-X ancillary qubits. If the state of any measure-Z/measure-X ancillary qubit is flipped in a cycle, a bit-flip/phase-flip error syndrome in adjacent data qubits is detected. If the state of a measurement ancillary qubit changes per two consecutive cycles, a measurement error syndrome is detected. Then a software is used to map the error syndromes to a graph which represents the error propagation model. Later a recovery operation is applied to restore the states. It is notable that the recovery operations are applied only in software by tracking error syndromes of all cycles, the software corrects the final data measurement by fixing, if necessary, the measured data [12].

In chapter 2 we further explain the Surface Code error syndrome detection circuits and present a new Surface Code memory architecture that realizes the required stabilizers utilizing our multi-qubit gates. Performing quantum computation in Surface Code is not in the scope of this work. The interested reader is referred to [13] for further information about Surface Code.

1.3 Simulation-based quantum gate design

In the absence of a physical quantum computer, one can design and optimize quantum gates utilizing a software that simulates the dynamics of the quantum system. There are three different formulations in quantum mechanics to study the evolution of quantum systems: the Schrödinger picture, the Heisenberg picture, and the Interaction picture (or Dirac picture). In the Schrödinger picture, the operators/observables are stationary, and the state vectors evolve in time. In the Heisenberg picture, the state vectors are considered stationary and the operators/observables evolve in time. In the Dirac picture, the Hamil-

tonian is separated to two parts, one which is time-independent and the other time-dependent. The time-independent portion of the Hamiltonian is used to evolve the state vector as in the Schrödinger picture, and the time-dependent portion of the Hamiltonian is used to evolve the observables/operators as in the Heisenberg picture [17]. Though these pictures are different mathematical formulations to represent the dynamics of the quantum system, using any of them should result in the same solution. In chapter 3 we explain our approach to use the Schrödinger picture to implement a quantum simulator.

To simulate the dynamics of a quantum system, a time-dependent Schrödinger equation needs to be solved. Knowing the Hamiltonian of the system H and the initial state $|\Psi(t_0)\rangle$, the time evolution of a quantum state is given by $|\Psi(t)\rangle = U(t)\,|\Psi(t_0)\rangle$ with $U(t) = e^{-iHt/\hbar}$ being the unitary transformation of the system. Throughout this dissertation, we consider $\hbar = 1$.

Any time dependent Hamiltonian that can be decomposed to m local interactions can be written as a summation of m local Hamiltonians [18-19]. This Hamiltonian can be efficiently simulated using a universal quantum computer [20]. If we decompose the Hamiltonian to m local non-commuting Hamiltonians, we can estimate the term $e^{\frac{-i}{\hbar}H(t)t}$ using Trotterization [18] as below.

$$e^{\frac{-i}{\hbar}Ht} \approx \left(e^{\frac{-i}{\hbar}H_1\frac{t}{n}}\; e^{\frac{-i}{\hbar}H_2\frac{t}{n}} \ldots e^{\frac{-i}{\hbar}H_m\frac{t}{n}}\right)^n \tag{1.17}$$

Where, the estimation can be accurate by choosing very small Trotter steps ($\frac{t}{n}$). The Trotterization methodology for quantum simulation can also be done in a classical computer for simulating the dynamics of a small quantum system. If we try to simulate a quantum system with many qubits using classical computational resources, we are limited by the memory and computational power. Once we have a simulator based on the Hamiltonian of our desired physical system, we can treat it as a framework for design, optimization, or test of new gates or

architectures.

1.3.1 Designing gates using an analytical approach

In chapter 2, we use an analytical approach to design multi-qubit controlled-unitary gates for quantum systems with always-on Ising interactions. We use the simulation environment as a test framework to check if the derived parameters result in the desired gate with high fidelity. Our analytical approach is based on a pulsed bias scheme [21] which uses a reduced Hamiltonian technique [22] to derive the required quantum system parameters. In this chapter, we show how the new multi-qubit gates can be applied in circuits implementing Surface Code quantum error correction cycles.

1.3.2 Designing gates using a heuristic approach

In chapter 3, we use the simulation environment as a design framework similar to an actual quantum hardware in an experimental lab. We present a heuristic methodology to design single-qubit and two-qubit gates in a quantum simulator which simulates the dynamics of the superconducting systems based on transmons in cQED. Then we apply the designed two-qubit gates to realize a quantum full-adder circuit. In NN architectures, in order to perform quantum gates between two non-neighbor qubits, we need to bring them adjacent to each other by applying extra gate operations named SWAP gates. This increases the gate counts, the complexity of the control electronics, the latency of the quantum circuits, and consequently increases the error rate. Therefore, we need to design an efficient mapping of a quantum circuit to the qubit layout such that the number of SWAP gates is minimized. In this chapter, we found the optimum mapping of the full-adder circuit on the qubit layout.

1.3.3 Designing gates using a machine learning approach

In chapter 4, we model the quantum gate design problem as a control/optimization problem and implement an intelligent simulation environment to design new three-qubit quantum gates for the superconducting systems based on transmons in cQED. Then we investigate the robustness of the new gates, and finally we apply these gates in a circuit to realize the Shor's nine-qubit error correction code [6] . The result of this chapter shows the benefit of applying multi-qubit gates in quantum error correction codes.

References

[1] M. A. Nielsen and I. L. Chuang. Quantum computation and quantum information. 2000.

[2] A. Peres. Reversible logic and quantum computers. Physical Rev. A., 32 (6):3266-3276, 1985.

[3] A. Barenco, C. H. Bennett, C. H. Cleve, D. P. DiVincenzo, N. Margolus, N. Shor, T. Sleator, J. A. Smolin, and H. Weinfurter. Elementary gates for quantum computation, Physical Review A. vol., 52, 1995.

[4] I. L. Markov 24. V. V. Shende. On the cnot-cost of toffoli gates. Quant. Inf. Comp., 9(5-6):461–486, 2009.

[5] Barbara M. Terhal. Quantum error correction for quantum memories. Rev. Mod.Phys. 87, 307, Apr 2015.

[6] Peter W. Shor. Scheme for reducing decoherence in quantum computer memory. Physical Review A, 52(4), Jan 1995.

[7] A. M. Steane. error correcting codes in quantum theory. Phys. Rev. Lett. 77, 793, July 1996.

[8] A. R. Calderbank and Peter W. Shor. Good quantum error-correcting codes exist. Physical Review A, 54(2):1098–1105, Jan 1996.

[9] Dave Bacon. Operator quantum error-correcting subsystems for self-correcting quantum memories. Physical Review A, 73(1), 2006.

[10] Eric Dennis. Topological quantum memory. Journal of Mathematical Physics, 43(4452), 2002.

[11] Daniel Eric Gottesman. arXiv:quant-ph/9705052v1, Caltech Ph.D. Thesis, May 1997.

[12] J. Kelly, R. Barends, A. G. Fowler, A. Megrant, E. Jeffrey, T. C. White, D. Sank, J. Y. Mutus, B. Campbell, Yu Chen, and et al. State preservation by repetitive error detection in a superconducting quantum circuit. Nature, 519(7541):66–69, 2015.

[13] Austin G. Fowler, Matteo Mariantoni, John M. Martinis, and Andrew N. Cleland. Surface codes: Towards practical large-scale quantum computation. Physical Review A, 86(3), 2012.

[14] John M Martinis. Qubit metrology for building a fault-tolerant quantum computer. npj Quantum Information, 1(1), 2015.

[15] L. S. Bishop. Circuit quantum electrodynamics ii. Yale University, 2010.

[16] Yu Tomita and Krysta M. Svore. Low-distance surface codes under realistic quantum noise. Physical Review A, 90(6), Nov 2014.

[17] A. Tokmakoff. Introductory quantum mechanics ii lecture notes. MIT Department of Chemistry, 2007.

[18] M h Yung, J. D. Whitfield, D. G. Tempel S. Boixo, and A. Aspuru-Guzik. Introduction to quantum algorithms for physics and chemistry. Advances in Chemical Physics, 21, March 2014.

[19] S. Lloyd. Universal quantum simulators. Science, 273(5278):1073–1078, 1996.

[20] R. P. Feynman. Int. J. Theor. Phys., 21 (6-7):467–488, 1982.

[21] E. C. Behrman J. E. Steck Z. Zhou P. K. Gagnebin, S. R. Skinner and S. Han. Quantum gates using a pulsed bias scheme. Phys. Rev. A, 72(042311), oct. 2005.

[22] P. Kumar, S. R. Skinner, and S. Daraeizadeh. Nearest-neighbor architecture to overcome effects of qubit precessions in gate operations. 2011 IEEE Congress of Evolutionary Computation (CEC), 2011.

2 Realization of Surface Code Quantum Memory on Systems with Always-On Interactions

Note: This chapter has been published: S. Daraeizadeh[1], S. Mostame[2], P. Kumar Eslami[3], X. Song[1], M. Perkowski[1], arXiv:1811.09011v3

We realize Surface Code quantum memories for nearest-neighbor qubits with always-on Ising interactions. This is done by utilizing multi-qubit gates that mimic the functionality of several gates. The previously proposed Surface Code memories rely on error syndrome detection circuits based on CNOT gates. In a two-dimensional planar architecture, to realize a two-qubit CNOT gate in the presence of couplings to other neighboring qubits, the interaction of the target qubit with its three other neighbors must cancel out. Here we present a new error syndrome detection circuit utilizing multi-qubit parity gates. In addition to speed up in the error correction cycles, in our approach, the depth of the error syndrome detection circuit does not grow by increasing the number of qubits in the logical qubit layout. We analytically design the system parameters to realize new five-qubit gates suitable for error syndrome detection in nearest-neighbor two-dimensional array of qubits. The five-qubit g ates a re designed such that the middle qubit is the target qubit and all four coupled neighbors are the control qubits. In our scheme, only one control parameter of the target

[1] Department of Electrical and Computer Engineering, Portland State University.
[2]IBM T.J. Watson Research Center, Yorktown Heights, NY 10598, USA
[3] Department of Electrical Engineering Computer Science, Wichita State University.

qubits must be adjusted to realize controlled-unitary operations. The gate operations are confirmed with a fidelity of >99.9% in a simulated system consists of nine nearest-neighbor qubits.

2.1 Introduction

One of the most important areas of research in the field of quantum computing is to design and implement highly efficient and fault-tolerant scalable quantum architectures. The quantum systems are intrinsically error-prone since the states of qubits can change by environmentally-induced errors. Therefore, to realize a quantum memory, it is required to apply Quantum Error Correction (QEC) schemes [1] to preserve the states of the qubits during idle times. One of the most promising QEC schemes is Surface Code [2]. The Surface Code architecture consists of Z and X stabilizers [3-4] and introduces ancillary qubits dedicated to these stabilizers. The code repetitively performs projective quantum non-demolition (QND) parity measurements on these ancillary qubits to measure the bit-flip and phase-flip errors of the data qubits [5]. The number of ancillary qubits in these measurements is approximately equal to the number of data qubits. Although it has been shown this approach results in storing information with a lower error rate, the Surface Code methodology has a high computational and resource overhead to realize the logical states and process the information. In this work, we propose a protocol to implement an efficient quantum memory based on Surface Code with applications in large scale 2-Dimensional (2D) nearest-neighbor (NN) quantum architectures with always-on interactions. This is possible due to our proposed five-qubit parity gates which can be applied in parallel on the entire array of qubits.

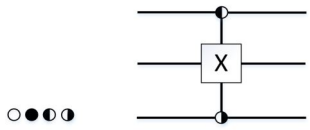
Parity gates can be used as an elementary gate in universal quantum com-

putation [6]. Kumar et. al. [7] designed a single-shot multi-qubit parity gate for quantum systems with Ising interactions. That can be utilized to generate efficient circuits for Mirror Inversion (MI) [8-10] as a sequence of controlled-unitary operations between 2D nearest-neighbor qubits with tunable couplings. This method significantly increases the efficiency by lowering the computational overhead since the state transfer can be achieved in fewer computational steps without requiring ancillary qubits. Furthermore, there is not any dephasing from idle qubits since all the qubits are used in the MI operation as target or control qubits. However, the method is limited to 2D systems with tunable couplings. Although it is easier to perform multi-qubit gates in the systems with tunable couplings, there are some disadvantages such as increased circuit complexity and more noise introduction. We generalize the previous approach to design five-qubit controlled-unitary gates to realize parity gates in 2D nearest neighbor layouts with always-on interactions.

In our model, each five-qubit parity gate consists of one target qubit which is coupled to four adjacent control qubits where each control qubit can act as an active control qubit or a dummy qubit. In the case of five-qubit parity gate with four active control qubits, the gate operates so that the state of the target qubit is flipped when the XOR of all four adjacent qubits is one. In other words, if the four adjacent control qubits have even or odd parity, the state of the middle target qubit is preserved or inverted, respectively. However, in the case of five-qubit parity gate with two active control qubits, the state of the target qubit flips when two adjacent qubits (two active control qubits) have odd parity, while the other two adjacent qubits act as dummy qubits and have no effect on the gate operation.

Here we introduce a new symbol to represent the multi-qubit parity gates.

As it is known, the symbol of a full-colored circle on a control qubit means when the logical state of control qubit is 1, the gate operation is performed on the target qubit. While the symbol of an empty circle means when the logical state of control qubit is 0, the gate operation is performed on the target qubit. We introduce the half-colored circles as shown in Fig. 2.1 (a) which means the logical state of the control qubit can be either 1 or 0. The half-colored circles are meaningful when applied in pairs to represent the opposite states of two control qubits resulting in a gate operation on target qubit. For example, in Fig. 2.1 (b) there is a left-half colored circle on top control qubit, while on the bottom control qubit there is a right-half colored circle. This means the two pairs of control qubits must be in opposite states for the target qubit to change its state (parity detection).



(a) (b)

Figure 2.1 (a) Notations to represent the state of the control qubits in controlled unitary operations (b) A 3-qubits parity gate. Half colored circles can be used in pairs to represent the parity of the states of a pair of control qubits

2.2 Physical model and the simulation method

To simulate the dynamics of a quantum system, a time-dependent Schrödinger equation needs to be solved. Knowing the Hamiltonian of the system H and the initial state $|\Psi(t_0)\rangle$, the time evolution of a quantum state is given by $|\Psi(t)\rangle = U(t)\,|\Psi(t_0)\rangle$ with $U(t) = e^{-iHt/\hbar}$ being the unitary transformation of the system. Throughout this paper, we consider $\hbar = 1$.

Consider a $(m \times n)$ two-dimensional system of qubits with always-on nearest-neighbor Ising interactions. Such a system can be described by the following Hamiltonian [10-13], where the qubits are labeled with j and k, for the rows and columns, respectively.

$$\begin{aligned} H &= \sum_{k=1}^{n} \sum_{j=1}^{m} \left(\Delta_{j,k}\, \sigma_x^{j,k} + \varepsilon_{j,k}\, \sigma_z^{j,k} \right) + \\ &+ \sum_{k=1}^{n} \sum_{j=1}^{m-1} \xi_{j,j+1}^{k}\, \sigma_z^{j,k}\, \sigma_z^{j+1,k} + \sum_{k=1}^{n-1} \sum_{j=1}^{m} \xi_{j}^{k,k+1}\, \sigma_z^{j,k}\, \sigma_z^{j,k+1}, \end{aligned} \tag{2.1}$$

where σ_x and σ_z are Pauli operators, $\Delta_{j,k}$ is the tunneling energy for the qubit located at the j-th row and k-th column, and $\varepsilon_{j,k}$ is the bias energy for the qubit. Here $\xi_{j,j+1}^{k}$ is the coupling energy between two adjacent vertically coupled qubits in column k. Similarly, $\xi_{j}^{k,k+1}$ is the coupling energy between two adjacent horizontally coupled qubits in row j. The Hamiltonian operator is a $2^{m \times n} \times 2^{m \times n}$ matrix, which scales exponentially with the number of qubits in the system. It is challenging to solve such a large matrix analytically in order to derive the system parameters. However, using a pulses bias scheme [14-15] and reduced Hamiltonian technique [16], we can solve the system parameters to realize a desired multi-qubit parity gate.

We consider a system of nine qubits as depicted in the black square in Fig. 2.2, where each qubit is interacting with 4 neighbors. We design a controlled-unitary gate where qubits A, B, C, and D are control qubits and T is the target qubit. In the architecture shown in Fig. 2.2, we consider four coupling strengths ξ_{A}, ξ_{B}, ξ_{C}, and ξ_{D} respectively between the target qubit T and the control qubits A, B, C, and D. By design, the coupling strengths between pairs of qubits are alternating in a row or column of the two-dimensional array of qubits. Therefore, if any qubit in the array be selected as the Target qubit, it is interacting with four

neighbors with four distinct coupling strength.

The evolution of a nine-qubit system, qubits A, B, C, D, E, F, G, H and T in Fig. 2.2, is described by a 512×512 Hamiltonian matrix. Qubits E, F, G, and H have direct interactions with control qubits but do not have any direct interaction with the target qubit. In order to study their impact on the dynamics of the system, we applied the same bias parameters on E, F, G, and H qubits as for control qubits A, B, C, and D. We observed that they do not affect the 5-qubit gate operation nor the gate operation affects the state of these qubits. Therefore, to find the parameters of a five-qubit gate operation on A, B, C, D and T, we analyze a 32×32 Hamiltonian matrix. Using the reduced Hamiltonian scheme [16-17], we break this Hamiltonian matrix to sixteen 2×2 Hamiltonian matrices. Each 2×2 Hamiltonian describes evolution of the target qubit T in a subspace depending on the states of the control qubits. Then for each of these 2×2 Hamiltonians, we generate a unitary matrix by integrating the Schrödinger equation, and then equating the generated unitary matrix to a desirable controlled unitary gate operation for that subspace. Next, we describe this in details.

The evolution of the target qubit T being directly coupled to the control qubits A, B, C, and D can be described by the reduced Hamiltonian:

$$\begin{aligned} H_{\text{red}} &= \Delta_{\text{T}}\,\sigma_x^{\text{T}} + \Big\{\varepsilon_{\text{T}} + \xi_{\text{A}}\langle\Phi|\sigma_z^{\text{A}}|\Phi\rangle + \xi_{\text{B}}\langle\Phi|\sigma_z^{\text{B}}|\Phi\rangle \\ &+ \xi_{\text{C}}\langle\Phi|\sigma_z^{\text{C}}|\Phi\rangle + \xi_{\text{D}}\langle\Phi|\sigma_z^{\text{D}}|\Phi\rangle\Big\}\,\sigma_z^{\text{T}} \end{aligned} \tag{2.2}$$

where the label "red" stands for the reduced evolution subspace and $|\Phi\rangle$ represents the initial state of four control qubits A, B, C, and D, each is initialized to $|0\rangle$ or $|1\rangle$. The parameters of the Hamiltonian are the same as Eq. (2.1) but for simplicity, we have dropped some labels for the 5-qubits system shown in Fig. 2.2: Δ_{T}, ε_{T} are the tunneling energy and the bias energy for the target

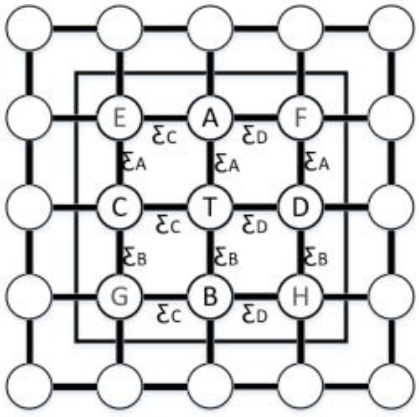


Figure 2.2 In non-tunable coupling systems, each qubit is interacting with 4 neighbor qubits. In this figure, we are interested to perform a Parity gate on the target qubit T where it is directly coupled to qubits A, B, C, and D.

qubit, respectively, and ξ_i is the coupling energy between control qubit "i" and the target qubit, where i = A, B, C, D. Depending on the initial state of $|\Phi\rangle$, the expectation value of σ_z^i can be +1 or -1. Note that here we have ignored the effect of the next-nearest-neighbor couplings. The qubits E, F, G, and H do not contribute to the evolution of the target qubit T as they don't have any direct coupling with T. In our simulations, the biases on non-interacting qubits E, F, G, and H are set such that their states are preserved.

Given the above reduced Hamiltonian, the unitary transformation on target qubit in terms of the system parameters can be derived as

$$U(t) = e^{i\theta} \begin{bmatrix} \cos(\omega t) - \frac{2\pi i E}{\omega}\sin(\omega t) & \frac{2\pi(-i\Delta_T)}{\omega}\sin(\omega t) \\ \\ \frac{2\pi(-i\Delta_T)}{\omega}\sin(\omega t) & \cos(\omega t) + \frac{2\pi i E}{\omega}\sin(\omega t) \end{bmatrix}, \tag{2.3}$$

with E being the effective bias and $\omega = 2\pi\sqrt{\Delta_T^2 + E^2}$ being the angular momentum of the gate operation. Here θ is a global phase factor.

Designing multiple-controlled unitary gates in a system with always-on in-

teractions requires careful attention to the connectivity or couplings between the qubits. To design a controlled-unitary gate where the target qubit is interacting with a set of neighbors but only a subset of neighbors have a control role; one needs to cancel out the effect of those neighbors who do not have a control role. For instance, in Fig. 2.2 consider designing a CNOT gate between qubits A and T, where A is the control qubit and T is the target qubit. Here, the qubits B, C, and D have direct interaction with qubit T but do not have a control role. Therefore, we need to design a five-qubit controlled-unitary gate with one target qubit T, one active control qubit A, and three dummy qubits B, C, and D. Note that the states of the dummy qubits should not effect the CNOT gate operation between A and T. Since there are 8 logical states $(000, 001, ..., 111)$ associated with the dummy qubits, to achieve the desired CNOT gate, one needs to realize a sequence of 8 five-qubit controlled-unitary operations, each taking the duration of τ. Where each five-qubit controlled-unitary operation configures T as the target qubit, A as the control qubit with logical state 1, and B, C, and D as the control qubits with one of the 8 logical states [16].

In most of the error correction codes such as Repetition Code [18], the bit-flip error syndrome detection circuit uses a sequence of two CNOT gates applied on two data qubits as control qubits and one measurement qubit as the target qubit. In a 2-dimensional system with always-on interaction, this results in a decomposition to a sequence of sixteen controlled-unitary operations (16τ). Utilizing the five-qubit parity gates with two active control qubits, we realize the same functionality while reducing the circuit depth to a sequence of only three controlled-unitary operations (3τ).

2.3 Five-qubit parity gates with two active control qubits

In this section we design a five-qubit parity gate in a 2D array of qubits where only two of the four control qubits have an active effect while the effects of two other control qubits are canceled. As discussed above and shown in Fig. 2.2, qubit T is the target qubit. Our goal here is to apply a parity gate to detect the parity of qubits A and B which are vertically coupled to the target qubit T. Therefore, we perform an X unitary operation on the target qubit $U_{\mathrm{T}} = X$ in the subspaces $Q_{\mathrm{A}}Q_{\mathrm{B}} = |10\rangle$ and $Q_{\mathrm{A}}Q_{\mathrm{B}} = |01\rangle$, irrespective of the states of the qubits C, and D. In the subspaces where $Q_{\mathrm{A}}Q_{\mathrm{B}} = |00\rangle$ or $Q_{\mathrm{A}}Q_{\mathrm{B}} = |11\rangle$ we will perform an Identity unitary operation on the target qubit $U_{\mathrm{T}} = I$. This is done by applying a sequence of four controlled-unitary gates as shown in Fig. 2.3 (a). Similarly, the circuit shown in Fig. 2.3 (b) realizes a parity detector gate where the qubits C and D are the two actively effective control qubits.

In Fig. 2.3 (a) and (b), the two gates located in the middle can be combined into one gate which operates on the target qubit if the qubits A vs B and C vs D are in different states. We can represent this gate with four half colored circles on the control qubits, where (A, B) and (C, D) are considered the pairs with opposite half-colored circles as shown in Fig. 2.4, where the pairs of half-colored circles are color-coded.

Note we consider only the qubits that have direct coupling with the target T. In Table 2.1 we list the effective bias in all 16 possible subspaces of $Q_{\mathrm{A}}Q_{\mathrm{B}} = |10\rangle$, $Q_{\mathrm{A}}Q_{\mathrm{B}} = |01\rangle$, $Q_{\mathrm{A}}Q_{\mathrm{B}} = |00\rangle$, and $Q_{\mathrm{A}}Q_{\mathrm{B}} = |11\rangle$, where qubits C, and D, have arbitrary values.

In the subspace $Q_{\mathrm{A}}Q_{\mathrm{B}} = |10\rangle$, where qubits C, and D have arbitrary values, the effective bias is $E = \varepsilon_{\mathrm{T}} - \xi_{\mathrm{A}} + \xi_{\mathrm{B}} \pm \xi_{\mathrm{C}} \pm \xi_{\mathrm{D}}$. To realize an X operation, we need to cancel the diagonal terms in Eq. (2.3), and force $\sin(\omega t) = 1$ and $\omega = 2\pi\Delta_{\mathrm{T}}$

Table 2.1 Effective bias under each subspace when qubit T is coupled to four neighbor qubits A, B, C, D.

	AB	CD	Effective Bias
0	$\|00\rangle$	$\|00\rangle$	$E = \varepsilon_T + \xi_{\mathrm{A}} + \xi_{\mathrm{B}} + \xi_{\mathrm{C}} + \xi_{\mathrm{D}}$
1	$\|00\rangle$	$\|01\rangle$	$E = \varepsilon_T + \xi_{\mathrm{A}} + \xi_{\mathrm{B}} + \xi_{\mathrm{C}} - \xi_{\mathrm{D}}$
2	$\|00\rangle$	$\|10\rangle$	$E = \varepsilon_T + \xi_{\mathrm{A}} + \xi_{\mathrm{B}} - \xi_{\mathrm{C}} + \xi_{\mathrm{D}}$
3	$\|00\rangle$	$\|11\rangle$	$E = \varepsilon_T + \xi_{\mathrm{A}} + \xi_{\mathrm{B}} - \xi_{\mathrm{C}} - \xi_{\mathrm{D}}$
4	$\|01\rangle$	$\|00\rangle$	$E = \varepsilon_T + \xi_{\mathrm{A}} - \xi_{\mathrm{B}} + \xi_{\mathrm{C}} + \xi_{\mathrm{D}}$
5	$\|01\rangle$	$\|01\rangle$	$E = \varepsilon_T + \xi_{\mathrm{A}} - \xi_{\mathrm{B}} + \xi_{\mathrm{C}} - \xi_{\mathrm{D}}$
6	$\|01\rangle$	$\|10\rangle$	$E = \varepsilon_T + \xi_{\mathrm{A}} - \xi_{\mathrm{B}} - \xi_{\mathrm{C}} + \xi_{\mathrm{D}}$
7	$\|01\rangle$	$\|11\rangle$	$E = \varepsilon_T + \xi_{\mathrm{A}} - \xi_{\mathrm{B}} + \xi_{\mathrm{C}} - \xi_{\mathrm{D}}$
8	$\|10\rangle$	$\|00\rangle$	$E = \varepsilon_T - \xi_{\mathrm{A}} + \xi_{\mathrm{B}} + \xi_{\mathrm{C}} + \xi_{\mathrm{D}}$
9	$\|10\rangle$	$\|01\rangle$	$E = \varepsilon_T - \xi_{\mathrm{A}} + \xi_{\mathrm{B}} + \xi_{\mathrm{C}} - \xi_{\mathrm{D}}$
10	$\|10\rangle$	$\|10\rangle$	$E = \varepsilon_T - \xi_{\mathrm{A}} + \xi_{\mathrm{B}} - \xi_{\mathrm{C}} + \xi_{\mathrm{D}}$
11	$\|10\rangle$	$\|11\rangle$	$E = \varepsilon_T - \xi_{\mathrm{A}} + \xi_{\mathrm{B}} - \xi_{\mathrm{C}} - \xi_{\mathrm{D}}$
12	$\|11\rangle$	$\|00\rangle$	$E = \varepsilon_T - \xi_{\mathrm{A}} - \xi_{\mathrm{B}} + \xi_{\mathrm{C}} + \xi_{\mathrm{D}}$
13	$\|11\rangle$	$\|01\rangle$	$E = \varepsilon_T - \xi_{\mathrm{A}} - \xi_{\mathrm{B}} + \xi_{\mathrm{C}} - \xi_{\mathrm{D}}$
14	$\|11\rangle$	$\|10\rangle$	$E = \varepsilon_T - \xi_{\mathrm{A}} - \xi_{\mathrm{B}} - \xi_{\mathrm{C}} + \xi_{\mathrm{D}}$
15	$\|11\rangle$	$\|11\rangle$	$E = \varepsilon_T - \xi_{\mathrm{A}} - \xi_{\mathrm{B}} - \xi_{\mathrm{C}} - \xi_{\mathrm{D}}$

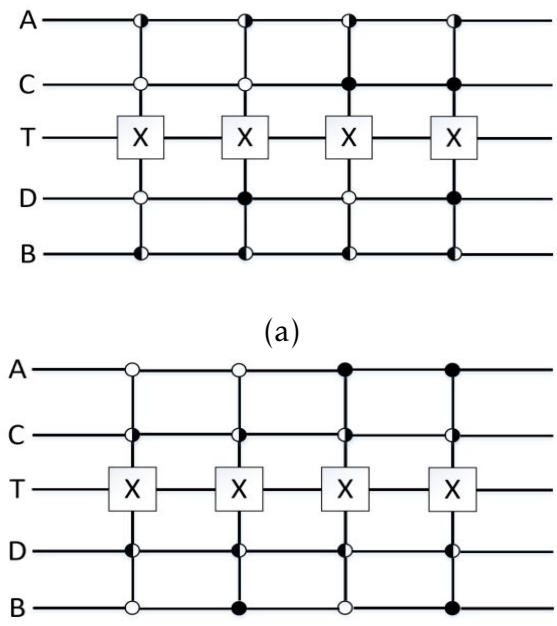


Figure 2.3 The circuits to realize parity gates with only two active vertical (a) or horizontal (b) control qubits.

which results in $-2\pi i\,\Delta_{\mathrm{T}}\sin(\omega t)/\omega = -i$, where $-i$ contributes as a phase factor of $3\pi/2$ on the target qubit. This extra phase on the target qubit can be tracked in the course of computation. The following conditions must be satisfied:

$$\cos(\omega t) - \frac{2\pi i E}{\omega}\sin(\omega t) = \cos(\omega t) + \frac{2\pi i E}{\omega}\sin(\omega t) = 0$$
$$\Longrightarrow E = 0, \quad \cos(\omega t) = 0 \tag{2.4}$$

Considering $t = \tau$ for the X operation time, we need to satisfy condition: $\omega\tau = (4n+1)\pi/2$, where n is an integer. For rf-SQUID (Superconducting Quantum Interference Devices) qubit systems, one set of parameters which satisfies the conditions above would be $\Delta_{\mathrm{T}} = 25$ MHz, $n = 0$, and $\tau = 10$ ns, while bias pulse magnitude can range up to 10 GHz [7,14,16]. Canceling out the effective bias ($E = 0$) we would also need the condition:

$$\varepsilon_{\mathrm{T}} = \xi_{\mathrm{A}} - \xi_{\mathrm{B}} \pm \xi_{\mathrm{C}} \pm \xi_{\mathrm{D}}\,. \tag{2.5}$$

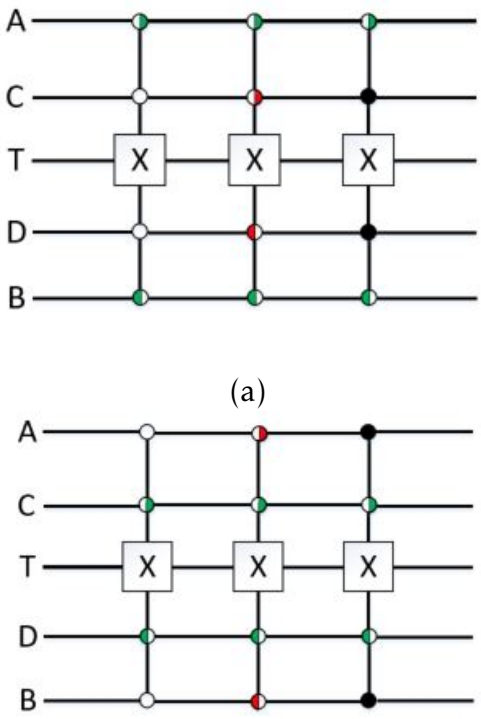


Figure 2.4 (a) Two active vertical control qubits. The red half-colored circles represent the different states of qubits C, and D. (b) Two active horizontal control qubits. The red half-colored circles represent the different states of qubits A, and B. The same principle applies to green half-colored circles.

As shown in Table 2.1, the effective bias in Eq. (2.5) – subspace $Q_A Q_B = |10\rangle$ – expands to four subspaces depending on the state of $Q_C Q_D$:

$$\begin{aligned} Q_C Q_D &= |00\rangle,\ \varepsilon_T = \xi_A - \xi_B - \xi_C - \xi_D \\ Q_C Q_D &= |01\rangle,\ \varepsilon_T = \xi_A - \xi_B - \xi_C + \xi_D \\ Q_C Q_D &= |10\rangle,\ \varepsilon_T = \xi_A - \xi_B + \xi_C - \xi_D \\ Q_C Q_D &= |11\rangle,\ \varepsilon_T = \xi_A - \xi_B + \xi_C + \xi_D . \end{aligned} \tag{2.6}$$

By choosing $\xi_A = \xi_B$ and $\xi_C = \xi_D$, for $Q_C Q_D = |01\rangle, |10\rangle$ we get $\varepsilon_T = 0$, while for $Q_C Q_D = |00\rangle, |11\rangle$ we have $\varepsilon_T = -\xi_C - \xi_D = -2\xi_D$, and $\varepsilon_T = \xi_C + \xi_D = 2\xi_D$.

A similar calculation can be done for subspace $Q_A Q_B = |01\rangle$ with the same results for the bias on the target qubit (ε_T). Therefore to realize an X operation on the target qubit, we keep biases on all control qubits at some arbitrary value such that it would not cancel the effect of couplings [16] $\varepsilon_A = \varepsilon_B = \varepsilon_C = \varepsilon_D = 2$ GHz, and apply a sequence of bias pulse steps on the target qubit as following, with each of them taking $\tau = 10$ ns:

$$\varepsilon_{T_1} = -\xi_C - \xi_D, \quad \varepsilon_{T_2} = 0, \quad \varepsilon_{T_3} = \xi_C + \xi_D \tag{2.7}$$

where ε_{T_i} represents the i-th bias magnitude on target qubit T. The order of applying these three pulse steps does not matter, since at the end after 30 ns, the desired gate operation has been realized. Table 2.2 summarizes all possible effective biases in each subspace under the three pulse steps given by Eq. (2.7). This table is derived by substituting the bias magnitude of the target qubit under each pulse step (ε_{T_1}, ε_{T_2}, ε_{T_3}) in the effective bias E given in Table 2.1.

Then to perform an X operation in subspaces where $Q_A Q_B = |10\rangle$ and $Q_A Q_B = |01\rangle$, we set the coupling values such that the effective bias is canceled out under one of the three pulse steps (ε_{T_1}, ε_{T_2}, ε_{T_3}) while an Identity operation is realized elsewhere (see Table 2.2). For all other subspaces where $Q_A Q_B = |00\rangle$ and

Table 2.2 Effective bias in each subspace under each pulse sequence

AB	CD	$\varepsilon_{T_1} = -\xi_C - \xi_D$	$\varepsilon_{T_2} = 0$	$\varepsilon_{T_3} = \xi_C + \xi_D$
$\|00\rangle$	$\|00\rangle$	$E = +\xi_A + \xi_B$	$E = \xi_A + \xi_B + \xi_C + \xi_D$	$E = \xi_A + \xi_B + 2\xi_C + 2\xi_D$
$\|00\rangle$	$\|01\rangle$	$E = \xi_A + \xi_B - 2\xi_D$	$E = \xi_A + \xi_B + \xi_C - \xi_D$	$E = \xi_A + \xi_B + 2\xi_C$
$\|00\rangle$	$\|10\rangle$	$E = \xi_A + \xi_B - 2\xi_C$	$E = \xi_A + \xi_B - \xi_C + \xi_D$	$E = \xi_A + \xi_B + 2\xi_D$
$\|00\rangle$	$\|11\rangle$	$E = \xi_A + \xi_B - 2\xi_C - 2\xi_D$	$E = \xi_A + \xi_B - \xi_C - \xi_D$	$E = +\xi_A + \xi_B$
$\|01\rangle$	$\|00\rangle$	$E = +\xi_A - \xi_B$	$E = \xi_A - \xi_B + \xi_C + \xi_D$	$E = \xi_A - \xi_B + 2\xi_C + 2\xi_D$
$\|01\rangle$	$\|01\rangle$	$E = \xi_A - \xi_B - 2\xi_D$	$E = \xi_A - \xi_B + \xi_C - \xi_D$	$E = \xi_A - \xi_B + 2\xi_C$
$\|01\rangle$	$\|10\rangle$	$E = \xi_A - \xi_B - 2\xi_C$	$E = \xi_A - \xi_B - \xi_C + \xi_D$	$E = \xi_A - \xi_B + 2\xi_D$
$\|01\rangle$	$\|11\rangle$	$E = \xi_A - \xi_B - 2\xi_C - 2\xi_D$	$E = \xi_A - \xi_B - \xi_C - \xi_D$	$E = +\xi_A - \xi_B$
$\|10\rangle$	$\|00\rangle$	$E = -\xi_A + \xi_B$	$E = -\xi_A + \xi_B + \xi_C + \xi_D$	$E = -\xi_A + \xi_B + 2\xi_C + 2\xi_D$
$\|10\rangle$	$\|01\rangle$	$E = -\xi_A + \xi_B - 2\xi_D$	$E = -\xi_A + \xi_B + \xi_C - \xi_D$	$E = -\xi_A + \xi_B + 2\xi_C$
$\|10\rangle$	$\|10\rangle$	$E = -\xi_A + \xi_B - 2\xi_C$	$E = -\xi_A + \xi_B - \xi_C + \xi_D$	$E = -\xi_A + \xi_B + 2\xi_D$
$\|10\rangle$	$\|11\rangle$	$E = -\xi_A + \xi_B - 2\xi_C - 2\xi_D$	$E = -\xi_A + \xi_B - \xi_C - \xi_D$	$E = -\xi_A + \xi_B$
$\|11\rangle$	$\|00\rangle$	$E = -\xi_A - \xi_B$	$E = -\xi_A - \xi_B + \xi_C + \xi_D$	$E = -\xi_A - \xi_B + 2\xi_C + 2\xi_D$
$\|11\rangle$	$\|01\rangle$	$E = -\xi_A - \xi_B - 2\xi_D$	$E = -\xi_A - \xi_B + \xi_C - \xi_D$	$E = -\xi_A - \xi_B + 2\xi_C$
$\|11\rangle$	$\|10\rangle$	$E = -\xi_A - \xi_B - 2\xi_C$	$E = -\xi_A - \xi_B - \xi_C + \xi_D$	$E = -\xi_A - \xi_B + 2\xi_D$
$\|11\rangle$	$\|11\rangle$	$E = -\xi_A - \xi_B - 2\xi_C - 2\xi_D$	$E = -\xi_A - \xi_B - \xi_C - \xi_D$	$E = -\xi_A - \xi_B$

$Q_A Q_B = |11\rangle$, we want to achieve Identity operation under all three pulse steps (ε_{T_1}, ε_{T_2}, ε_{T_3}). By choosing $\xi_A = \xi_B$ and $\xi_C = \xi_D$, most of the equations in Table 2.2 simplify or cancel out and only 7 effective bias equations remain which are listed below

$$\begin{aligned} E &= 2\xi_B, \quad E = 2\xi_D, \quad E = 4\xi_D, \quad E = 2\xi_B + 2\xi_D \\ E &= 2\xi_B - 2\xi_D, \quad E = 2\xi_B + 4\xi_D, \quad E = 2\xi_B - 4\xi_D \end{aligned} \tag{2.8}$$

Under the above effective biases, we like to achieve an Identity gate operation. Therefore, we should choose the coupling values such that the off-diagonal terms in Eq. 2.3 are zero and diagonal terms are 1. This results in

$$cos(\omega t) = 1 \quad \Rightarrow \quad \omega = \frac{2\pi n}{\tau} \tag{2.9}$$

where n is an integer. For $\xi \gg \Delta_T$, we can ignore Δ_T^2 in $\omega = 2\pi\sqrt{\Delta_T^2 + E^2}$, which results in $\omega = 2\pi E = 2\pi n/\tau$. Therefore, we choose the effective biases in the equations above as multiples of some integers τ such that

$$\begin{aligned} &2\xi_B = 2\frac{v}{\tau}, \quad 2\xi_D = 2\frac{w}{\tau}, \quad 4\xi_D = 4\frac{w}{\tau} \quad \Rightarrow \\ &2\xi_B \pm 2\xi_D = 2\frac{v \pm w}{\tau}, \quad 2\xi_B \pm 4\xi_D = 2\frac{v \pm 2w}{\tau} \end{aligned} \tag{2.10}$$

where v and w are integers. One set of values for a system with tunneling energy $\Delta_T = 25$ MHz and $\tau = 10$ ns are the coupling values $\xi_A = \xi_B = 0.6$ GHz and $\xi_C = \xi_D = 0.4$ GHz. The above set of parameters realizes a parity gate that detects the parity of qubits A vs B (vertical) while ignoring the states of D and C (horizontal). Similar calculations can be done to design a parity gate that detects the parity of qubits D vs C (horizontal) while ignoring the states of A and B (vertical). Here, we would like to perform an X unitary operation on the target qubit ($U_T = X$) in subspaces $Q_C Q_D = |10\rangle$ and $Q_C Q_D = |01\rangle$, no matter

what the states of qubits A and B are. In all other subspaces where $Q_C Q_D = |00\rangle$ or $Q_C Q_D = |11\rangle$, we want to perform an Identity operation on qubit T ($U_T = I$). Using the set of parameters as discussed previously, we can apply a bias pulse on the target qubit with the following three magnitudes, each taking $\tau = 10$ ns

$$\varepsilon_{T_1} = -\xi_A - \xi_B, \quad \varepsilon_{T_2} = 0, \quad \varepsilon_{T_3} = \xi_A + \xi_B \tag{2.11}$$

2.4 Five-qubit parity gate with four active control qubits

Now consider the case where the target qubit detects the even or odd parity of the four control qubits. For the target qubit to flip when the parity of four control qubits is odd, we need to treat all four control qubits equally, therefore, we set the coupling values connected to the target qubit equal $\xi_A = \xi_B = \xi_C = \xi_D = \xi$. To realize an Identity operation on subspaces when the parity of four control qubits is even (subspaces in rows 0, 3, 5, 6, 9, 10, 12, and 15 from Table 2.1), the effective bias on the target qubit must be chosen such that the angular frequency of the target qubit equals an integer multiple of 2π over the Identity operation duration, say $\omega = 2\pi E = 2\pi n/\tau$, with n being an integer. This results in

$$E = \varepsilon_T = \frac{v}{\tau}, \quad E = \varepsilon_T \pm 4\xi = \frac{w}{\tau} \tag{2.12}$$

where v and w are integers. Using the same parameters derived in the previous section for the initial bias $\varepsilon_T = 2$ GHz and tunneling $\Delta_T = 25$ MHz, $\tau = 10$ ns, and $\xi = 0.4$ GHz the conditions above are met. That would be true even if we change the coupling strength to $\xi = 0.6$ GHz.

In order to realize an X operation on the target qubit, the effective bias is set to zero on the desired subspaces (rows 1, 2, 4, 7, 8, 11, 13, and 14 in Table 2.1).

This results in $E = \varepsilon_T \pm 2\xi = 0$ leading to

$$\varepsilon_{T_1} = 2\xi, \quad \varepsilon_{T_2} = -2\xi. \tag{2.13}$$

Note that this gate operation also results in a phase factor $3\pi/2$ on the target qubit which can be tracked in the course of computation. As we showed the multi-qubit gate that detects the parity of four can be performed in a sequence of two controlled-unitary operations (2τ) which is faster than detecting the parity of two out of four which takes a sequence of three controlled-unitary operations (3τ).

2.5 Surface Code error syndrome detection based on multi-qubit gates

In Surface Code scheme [5], a 2D array of physical qubits is constructed with interleaving data qubits and measurement qubits called measure-Z and measure-X ancillary qubits, and a methodology is presented to protect the architecture from both bit-flip and phase-flip errors at the same time. The measure-X and measure-Z qubits detect phase-flip and bit-flip parities, respectively. As shown in Fig. 2.5, each data qubit in Surface Code is surrounded with 4 measurement qubits while each measurement qubit is surrounded with 4 data qubits. At the start, all measurement qubits are initialized to zero. At each error correction cycle, we perform measurements only on ancillary measurement qubits which stabilize the data qubits. Note that the states of data qubits are not perturbed by the measurement. A software maps the detected error syndromes (bit-flip, phase-flip, measurement error) to a graph model which keeps track of errors and fixes the errors [5,18].

For instance, in Fig. 2.5, the qubit Zb forces the data qubits Df, De, Dc, and Db to an eigenstate of $\hat{Z}_{Df}\hat{Z}_{De}\hat{Z}_{Dc}\hat{Z}_{Db}$ operator, while the qubit Xc forces

the data qubits Di, Df, Dh, and De to an eigenstate of $\hat{X}_{\mathrm{Di}}\hat{X}_{\mathrm{Df}}\hat{X}_{\mathrm{Dh}}\hat{X}_{\mathrm{De}}$ operator. Note the chosen stabilizer operators $\hat{X}_{\mathrm{Di}}\hat{X}_{\mathrm{Df}}\hat{X}_{\mathrm{Dh}}\hat{X}_{\mathrm{De}}$ and $\hat{Z}_{\mathrm{Df}}\hat{Z}_{\mathrm{De}}\hat{Z}_{\mathrm{Dc}}\hat{Z}_{\mathrm{Db}}$ must commute with one another to force the projective measurement outcome of the system into a unique eigenstate of all the stabilizers. Moreover, the order of applying $\hat{X}$ and $\hat{Z}$ operators on data qubits is important. The order must be chosen to ensure that we are not measuring the result of $\hat{X}$ and $\hat{Z}$ operators of any data qubit simultaneously. Failure to keep the commutation relationship of neighbor stabilizers results in random measurements [5]. In our example, the order of $\hat{X}$ and $\hat{Z}$ in $\hat{X}_{\mathrm{Di}}\hat{X}_{\mathrm{Df}}\hat{X}_{\mathrm{Dh}}\hat{X}_{\mathrm{De}}$ and $\hat{Z}_{\mathrm{Df}}\hat{Z}_{\mathrm{De}}\hat{Z}_{\mathrm{Dc}}\hat{Z}_{\mathrm{Db}}$ guarantees that the two stabilizers are commuting as well as the shared data qubits Df and De between the two stabilizer types $\hat{X}_{\mathrm{Di}}\hat{X}_{\mathrm{Df}}\hat{X}_{\mathrm{Dh}}\hat{X}_{\mathrm{De}}$ and $\hat{Z}_{\mathrm{Df}}\hat{Z}_{\mathrm{De}}\hat{Z}_{\mathrm{Dc}}\hat{Z}_{\mathrm{Db}}$ are interacting with one ancilla qubit of a type (Xc or Zb) at a time. This ensures the robustness of Surface Code to ancilla errors [19].

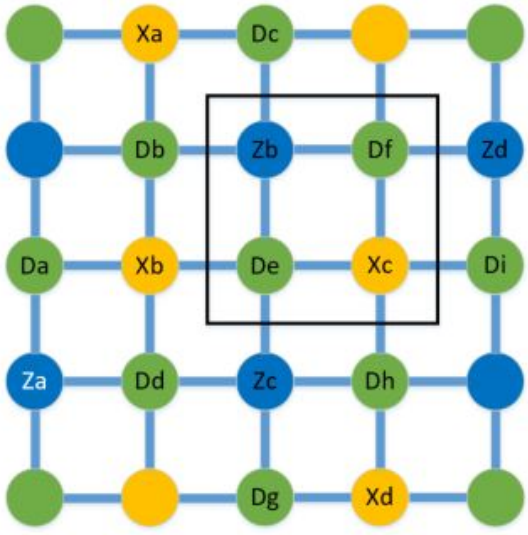


Figure 2.5 A 2D array of qubits with nearest-neighbor couplings forming a Surface-17 planar code logical qubit. Here 17 physical qubits (labeled) are required to form a logical qubit, 9 of which are data qubits and 8 of them are measurement ones. The box shows two data qubits De and Df in green, one measure-Z qubit Zb in blue and one measure-X qubit Xc in orange.

The quantum circuits to detect bit-flip error or phase-flip error during one cycle of Surface Code are based on applying CNOT gates. In some systems, one could perform a CNOT gate on any pair of neighboring qubits while the unwanted couplings to the other neighboring qubits are shut off or sufficiently detuned such that their interaction with the target qubit can be neglected. However, in systems with always-on interactions, the coupling values can not be tuned or shut off during the error syndrome detection and performing CNOT gates are costly. Here we consider designing new multi-qubit gates to facilitate error syndrome detection in such systems. In this section, we discuss different scenarios to realize Surface Code error correction for systems with always-on interactions using the introduced multi-qubit gates.

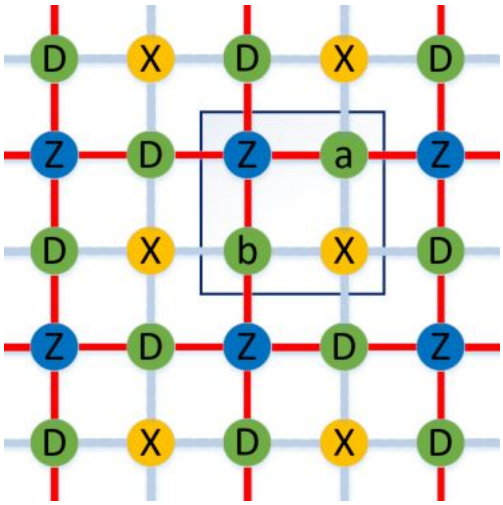


Figure 2.6 A 2D array of Surface Code where each pair of data qubits share a measure-X qubit and a measure-Z qubit as shown in the square box. All measure-Z qubits are coupled to data qubits with the same coupling strength shown in red, while all measure-X qubits are coupled to data qubits with the coupling strength shown in light grey.

Consider a large fabric of Surface Code with the proposed architecture shown in Fig. 2.6. Here, all measure-Z qubits are coupled to the surrounded data qubits using the same coupling strength $\xi_A = 0.4$ GHz and all measure-X qubits

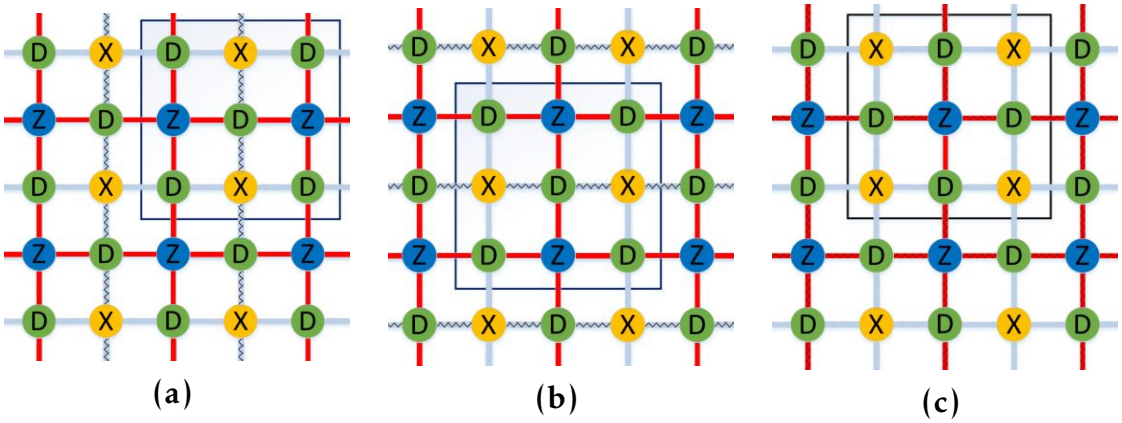


Figure 2.7 A two-dimensional array of Surface Code. (a) Applying multi-qubit X operators on all vertical columns in 2D array of qubits shown in dotted lines. (b) Applying multi-qubit X operator on all horizontal rows in 2D array of qubits shown in dotted lines. (c) Applying multi-qubit Z operators on 2D array of qubits shown in dotted red lines.

are coupled to the surrounded data qubits by the same coupling strength of $\xi_B = 0.6$ GHz. In this architecture, we can use our multi-qubit gates plus Hadamard gates to realize one cycle of error syndrome detection. The order of applying multi-qubit gates is given below – note that Hadamard gates are applied on all measure-X qubits at the beginning and end of each cycle:

A. Apply five-qubit parity gates where data qubits are the target qubits, and top and bottom X stabilizers are the active control qubits, while the left and right Z stabilizers are the dummy qubits (see Fig. 2.7 (a)).

B. Apply five-qubit parity gates where data qubits are the target qubits, and top and bottom Z stabilizers are the active control qubits, while the left and right X stabilizers are the dummy qubits (see Fig. 2.7 (b)).

C. Apply five-qubit parity gates where all Z stabilizers are the target qubits and all data qubits are the active control qubits (see Fig. 2.7 (c)).

Note that the order of applying these multi-qubit operators is important. Next we use the Heisenberg representation [5] and work on the stabilizer formalism to confirm the correct order by evaluating two different choices. For simplicity, we consider a small subspace of two data qubits a and b, and two measurement qubits Z and X as shown in Fig. 2.6. Here, we analyze the effect of our multi-qubit gates acting on the small subspace of interest step by step.

Let us start with the order of A, C, B from above; first a multi-qubit vertical $\hat{X}$ operator, second a $\hat{Z}$ operator, third a horizontal $\hat{X}$ operator. Consider the box of four qubits as depicted in Fig. 2.6. Initially, the measure-X and measure-Z qubits are initialized to $|+\rangle$ and $|0\rangle$, respectively, and the system is in a simultaneous eigenstate of the two operator products $\hat{X}_X \hat{I}_a \hat{I}_b \hat{I}_Z$ and $\hat{I}_X \hat{I}_a \hat{I}_b \hat{Z}_Z$. There is a tensor product between each pair of single-qubit operators but is removed for simplicity. Then applying the vertical $\hat{X}$ operator results in $\hat{X}_X \hat{X}_a \hat{I}_b \hat{I}_Z$ and $\hat{I}_X \hat{I}_a \hat{I}_b \hat{Z}_Z$. Applying the $\hat{Z}$ operator leads to $\hat{X}_X \hat{X}_a \hat{I}_b \hat{X}_Z$ and $\hat{I}_X \hat{Z}_a \hat{Z}_b \hat{Z}_Z$. And finally applying the horizontal $\hat{X}$ operator results in

$$\hat{X}_X \hat{X}_a \hat{X}_b \hat{X}_Z \qquad \text{and} \qquad \hat{Z}_X \hat{Z}_a \hat{Z}_b \hat{Z}_Z . \tag{2.14}$$

The order chosen above will not work since the resulted stabilizers in Eq. 2.14 do not commute and the single measurements of $\hat{X}$ and $\hat{Z}$ operators give us random results.

Next we consider the order of A, B, C from the above; first a multi-qubit vertical $\hat{X}$ operator, second a horizontal $\hat{X}$ operator, and third a $\hat{Z}$ operator. Initially, we have $\hat{X}_X \hat{I}_a \hat{I}_b \hat{I}_Z$ and $\hat{I}_X \hat{I}_a \hat{I}_b \hat{Z}_Z$. Applying the vertical X operator results in $\hat{X}_X \hat{X}_a \hat{I}_b \hat{I}_Z$ and $\hat{I}_X \hat{I}_a \hat{I}_b \hat{Z}_Z$. Then applying the horizontal X operator leads to $\hat{X}_X \hat{X}_a \hat{X}_b \hat{I}_Z$ and $\hat{I}_X \hat{I}_a \hat{I}_b \hat{Z}_Z$. Finally, applying the Z operator results in

$$\hat{X}_X \hat{X}_a \hat{X}_b \hat{I}_Z \qquad \text{and} \qquad \hat{I}_X \hat{Z}_a \hat{Z}_b \hat{Z}_Z . \tag{2.15}$$

This order of applying multi-qubit gates guarantees that each two data qubits share a pair of $\hat{X}$ and $\hat{Z}$ stabilizers and the measurements after each cycle are valid.

In all three steps above, the states of all qubits that are not used in the multi-qubit gate operations are frozen. The error correction cycle is performed by applying 3 sequences of multi-qubit parity gates as ordered in Fig. 2.7 (a), (b), (c). We can add/remove an arbitrary number of multi-qubit parity gates to scale up or down these gates in a larger 2D array of qubits when realizing a large-scale Surface Code memory.

The proposed protocol can be further improved by considering a new phase-error-detection circuit utilizing a five-qubit parity detection gate with four active control qubits. The conventional phase-error-detection circuit is shown in Fig. 2.8 (a). One can reach the same functionality by reversing the direction of each CNOT gate and sandwiching it between Hadamard gates on both control and target qubits. Canceling out the consecutive pair of Hadamard gates on measure-X qubit, the equivalent circuit is shown in Fig. 2.8 (b). Moreover, the functionality of the four CNOT gates shown in Fig. 2.8 (b) can be achieved by a five-qubit parity gate with four active control qubits. As depicted in Fig. 2.8 (c), to realize a phase-flip detection circuit, first, we apply Hadamard gates on data qubits surrounding the measure-X qubit, then we apply a five-qubit parity gate with four data qubits acting as active control qubits and the measure-X qubit acting as the target qubit. Finally, we apply Hadamard gates on the four data qubits.

Utilizing the five-qubit parity gates with four active control qubits plus single-qubit Hadamard gates, one can realize Surface Code error detection cycles in a sequence of only two multi-qubit gates plus single-qubit gates. Here the order

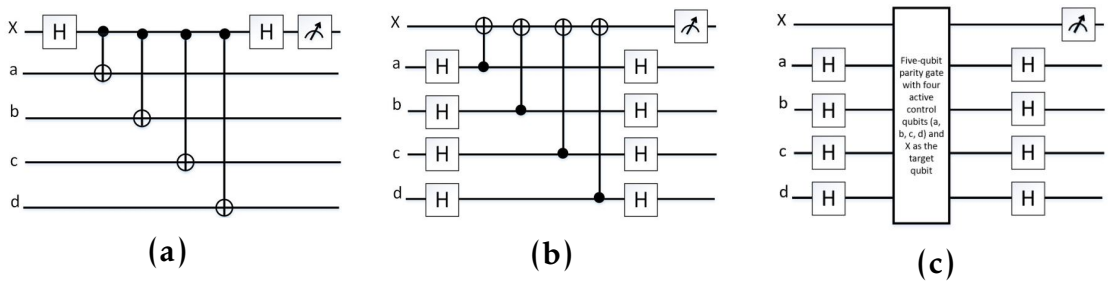


Figure 2.8 The logically equivalent phase error syndrome detection circuits (a) The conventional phase-error syndrome detection circuit (b) The equivalent phase error syndrome detection circuit where the direction of CNOT gates are reversed and some consecutive pair of Hadamard gates are canceled out (c) The equivalent phase error syndrome detection circuit utilizing the five-qubit parity gate with four active control qubits (a, b, c, d) and X as the target qubit

of applying multi-qubit gates is not important and Hadamard gates are applied on all data qubits at the beginning and end of each cycle.

A. Apply five-qubit parity gate where the X stabilizers are the target qubits and the surrounded data qubits are the active control qubits.

B. Apply five-qubit parity gates where the Z stabilizers are the target qubits and the surrounded data qubits are the active control qubits.

2.6 Discussion

We use our derived gate parameters in a MATLAB simulator that performs time evolution of a nine-qubit system as shown in Fig. 2.2. The simulator solves the Schrödinger equation based on trotterization [20] method with 0.1 ns trotter steps . We consider qubits E, F, G, H in the simulation to show that their states remain unchanged during the five-qubit gate operations. We use the following equations for calculating the gate fidelity:

$$\text{Fid} = \frac{\left|\text{Tr}\left(U_{\text{ideal}}^{\dagger} U\right)\right|}{d}, \tag{2.16}$$

$$\text{Fid} + \text{Unit} = \frac{\text{Tr}\left(U^{\dagger}U\right) + \left|\text{Tr}\left(U_{\text{ideal}}^{\dagger}U\right)\right|^{2}}{d \times (d+1)} \tag{2.17}$$

where $d = 2^9$ is the dimentionality of the computational space, U_{ideal} is the unitary transformation of the desired ideal gate, and U is the achieved unitary transformation calculated from the time evolution of the system:

$$U = e^{-i\int_0^{\tau_{\text{total}}} H(t)dt}, \tag{2.18}$$

with τ_{total} being the overall duration of the gate operation and $H(t)$ being the Hamiltonian of the system at time t. The fidelity equation in Eq. 2.17 accounts for checking the unitary condition of the quantum operation [21] and reports slightly lower gate fidelity as it is depicted in Fig. 2.9.

Note that in an experimental setup, one can realize the presented gates by choosing a different set of parameters which match with their physical system. One may choose different integers or multiply each parameter by a scaling factor such that the conditions explained in sections 2.3 and 2.4 remain satisfied [7,14]. For example, another set of parameters satisfying Eq. 2.4 would be $\Delta_T = 25$ MHz, $n = 1$, and $\tau = 50$ ns. Or the same gate fidelity can be achieved by keeping $n = 0$, but changing τ to 20 ns and reducing the tunneling parameter to $\Delta_T = 12$ MHz instead. An example of implementing controlled-unitary gates deriven by bias pulse scheme on an rf SQUIDs physical system has been presented in Ref. [14]. The parameters such as tunneling, coupling, bias pulse magnitude and duration chosen in this paper are in the same range as discussed in Ref. [14], where it is shown how these parameters can be adjusted to realize the controlled-unitary gates on the hardware.

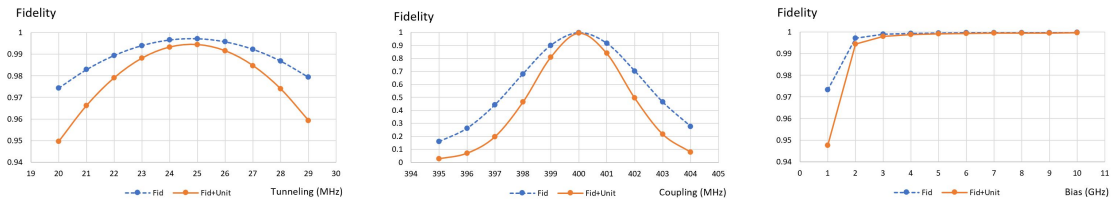
Figure 2.9 shows the sensitivity of the parity gate with four active control qubits on the different parameters. As depicted in Fig. 2.9 (a), a mis-

match of up to 2 MHz in the tunneling value results in the fidelity drop of $< 1\%$, however this can be compensated by adjusting the bias pulse width τ on the target qubit. In Fig. 2.9 (b) we swept away the coupling value of all four control qubits from the designed value $\xi = 0.4$ GHz and plotted the fidelity change. As it can be seen, if we use the same bias pulse magnitudes from Eq. 2.13, $\varepsilon_{T_1} = 0.8$, $\varepsilon_{T_2} = -0.8$, the gate fidelity drops significantly. However, if we change the magnitudes of the bias pulse according to the new ξ values, we can achieve a high fidelity gate again. As we discussed, any error from the parameter mismatch in tunneling and couplings can be respectively recovered by adjusting the bias pulse duration and magnitude. Therefore, the control circuitry is greatly reduced since by only adjusting one control parameter (bias pulse), one can achieve a high fidelity gate.

Our simulation shows that increasing the chosen bias value on control qubits would result in better gate fidelity. In Fig. 2.9 (c) the resulted fidelity vs bias values varying from 1 GHz to 10 GHz is plotted. For instance, with bias on control qubits as 2 GHz, the fidelity of the parity gate with four active control qubits (with $\Delta_T = 25$ MHz, $\xi = 0.4$ GHz, and $\tau = 10$ ns), was 0.9972 and 0.9944 based on Eq. 2.16 and Eq. 2.17, respectively. However, changing the bias on control qubits to 3 GHz resulted in gate fidelity of 0.999 and 0.998, respectively.

In the physical realizations, the bias pulses are not ideal and have some rise/fall times depending on the control electronics. The effect of the rise/fall times can be compensated by slightly changing the gate duration times [14]. Ideally one can use the analytical methods to design the ideal bias pulses and then use optimization methods to optimize the rise/fall times and bias pulse shapes based on their physical system to achieve the highest fidelity.

In this work, we considered the Hamiltonian with Ising interactions, how-



(a) **(b)** **(c)**

Figure 2.9 The effect of changing the system parameters on the fidelity of the parity gate with four active control qubits. Initial parameters are set as $\Delta_T = 25$ MHz, $\tau = 10$ ns, $\xi = 0.4$ GHz, and bias on control qubits as 2 GHz, then one parameter (tunneling, coupling or bias) is changed while all others are constant. Here we considered two different fidelity formulas as discussed in the main text. Fid represents the fidelity based on Eq. 2.16, and Fid+Unit represents the fidelity based on Eq. 2.17 where the unitary condition of the gate is also evaluated. (a) The effect of changing the tunneling (b) The effect of equally changing the coupling strengths (c) The effect of changing the bias on control qubits.

ever, the proposed gates can be realized for Hamiltonians with XX and YY interactions by simply interchanging the tunneling and bias values while coupling values and other parameters remain unchanged [7,15]. Furthermore, here we considered an arbitrary size 2D array of qubits to represent the application of multi-qubit parity gates in Surface Code schemes. However, it is often required to perform a reduced $\hat{X}$ or $\hat{Z}$ stabilizers on the borders of a logical qubit. To realize a two-terminal stabilizer, one can use the five-qubit parity gate with two active controls. Also realizing a three-terminal stabilizer is possible using a five-qubit parity gate with three active control qubits. Designing a five-qubit parity gate with three active control qubits using the methods discussed here is straightforward. Note that different coupling strengths are engineered depending on the number of the active control qubits in a multi-qubit gate and this effects on the architectural design decisions of the Surface Code array in systems with always-on interactions.

2.7 Conclusion

We designed new five-qubit parity gates with the fidelity of $> 99.9\%$ for nearest-neighbor architectures with always-on Ising interactions. There are many applications for these new gates, such as performing quantum state transfer in blocks of two-dimensional (2D) array of qubits. In this paper, we utilized these gates in error-syndrome-detection circuits. We designed a new quantum memory architecture for systems with always-on interactions, and presented a Surface Code protocol based on multi-qubit gates. The five-qubit parity gates can simultaneously be applied on many qubits in the array of Surface Code by adjusting only one control parameter (bias on the target qubits). Here, the Surface Code cycles can be achieved by applying two sequences of five-qubit parity gates across the entire qubit array, with the duration of each sequence being 2τ plus the timing required for single-qubit gates and measurements. The conventional Surface Code schemes based on two-qubit gates use the same timing for single-qubit gates and measurements, however, they need at least three sequences of CNOT gates across the qubit array. In the 2D qubit systems with always-on interactions, each CNOT gate takes 8τ which adds up to 24τ for a full Surface Code cycle. The advantages of using our proposed Surface Code memory architecture can be summarized in four main points:

1. It extensively simplifies the control circuitry.

2. It achieves a much faster error-correction cycle compared to the error syndrome detection circuits based on two-qubit gates.

3. It is expandable to large-scale Surface Code architectures with a fixed circuit depth for any size of a 2D array of qubits.

4. It removes the possibility of developing relative phases (dephasing) during idle times since there are no idle qubits in this scheme.

References

[1] Barbara M. Terhal. Quantum error correction for quan- tum memories. Rev. Mod. Phys. 87, 307, Apr 2015.

[2] John M Martinis. Qubit metrology for building a fault-tolerant quantum computer. npj Quantum Information,1(1), 2015.

[3] Daniel Eric Gottesman. arXiv:quant-ph/9705052v1, Cal-tech Ph.D. Thesis, May 1997.

[4] A. M. Steane. error correcting codes in quantum theory. Phys. Rev. Lett. 77, 793, July 1996.

[5] Austin G. Fowler, Matteo Mariantoni, John M. Martinis, and Andrew N. Cleland. Surface codes: Towards practical large-scale quantum computation. Physical Review A, 86(3), 2012.

[6] Radu Ionicioiu. Entangling spins by measuring charge: A parity-gate toolbox. Physical Review A, 75(3), 2007.

[7] P. Kumar and S. Daraeizadeh. Parity-based mirror inversion for e fficient quantum state transfer and computation in nearest-neighbor arrays. Physical Review A, 91(4), Sep 2015.

[8] Claudio Albanese, Matthias Christandl, Nilanjana Datta, and Artur Ekert. Mirror inversion of quantum states in linear registers. Physical Review Letters, 93(23), 2004.

[9] Peter Karbach and Joachim Stolze. Spin chains as perfect quantum state mirrors. Physical Review A, 72(3), 2005.

[10] Robert Raussendorf. Quantum computation via translation-invariant operations on a chain of qubits. Physical Review A , 72(5), Jan 2005.

[11] Xingxiang Zhou, Zheng-Wei Zhou, Guang-Can Guo, and Marc J. Feldman. Quantum computation with untunable couplings. Physical Review Letters , 89(19), 2002.

[12] Simon C. Benjamin and Sougato Bose. Quantum computing with an always-on heisenberg interaction. Physical Review Letters, 90(24), 2003.

[13] Joseph Fitzsimons and Jason Twamley. Globally controlled quantum wires for perfect qubit transport, mirroring, and computing. Physical Review Letters, 97(9), Jan 2006.

[14] E. C. Behrman J. E. Steck Z. Zhou P. K. Gagnebin, S.R. Skinner and S. Han. Quantum gates using a pulsed bias scheme. Phys. Rev. A, 72(042311), oct. 2005.

[15] P. Kumar and S. R. Skinner. Simplified approach to implementing controlled-unitary operations in a two-qubit system. Physical Review A, 76(2), 2007.

[16] Preethika Kunmar, Steven Skinner, and Sahar Daraeizadeh. Reduced hamiltonian technique for gate design in strongly coupled quantum systems. The Fifth International Conference on Quantum, Nano and Micro Technologies , 2011.

[17] Preethika Kumar. Direct implementation of an n-qubit controlled-unitary gate in a single step. Quantum Information Processing, 12(2):12011223, 2012.

[18] J. Kelly, R. Barends, A. G. Fowler, A. Megrant, E. Jeffrey, T. C. White, D. Sank, J. Y. Mutus, B. Campbell, Yu Chen, and et al. State preservation by repetitive error detection in a superconducting quantum circuit. Nature, 519(7541):6669, 2015.

[19] Yu Tomita and Krysta M. Svore. Low-distance surface codes under realistic quantum noise. Physical Review A, 90(6), Nov 2014.

[20] Naomichi Hatano and Masuo Suzuki. Finding exponential product formulas of higher orders. Quantum Annealing and Other Optimization Methods Lecture Notes in Physics, page 3768, 2005.

[21] Line Hjortshj Pedersen, Niels Martin Moller, and Klaus Molmer. Fidelity of quantum operations. Physics Letters A, 367(1-2):4751, 2007.

3 Designing Gates to Realize a Full Adder Quantum Circuit in cQED Transmon Systems

Note: This chapter has been republished with permission of Taylor and Francis Group LLC (Books) US, from "Nanoengineering, Quantum Science, and Nanotechnology Handbook", edited by Sergey Edward Lyshevski, Taylor Francis Group LLC, S. Daraeizadeh[1,2], S. P. Premaratne[2], M. Perkowski[1], A. Y. Matsuura[2], chapter 20, first edition, 2019, ISBN 9780367197513; permission conveyed through Copyright Clearance Center, Inc."

In this chapter, we present the methodology of designing single-qubit and two-qubit gates and implementing circuits for transmons within the circuit quantum electrodynamics paradigm. We implement a Hamiltonian-level simulator to design quantum gates and use these gates to realize a reversible logic quantum circuit of a full adder. The logic circuit is mapped to the physical constraints of a transmon chip, and we propose an alternative physical layout to reduce the communication overhead.

3.1 Introduction

Quantum entanglement and the superposition of states offer the potential of great computational power beyond the capabilities of transistor-based classical computers. However, simulation of quantum circuits using classical resources

[1] Department of Electrical and Computer Engineering, Portland State University.
[2] Intel Labs, Intel Corporation.

is difficult, due to exponential overhead [1]. This is unfortunate, as simulation is a fruitful practice in classical computing for architectural design. However, simulation of small qubit systems is still possible and can be used as a crucial tool for the architectural design of quantum computers and algorithms.

3.2 Superconducting Qubits

There exist different physical realizations of qubits such as trapped ions, nitrogen-vacancy centers in diamond, quantum dots, nuclear magnetic resonance, neutral atoms in optical lattices, superconducting devices, etc. Currently the transmon [2] is the most popular implementation of superconducting qubits. The transmon's quantum level structure is that of a weakly anharmonic oscillator to a good approximation. The relative anharmonicity is usually 3% — 5% of the characteristic system frequency [3].

To realize a single-qubit gate Hamiltonian, we can take the effective Hamiltonian and add a drive Hamiltonian term. Consider a drive as follows [4]:

$$\Omega(\mathrm{t}) = \begin{cases} \Omega_x(t)\cos(\omega_\mathrm{d} t) + \Omega_y(t)\sin(\omega_\mathrm{d} t) & 0 < t < t_\mathrm{g} \\ 0 & \text{otherwise} \end{cases} \tag{3.1}$$

where t_g refers to the gate duration time, ω_d is the drive frequency, and $\Omega_x(t)$ and $\Omega_y(t)$ are two independent quadrature controls. The effective Hamiltonian in the rotating frame of the drive is as follows:

$$\mathcal{H}_\mathrm{eff} = \hbar\widetilde{\omega}_\mathrm{r} a^\dagger a + \frac{1}{2}\hbar\widetilde{\omega}_\mathrm{q}\sigma_z + \hbar\chi\sigma_z a^\dagger a + \frac{1}{2}\hbar\left(\Omega_x(t)\sigma_x + \Omega_y(t)\sigma_y\right) \tag{3.2}$$

where $a^\dagger$ and a are resonator's raising and lowering operators, $\chi \simeq g^2/|\omega_\mathrm{r} - \omega_\mathrm{q}|$, and the resonator angular frequency ω_r and the angular frequency of the qubit transition ω_q are shifted by the drive frequency $\widetilde{\omega}_\mathrm{r} = \omega_\mathrm{r} - \omega_\mathrm{d}$, and $\widetilde{\omega}_\mathrm{q} = \omega_\mathrm{q} - \omega_\mathrm{d}$ in the rotating frame. Since the drive frequency is chosen to be far from the

frequency band $\omega_r \pm \chi$ where the resonator population can be large, we can assume $\langle a^\dagger a \rangle \sim 0$. Now by choosing $\omega_\mathrm{d} = \omega_\mathrm{q}$ for an on-resonance gate $\left(\widetilde{\omega}_\mathrm{q} = 0\right)$, the effective Hamiltonian is reduced to:

$$\mathcal{H}_\mathrm{eff} = \frac{1}{2}\hbar(\Omega_x(\mathrm{t})\,\sigma_x + \Omega_y(\mathrm{t})\,\sigma_y) \tag{3.3}$$

Now by adjusting $\Omega_x(t)$ and $\Omega_y(t)$ different rotations around x or y axes can be realized. Furthermore, choosing an off-resonance drive such that $\omega_\mathrm{d} = \omega_\mathrm{q} - \Omega_x(t)$ results in a Hadamard gate with the following Hamiltonian:

$$\mathcal{H}_\mathrm{eff} = \frac{1}{2}\hbar\Omega_x(\mathrm{t})\,(\sigma_z + \sigma_x) \tag{3.4}$$

Using the effective Hamiltonians for single transmon gates, we can simulate X, Y, and Hadamard gates which suffice for building a library of single qubit gates for quantum computation.

Redefining the ground-state energy in the Tavis-Cummings Hamiltonian for multi-level transmons, the effective Hamiltonian for two transmons can be written as [5]:

$$\widetilde{\mathcal{H}}_\mathrm{TC} = \hbar\omega_\mathrm{r}\left(a^\dagger a\right) + \sum_j^2 \left[\frac{1}{2}\hbar\sum_m \omega_m^{(j)} |m\rangle_{(j)}\langle m| + \hbar\sum_n g_{n,n+1}^{(j)} (a\,|\,n+1\rangle_{(j)}\langle n| + a^\dagger |n\rangle_{(j)}\langle n+1|)\right] \tag{3.5}$$

The last term in the above Hamiltonian describes the coupling between the transmons mediated through the resonator, where $g_{n,n+1}^{(j)}$ represents the transmon-resonator coupling value associated with the energy exchange between the energy levels (n and $n+1$) of transmon j and resonator. The last term can be replaced by the direct coupling term as follows:

$$\widetilde{\mathcal{H}}_\mathrm{coupling} = \hbar\sum_{j_1,j_2} \sqrt{j_1+1}\,\sqrt{j_2+1} J_{j_1,j_2} (|\,j_1, j_2+1\rangle\langle j_1+1, j_2| + |\,j_1+1, j_2\rangle\langle j_1, j_2+1|) \tag{3.6}$$

where $J_{j_1,j_2}$is the direct coupling between level j_1 from the first transmon and level j_2 from the second transmon. The coupling coefficients for different allowed energy levels of two transmons can be calculated as [5]:

$$J_{j_1,j_2} = \frac{g_1 g_2 \left(\omega_q^{(1)} + \delta_1 j_1 - \omega_r + \omega_q^{(2)} + \delta_2 j_2 - \omega_r\right)}{2\left(\omega_q^{(1)} + \delta_1 j_1 - \omega_r\right)\left(\omega_q^{(2)} + \delta_2 j_2 - \omega_r\right)} \tag{3.7}$$

where δ_1 and δ_2 are the anharmonicity values associated with transmons 1 and 2, respectively. To realize a two-qubit controlled-phase gate we need to consider the energy levels up to the second excitation manifold of each qubit. All the couplings except for those involving $|22\rangle$ (which results in a total of 4 system excitations) are considered. Then the effective Hamiltonian for two transmons can be represented by the following matrix. The order of the levels in the matrix is $\{|00\rangle, |01\rangle, |02\rangle, |10\rangle, |11\rangle, |12\rangle, |20\rangle, |21\rangle, |22\rangle\}$.

$$H = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \widetilde{\omega}_1^{(2)} & 0 & J_{0,0} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \widetilde{\omega}_2^{(2)} & 0 & \sqrt{2}J_{0,1} & 0 & 0 & 0 & 0 \\
0 & J_{0,0} & 0 & \widetilde{\omega}_1^{(1)} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \sqrt{2}J_{0,1} & 0 & \widetilde{\omega}_1^{(1)} + \widetilde{\omega}_1^{(2)} & 0 & \sqrt{2}J_{1,0} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \widetilde{\omega}_1^{(1)} + \widetilde{\omega}_2^{(2)} & 0 & 2J_{1,1} & 0 \\
0 & 0 & 0 & 0 & \sqrt{2}J_{1,0} & 0 & \widetilde{\omega}_2^{(1)} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 2J_{1,1} & 0 & \widetilde{\omega}_2^{(1)} + \widetilde{\omega}_1^{(2)} & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \widetilde{\omega}_2^{(1)} + \widetilde{\omega}_2^{(2)}
\end{bmatrix} \tag{3.8}$$

Here $\widetilde{\omega}_{(j)}^{(k)}$ represents the angular frequency associated with the k^{th} transmon at

energy level j and is given [5] as follows:

$$\widetilde{\omega}_j^{(k)} \equiv j\omega_{\mathrm{q}}^{(k)} + \frac{\delta_{\mathrm{k}}}{2}(j-1)\,j + \frac{j{g_{\mathrm{k}}}^2}{\omega_{\mathrm{q}}^{(k)} - \omega_{\mathrm{r}} + (j-1)\delta_{\mathrm{k}}} \tag{3.9}$$

Since the couplings are compared to the qubit transition frequencies, they can be considered as perturbations to the system. A Schrieffer-Wolff transformation is used to perform a full diagonalization of the Hamiltonian to second order in J [5-6].

To design a two-qubit cPhase gate, the frequency difference between the two qubit is adjusted to reach the avoided level crossing region between levels $|11\rangle$ and $|02\rangle$, where couplings between levels introduces a phase shift. When the desired phase is collected the transmons are brought back to the original frequencies. The behavior of the system depends on the speed of the frequency change operation. In the adiabatic regime, the crossing is approached slowly to ensure the population of each energy level does not change.

To realize the adiabatic regime, the frequency of qubits should change continuously with a smoothly shaped pulse when approaching the avoided crossing region. It should be adiabatic with respect to the timescale given by the level splitting [7]. By varying the frequencies of the pulses applied, the effective Hamiltonian described in Eq. 3.8 changes in a perturbative manner. Using a simulator, we can design the required pulses to achieve the controlled-rotation gates between two transmons. We consider two transmons with fundamental frequencies of 11.2 GHz and 9.6 GHz, with the resonator frequency fixed at 6.5 GHz. We assume both transmons have the same anharmonicity δ equal to -300 MHz, and the transmon-resonator coupling g is 200 MHz for both.

3.3 Quantum simulation

Here we simulate the dynamics of a quantum system in a decoherence-free subspace where we consider pure state simulation. If we want to consider noise, decoherence, and mixed states, a master equation simulation using a density matrix formalism is usually preferred [8]. The time evolution of the state of a quantum system is described by a time-dependent Schrödinger Equation (TDSE) or a related equation (e.g. Liouville–von Neumann equation or Lindblad equation). In a decoherence-free subspace, knowing the initial state of the system and the Hamiltonian, the solution to the Schrödinger equation is as follows:

$$\left|\psi(t)\right\rangle = \hat{U}\left|\psi(t_0)\right\rangle \tag{3.10}$$

Here, the state $\left|\psi(t)\right\rangle$ represents the vector for the probability amplitudes of different eigenstates, and $\hat{U}$ is the time evolution operator. $\hat{U}(t,\ \mathrm{t0}\ )$ is a unitary transformation operator which maps the initial state at time t0, to the final state at time t. $\hat{U}(t,\ \mathrm{t0}\ ) = e^{-\frac{i\hat{H}t}{\hbar}}$, where $\hat{H}$ is the Hamiltonian operator that describes the energy of the system. For instance, in case of single qubit gate operation, one can realize a 2×2 unitary transformation matrix $\hat{U}$ [9] as follows:

$$\hat{U}{=}e^{i\alpha}R_z(\beta)R_y(\gamma)R_z(\delta) = e^{i\alpha}\begin{bmatrix} e^{-\frac{i\beta}{2}} & 0 \\ 0 & e^{\frac{i\beta}{2}} \end{bmatrix}\begin{bmatrix} cos\frac{\gamma}{2} & -sin\frac{\gamma}{2} \\ sin\frac{\gamma}{2} & cos\frac{\gamma}{2} \end{bmatrix}\begin{bmatrix} e^{-\frac{i\delta}{2}} & 0 \\ 0 & e^{\frac{i\delta}{2}} \end{bmatrix} \tag{3.11}$$

where α is the global phase factor, $R_z(\beta)$ and $R_z(\delta)$ are the rotations β and δ around the z-axis, and γ is the rotation around the y-axis when the qubit state is visualized on a three-dimensional Bloch Sphere [8].

Any time dependent Hamiltonian that can be decomposed to m local interactions can be written as a summation of m local Hamiltonians $\hat{H}{=}\ \hat{H}_1 + \hat{H}_2 + \cdots + \hat{H}_m$, [10-11]. This Hamiltonian can be efficiently simulated using a uni-

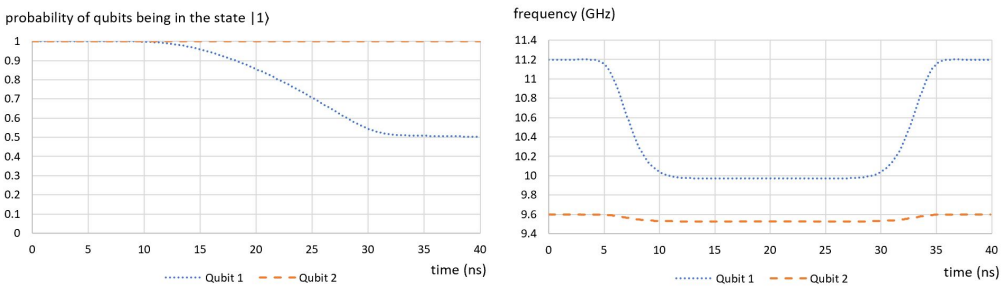
versal quantum computer [12]. If we decompose the Hamiltonian to m local non-commuting Hamiltonians, we can estimate the term $e^{\frac{-i}{\hbar}\hat{H}(t)t}$ using Trotterisation as below [10]:

$$e^{\frac{-i}{\hbar}\hat{H}t} \approx \left(e^{\frac{-i}{\hbar}\hat{H}_1\frac{t}{n}}\ e^{\frac{-i}{\hbar}\hat{H}_2\frac{t}{n}}\ \dots\ e^{\frac{-i}{\hbar}\hat{H}_m\frac{t}{n}}\right)^n \tag{3.12}$$

where $n \to \infty$, i.e. the accuracy of this estimation is increased by choosing very small Trotter steps $\left(\frac{t}{n}\right)$. In this method, we approximate the time evolution operator as $\hat{U}(t,\ \mathrm{t}_0) = \hat{U}(t,\ \mathrm{t}_{n-1})\ \dots\ \hat{U}(t,\ \mathrm{t}_1)\ \hat{U}(t,\ \mathrm{t}_0)$, $\mathrm{t}_0 < \mathrm{t}_1 < \cdots < \mathrm{t}_{n-1} < \mathrm{t}$, where at each time interval $\left(\frac{t}{n}\right)$; the Hamiltonian is considered piecewise constant. The Trotterisation methodology for quantum simulation can also be done in a classical computer for simulating the dynamics of a small quantum system, since simulation of quantum systems with many qubits using classical computational resources is limited by the memory and computational power.

3.4 Designing c $\sqrt{\text{NOT}}$ and c $\sqrt{\text{NOT}}^{\dagger}$ gates for transmons in cQED

It was found that large binary reversible quantum gates such as Toffoli with high number of inputs or circuits such as arithmetic quantum circuits used in Grover algorithm [9] can be efficiently realized from unitary gates such as CNOT and CCNOT (Toffoli), c $\sqrt{\text{NOT}}$, c $\sqrt{\text{NOT}}^{\dagger}$, cc $\sqrt{\text{NOT}}$, cc $\sqrt{\text{NOT}}^{\dagger}$. Like the c $\sqrt{\text{NOT}}$ gate, analogous gates that control the n^{th}-order root of NOT (controlled- $\sqrt[n]{\text{NOT}}$), are important in reversible quantum circuits [13-16]. In this section we describe designing of c $\sqrt{\text{NOT}}$, and c $\sqrt{\text{NOT}}^{\dagger}$ gates for transmons in cQED technology. In some literature, $\sqrt{\text{NOT}}$ and $\sqrt{\text{NOT}}^{\dagger}$are introduced by symbols V and $V^{\dagger}$, moreover c $\sqrt{\text{NOT}}$, and c $\sqrt{\text{NOT}}^{\dagger}$ are denoted as cV, and $cV^{\dagger}$, respectively [5]. The unitary transformations of $V = \sqrt{\text{NOT}}$ and $V^{\dagger} = \sqrt{\text{NOT}}^{\dagger}$ gates are given as



(a) **(b)**

Figure 3.1 A controlled rotation around the Z axis for 90° $cR_z\left(\frac{\pi}{2}\right)$, sandwiched between two Hadamard operations on the first qubit results in a c $\sqrt{\text{NOT}}$(Q2, Q1) gate. (a) Probabilities of qubits Q1 and Q2, respectively, being in state $|1\rangle$. (b) The required fundamental frequency changes in GHz for Qubits. The horizontal axes show time in 100 picoseconds units.

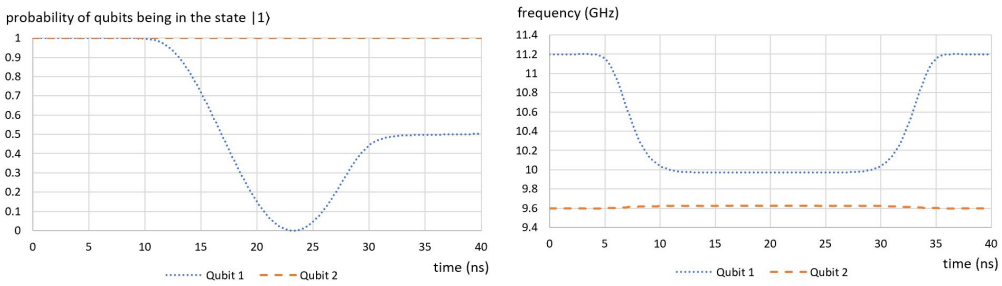
follows:

$$V = \sqrt{\text{NOT}} = \frac{1+i}{2}\begin{bmatrix} 1 & -i \\ -i & 1 \end{bmatrix} \tag{3.13}$$

$$V^\dagger = \sqrt{\text{NOT}}^\dagger = \frac{1-i}{2}\begin{bmatrix} 1 & i \\ i & 1 \end{bmatrix} \tag{3.14}$$

In transmons in cQED physical systems realizing a cPhase gate is more convenient than a native CNOT gate. It can be shown that CNOT, c $\sqrt{\text{NOT}}$, and c $\sqrt{\text{NOT}}^\dagger$ gates can be realized using controlled-phase gates where the target qubit will be sandwiched between two Hadamard gates before and after the controlled-phase gate [13]. A CNOT gate can be realized using controlled-$R_z(\pi)$ gate since CNOT = $HcR_z(\pi)H$. This idea can be generalized, in particular, c $\sqrt{\text{NOT}} = HcR_z\left(\frac{\pi}{2}\right)H$ and c $\sqrt{\text{NOT}}^\dagger = HcR_z\left(\frac{3\pi}{2}\right)H$.

The simulation results for $cR_z\left(\frac{\pi}{2}\right)$ and $cR_z\left(\frac{3\pi}{2}\right)$ gates for transmons are shown in Fig. 3.1 and Fig. 3.2, respectively. We designed these gates by considering a fixed gate time of 40 ns. Once the two transmons reach the avoided crossing region, the wait time is adjusted to ensure a phase collection of 90° for the $cR_z\left(\frac{\pi}{2}\right)$ gate,



(a) **(b)**

Figure 3.2 A controlled rotation around the Z axis for 270° $cR_z\left(\frac{3\pi}{2}\right)$, sandwiched between two Hadamard operations on the first qubit results in a c $\sqrt{\text{NOT}}^{\dagger}$(Q2, Q1) gate. (a) Probabilities of qubits Q1 and Q2, respectively, being in state $|1\rangle$. (b) The required fundamental frequency changes in GHz for Qubits. The horizontal axes show time in 100 picoseconds units.

and 270° for the $cR_z\left(\frac{3\pi}{2}\right)$ gate.

3.5 Realizing a full adder quantum circuit

3.5.1 Quantum logic synthesis

The research area of quantum logic synthesis consists in developing methods and algorithms to decompose larger matrices to elementary gates from predefined gate libraries in such a way that the quantum circuit has the smallest total gate cost, has the maximum speed, and the reduced number of ancilla qubits. The implementation of operations such as Peres [17], Toffoli, and Fredkin gates, as well as adders, multi-controlled AND gates, symmetric functions, and arithmetic functions based on controlled- $\sqrt[n]{\text{NOT}}$ have been proposed [15, 18-23]. In this chapter, we demonstrate a methodology to design small quantum circuits of this type in cQED transmons systems and how to map the circuit to the physical layout of the qubits. The principle of using only two qubit gates is represented on the full adder design given as Fig. 3.3 [23]. In the figures, for simplicity we

use the notations of V and $V^{\dagger}$, for $\sqrt{\text{NOT}}$ and $\sqrt{\text{NOT}}^{\dagger}$, respectively.

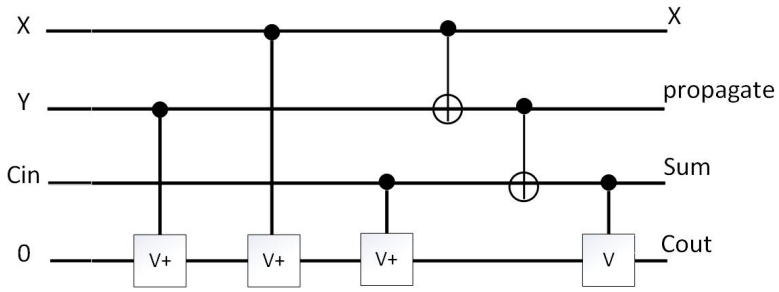


Figure 3.3 A full adder quantum circuit using cV (c $\sqrt{\text{NOT}}$), c$V^{\dagger}$ (c $\sqrt{\text{NOT}}^{\dagger}$), and CNOT gates, this circuit is taken from Ref. [23] that explains also quantum logic synthesis of similar quantum circuits.

In the design flow of the classical devices such as CMOS circuits in Very Large-Scale Integration (VLSI) or other technologies, the logic synthesis is a stage (abstraction layer) where a gate-level netlist is provided based on Register Transfer Level (RTL) design, and it happens before the physical design stage. Currently, the quantum logic synthesis is tightly bound to the research area of optimizing physical design (placement and routing) of the quantum circuit on the quantum chip layout. Therefore, in the next sections, we perform logic synthesis considering the quantum chip layout in a one-dimensional and two-dimensional space.

3.5.2 Quantum circuit implementation on a one-dimensional layout

To implement the quantum circuit in Fig. 3.3, let's consider a chip consisting of four transmons which are coupled through a waveguide resonator [24]. An example of analytically/Experimentally designing a three-qubit gate for this layout is found in Ref. [24], where the chip is made of a linear array of four transmons with frequencies 6 GHz, 7 GHz, 7.85 GHz, and 13 GHz, respectively. Note that when operating a two-qubit cPhase gate between qubits i and j, the

frequency of either qubit must not cross any other fundamental frequency of non-participating transmons. This criterion must be ensured between all neighboring transmons. An example for coupling multiple transmons is shown in Fig. 3.4 where four transmons are coupled in a linear array. In Fig. 3.4, the circles represent the transmons and the boxes represent the direct coupling between two transmons.

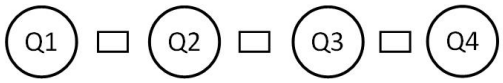


Figure 3.4 A one-dimensional nearest neighbor (NN) layout of transmons in cQED. The squares represent the coupling element, and circles represent the transmon qubits.

In our simulation for the full adder quantum circuit, we consider the fundamental frequencies of the transmons Q1-Q4 to be 8.4 GHz, 9.6 GHz, 11.2 GHz, and 12.6 GHz, respectively. In the NN architectures, in order to realize a two-qubit gate between two non-neighbor qubits, a SWAP operation is used to transfer information along the connected chain to yield the desired operation. Considering the full adder quantum circuit shown in Fig. 3.3, a possible placement choice is to assign Q1, Q2, Q3, and Q4 is as 0, C_{in}, Y, and X, respectively. The placement is simply assigning logical variables to qubits on the chip. Then the first c $\sqrt{\text{NOT}}^{\dagger}$ gate is operating between Q1 and Q3 which are not neighbors due to the current assignment. Therefore, a c $\sqrt{\text{NOT}}^{\dagger}$(Q3, Q1) is physically realized by sandwiching a c $\sqrt{\text{NOT}}^{\dagger}$(Q2, Q1) between two SWAP(Q_3, Q_2) gates. Similarly, c $\sqrt{\text{NOT}}^{\dagger}(Q_4, Q_1)$ is physically performed using 4 extra SWAP gates. In total, 6 extra SWAP gates are required to realize our full adder quantum circuit. However, it is possible to find a more optimal assignment of qubits considering connectivity of the logical quantum circuit. As input 0 has a direct interaction with all other three qubits, it is advantageous to assign it to either

Q2 or Q3. It is also advantageous to assign X and C_{in} for the extremities, since they have no mutual gate. Then an efficient qubit assignment for Q1-Q4 is C_{in}, 0, Y, and X, respectively. Using this efficient qubit assignment, the number of SWAP gates is reduced from 6 to 4. The full adder circuit realized in terms of all NN interactions is shown in Fig. 3.5. (see Fig. 3.5).

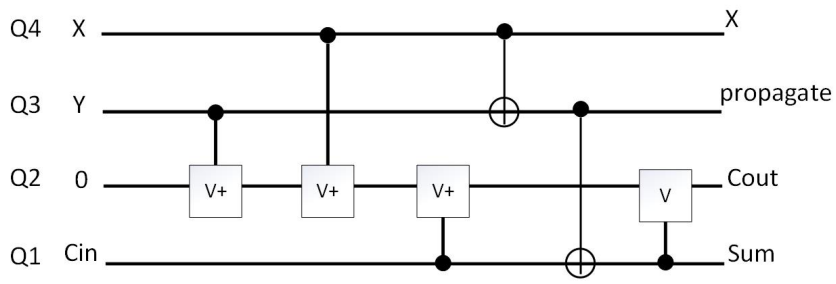


Figure 3.5 A placement option for realizing the full adder quantum circuit on the layout shown in Fig. 3.4 is assigning C_{in}, 0, Y, and X to Q1, Q2, Q3, and Q4, respectively.

The SWAP operation is not a native gate available in transmon systems. Thus, a SWAP gate can be constructed using three CNOT gates. This means the SWAP is an expensive operation when considering the relative cost of operations in terms of manipulation time. By changing the order of applying gates which are commuting and using known logic synthesis methods, the full adder circuit shown in Fig. 3.5 can be further simplified to reduce the number of SWAP gates and extra CNOT gates. The full adder circuit optimized for one-dimensional NN architecture is shown in Fig. 3.6. The detailed step-by-step optimization procedure is explained in the published work in [25].

The functionality of the quantum circuit shown in Fig. 3.6 matches with the functionality of circuit shown in Fig. 3.5, and both are confirmed by simulation. This example illustrates how the abstract quantum logic synthesis, the one-dimensional logic synthesis, and the final placement to the actual layout of quantum chip are interrelated. In the next section we propose a two-

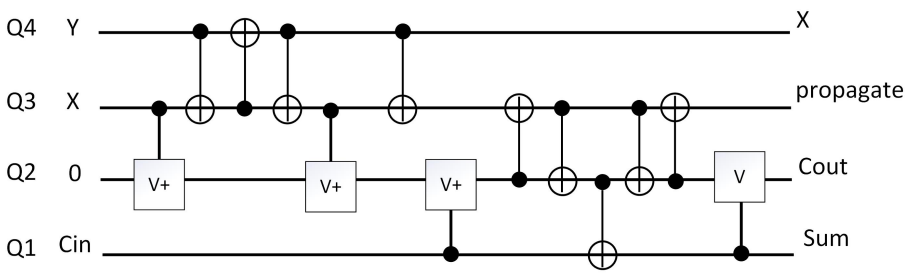


Figure 3.6 The full adder circuit following some logic synthesis and optimizing for the one-dimensional layout in Fig.3.4

dimensional qubit layout that requires no SWAP operation to implement a full adder quantum circuit.

3.5.3 Quantum circuit implementation on a two-dimensional layout

Suppose we have a two-dimensional nearest neighbor layout of transmons with resonator couplings as shown in Fig. 3.7. In the two-dimensional layout, the frequencies of the resonators between each transmon pair is set differently so that they will not be resonant with each other. Since a resonator is shared only between a unique pair of qubits, turning on an interaction may cause only weak unwanted interactions between neighbor qubits that we can neglect to lowest order.

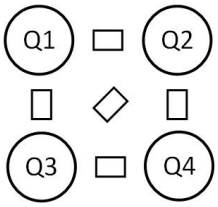


Figure 3.7 Two-dimensional layout for coupling four transmons Q1-Q4. The squares represent the resonators, and circles represent the transmon qubits.

Here if we place Qubits as shown in Fig. 3.8 (a), (b), and (c), the quantum cost will be different. As it can be seen from the original full adder circuit in

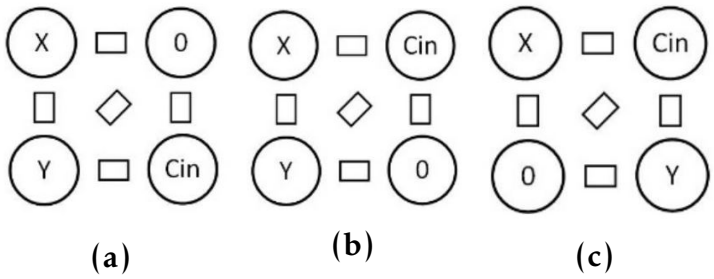


Figure 3.8 Some placement options to realize circuit shown in Fig. 3.3 on the layout shown in Fig. 3.7

Fig. 3.3, there is no interaction between X and C_{in}, while all other qubit pairs have direct interactions. Therefore, placement (a) is an optimal placement since the qubit pairs that are interacting directly, are coupled directly in the layout as well, and there is no SWAP operation needed at all. In Fig. 3.8, both (b) and (c) choices have equal cost and require two SWAP gates. If the optimal assignment shown in Fig. 3.8 (a) is chosen for layout shown in Fig 19, the full adder circuit can be expressed without SWAP operations resulting in a quantum cost of 6.

A more systematic way of finding the optimal physical design given the constraints of the physical system, is by mapping the problem to a graph theory problem and using graph theory algorithms already being used in more mature fields such as Very Large-Scale Integration-circuits (VLSI) physical design tools. For example, the physical layout of the qubits can be mapped to a *connectivity graph*, where vertices represent the qubits, and edges represent the direct coupling between qubits. The connectivity graph for the layout in Fig. 3.7 is given in Fig. 3.9 (a). Similarly, the quantum circuit can be modeled as an *interaction graph* where vertices represent the qubits, and edges represent the connection between qubits through the gates. The number of gates between two qubits can be specified as the weight on the edge between the two qubits. The interaction graph of the layout in Fig. 3.3 is given in Fig. 3.9 (b).

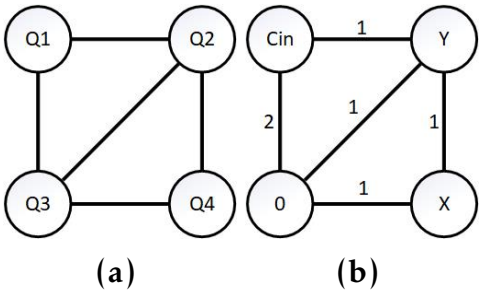


Figure 3.9 (a) The connectivity graph of the layout in Fig. 3.7, (b) The interaction graph of the circuit in Fig. 3.3

Now the qubit assignment problem is mapped to a graph embedding problem, where the interaction graph must be embedded into the connectivity graph such that the total distance between adjacent vertices in the interaction graph is minimized [26]. For this particular example, the optimal embedding is trivial and can be performed visually. However, when the number of qubits in the circuit grows, this graph problem becomes a NP-complete problem [27].

3.6 Conclusion

In this work, We presented a methodology to build a quantum simulator and use this simulator to design two-qubit gates such as $cR_z\left(\frac{\pi}{2}\right)$ and $cR_z\left(\frac{3\pi}{2}\right)$. Finally, we used a full adder reversible logic circuit as an example to show what kind of constraints one may consider when implementing the circuit on an actual qubit chip. This study illustrates the need for Computer-Aided Design (CAD) tools for physical design of quantum circuits with larger numbers of qubits, in order to optimize the placement and routing of qubits and to reduce the communication overhead introduced by SWAP gates. As quantum devices mature and scale-up, it will be increasingly important to create new quantum logic synthesis methods and physical design techniques that will be tailored to the specific constraints of emerging quantum nanotechnology devices [26, 28-31].

References

[1] D. P. DiVincenzo, "Topics in Quantum Computers", in Mesoscopic Electron Transport, ed. L. Kowenhove, G. Schon and L. Sohn, 1997

[2] J. Koch, T. M. Yu, J. Gambetta, A. A. Houck, D. I. Schuster, J. Majer, A. Blais, M. H. Devoret, S. M. Girvin, R. J. Schoelkopf , Charge-insensitive qubit design derived from the Cooper pair box. Phys. Rev. A 76, 042319 (2007).

[3] L. S. Bishop, Circuit quantum electrodynamics II. (Yale University, 2010).

[4] J. M. Chow, Quantum Information Processing with Superconducting Qubits, PhD thesis, Yale University, May 2010

[5] S. Richer, Master thesis, Perturbative analysis of two-qubit gates on transmon qubits, September 2013

[6] J. M. Gambetta. "Control of Superconducting Qubits". In: Lecture Notes of the 44th IFF Spring School. Ed. by David DiVincenzo. Schriften des Forschungszentrums J¨ulich. 2013.

[7] L. DiCarlo, J. M. Chow, J. M. Gambetta, Lev S. Bishop, B. R. Johnson, D. I. Schuster, J. Majer, A. Blais, L. Frunzio, S. M. Girvin, R. J. Schoelkopf, Demonstration of two-qubit algorithms with a superconducting quantum processor. Nature 460, 240–244 (2009).

[8] D. A. Steck, Quantum and Atom Optics. (http://steck.us/teaching (0.12.3, 25 October 2018).

[9] M. A. Nielsen and I. L. Chuang, "Quantum Computation and Quantum Information", 2000

[10] M-H Yung, J. D. Whitfield, S.Boixo, D.G. Tempel, A. Aspuru-Guzik, Introduction to Quantum Algorithms for Physics and Chemistry, Advances in Chemical Physics, 21 March 2014, doi:10.1002/9781118742631.ch03

[11] S. Lloyd, Science 273(5278), 1073–1078 (1996).

[12] R. P. Feynman, Int. J. Theor. Phys. 21(6-7), 467–488 (1982).

[13] A. Barenco, C. H. Bennett, C. H. Cleve, D. P. DiVincenzo, N. Margolus, N. Shor, T. Sleator, J. A. Smolin, and H. Weinfurter, Elementary Gates for Quantum Computation, Physical Review A, vol. 52, 1995, pp. 3457–3467.

[14] J. A. Smolin, and D. P. DiVincenzo, Five two-bit quantum gates are sufficient to implement the quantum Fredkin gate, Received 18 September 1995, 1996 American Physical Society, doi: 10.1103/PhysRevA.53.2855,

[15] M. Szyprowski and P. Kerntopf, Low Quantum Cost Realization of Generalized Peres and Toffoli Gates with Multiple-Control Signals, Proceedings of the 13th IEEE International Conference on Nanotechnology Beijing, China, August 5-8, 2013

[16] Soonchil Lee, Seong-Joo Lee, T. Kim, Jae-Seung Lee, J. Biamonte, M. Perkowski. (2006). The cost of quantum gate primitives. Journal of Multiple-Valued Logic and Soft Computing. 12. 561-573

[17] A. Peres, Reversible logic and quantum computers, Physical Review A, vol. 32, no. 6, 1985, pp. 3266– 3276.

[18] D. Maslov and G. W. Dueck, Improved Quantum Cost for n-bit To oli

Gates, IEEE Electronic Letters, vol. 39, no. 25, 2003, pp. 1790–1791.

[19] D. M. Miller, R. Wille, and Z. Sasanian, Elementary Quantum Gate Realizations for Multiple-Control Toffoli Gates, Proc. 41st IEEE International Symposium on Multiple-Valued Logic, 2011, pp. 288–293.

[20] 20. D. M. Miller, Lower Cost Quantum Gate Realizations of Multiple-Control Toffoli Gates, IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, 2009, pp. 308–313.

[21] D. M. Miller and Z. Sasanian, Lowering the Quantum Gate Cost of Reversible Circuits, 53rd IEEE International Midwest Symposium on Circuits and Systems, 2010, pp. 260–263.

[22] C. Moraga and F. Z. Hadjam, On Double Gates For Reversible Computing Circuits, Proc. 10th Workshop on Boolean Problems, 2012, pp. 131138.

[23] W. NN Hung, X.Song, G. Yang, J. Yang, M. Perkowski, Optimal synthesis of multiple output Boolean functions using a set of quantum gates by symbolic reachability analysis, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Volume: 25 , Issue: 9 , Sept. 2006, DOI: 10.1109/TCAD.2005.858352

[24] M. D. Reed, L. DiCarlo, S. E. Nigg, L. Sun, L. Frunzio, S. M. Girvin, R. J. Schoelkopf , Realization of Three-Qubit Quantum Error Correction with Superconducting Circuits, Nature 482, 382-385 (2012), DOI: 10.1038/nature10786

[25] S. Daraeizadeh, S. P. Premaratne, M. Perkowski, A. Y. Matsuura, " Designing Gates to Realize a Full Adder Quantum Circuit in cQED Transmon Systems", chapter 20 from "Nanoengineering, Quantum Science, and Nanotechnology Handbook", edited by Sergey Edward Lyshevski, CRC Press,

Dec. 2019, ISBN 9780367197513

[26] A. Shafaei, M. Saeedi, and M. Pedram, Qubit Placement to Minimize Communication Overhead in 2D Quantum Architecturs, 2014 19th Asia and South Pacific Design Automation Conference (ASP-DAC), January 2014, DOI: 10.1109/ASPDAC.2014.6742940,

[27] S. N. Bhatt, S. S. Cosmadakis, The complexity of minimizing wire lengths in VLSI layouts, https://doi.org/10.1016/0020-0190(87)90173-6

[28] T. Kutzschebauch ; L. Stok, Congestion aware layout driven logic synthesis, IEEE/ACM International Conference on Computer Aided Design. ICCAD 2001. DOI: 10.1109/ICCAD.2001.968621

[29] M. Perkowski, P. Kerntopf, A. Buller, M. Chrzanowska-Jeske, A. Mishchenko, X. Song, A. Al-Rabadi, L. Jozwiak, A. Coppola, and B. Massey, Regular Realization of Symmetric Functions Using Reversible Logic, IEEE Proceedings Euromicro Symposium on Digital Systems Design, 2012, DOI: 10.1109/DSD.2001.952289

[30] L. Amarú, P. Gaillardon, S. Mitra, G. De Micheli, New Logic Synthesis as Nanotechnology Enabler, Proceedings of the IEEE (Volume: 103 , Issue: 11 , Nov. 2015), DOI: 10.1109/JPROC.2015.2460377

[31] N. Mohammadzadeh, M. Sedighi, M. Saheb Zamani, Quantum physical synthesis: Improving physical design by netlist modifications, doi:10.1016/j.mejo.

4 Machine-Learning-Based Three-Qubit Gate Design for Toffoli and Parity Check in Transmon Systems

Note: This chapter has been published: S. Daraeizadeh[1,2], S. P. Premaratne[2], N. Khammassi[2], X. Song[1], M. Perkowski[1], A. Y. Matsuura[2], arXiv:1908.01092

We use machine learning techniques to design three-qubit flux-tunable entangling gates with fidelities of >99.9% and duration of 50 ns for nearest-neighbor coupled transmons in circuit quantum electrodynamics architectures. The gate design procedure enforces realistic constraints and analyzes the robustness of the new gates under decoherence, distortion, and random noise. The controlled-controlled-phase gate in combination with two single-qubit gates realizes a Toffoli gate which is widely used in quantum circuits, logic synthesis, and quantum error correction. We also introduce a new three-qubit entangling Parity Checker gate which has applications in quantum arithmetic circuits and quantum error correction schemes. Using these three-qubit gates, we design a new circuit for Shor's nine-qubit quantum error correction code and compare its performance to conventional realizations.

4.1 Introduction

Circuit quantum electrodynamics (cQED) [1,2] systems utilizing transmons [3–5] are potential candidates for realizing gate model quantum computers [6],

[1] Department of Electrical and Computer Engineering, Portland State University.
[2] Intel Labs, Intel Corporation.

with qubit coherence times of hundreds of microseconds [7] and the potential to scale up facilitated by quantum error correction schemes [8,9]. In these systems, the realization of high fidelity single-qubit and two-qubit entangling controlled-phase (CZ) gates enables universal quantum computation [7]. However, it is desirable to design three-qubit entangling gates to achieve better performance in certain quantum circuits.

Multi-qubit controlled-phase gates in transmons are typically designed by detuning the qubit transition frequencies to approach the avoided-level-crossing regions. In this regime, state mixing or level shifting due to non-computational quantum levels allows non-uniform phase collection within the computational subspace. This gives rise to entangling operations between qubits [10-15]. Finding the optimal transmon frequency detuning to achieve the desired avoided level crossings between system energy levels is a complex task which can take advantage of machine learning (ML) approaches [16-18]. Designing quantum gates and optimized control pulses using ML techniques and optimization theory has been demonstrated for various quantum systems [19-22]. We model the quantum gate design problem as a supervised ML exercise, by adjusting the system control parameters to converge to the target gate [17]. In this model, the training set is the desired unitary matrix and the cost function is the gate fidelity. We realize two types of three-qubit gates, the controlled-controlled-phase (CCZ) and the Parity Checker (ParChe) gates for transmon systems based on resonator-mediated nearest-neighbor (NN) couplings [23]. The CCZ gate collects a π rotation when all three qubits are in the state $|1\rangle$. While the ParChe gate collects a π rotation when the middle qubit is in the state $|1\rangle$ and the two distant qubits are in the opposite states. Both gates are designed with fidelity >99.9% and duration of 50 ns.

In sections 4.2, 4.3, and 4.4, we explain the motivation behind our work and introduce the CCZ and ParChe gates from the logic perspective. In sections 4.5 and 4.6, our simulation method and gate design methodology are respectively explained. The new gates are characterized in sections 4.7 and 4.8. Finally, in section 4.9, we show how quantum error correction circuits can benefit from these gates by applying them to the circuit for Shor's nine-qubit error correction code.

4.2 Motivation

The ParChe gate can be used in various quantum error correction (QEC) schemes [9] where a parity check is used to detect error syndromes. Example quantum circuits utilizing CNOT gates for detecting error syndromes are shown in Fig. 4.1 (a) and Fig. 4.1 (b) for bit-flip and phase-flip error detection, respectively. The circuits illustrated in Fig. 4.1 (a), and Fig. 4.1 (b) are utilized in many QEC stabilizer codes, including Surface Code [24-26]. Previous works have shown that the combinations of CNOT gates as shown in Fig. 4.1 (a), and Fig. 4.1 (b) can in principle be replaced by individual three-qubit gates [17,27]. However, a realistic physical gate design for resonator-coupled transmons has not been presented. Typically, in such systems, CNOT gates are realized using two-qubit CZ gates in combination with single-qubit rotations as shown in Fig. 4.1 (c) and Fig. 4.1 (d). From the logic perspective, the proposed three-qubit ParChe gate is equivalent to two consecutive 2-qubit CZ gates (compare Fig. 4.1 (c) and Fig. 4.1 (e)). Bit-flip and phase-flip error syndrome detections using ParChe gates are depicted in Fig. 4.1 (e), and Fig. 4.1 (f), respectively.
Following Reed et al. [15] and Fedorov et al. [14], we use the symbol of three filled dots for the three-qubit CCZ gates. Here, we introduce a new symbol for

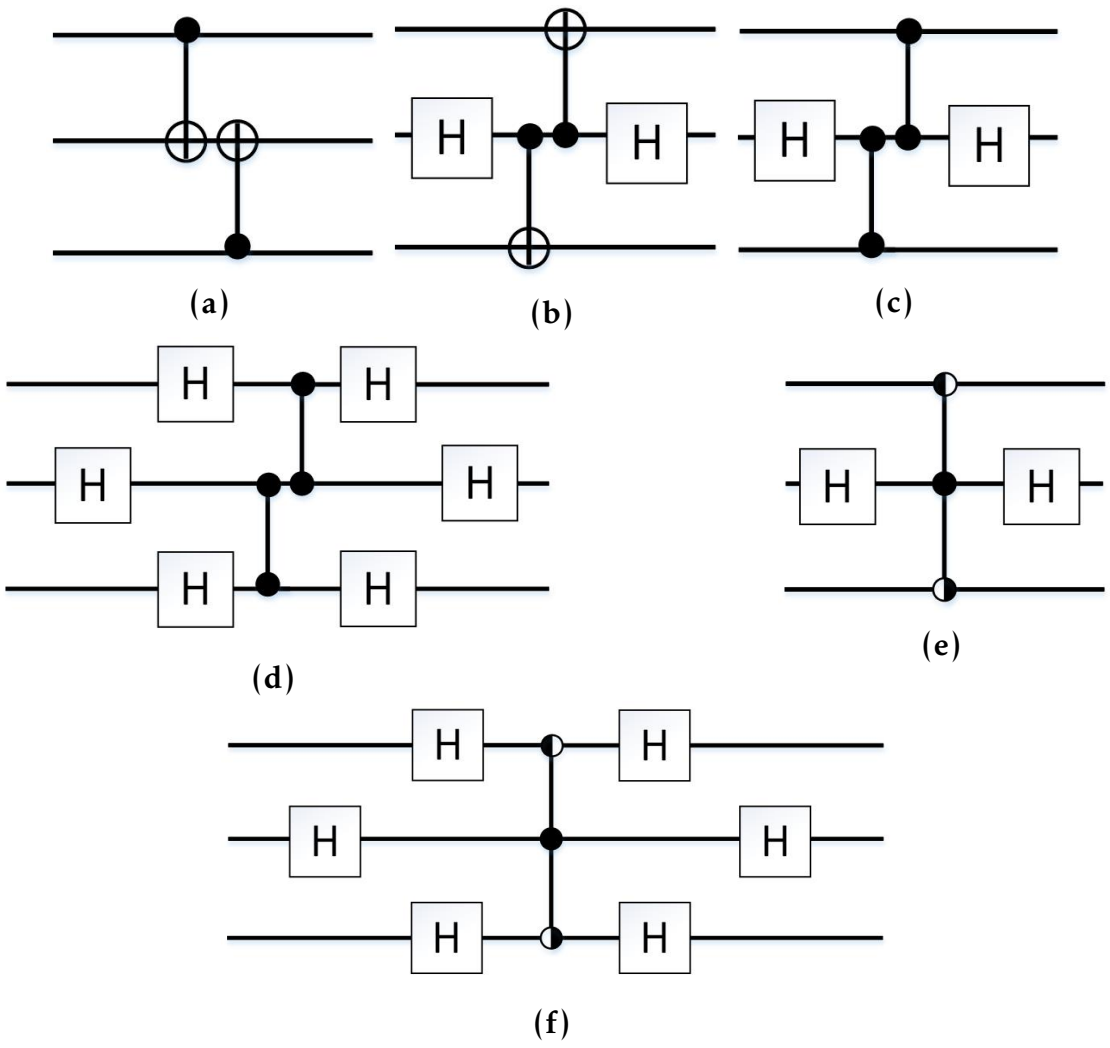


Figure 4.1 Error syndrome detection circuits (a) The classical bit-flip error syndrome detection circuit, (b) the classical phase-flip error syndrome detection circuit, (c) the classical bit-flip error syndrome detection circuit using controlled-phase gates, (d) the classical phase-flip error syndrome detection circuit using controlled-phase gates, (e) the new bit-flip error syndrome detection circuit using the parity checker gate, (f) the new phase-flip error syndrome detection using the parity checker gate

the ParChe gate as shown in Fig. 4.1 (e) and Fig. 4.1 (f). The first and the last dots are half-filled with opposite orientation, indicating the first and the last control qubits are in opposite states. While the full dot on the middle qubit means that it must be in the state $|1\rangle$ for phase collection.

Utilizing ParChe and CCZ gates, one can efficiently realize a family of majority-based reversible gates [28]. For example, the ParChe gate in combination with one CCZ gate and single-qubit gates as depicted in Fig. 4.2 (a) can realize a majority function of three inputs which is the 'carry-out' in Full adder circuits [29]. As illustrated in Fig. 4.2 (b), using four ParChe gates, one can reverse the order of the three qubits. This method can be generalized to efficiently perform quantum state transfer in NN architectures [27]. In technologies where the SWAP operations are decomposed to CNOT or CZ gates, using ParChe gates to perform mirror inversion [30] operations improve performance significantly. Following Reed et al. [15] and Fedorov et al. [14], we use the symbol of three filled dots for the three-qubit CCZ gates. Here, we introduce a new symbol for the ParChe gate as shown in Fig. 4.1 (e) and Fig. 4.1 (f). The first and the last dots are half-filled with opposite orientation, indicating the first and the last control qubits are in opposite states. While the full dot on the middle qubit means that it must be in the state $|1\rangle$ for phase collection.

Utilizing ParChe and CCZ gates, one can efficiently realize a family of majority-based reversible gates [28]. For example, the ParChe gate in combination with one CCZ gate and single-qubit gates as depicted in Fig. 4.2 (a) can realize a majority function of three inputs which is the 'carry-out' in Full adder circuits [29]. As illustrated in Fig. 4.2 (b), using four ParChe gates, one can reverse the order of the three qubits. This method can be generalized to efficiently perform quantum state transfer in NN architectures [27]. In technologies where

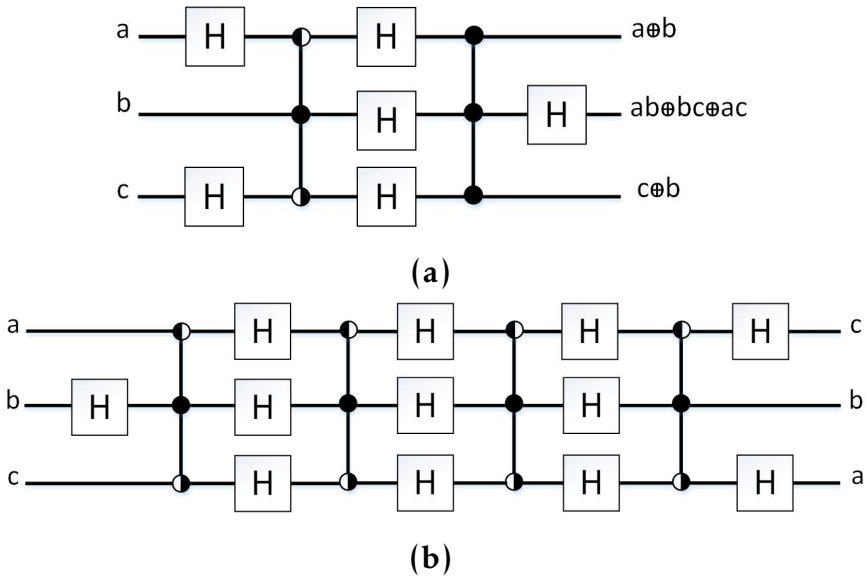


Figure 4.2 Quantum circuits utilizing ParChe gate (a) Majority function of three inputs is realized on the middle qubit. (b) The states of the first and the last qubits are swapped while the state of the middle qubit is unchanged.

the SWAP operations are decomposed to CNOT or CZ gates, using ParChe gates to perform mirror inversion [30] operations improve performance significantly.

4.3 Realization of Toffoli gate

The Toffoli (controlled-controlled-NOT) gate has broad applications in many quantum circuits. The most common decomposition of the Toffoli gate using standard single- and two-qubit gates [31] requires multiple single-qubit gates (H, T, and $T^{\dagger}$) and six two-qubit CNOT gates as shown in Fig. 4.3 (a). In this decomposition, at least two of the CNOT gates are applied to non-neighboring qubits which results in the addition of four SWAP gates in a NN-coupled architecture. The total circuit depth in these architectures is 16, including 10 two-qubit gates steps and 6 single-qubit gates steps.

Another decomposition of the Toffoli gate with circuit depth of three is pos-

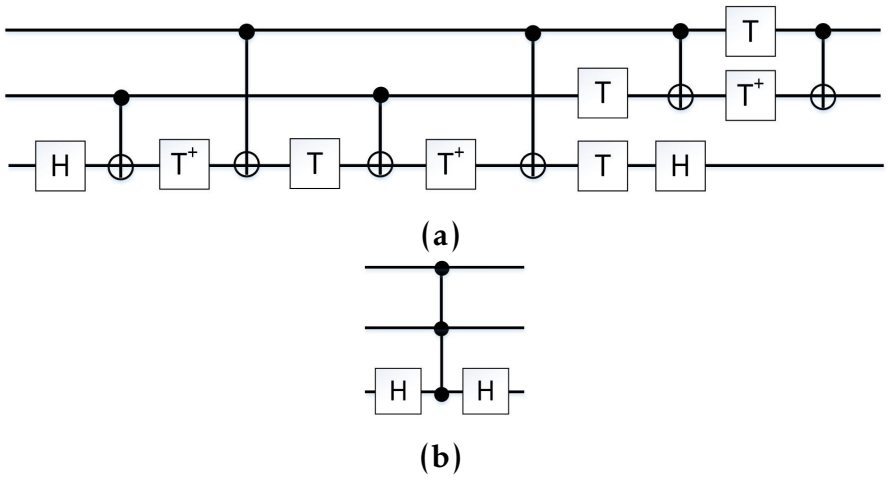


Figure 4.3 Quantum circuits utilizing ParChe gate (a) Majority function of three inputs is realized on the middle qubit. (b) The states of the first and the last qubits are swapped while the state of the middle qubit is unchanged.

sible based on a three-qubit CCZ gate and two single-qubit gates (Hadamard or single-qubit rotation gates) as shown in Fig. 4.3 (b) [32]. Using the latter decomposition, we show that a Toffoli gate can be realized for a resonator-mediated NN-coupled transmon system utilizing single-qubit gates, and our high fidelity CCZ gate (50 ns) with realistic frequency detuning sequences and system parameters.

In our simulations, we consider the lowest four energy levels (labeled $|0\rangle$ to $|3\rangle$) to ensure system evolution within the full triple-excitation manifold [15]. However, the cost function evaluation for the ML approach is performed only within the qubit subspace. The CCZ gate is designed to collect a π phase only on the $|111\rangle$ computational state (i.e. when all three qubits are in the $|1\rangle$ state).

The ideal CCZ gate in matrix form is:

$$U_{\mathrm{CCZ}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix} \tag{4.1}$$

where the ordering of the states is $|000\rangle$ to $|111\rangle$ in binary increments.

4.4 Realization of ParChe gate

We introduce a novel three-qubit gate (ParChe gate), which can be used as an elementary gate in universal quantum computing. Consider an array of three NN-coupled qubits (L, M, and R). The ParChe gate applies a π rotation only if qubit M is in state $|1\rangle$, and the first and the third qubits are in different states. In other words, if the XOR of the states of qubits L and R is 1, and qubit M is in state 1, then a π phase is collected. The matrix representation of the ideal

ParChe gate is

$$U_{\text{ParChe}} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{4.2}$$

The typical avoided crossing used for two-qubit gates in transmons is between eigenstates $|11\rangle$ and $|02\rangle$ in a two-transmon system and these levels occupy the double-excitation manifold [10]. Here, we generalize this idea to a three-transmon system and consider the primary interactions up to the triple-excitation manifold. We steer the energy levels of the three-transmon system by detuning qubit transition frequencies such that the computational states $|011\rangle$ and $|110\rangle$ each pick up a phase factor π, while all other energy levels collect trivial phases.

4.5 Simulation of the system dynamics

The effective Hamiltonian for our model with n transmons, when the coupling resonators are not populated, can be described as follows [33]:

$$\mathcal{H} = \sum_{k=1}^{n-1} \widetilde{\mathcal{H}}_c^{(k,k+1)} + \sum_{k=1}^{n} \widetilde{\mathcal{H}}_t^{(k)} \tag{4.3}$$

Here, the Hamiltonian of each transmon k is:

$$\widetilde{\mathcal{H}}_t^{(k)} = \sum_j \widetilde{\omega}_j^{(k)} |j\rangle_{(k)} \langle j| \tag{4.4}$$

where $\widetilde{\omega}_{(j)}^{(k)}$ is the dressed transition frequency associated with the k^{th} transmon at energy level j and is given by

$$\widetilde{\omega}_{j}^{(k)} = j\omega_{\text{q}}^{(k)} + \frac{\delta_{\text{k}}}{2}(j-1)j + \frac{j{g_{\text{k}}}^{2}}{\omega_{\text{q}}^{(k)} - \omega_{\text{r}} + (j-1)\delta_{\text{k}}} \tag{4.5}$$

where $\omega_{\text{q}}^{(k)}$ is the bare transition frequency associated with qubit k; g_k is the coupling strength between transmon k and the connected resonator, and ω_{r} represents the frequency of the coupled resonator. The last term in Eq. 4.5 is repeated for each transmon with appropriate modifications depending on the number of coupled resonators.

For any pair of resonator-coupled transmons, we estimate the direct coupling between two transmons $(k,\ k+1)$ as:

$$\widetilde{\mathcal{H}}_{c}^{(k,k+1)} = \sum_{j_{\text{k}},\ j_{\text{k+1}}} \sqrt{j_{\text{k}}+1}\,\sqrt{j_{\text{k+1}}+1}\,J_{j_{\text{k}},j_{\text{k+1}}}(|\,j_{\text{k}}, j_{\text{k+1}}+1\rangle\langle j_{\text{k}}+1, j_{\text{k+1}}| + |\,j_{\text{k}}+1, j_{\text{k+1}}\rangle\langle j_{\text{k}}, j_{\text{k+1}}+1|) \tag{4.6}$$

where $J_{j_{\text{k}},j_{\text{k+1}}}$is the direct coupling between the level j_{k} from the k^{th} transmon and level $j_{\text{k+1}}$ from the $(k+1)^{\text{th}}$ transmon.

$$J_{j_{\text{k}},\ j_{\text{k+1}}} = \frac{g_{\text{k}}g_{k+1}\left(\omega_{\text{q}}^{(k)} + \ \delta_{\text{k}}j_{\text{k}} - \omega_{\text{r}} + \omega_{\text{q}}^{(k+1)} + \ \delta_{\text{k+1}}j_{\text{k+1}} - \omega_{\text{r}}\right)}{2\left(\omega_{\text{q}}^{(k)} + \ \delta_{\text{k}}j_{\text{k}} - \omega_{\text{r}}\right)\left(\omega_{\text{q}}^{(k+1)} + \ \delta_{\text{k+1}}j_{\text{k+1}} - \omega_{\text{r}}\right)} \tag{4.7}$$

where δ_{k} and $\delta_{\text{k+1}}$ are the anharmonicity values associated with transmons k and $k+1$, respectively.

Using the time-dependent Hamiltonian, the time evolution equation of the system is solved to obtain the unitary transformation U:

$$U(t) = \exp\left\{-\frac{i}{\hbar}\int_{0}^{t}\mathcal{H}(\tau)\ d\tau\right\} \tag{4.8}$$

Here t is the time; $\mathcal{H}$ is the Hamiltonian of the system, and $\hbar$ is the reduced Planck's constant. To solve Eq. 4.8, we employ Trotterization [34]. Hence, the

final unitary transformation is estimated as follows [35]:

$$U(t_k) = U_k U_{k-1} U_{k-2}\ U_2 U_1 U_0 \tag{4.9}$$

Here U_i for $i = \{0,\ 1,\ ,\ k\}$ is calculated using Eq. 4.8 for the newly time-independent Hamiltonian at each timestep i, where $U_0 = I$ and k is the total number of steps. The Trotter step size is T/k, where T is the gate evolution time. In our simulations, the Trotter step size was 100 ps.

When solving the time evolution equation, we considered a smaller subspace to reduce the computational expenses. The Hamiltonian for n transmons with four energy levels spans a 4^n-dimensional Hilbert space. For a system composed of three transmons ($n = 3$), the Hamiltonian is a 64×64 matrix operator. Solving the Schrödinger equation for this large operator is computationally expensive, and there are numerous energy levels that have a minimal impact on the evolution of the gate of interest. Thus, we project this larger Hamiltonian to a smaller subspace where at most three excitations are allowed, resulting in a 20×20 matrix [17]. The 20 states considered are $\{|000\rangle, |001\rangle, |002\rangle, |003\rangle, |010\rangle, |011\rangle, |012\rangle, |020\rangle, |021\rangle, |030\rangle, |100\rangle, |101\rangle, |102\rangle, |110\rangle, |111\rangle, |120\rangle, |200\rangle, |201\rangle, |210\rangle, |300\rangle\}$.

The reduced Hamiltonian is evolved based on the qubit transition frequencies. The resulting unitary is projected [17] to the 8×8 computational subspace that includes the states $\{|000\rangle, |001\rangle, |010\rangle, |011\rangle, |100\rangle, |101\rangle, |110\rangle, |111\rangle\}$. Single-qubit phase compensation [13,16,17] is performed on this resultant unitary using the diagonal compensation matrix

$$M = e^{-i\theta_0}\ \mathrm{diag}(1, e^{-i\theta_1}, e^{-i\theta_2}, e^{-i(\theta_1+\theta_2)}, e^{-i\theta_4},$$

$$e^{-i(\theta_1+\theta_4)}, e^{-i(\theta_2+\theta_4)}, e^{-i(\theta_1+\theta_2+\theta_4)}) \tag{4.10}$$

where θ_0 represents the global phase, and θ_1, θ_2, and θ_4 represent the relative single qubit phases of states $|001\rangle$, $|010\rangle$, and $|100\rangle$, respectively.
The single-qubit phases are canceled out by multiplying matrix M with the projected unitary matrix in the computational subspace:

$$U_{\text{final}} = U_{\text{proj}} M \tag{4.11}$$

Finally, we calculate the gate fidelity $\mathcal{F}$ considering unitarity and closeness to the target ideal operation [36]:

$$\mathcal{F} = \frac{\text{Tr}\left(U_{\text{final}}^{\dagger} U_{\text{final}}\right) + \left|\text{Tr}\left(U_{\text{ideal}}^{\dagger} U_{\text{final}}\right)\right|^2}{d(d+1)}, \tag{4.12}$$

where $d = 2^3$ is the dimensionality of the computational subspace.

4.6 The gate design methodology based on machine learning

There are many machine learning and optimization algorithms one can choose to solve the optimal control problem for designing quantum gates. We design the system parameters to realize the CCZ and ParChe gates by combining two learning methods:

1. A machine learning method based on differential evolution [37] named Subspace-Selective Self-Adaptive Differential Evolution (SUSSADE) [16,17].

2. Our new local search algorithm.

In both learning procedures, the gate fidelity (Eq. 4.12) is considered as the fitness function to achieve the optimal control parameters for the given ideal unitary matrix. During the learning procedure, all parameters are assumed to be fixed, except for the frequency detuning of transmons.

In our simulations, the resonator-transmon couplings are $g = 0.2$ GHz, and the anharmonicity of each transmon is $\delta = -0.3$ GHz. The three transmons (L, M, R) with reference transition frequencies set to 5, 6, and 7 GHz, realize an identity operation with fidelity 99.9% when idling for 10 ns. Transmons L and M are coupled with an 8.05 GHz resonator, and transmons M and R are coupled with an 8.2 GHz resonator.

To reduce the search space during the learning procedure, the reference transition frequencies of the qubits are set closer during the ML algorithms search; $f_{\mathrm{L}} = 5.61$ GHz, $f_{\mathrm{M}} = 6$ GHz, and $f_{\mathrm{R}} = 6.39$ GHz, respectively. The maximum frequency detuning ranges permitted from the reference frequency of each qubit are set to [0, 0.5), (-0.5, 0.5), and (-0.5, 0], for qubits L, M, and R, respectively. These constraints help further reduce the search space and increase the efficiency of the learning process by removing the trial of detuning values far away from the interaction region.

Note that we further impose the following constraints during learning to ensure that the optimal frequency detuning sequences are experimentally realistic, achievable and that the target gate is robust. We enforce these constraints by:

1. Limiting the maximum point-to-point variation of the frequency detuning of each qubit to 220 MHz within the sequence to prevent undesired excitations in the quantum system. To take into account the limitations of physical signal instrumentation [38], the initial and the final points of the sequence are limited to a maximum point-to-point variation of 500 MHz from the initial reference transition frequencies of 5, 6, and 7 GHz.

2. Limiting the minimum difference between transition frequencies of two

adjacent qubits to 0.21 GHz/0.309 GHz for the CCZ/ParChe gate; primarily to prevent interactions within the single-excitation manifold.

Here, we briefly describe how the SUSSADE algorithm [16,17] was used to generate the qubit transition detuning sequences. First, a population of 200 random frequency detuning sequences (chromosomes) is generated in which each sequence contains 150 frequencies (50 per qubit). For a gate duration of $T = 50$ ns, the detuning sequence of each qubit is discretized to 50 amplitudes. After generating the initial population, we perform SUSSADE by randomly modifying the values of detuning sequences using the differential evolution operations such as mutation, crossover, and selection [17,37]. Finally, the fidelity of the resulting gate is calculated using Eq. 4.12. For any modified detuning sequence, if the new fidelity value is larger than the initial one, the new detuning sequence survives to the next generation. This procedure repeats until we reach our choice of either the fidelity threshold value (99.99%) or the maximum number of iterations (one million cycles). We use the Message Passing Interface (MPI) to distribute the simulation to 200 nodes on a computer cluster [39] such that each node is performing a full cycle of solving the time evolution and fidelity calculation for each member of the population.

SUSSADE was successfully used to obtain the frequency detuning sequences for 50 ns three-qubit gates with a fidelity of 98.8%, but any further progress was slow. Thus, a local search algorithm was implemented to refine the detuning sequences and achieve a gate fidelity of >99.9%. Note that the local search algorithm is efficient once the search space has been reduced by other learning algorithms.

The local search algorithm consists of the following steps:

1. At the beginning of the learning process, we define the largest (100 MHz)

and the smallest (1 kHz) change in frequency detuning allowed per data point. This is referred to as the optimization step size ϵ. We also set the maximum number of iterations (1000), the desired fidelity (99.99%), and all constraints enforced during SUSSADE.

2. While the constraints are met and the desired fidelity or the maximum number of iterations have not been reached, the following procedure is repeated:

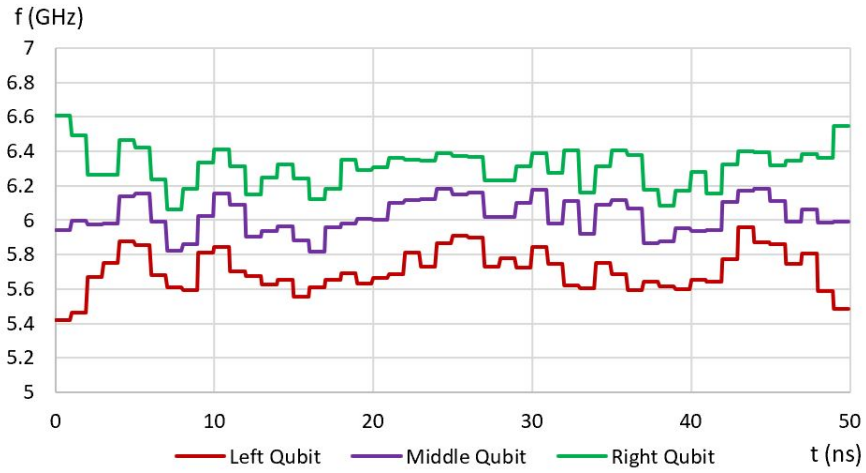
3. A local search window is moved from the first data point toward the last data point.

4. At each window, we recursively vary the frequency detuning value up or down by the optimization step size ϵ as long as it keeps improving the gate fidelity.

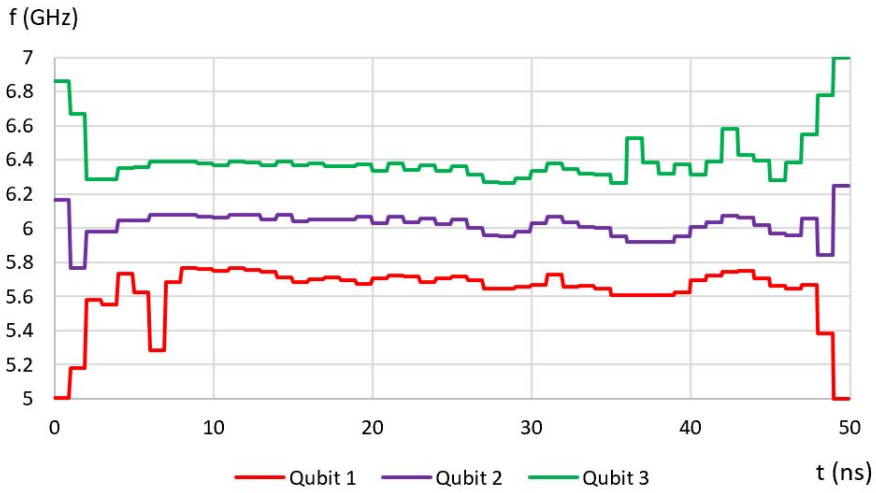
5. Once the local search window has covered all data points of the detuning sequence of all qubits, we reduce ϵ for a finer grain optimization ($\epsilon_{\text{new}} = 0.1\epsilon_{\text{old}}$).

6. If the optimization is already completed for the smallest predefined ϵ during the iteration, we increase the iteration number by one, reset ϵ to the largest predefined value, and repeat from step a.

The three-qubit gate duration is set to 50 ns for evolution, and the learning algorithms operate on 1 ns step size. The learned frequency detuning sequences are kept constant during each 1 ns step to obtain piecewise-constant pulse forms as shown in Fig. 4.4.



(a)



(b)

Figure 4.4 The frequency (f) vs. time (t) plots for learned transition frequency detuning sequences. The piecewise constant forms are generated from the learned frequency detuning sequences (50 learned data points per each transmon). (a) CCZ gate (b) ParChe gate

4.7 Gate verification and impact of decoherence

Simulated quantum process tomography (QPT) was used to independently evaluate gate performances by using master equation simulations. QPT is an excellent tool to evaluate the dynamics of a quantum system due to any process [40], in this case, the CCZ and ParChe gates. Given that this is QPT within the simulation, state preparation and measurement errors do not affect the methodology. Hence the results from QPT enable us to fully characterize the introduced gates.

Initial verification was performed assuming no decoherence in the system by using the Von Neumann equation for time evolution:

$$i\hbar\frac{\partial\rho}{\partial t} = \mathcal{H}\rho - \rho\mathcal{H}, \tag{4.13}$$

where the Hamiltonian $\mathcal{H}$ is the same as that given in Eq. 4.3 with the number of levels in each transmon set to $j_{\max} = 4$ and ρ is the density matrix for the three transmon system.

The three transmon system was evolved using the generated resonance frequency detuning sequences from learning algorithms. The evolution was performed on all the initial states given by $\left\{I, R_x^{0.5\pi}, R_y^{0.5\pi}, R_x^{\pi}\right\}^{\otimes 3}|000\rangle$ resulting in 64 density matrices. Unlike experimental QPT, it was not necessary to perform quantum state tomography to reconstruct these density matrices for the final states. These results were used to perform QPT by imposing constraints that the process matrix χ must satisfy [41,42]. The χ matrix completely characterizes the underlying process and is positive-Hermitian by definition [40].

We use the following metrics as defined in Ref. [41] to evaluate the performance of the new gates:

$$\text{Process fidelity}: \ \mathcal{F}_{\text{p}} = \text{Tr}\left(\chi^{(\text{ideal})}\chi\right) \tag{4.14}$$

Table 4.1 Table of QPT matrices for simulations under different conditions

Conditions	$\mathcal{F}_{\mathbf{p}}$ **CCZ/ParChe**	$\mathcal{F}_{\mathbf{g}}$ **CCZ/ParChe**	$\overline{\mathbf{Tr}(\rho^2)}$ **CCZ/ParChe**
$k_{\max}$= 4, $T_1 = T_2 = \infty$	0.999/0.999	0.999/0.999	0.999/0.999
$k_{\max}$= 3, $T_1 = T_2 = \infty$	0.998/0.996	0.998/0.997	0.999/0.999
$k_{\max}$= 4, $T_1 = T_2 = 20\ \mu$s	0.995/0.994	0.995/0.995	0.991/0.991
$k_{\max}$= 3, $T_1 = T_2 = 20\ \mu$s	0.993/0.992	0.994/0.993	0.991/0.991

$$\text{Average gate fidelity}: \ \mathcal{F}_{\mathrm{g}} = \frac{\mathrm{d}\mathcal{F}_{\mathrm{p}} + 1}{\mathrm{d} + 1} \tag{4.15}$$

$$\text{Average purity} \ : \overline{\mathrm{Tr}(\rho^2)} = \frac{\mathrm{d}\ \mathrm{Tr}\left(\chi^2\right) + 1}{\mathrm{d} + 1} \tag{4.16}$$

where χ is the experimentally determined process matrix; $\chi^{(\text{ideal})}$ is the ideal process matrix for the new gates, and $d = 2^3$ is the dimensionality of the computational subspace of the system. The results from the evaluation are given in Table 4.1.

The simulations incorporating decoherence were performed using the Lindblad-Kossakowski form of the master equation [43,44]. The appropriate operators for the dephasing portion of the master equation were obtained as in Refs. [45,46]. For convenience in simulation, T_1 and T_2 were both set to 20 s, assuming coherence times independent of the flux-tuning of the transmons [47].

Please refer to supplementary material for the full process matrices resulting from QPT. Comparison of results for $k_{\max} = 3$ ($\{|0\rangle, |1\rangle, |2\rangle\}$ levels) and $k_{\max} = 4$ ($\{|0\rangle, |1\rangle, |2\rangle, |3\rangle\}$ levels) from Table 4.1 indicates that the fourth level ($|3\rangle$) also plays a limited role in the system evolution.

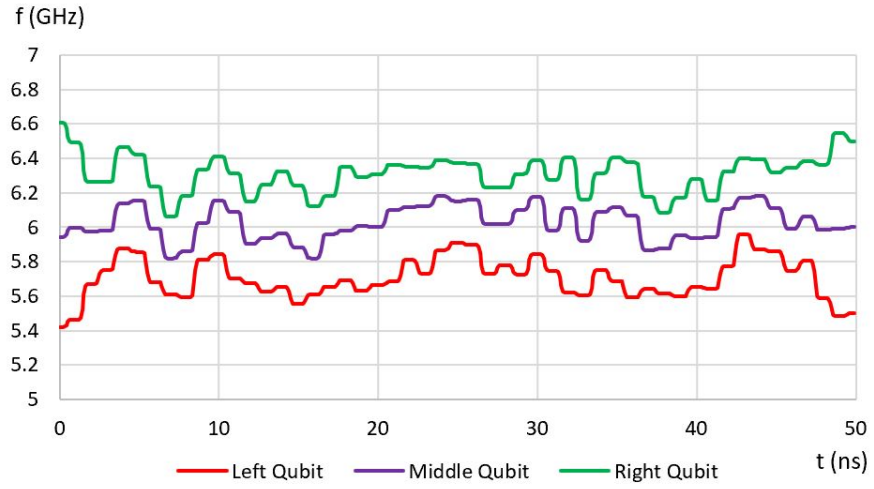
4.8 Robustness evaluation

The frequency detuning sequences derived from the learning algorithms have a piecewise-constant form. To investigate the effect of the first-order distortion due to control electronics, we use the following pulse reshaping method [13,17] to smooth the frequency detuning sequences:

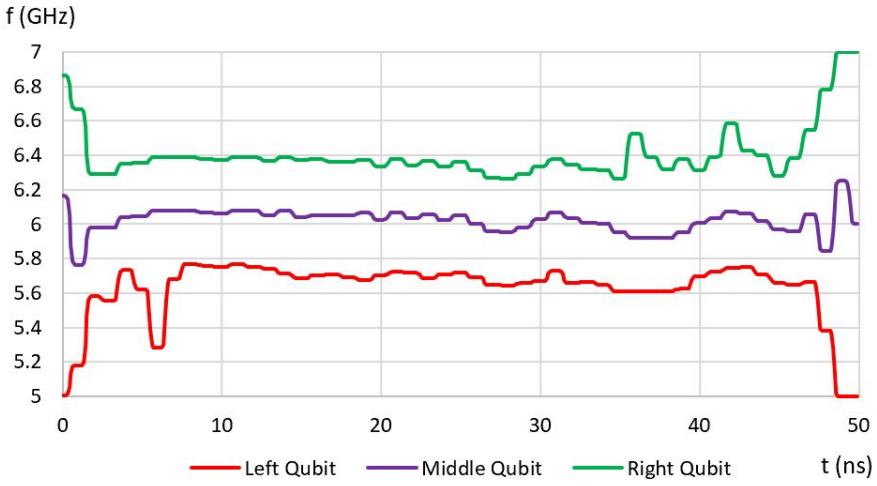
$$\omega_k(t) = \frac{\omega_{k_i} + \omega_{k_{i+1}}}{2} + \frac{\omega_{k_{i+1}} - \omega_{k_i}}{2}\left[\mathrm{Erf}\left(\frac{t - \left(\frac{t_{\mathrm{ramp}}}{2}\right)}{\sqrt{2}\tau}\right)\right], \tag{4.17}$$

where $\omega_k(t)$ represents the distorted frequency detuning of qubit k during $t_i \leq t \leq t_{i+1}$, and t_i represents the i^{th} time step. Here $\mathrm{Erf}(t) \equiv \frac{2}{\sqrt{\pi}} \int_0^t e^{-x^2} dx$ is the error function value of t, $t_{\mathrm{ramp}} = 1$ ns, and $\tau = \frac{t_{\mathrm{ramp}}}{4\sqrt{2}}$ [13]. The distorted sequences are shown in Fig. 4.5. Fidelity >99% is observed for both gates with smoothed frequency detuning distortions.

To investigate the effect of random noise on the CCZ and ParChe gates, we plot the average fidelity while increasing the random noise with amplitudes varying from 0 to 10 MHz. For each amplitude value, random noise is generated from a uniform distribution $(-1, 1)$, multiplied by the noise amplitude and added to the optimized detuning sequence. The latter step is repeated 10000 times and at each iteration, the system Hamiltonian is evolved, and the gate fidelity is calculated. The averaged fidelity of the 10000 results is reported as the average fidelity at each noise amplitude. Fig. 4.6 illustrates the gates' robustness against random noise and demonstrates fidelity >99% with random noise



(a)



(b)

Figure 4.5 The frequency (f) vs. time (t) plots for the smoothed learned frequency detuning sequences of qubits. (a) CCZ gate (b) ParChe gate

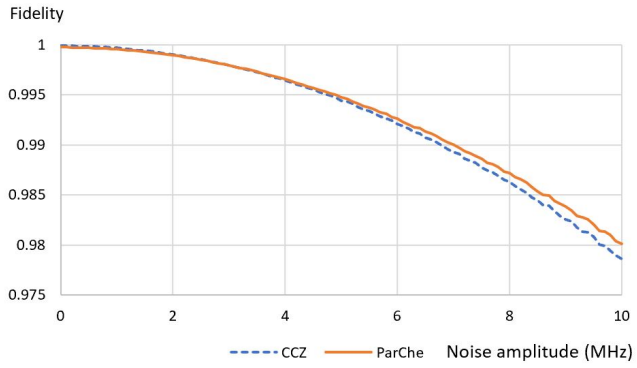


Figure 4.6 Average fidelities of CCZ gate and ParChe gate over 10000 samples under the effect of random noise with amplitudes ranging in 0 to 10 MHz.

amplitudes of up to 6.7 MHz for CCZ gate and up to 7 MHz for the ParChe gate.

4.9 Shor's nine-qubit QEC circuit based on CCZ and ParChe gates

In order to evaluate the performance of the new parity check gate in comparison with the conventional CZ-based parity check, we design the Shor 9-qubit error correction code using ParChe and CCZ gates. Fig. 4.7 shows a circuit design of Shor's 9-qubit code using Hadamard, ParChe, and CCZ gates.

We use the QX Simulator [48] to simulate the different designs of the Shor's error correction code under noise, where we use the Pauli-Twirling Approximation (PTA) error model with qubit relaxation time $T_1 = 20$ µs and an echo time $T_2 = 20$ µs. The duration of the single-qubit gates, two-qubit CZ gates, and three-qubit (CCZ and ParChe) gates are respectively 20 ns, 40 ns, and 50 ns.

The Pauli-Twirling channel [49,50] allows the approximation of the decoherence channel as an asymmetric depolarizing channel where the decohering qubit suffers from discrete Pauli errors (X, Y, Z) with respective probabilities (p_X, p_Y, p_Z) [51]. The error probabilities are expressed in terms of the gate

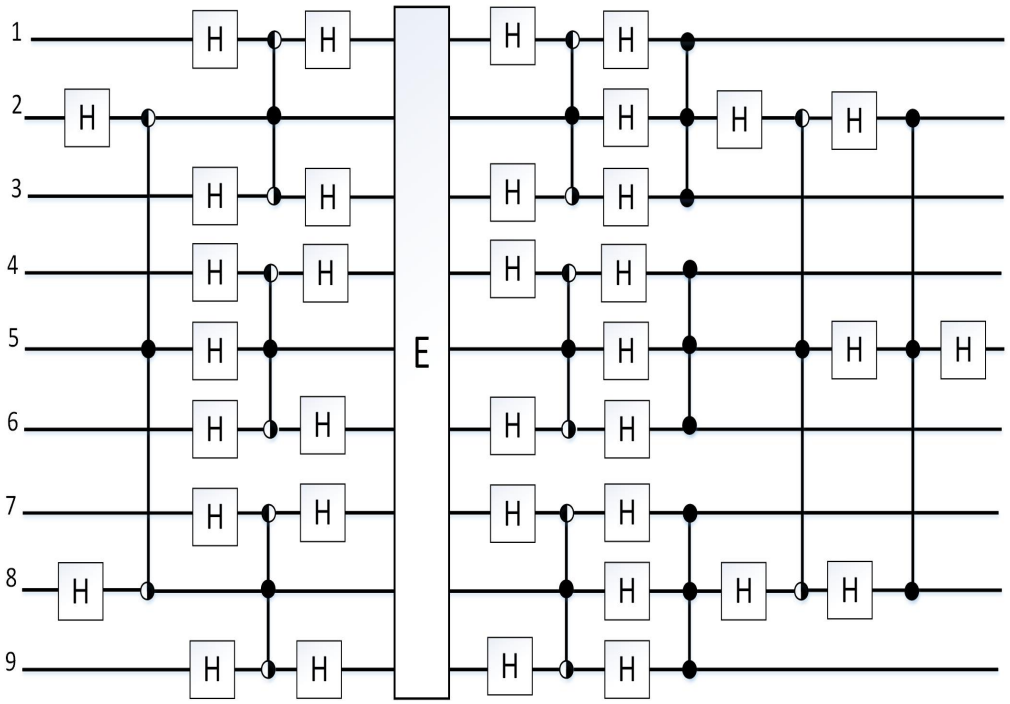


Figure 4.7 The proposed Shor's nine qubit error correction circuit using new parity checker gates and CCZ gates

execution time and the qubit coherence times T_1 and T_2:

$$p_Y = \frac{1}{4}\left[1 - \exp\left(\frac{-t}{T_1}\right)\right] \tag{4.18}$$

$$p_X = \frac{1}{4}\left[1 - \exp\left(\frac{-t}{T_1}\right)\right] \tag{4.19}$$

$$p_Z = \frac{1}{2}\left[1 - \exp\left(\frac{-t}{T_2}\right)\right] - \frac{1}{4}\left[1 - \exp\left(\frac{-t}{T_1}\right)\right] \tag{4.20}$$

The approximated amplitude damping (AD) channel is given by

$$\rho \rightarrow e_{\mathrm{AD}}(\rho) = E_1^{\mathrm{AD}} \rho E_1^{\mathrm{AD}\dagger} + E_2^{\mathrm{AD}} \rho E_2^{\mathrm{AD}\dagger}. \tag{4.21}$$

Here E_1^{AD} and $E_2^{\mathrm{AD}}$are the Kraus matrices for the amplitude damping channel:

$$E_1^{AD} = \begin{pmatrix} 1 & 0 \\ 0 & \sqrt{1-p_{\mathrm{AD}}} \end{pmatrix} \tag{4.22}$$

$$E_2^{AD} = \begin{pmatrix} 0 & \sqrt{p_{\mathrm{AD}}} \\ 0 & 0 \end{pmatrix} \tag{4.23}$$

where p_{AD} corresponds to the probability of single photon emission from the qubit. Similarly, the phase damping channel can be expressed in terms of Kraus matrices and the action of the two channels combined within a single channel as in [50].

We compare three designs of the Shor's code:

1. The CZ-based design where both the parity check stage and the Toffoli gate are implemented in terms of CZ and single-qubit gates.

2. Replacing only the parity check stage of 1) with the ParChe-based design.

3. The full three-qubit gate implementation using ParChe-based parity checking and a CCZ-based Toffoli gate.

For each of the Shor's code implementations, we apply many error detection and correction cycles and measure the logical error rate and thus the fidelity. Figure 8 shows the fidelity decay of the logical qubit through the correction cycles of the Shor's code. While the fidelity of the logical state decays over the correction cycles due to the low coherence time and the high physical error rate of current systems used in this simulation, the fidelity is significantly improved after introducing the ParChe gate for performing faster parity checks with higher fidelity. The use of the ParChe gate in combination with the CCZ-based Toffoli provides further improvement and results in a lower logical error rate.

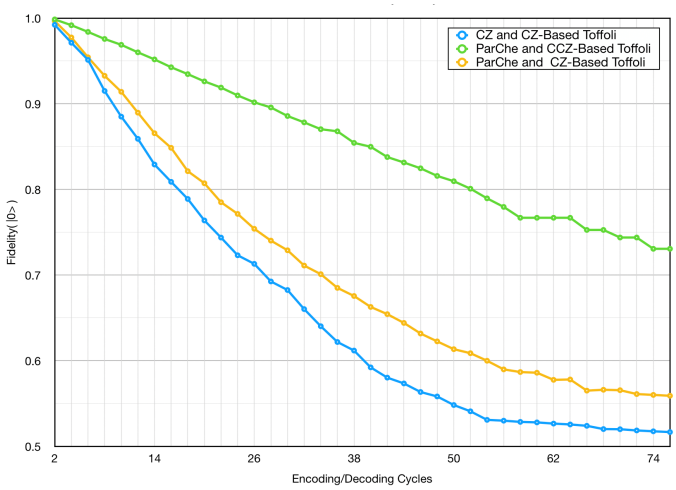


Figure 4.8 Performance of a Shor's error correction code under noise when using the ParChe and CCZ gates in comparison with the traditional CZ-based design.

4.10 Conclusion

We designed two fast, high-fidelity, and robust three-qubit entangling (CCZ and ParChe) gates for resonator-mediated NN-coupled transmons. We described the gate design methodology using simulation and machine learning techniques and presented a new local search algorithm for optimal quantum control applicable to small search spaces. The operations of the CCZ gate and the ParChe gate were confirmed by a C++ simulator that solves the Schrödinger equation for the time-dependent Hamiltonian of the system. Moreover, the operations of the gates were verified independently via quantum process tomography in both the presence and absence of decoherence. The robustness of gates was examined using random noise injection and frequency detuning distortion.

The presented gate design procedure, verification, and robustness investigation can be applied to designing new gates for other quantum systems as well. We showed that our designed gates can significantly increase the performance of the Shor's nine qubit error correction circuit, compared to the traditional

circuits based on two-qubit controlled-phase gates. The ParChe gate can be considered as an elementary gate for universal quantum computing and can be used in quantum arithmetic circuits and many QEC schemes.

4.11 Supplementary Material

Here the full process matrices resulted from simulated quantum process tomography are presented.

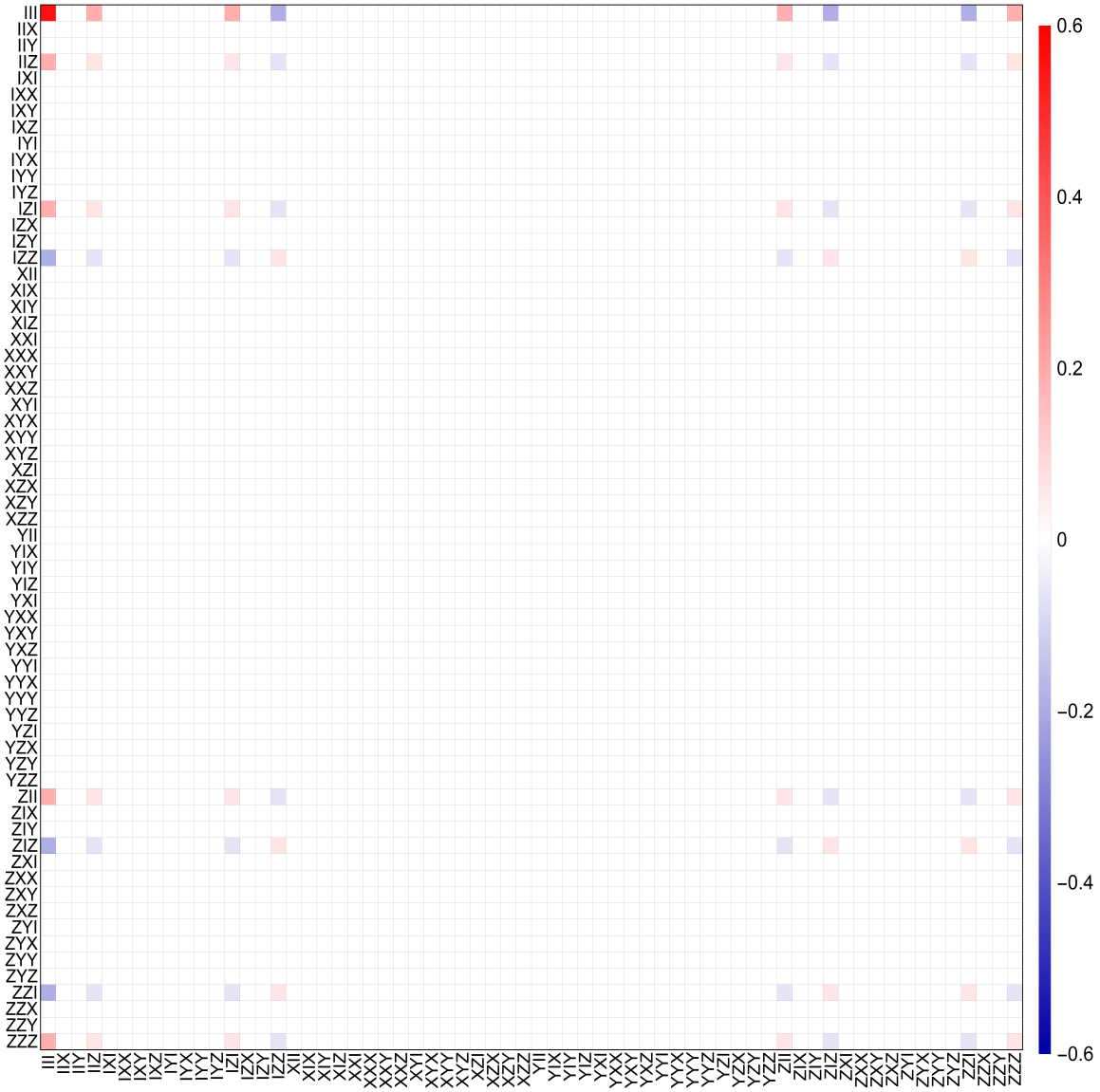


Figure 4.9 Real part of the process matrix $\chi^{(\text{ideal})}$ for the ideal CCZ operation. The Imaginary component is identically 0 for all elements. The process matrix is expressed in terms of the complete basis set of 64 three-qubit Pauli matrices; I,X,Y,Z represent the matrices σ_0, σ_x, σ_y, σ_z.

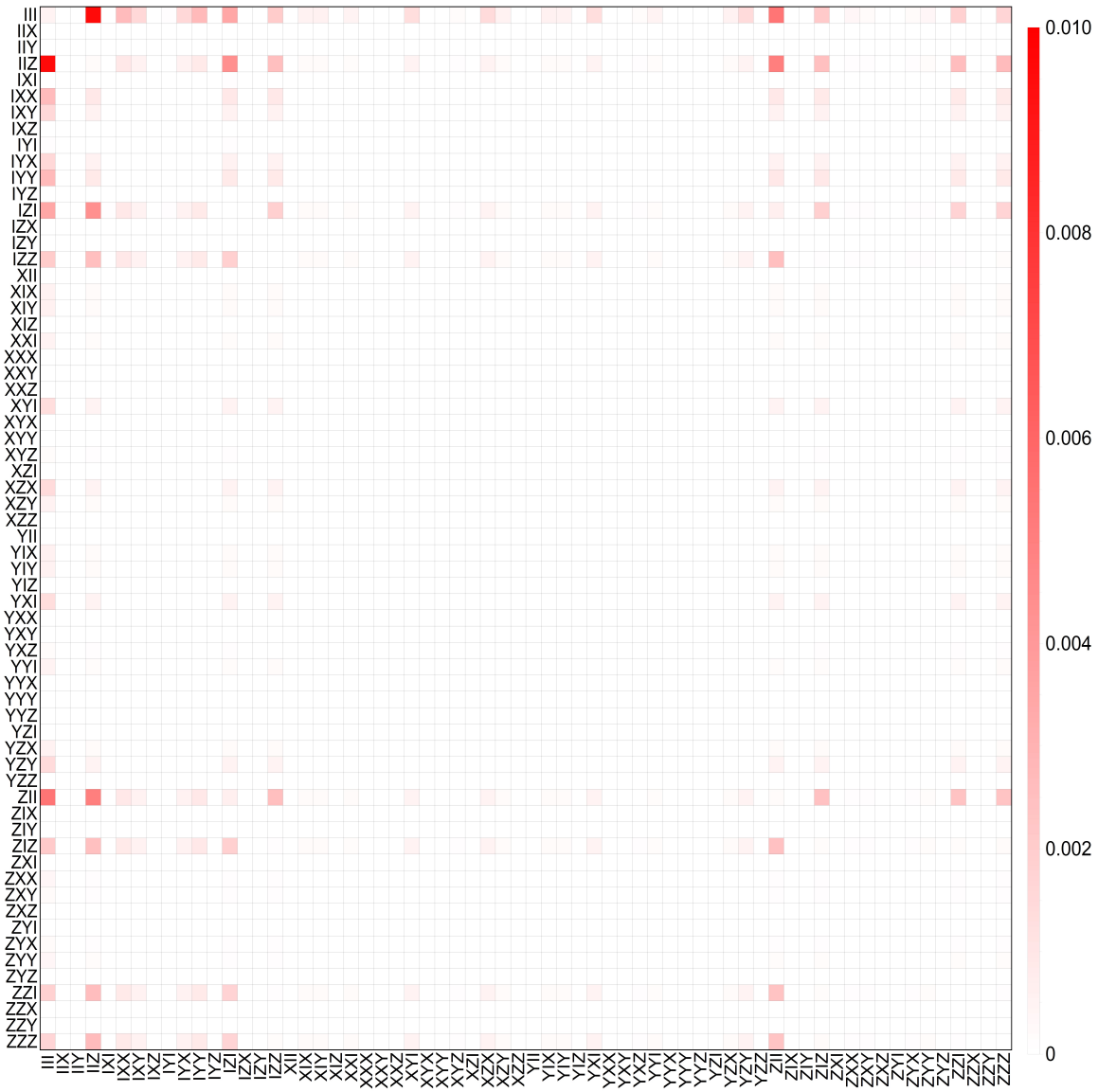


Figure 4.10 Absolute values of the differences between elements of ideal and simulated process matrices $\left[\chi^{(\text{ideal})} - \chi\right]$ for the ideal CCZ operation. Note the change in legend scale to increase clarity of error terms. The process matrix is expressed in terms of the complete basis set of 64 three-qubit Pauli matrices; I,X,Y,Z represent the matrices σ_0, σ_x, σ_y, σ_z.

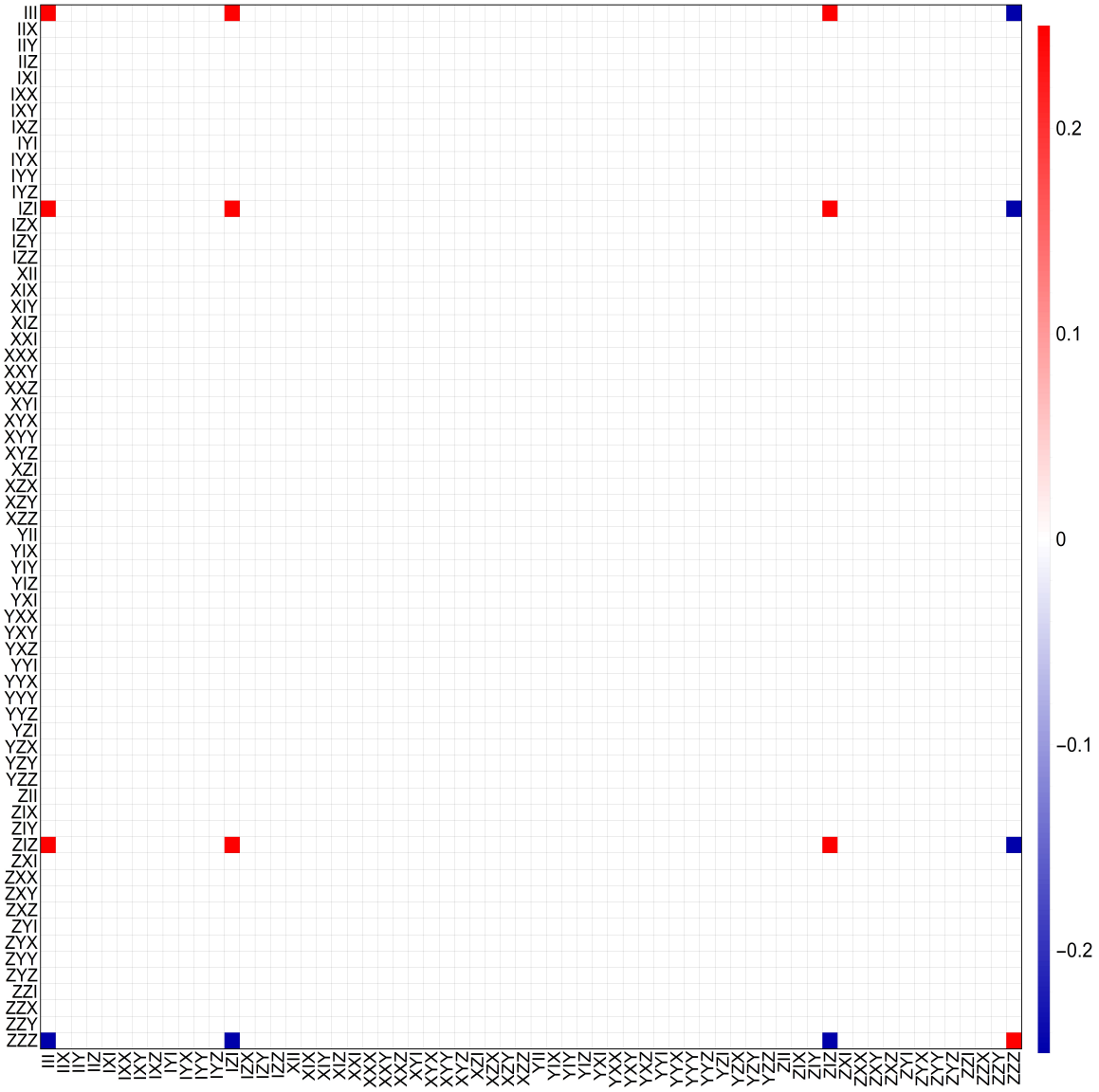


Figure 4.11 Real part of the process matrix $\chi^{(\text{ideal})}$ for the ideal ParChe operation. The Imaginary component is identically 0 for all elements. The process matrix is expressed in terms of the complete basis set of 64 three-qubit Pauli matrices; I,X,Y,Z represent the matrices σ_0, σ_x, σ_y, σ_z.

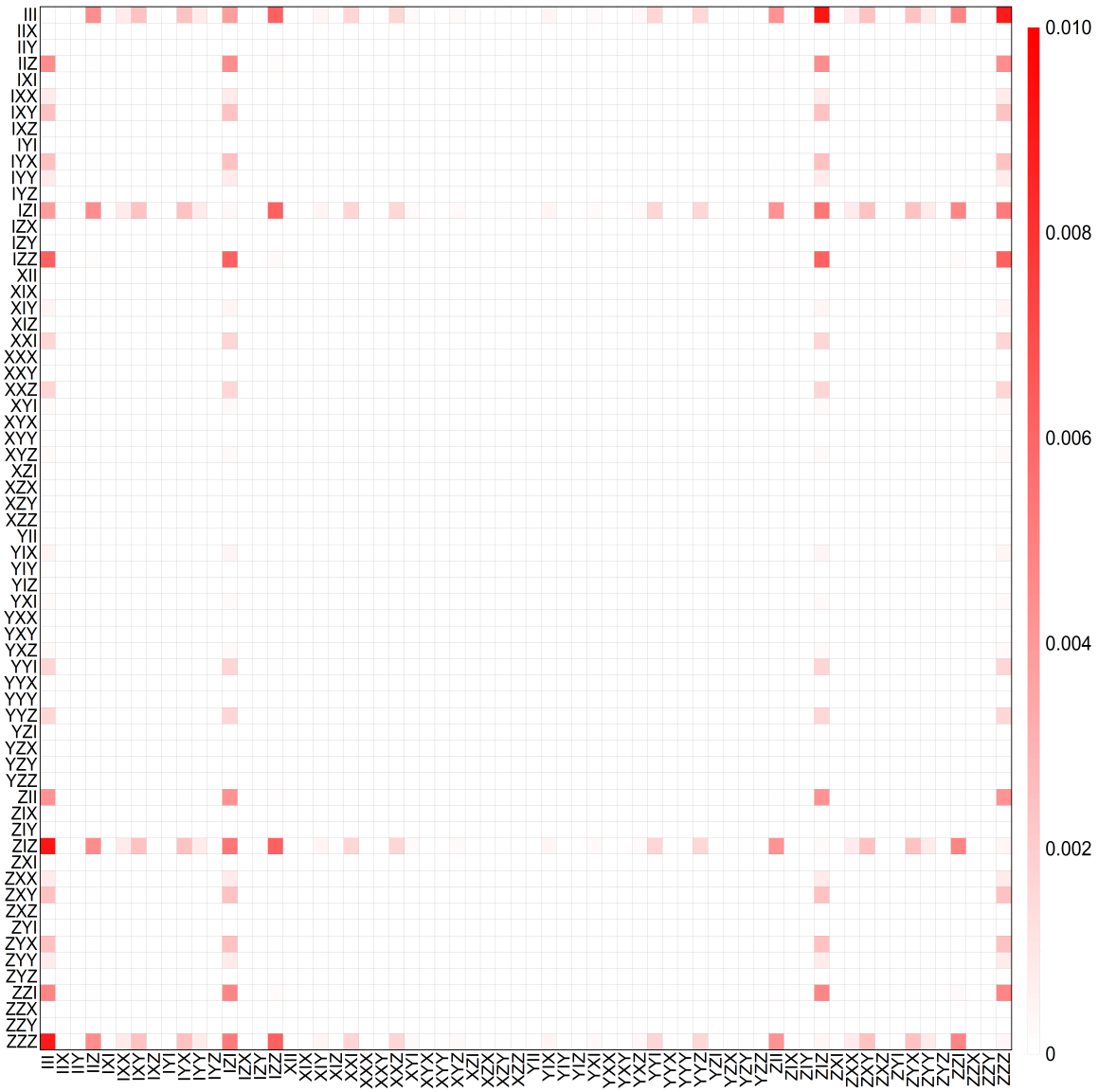


Figure 4.12 Absolute values of the differences between elements of ideal and simulated process matrices $\left[\chi^{(\text{ideal})} - \chi\right]$ for the ideal ParChe operation. Note the change in legend scale to increase clarity of error terms. The process matrix is expressed in terms of the complete basis set of 64 three-qubit Pauli matrices; I,X,Y,Z represent the matrices σ_0, σ_x, σ_y, σ_z.

References

[1] A. Blais, R. S. Huang, A. Wallraff, S.M. Girvin, R. J. Schoelkopf, "Cavity quantum electrodynamics for superconducting electrical circuits: An architecture for quantum computation", Phys. Rev. A 69, 062320, 2004.

[2] A. Wallraff, D. I. Schuster, A. Blais, L. Frunzio, R.- S. Huang, J. Majer, S. Kumar, S. M. Girvin & R. J. Schoelkopf , "Strong coupling of a single photon to a superconducting qubit using circuit quantum electrodynamics", Nature 431, 162–167, 2004.

[3] J. Koch, T. M. Yu, J. Gambetta, A. A. Houck, D. I. Schuster, J. Majer, A. Blais, M. H. Devoret, S. M. Girvin, R. J. Schoelkopf , "Charge-insensitive qubit design derived from the Cooper pair box", Phys. Rev. A 76, 042319 (2007).," Phys. Rev. A 76, 042319, 2007.

[4] H. Paik, D. I. Schuster, L. S. Bishop, G. Kirchmair, G. Catelani, A. P. Sears, B. R. Johnson, M. J. Reagor, L. Frunzio, L. I. Glazman, S. M. Girvin, M. H. Devoret, R. J. Schoelkopf, "Observation of High Coherence in Josephson Junction Qubits Measured in a Three-Dimensional Circuit QED Architecture", Phys. Rev. Lett. 107, 240501, 2011.

[5] R. Barends, J. Kelly, A. Megrant, D. Sank, E. Jeffrey, Y. Chen, Y. Yin, B. Chiaro, J. Mutus, C. Neill, P. O'Malley, P. Roushan, J. Wenner, T. C. White, A. N. Cleland, and John M. Martinis, "Coherent Josephson Qubit Suitable for Scalable Quantum Integrated Circuits", Phys. Rev. Lett. 111, 080502 (2013).," Phys. Rev. Lett. 111, 080502, 2013.

[6] J. M. Gambetta, J. M. Chow, J. M. Steffen, "Building logical qubits in a superconducting quantum computing system", npj Quantum Inf. 3, 2,

2017.

[7] A. Blais, J. Gambetta, A. Wallraff, D. I. Schuster, S. M. Girvin, M. H. Devoret, and R. J. Schoelkopf, "Quantum-information processing with circuit quantum electrodynamics", Phys. Rev. A 75, 032329 – Published 22 March 2007.

[8] J. Preskil, "Quantum Error Correction," Physics 219, lecture notes chapter 7, Caltech, http://theory.caltech.edu/people/preskill/ph229/notes/chap7

[9] B. M. Terhal, "Quantum Error Correction for Quantum Memories", Rev. Mod. Phys. 87, 307 – Published 7 April 2015.

[10] L. DiCarlo, J. M. Chow, J. M. Gambetta, Lev S. Bishop, B. R. Johnson, D. I. Schuster, J. Majer, A. Blais, L. Frunzio, S. M. Girvin, R. J. Schoelkopf, "Demonstration of two-qubit algorithms with a superconducting quantum processor", Nature 460, 240–244 (2009).

[11] F. W. Strauch, Ph. R. Johnson, A. J. Dragt, C. J. Lobb, J. R. Anderson, and F. C. Wellstood, "Quantum Logic Gates for Coupled Superconducting Phase Qubits", Phys. Rev. Lett. 91, 167005, 2003.

[12] R. Barends et. al., "Superconducting quantum circuits at the surface code threshold for fault tolerance", Nature 508, 500–503, 2014.

[13] J. Ghosh, A. Galiautdinov, Z. Zhou, A. N. Korotkov, J. M. Martinis, and M. R. Geller, "High-?delity controlled-σ Z gate for resonator-based superconducting quantum computers", Phys. Rev. A.87.022309, 2013.

[14] A. Fedorov, L. Steffen, M. Baur, M. P. da Silva, A. Wallraff, "Implementation of a Toffoli gate with superconducting circuits", Nature. 2011 10.1038/na-

ture10713, 2011.

[15] M. D. Reed, L. DiCarlo, S. E. Nigg, L. Sun, L. Frunzio, S. M. Girvin & R. J. Schoelkopf, "Realization of three-qubit quantum error correction with superconducting circuits", Nature. 2012 10.1038/nature10786, 2012.

[16] E. Zahedinejad, J. Ghosh, B. C. Sanders , "High-Fidelity Single-Shot Toffoli Gate via Quantum Control", 10.1103/ PhysRevLett. 114.200502.

[17] E. Zahedinejad, J. Ghosh, B. C. Sanders, "Designing High-Fidelity Single-Shot Three-Qubit Gates: A Machine-Learning Approach", 10.1103/PhysRevApplied.6.054005.

[18] M. H. Goerz, F. Motzoi, K. B. Whaley, C. P. Koch, "Charting the circuit QED design landscape using optimal control theory", npj Quantum Information (2017) 3:37 ; doi:10.1038/s41534-017-0036-0, 2017.

[19] S. Machnes, U. Sander, S.J. Glaser, P. de Fouquières, A. Gruslys, S. Schirmer, and T. Schulte-Herbrüggen, "Comparing, optimizing, and benchmarking quantum-control algorithms in a unifying programming framework",

10.1103/PhysRevA.84.022305 (2011), 2011.

[20] R. J. Spiteri, M. Schmidt, J. Ghosh, E. Zahedinejad, and B. C. Sanders, "Quantum control for high-fidelity multi-qubit gates," 2018 New J. Phys. 20 113009, 2018.

[21] N. Khaneja, T. Reiss, C. Kehlet, T. Schulte-Herbrüggen, and S. J. Glaser, J. Magn. Reson, "Optimal control of coupled spin dynamics: design of NMR pulse sequences by gradient ascent algorithms",

DOI: 10.1016/j.jmr.2004.11.004, 2004.

[22] N. Rach, M. M. Müller, T. Calarco, and S. Montangero, "Dressing the chopped-random-basis optimization: A bandwidth-limited access to the trap-free landscape", Phys. Rev. A. 92, 062343 (2015).

doi:10.1103/PhysRevA.92.062343, 2015.

[23] R. Versluis, S. Poletto, N. Khammassi, N. Haider, D. J. Michalak, A. Bruno, K. Bertels, L. DiCarlo, "Scalable Quantum Circuit and Control for a Superconducting Surface Code", 10.1103/PhysRevApplied.8.034021.

[24] S. B. Kitaev, A. Y. Bravyi, "Quantum codes on a lattice with boundary," arXiv:quant-ph/9811052, 1998.

[25] E. Dennis, A. Y. Kitaev, A. Landahl, and J. Preskill, "Topological Quantum Memory," J. Math. Phys. 43, 4452, 2002.

[26] 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 – Published 18 September 2012.

[27] P. Kumar, S. Daraeizadeh, "Parity-based mirror inversion for ef?cient quantum state transfer and computation in nearest-neighbor arrays," Phys. Rev. A. 91. 042310, 2015.

[28] G. Yang, W. N. N. Hung, X. Song, M. Perkowski, "Majority-based reversible logic gates," TheoreticalComputerScience334(2005)259–274, 2005.

[29] S. A. Cuccaro, T. G. Draper, S. A. Kutin, D. P. Moulton, "A new quantum ripple-carry addition circuit," 8th Workshop on Quantum Information Processing arXiv:quant-ph, 0410184v1, 2005.

[30] C. Albanese, M. Christandl, N. Datta, and A. Ekert, "Mirror Inversion of Quantum States in Linear Registers," Phys. Rev. Lett. 93, 230502 , 2004.

[31] V. V. Shende, I. L. Markov, "On the CNOT-cost of TOFFOLI gates", Vols. Volume 9 Issue 5, May 2009 Pages 461-486.

[32] A. Barenco, C. H. Bennett, R. Cleve, D. P. DiVincenzo,N. Margolus,P. Shor, T. Sleator, J. A. Smolin, and H. Weinfurter, "Elementary gates for quantum computation", PHYSICAL REVIEW A VOLUME 52, NUMBER 5, 1995.

[33] S. Richer, "Perturbative analysis of two-qubit gates on Transmon qubits", Masters thesis, RWTH Aachen Univer, 2013.

[34] N. Hatano, M. Suzuki, "Finding Exponential Product Formulas of Higher Orders," In: Das A., K. Chakrabarti B. (eds) Quantum Annealing and Other Optimization Methods. Lecture Notes in Physics, vol 679. Springer, Berlin, Heidelberg, 2005.

[35] M. H. Yung, J. D. Whitfield, S. Boixo, D. G. Tempel, A. Aspuru-Guzik, "Introduction to Quantum Algorithms for Physics and Chemistry", Quantum Physics (quant-ph);Mesoscale and Nanoscale Physics, Advances in Chemical Physics Volume 154 (ed S. Kais), John Wiley & Sons, Inc., Hoboken, New Jersey (2014), 10., 2014.

[36] L. H. Pedersen, K. Molmer, N. M.Moller, , "Fidelity of quantum operations", Phys Lett. A, 367, 47 (2007).

[37] R. Storn, K. Price, "Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces", Journal of Global Optimization (1997) 11: 341. https://doi.org/10.1023/A:1008202821328, 1997.

[38] M. A. Rol, F. Battistel, F. K. Malinowski, C. C. Bultink, B. M. Tarasinski, R. Vollmer, N. Haider, N. Muthusubramanian, A. Bruno, B. M. Terhal, and L. DiCarlo, "A fast, low-leakage, high-?delity two-qubit gate for a programmable superconducting quantum computer", arXiv:1903.02492v1 , March 2019.

[39] "Coeus Cluster," [Online]. Available: http://www.pi4cs.org/equipment.

[40] M. A. Nielsen, I.L. Chuang, "Quantum Computation and Quantum Information", CambridgeUniversity Press, Cambridge, 2001.

[41] J. L. O'Brien, G. J. Pryde, A. Gilchrist, D. F. V. James, N. K. Langford, T. C. Ralph, A. G. White, "Quantum Process Tomography of a Controlled-NOT Gate", Phys. Rev. Lett., vol. 93, no. 8, p. 080502, Aug. 2004.

[42] S. P. Premaratne, J. H. Yeh, F. C. Wellstood, B. S. Palmer, "Implementation of a generalized controlled-NOT gate between fixed-frequency transmons", Phys. Rev. A, vol. 99, no. 1, p. 012317, Jan. 2019.

[43] G. Lindblad, "On the generators of quantum dynamical semigroups", Commun. Math. Phys., vol. 48, no. 2, pp. 119–130, Jun. 1976.

[44] A. Kossakowski, "On quantum statistical mechanics of non-Hamiltonian systems Reports", Math. Phys., vol. 3, no. 4, pp. 247–274, Dec. 1972.

[45] G. Jaeger, and K. Ann, "Disentanglement and decoherence in a pair of qutrits under dephasing noise", J. Mod. Opt., vol. 54, no. 16–17, pp. 2327–2338, 2007.

[46] Y. Yang, and A. M. Wang, "Quantum discord for a qutrit-qutrit system under depolarizing and dephasing noise", Chinese Phys. Lett., vol. 30, no.

8, 2013.

[47] M. D. Hutchings, Jared B. Hertzberg, Yebin Liu, Nicholas T. Bronn, George A. Keefe, M. Brink, Jerry M. Chow, B. L. T. Plourde, "Tunable Superconducting Qubits with Flux-Independent Coherence", Phys. Rev. Appl., vol. 8, no. 4, p. 044003, Oct. 2017.

[48] N. Khammassi, I. Ashraf, X. Fu, C. G. Almudever and K. Bertels, "QX: A high-performance quantum computer simulation platform" pp. 464-469. doi: 10.23919/DATE.2017.7927034, 2017.

[49] M. Silva, E. Magesan, D. W. Kribs, and J. Emerson, "Scalable protocol for identification of correctable codes" Phys. Rev. A 78, 012347 , 2008.

[50] J. Ghosh, A. G. Fowler, and M. R. Geller, "Surface code with decoherence: An analysis of three superconducting architectures", Phys. Rev. A 86, 062318, 2012.

[51] P. K. Sarvepalli, A. Klappenecker, and M. Rötteler, "Asymmetric quantum codes: constructions, bounds and performance", *Proc. R. Soc. A 465*, *1645*, 2009.

Acknowledgments

This work was made possible in part thanks to Portland Institute for Computational Science and its resources acquired using NSF Grant # DMS 1624776 and ARO Grant #W911NF-16-1-0307.

5 Conclusion and Future Work

5.1 Conclusion

In this research work, different software environments were developed to simulate the dynamics of the superconducting quantum systems. We used the software simulators to design, verify, and optimize new quantum gates and architectures. We explored three quantum gate design methodologies depending on the characteristics of the desired gate, the number of qubits involving in the gate operation, and the Hamiltonian describing the quantum system. The new gates proposed in this dissertation have many applications in quantum circuits and have significant impact on various design decisions including logic synthesis, quantum compiler/mapper design, quantum error correction codes, and quantum games and algorithms.

Designing scalable and efficient quantum architectures based on error correction codes is an active research area in the field of quantum information processing. In chapter 2, we used analytical methods to derive the required system parameters to realize five-qubit controlled-unitary entangling gates for two-dimensional nearest-neighbor quantum systems with non-tunable Ising type interactions. Then we used our new multi-qubit gates to realize an efficient and scalable quantum memory based on Surface Code. The derived system parameters were confirmed to achieve high fidelity gates in a simulated environment, and the sensitivity of the gates to the system parameters were evaluated. It is notable that adjusting only one control parameter is required to achieve high-

fidelity multi-qubit entangling gates, where these gates mimic the functionality of several gates at once. Therefore, utilizing these gates in our quantum memory results in better performance while the control electronics resources are extensively reduced. To our knowledge, this is the first time a memory architecture based on Surface Code is presented for the systems with always-on Ising interactions. Furthermore, the presented multi-qubit gates can be realized in other physical systems as well.

Moreover, realizing scalable quantum computers based on transmons in circuit Quantum Electrodynamics (cQED) systems is one of the most active areas of research in the field of quantum information processing. In chapter 3, the physics of superconducting systems based on transmons in cQED regime is explained. Moreover, the method of developing a quantum simulator for such systems is described. Using a heuristic methodology, the simulator is used to design two-qubit gates c $\sqrt{NOT}$ and c $\sqrt{NOT}^{\dagger}$ which are not among standard gates in a universal set of gates for quantum computation but they can extensively be used in quantum logic synthesis and quantum arithmetic circuits. Then for the first time in transmons in cQED systems, we utilized c $\sqrt{NOT}$ and c $\sqrt{NOT}^{\dagger}$ gates to realize an efficient quantum full adder circuit. Moreover, we mapped the full adder quantum circuit to the physical layout of an existing quantum chip, and we proposed a new connectivity for the quantum layout such that the full adder circuit can be efficiently realized.

In chapter 4, we used machine learning techniques and high performance computing in our simulation environment to design two realistic high fidelity three-qubit entangling gates with duration of only 50 ns. The theoretical results backed with extensive simulations presented in this chapter demonstrate the feasibility of implementing robust, high-fidelity, and fast single-shot three-

qubit entangling gates in quantum systems of transmons in cQED regime. To our knowledge, this is the first time that high fidelity three-qubit entangling gates taking realistic experimental constraints into account are reported for transmons in cQED systems.

5.1.1 Accomplishments

This research work has resulted in the following achievements and publications:

Achievements

- Invented a new multi-qubit gate for rf-SQUID-based quantum systems with non-tunable couplings
- Invented a new quantum memory based on surface code architecture for quantum systems with always-on Ising couplings which include the known rf-SQUID systems
- Designed a ccPhase gate for transmon in cQED regime technology that in combination with Hadamard gates can realize a high fidelity Toffoli gate in only 50 ns.
- Invented a new 3-qubit controlled-unitary phase gate named ParChe gate for transmon-based technology that in combination with Hadamard gates can perform parity check
- Invented a new Shor's 9-qubit code architecture based on the new ccPhase gate and new ParChe gate
- Invented a new Topological code based on multi-qubit gates that can cut the latency of the parity detection circuits to half

- Created several gate design simulators in Matlab and C++
- Created a linear algebra library (in C++) based on Dirac notation which can be used for simulation of different physical quantum systems
- Realized a full adder circuit for the first time on Transmons in cQED technology

5.1.2 Publications

- P. Kumar, S. R. Skinner and, S. Daraeizadeh, "Reduced Hamiltonian Technique for Gate Design in Strongly Coupled Quantum Systems", ICQNM 2011: The Fifth International Conference on Quantum, Nano and Micro Technologies,ISBN: 978-1-61208-151-9, 2011
- P. Kumar, S. R. Skinner, S. Daraeizadeh, "Nearest-Neighbor Architecture to Overcome Effects of Qubit Precessions in Gate Operations", 2011 IEEE Congress of Evolutionary Computation (CEC), DOI 10.1109/CEC.2011.5949960, 2011
- P. Kumar, S. R. Skinner, S. Daraeizadeh, "A Nearest Neighbor Quantum Architecture to Overcome Dephasing", Quantum Inf. Process. 12:157–188 DOI 10.1007/s11128-012-0365-z, 2013

  P. Kumar, S. Daraeizadeh, "Parity-based mirror inversion for efficient quantum state transfer and computation in nearest-neighbor arrays", Phys. Rev. A. 91. 042310, 2015
- S. Daraeizadeh, A. Matsuura, J. Hogaboam, published patent US 20190042974, Oct. 2018
- S. Daraeizadeh, A. Matsuura, C. Zou, S. Johri, submitted patent, Jan. 2019

- S. Daraeizadeh, S. Premaratne, A. Matsuura, M. Perkowski, “Designing Gates to Realize a Full Adder Quantum Circuit in cQED Transmon Systems”, Book Chapter in “Handbook: Nanoengineering, Quantum Sciences and Nanotechnolgies”, Editor: Sergey Lyshevski, CRC Press by Taylor and Francis Group, LLC, ISBN 9780367197513, Dec. 2019

- S. Daraeizadeh, S. Premaratne, N. Khammassi, X. Song, M. Perkowski, A. Matsuura, "Machine-learning-based three-qubit gate design for Toffoli and parity check in transmon systems", submitted to Phys. Rev. A. on Aug. 2019

- S. Daraeizadeh, S. Mostame, P. Kumar Eslami, M. Perkowski, X. Song, “Realization of Surface Code Quantum Memory on Systems with Always-On Interactions”, submitted to Phys. Rev. A. on Oct. 2019

5.2 Future work

- We proposed a Surface Code architecture for quantum systems with always-on (non-tunable) Ising interactions. The multi-qubit gates designed in chapter 2 as well as the presented Surface Code protocol can also be designed for transmon-based superconducting quantum systems.

- Using the simulator for transmons in cQED systems, we can design any controlled-$\sqrt[n]{\text{NOT}}$ gate through controlled-arbitrary-rotation gates. Moreover, the simulator can be utilized to design and verify different quantum circuits.

- We designed three-qubit entangling gates using a simulator for transmons with resonator couplings. The simulator developed in this work can be

extended to realize larger multi-qubit gates with higher number of transmons involved. Moreover, it can easily be extended to be used as an automated quantum gate design tool for other quantum physical systems as well.