Dancing the Quantum Waltz: Compiling Three-Qubit Gates on Four Level Architectures

Andrew Litteken
litteken@uchicago.edu
University of Chicago
Chicago, Illinois, USA

Lennart Maximilian Seifert
lmseifert@uchicago.edu
University of Chicago
Chicago, Illinois, USA

Jason D. Chadwick
jchadwick@uchicago.edu
University of Chicago
Chicago, Illinois, USA

Natalia Nottingham
nottingham@uchicago.edu
University of Chicago
Chicago, Illinois, USA

Tanay Roy
roytanay@fnal.gov
Fermilab
Batvia, Illinois, USA

Ziqian Li
ziqianli@stanford.edu
Stanford University
Stanford, California, USA

David Schuster
dschus@stanford.edu
Stanford University
Stanford, California, USA

Frederic T. Chong
chong@cs.uchicago.edu
University of Chicago
Chicago, Illinois, USA

Jonathan M. Baker
jonathan.baker@duke.edu
Duke University
Durham, North Carolina, USA

ABSTRACT

Superconducting quantum devices are a leading technology for quantum computation, but they face several challenges. Gate errors, coherence errors and a lack of connectivity all contribute to low fidelity results. In particular, connectivity restrictions enforce a gate set that requires three-qubit gates to be decomposed into one- or two-qubit gates. This substantially increases the number of two-qubit gates that need to be executed. However, many quantum devices have access to higher energy levels. We can expand the qubit abstraction of $|0\rangle$ and $|1\rangle$ to a ququart which has access to the $|2\rangle$ and $|3\rangle$ state, but with shorter coherence times. This allows for two qubits to be encoded in one ququart, enabling increased virtual connectivity between physical units from two adjacent qubits to four fully connected qubits. This connectivity scheme allows us to more efficiently execute three-qubit gates natively between two physical devices.

We present direct-to-pulse implementations of several three-qubit gates, synthesized via optimal control, for compilation of three-qubit gates onto a superconducting-based architecture with access to four-level devices with the first experimental demonstration of four-level ququart gates designed through optimal control. We demonstrate strategies that temporarily use higher level states to perform Toffoli gates and always use higher level states to improve fidelities for quantum circuits. We find that these methods improve expected fidelities with increases of 2x across circuit sizes using intermediate encoding, and increases of 3x for fully-encoded ququart compilation.

ISCA '23, June 17–21, 2023, Orlando, FL, USA

ACM ISBN 979-8-4007-0095-8/23/06…$15.00
https://doi.org/10.1145/3579371.3589106

CCS CONCEPTS

• **Hardware** → **Quantum computation**; • **Computer systems organization** → **Quantum computing**.

KEYWORDS

quantum computing, qudit, compilation

ACM Reference Format:
Andrew Litteken, Lennart Maximilian Seifert, Jason D. Chadwick, Natalia Nottingham, Tanay Roy, Ziqian Li, David Schuster, Frederic T. Chong, and Jonathan M. Baker. 2023. Dancing the Quantum Waltz: Compiling Three-Qubit Gates on Four Level Architectures. In *Proceedings of the 50th Annual International Symposium on Computer Architecture (ISCA '23), June 17–21, 2023, Orlando, FL, USA.* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3579371.3589106

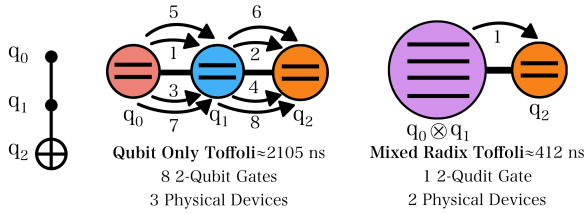

Figure 1: A comparison of a Toffoli gate execution on a three-qubit-only system versus a Toffoli gate execution on a ququart and qubit in a mixed-radix system. In a qubit-only system, we must use a decomposition that uses eight two-qubit gates that can be reduced to one two-qudit gate that has a shorter duration.

1 INTRODUCTION

Quantum systems are rapidly developing - stimulating the design and optimization of available hardware to maximize utilization of current resources for near-term quantum algorithms and the transition to quantum error correction [10, 18]. Error-prone gates,

sparse connectivity and low coherence times (the approximate computation time of any given device) are challenges currently facing quantum systems [49]. Even smaller quantum algorithms which fit on current hardware push devices to their limit. These limitations require improved optimization frameworks to make them useful in the near-term, prior to quantum error correction.

Success of quantum algorithms depends on how many error-prone gates are used and the total program duration. In most competitive quantum systems, e.g. superconducting systems, trapped ions, and neutral atoms, gates which act on many qubits simultaneously ($\geq$ 3 operands) must be decomposed, increasing both gate counts and circuit depth. In this work, we focus primarily on superconducting systems, where limited connectivity between devices further exacerbates the decomposition problem. Many circuits include gates, such as the Toffoli gate, to perform reversible arithmetic calculations; thus, three-qubit operations are common across implementations of quantum algorithms [4, 13, 21, 23]. Finding gate implementations without having to reduce them to more elementary gates saves valuable computational resources.

Currently, most quantum devices use qu*bits*, which have two energy levels, used to represent the $|0\rangle$ and $|1\rangle$ state. Recently, there have been several explorations into using the higher energy levels such as $|2\rangle$ and $|3\rangle$ to reduce the number of gates required to perform computation. While there are many examples of exploiting this concept of qu*dits*, such as using qutrits (3 logical states) to implement the multi-control Toffoli gate [19, 35], implementing higher-radix adders [55], and other applications [28], these use cases are the result of hand optimization, making their general use limited.

Another proposed use of higher-radix states is to fully encode data from two qubits into one physical unit with four logical levels, called a ququart. Previously, this strategy was avoided due to more error-prone operations [11, 56] and lower coherence times. Though coherence time is a limited resource, we can solve this problem by developing a set of operations which make better use of additional logical levels.

In this work, we observe that one ququart is equivalent to two qubits; thus the information of two qubit devices can instead occupy a single device which has access to four logical states [4]. This has significant advantages on the relative connectivity of the qubit information: by performing this compression, we can access three qubits worth of information by interacting only two physical devices in a single operation rather than directly interacting three physical devices in a single operation. This type of *mixed-radix* gate (four-level system interacting with an adjacent two-level system) is equivalent to performing a three-qubit gate. Similarly, we can consider two adjacent ququarts which allows us to perform interactions on up to four qubits worth of information by controlling only two physical devices; we call these *full-ququart* gates. Our strategy could remove the need to perform expensive decompositions of three- or four-qubit gates, as visualized in Figure 1 potentially improving circuit fidelity of circuits containing multiqubit gates through the direct execution of three-qubit gates. We are primarily focused on common three-qubit interactions since they appear more commonly in real applications, unlike four-qubit gates.

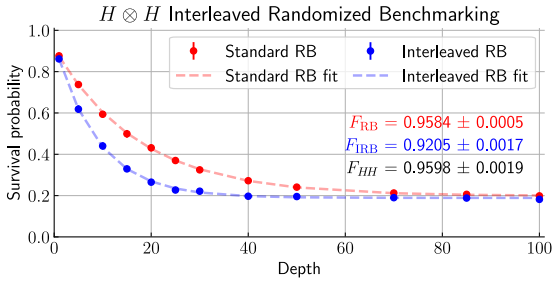

Figure 2: Interleaved Randomized Benchmarking for an optimal control $H \otimes H$ pulse on a superconducting transmon ququart following our qubit encoding. We use two-qubit Clifford sequences of gate depth up to 100 and average each data point over 10 samples. Error bars show the standard deviation of the mean but they are smaller than the mean markers. Red: Standard two-qubit Randomized Benchmarking to estimate the average Clifford gate fidelity to be $F_{\mathrm{RB}} \approx 95.8\%$. Blue: Interleaving the $H \otimes H$ pulse between the RB Cliffords yields a combined per-operation fidelity of $F_{\mathrm{IRB}} \approx 92.1\%$, resulting in an $H \otimes H$ fidelity $F_{HH} \approx 96.0\%$.

We examine using ququarts to dynamically encode and decode gates to perform native three-qubit gates on ququarts on a simulated superconducting device in a compilation pipeline called the Quantum Waltz, a dance done in three-four time. In particular, the major contributions are the following:

- A collection of mixed-radix and full-ququart gates that are logically equivalent to qubit-only gates, allowing for translation between qubit and mixed-radix operation.
- Demonstrating viability of ququart operations via pulses generated optimal control on hardware not previously designed for ququart pulses, Figure 2
- Identifying specific relationships between the controls and targets of three-qubit gates that allow for more efficient execution of mixed-radix and full-ququart gates with a compiler that choreographs three-qubit gates into particular configurations on ququarts for better performance and as a viable alternative to qubit-only strategies.
- Demonstrating, in simulation, how three-qubit gates on ququarts can achieve a 2x improvements in simulation in a mixed-radix environment and up to 3x fidelity improvements in a full ququart environment, as well as insights into the right situations to implement these gates.

2 BACKGROUND

2.1 Quantum Circuits

Quantum computation focuses on the use and manipulation of the qubit states $|0\rangle$ and $|1\rangle$, which can exist in a superposition of these states as $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ prior to measurement. N qubits exist in a superposition of 2^N basis states given by bitstrings of length N. These states are manipulated through the use of quantum logic gates in quantum circuits.

In principle, gates can act on any number of qubits. We mainly focus on single-, two- and three-qubit gate. Multi-qubit gates often use controls, meaning the state of another qubit only changes when

the value of the other qubits is in a specific state. For three-qubit gates, this can mean there are multiple controls, like the Toffoli gate shown in Figure 1. Similarly, multiple qubits can be controlled by one qubit. For comprehensive review of quantum gates we refer to [44].

2.2 Higher Radix Computation

Most abstractions of quantum computing are binary, focusing on the superposition of only two computational states. Many physical quantum technologies have access to higher energy levels which can be used to represent additional logical states as qu*dits* which use the lowest $d-1$ energy states which are increasingly harder to control. In this work, we constrain ourselves to at most four logical states, a *ququart*, which balances the potential computational benefit with its increasing error and time cost. In its naive use-case, additional levels have the same computational benefit as in classical - at most constant reductions in circuit depth and gate counts [46].

Some work [19, 35] has demonstrated specific applications that take advantage of extra computational states to reduce space requirements and improve execution time. These strategies are not generally applicable as it requires hand optimization for those circuits. Other work [4] attempted to generalize these improvements through compression, which stores multiple qubits worth of information in a smaller number of qudits. However, the usefulness of this strategy for general applications has not been explored and did not consider direct-to-pulse implementations of multi-qudit gates.

2.3 Quantum Optimal Control

The state of qudits is manipulated through external hardware-specific control fields $f_k(t)$. We consider superconducting devices, so these control fields are analog microwave pulses. Given a target unitary operation U, quantum optimal control finds controls f_k which realize U. Many optimal control algorithms and toolboxes have been developed [24, 30, 47, 54], and here we make use of the open-source software package Juqbox [47, 48]. We find control pulses of shortest duration which realize gates of interest up to competitive fidelity, 0.99 for two-qudit gates and 0.999 for single-qudit gates. Juqbox achieves this by minimizing the objective $J[f_k] = 1 - F[f_k] + L[f_k]$ where

$$F[f_k] = \frac{1}{h^2}\left|\mathrm{Tr}\left\{U_T^\dagger[f_k]\, V\right\}\right|^2 \tag{1}$$

quantifies the gate fidelity between target unitary V and the applied transformation $U_T[f_k]$. Here h is the Hilbert space dimension of the logical subspace (in our case $h = d$) and T denotes the allotted gate time. This task is solved by repeatedly solving the Schrödinger equation and adjusting the control fields to minimize J. Higher energy levels are sometimes included in the simulation in order to accurately capture their effect on the state evolution and reduce errors from truncating high-dimensional systems. These guard states are not logical states, therefore populating them is penalized with a leakage term $L[f_k]$. Currently, Juqbox only allows pulse optimization for a fixed gate time T, therefore we minimize pulse durations by applying an iterative re-optimization technique [51].

3 COMPRESSION AND GATE SET

3.1 Information Compression

Information *compression* in the context of this work refers to the storage of many qubits worth of information which deviates slightly from the typical classical understanding. The goal of this compression is to reduce the total number of physical units required to realize a given quantum algorithm.

Rather than designing algorithms that specifically use higher level states, we encode the data of two individual qubits into one four-level computational unit, called a ququart, given as $|\psi\rangle_4 = \alpha|0\rangle + \beta|1\rangle + \gamma|2\rangle + \delta|3\rangle$. This can be seen as equivalent to $|\psi_1\rangle_2 \otimes |\psi_2\rangle_2 = \alpha_1\alpha_2|00\rangle + \alpha_1\beta_2|01\rangle + \beta_1\alpha_2|10\rangle + \beta_1\beta_2|11\rangle$ by the following mapping:

$$|00\rangle \rightarrow |0\rangle \quad |01\rangle \rightarrow |1\rangle \quad |10\rangle \rightarrow |2\rangle \quad |11\rangle \rightarrow |3\rangle$$

Therefore, $\alpha = \alpha_1\alpha_2$, $\beta = \alpha_1\beta_2$ etc. This compression does not result in the loss of any information since the transformation from $|\psi_1\rangle_2 \otimes |\psi_2\rangle_2$ to $|\psi\rangle_4$ is unitary and therefore invertible. This follows a modification of the scheme from [4].

This compression does not require a circuit to explicitly use the $|2\rangle$ and $|3\rangle$ states for the compiler to make use of ququarts like in [19, 28, 56]. We are able to adapt qubit-compiler pipelines to compile a circuit and encode qubits into a ququart and keep track of the original qubits without requiring changes in the original circuit.

3.2 Qubit Gates on Ququarts

Past work has studied higher level systems by generalizing operations on a qubit circuit. For example, the X gate, is generalized to a $+1 \mod d$ instead, where d is the dimension of the qudit. Multi-qubit gates generalize similarly; for example a CNOT can be viewed as a $|1\rangle$-controlled $+1 \mod 2$ gate and therefore in general we can consider $|c\rangle$-controlled $+m \mod d$ gates, $0 \le c, m \le d-1$ [37].

While possible to use this generalized gate set to perform computation, it is not concise. For example, to perform a CNOT between the second encoded qubits encoded in different ququarts we would need to apply two $|1\rangle$-controlled +1 gates and two $|3\rangle$-controlled +1 gates. We could instead generate and calibrate a more expressive gate set that directly performs this operation.

We develop a gate set which performs qubit operations directly on ququarts. For a single-qubit gate U acting on two encoded qubits in the state $|q_0q_1\rangle$, we use the unitary $U^0 = U \otimes \mathbb{1}$ to act on qubit q_0, $U^1 = \mathbb{1} \otimes U$ to act on qubit q_1, and $U^{0,1} = U \otimes U$ to act on both qubits simultaneously.

For two-qubit gates, there are several important classes of operations. The first is the interaction between the two compressed qubits which we call an *internal* operation. For example, a CX^0 is a CNOT controlled on the second qubit targeting the first; this is equivalent to the single ququart gate which swaps the states $|1\rangle$ and $|3\rangle$. CX^1 controls on the first and targets the second encoded qubit. A SWAP operation exchanges the order of the encoding, i.e. $\mathrm{SWAP}|q_1q_2\rangle = |q_2q_1\rangle$. The second are gates which act on qudits in different, but adjacent, physical locations. These *partial* gates interact a non-encoded qubit and a qubit in an encoded pair in adjacent locations; all gates of this type we call *mixed-radix* gates. For these

Table 1: Durations for one-qubit, two-qubit and *i*Toffoli gates synthesized in qubit-only, mixed-radix and full-ququart environments.

(a) Qudit (ns)				(b) Qubit Only (ns)		(c) Mixed-Radix (ns)				(d) Full-Ququart (ns)			
U	35	U^0	87	CX_2	251	CX^{0q}	560	CX^{q0}	880	CX^{00}	544	CX^{01}	544
U^1	66	$U^{0,1}$	86	CZ_2	236	CX^{1q}	632	CX^{q1}	812	CX^{10}	700	CX^{11}	700
CX^0	83	CX^1	84	$CS_2^\dagger$	126	CZ^{q0}	384	CZ^{q1}	404	CZ^{00}	392	CZ^{01}	488
$SWAP^{in}$	78			$SWAP_2$	504	$SWAP^{q0}$	680	$SWAP^{q1}$	792	CZ^{11}	776	$SWAP^{00}$	916
				$i\text{Toffoli}_3$	912	ENC	608			$SWAP^{01}$	892	$SWAP^{11}$	964

gates, order matters, i.e. the gate behaves differently depending on which qubit is the target.

The four CX gates are $\{CX^{q0}, CX^{q1}, CX^{0q}, CX^{1q}\}$ where the first index indicates the control and the second the target object, and q is the qubit. We also define two mixed-radix SWAPs $\{SWAP^{q0}, SWAP^{q1}\}$ which are the same regardless of direction. The *full-ququart* gates follow from the mixed-radix gates defining the four CX gates: $\{CX^{00}, CX^{01}, CX^{10}, CX^{11}\}$ and three SWAPs: $\{SWAP^{00}, SWAP^{01}, SWAP^{11}\}$.

3.3 Generating Pulses

Using quantum optimal control we directly synthesize each of the gates in our new mixed-radix and full-ququart gate set and baseline comparisons. We use a realistic superconducting device Hamiltonian inspired by IBM hardware [52].

We consider up to three weakly coupled, anharmonic transmons [32]:

$$H(t) = \sum_{k=1}^{3}\left[\omega_k a_k^\dagger a_k + \frac{\xi_k}{2} a_k^\dagger a_k^\dagger a_k a_k\right] + \sum_{k=1}^{3}\sum_{l>k} J_{kl}(a_1^\dagger a_2 + a_2^\dagger a_1) + \sum_{k=1}^{3} f_k(t)(a_k + a_k^\dagger). \quad (2)$$

The static terms describe the individual qudits and their pairwise couplings, while the last term captures the effect of driving the system through external control fields $f_k(t)$. The transmons are designed with $|0\rangle$-$|1\rangle$ transition frequencies $\omega_1/2\pi = 4.914\,\text{GHz}$, $\omega_2/2\pi = 5.114\,\text{GHz}$, and $\omega_3/2\pi = 5.214\,\text{GHz}$, and with equal anharmonicities $\xi_k/2\pi = -330\,\text{MHz}$. We consider linear connectivity with static couplings given by $J_{12}/2\pi = J_{23}/2\pi = 3.8\,\text{MHz}$. The drive power is limited to $f_{\max} = 45\,\text{MHz}$ to avoid substantial leakage into higher energy states, and we restrict ourselves to the $k = 1$ subspace when synthesizing single-qudit gates.

A full list of the gates synthesized and the minimal found duration of these gates can be found in Table 1. We reiterate the importance of short gate times - quantum systems are subject to a variety of both coherent and incoherent errors. By minimizing the total execution time of any given gate we reduce the circuit duration, reducing the effects of incoherent noise.

The closed system considered does not account for the full dynamics of a real quantum device. We have not specifically optimized these pulses under a more detailed model due to the increased computational cost of these optimizations, especially for the large Hilbert spaces involved in two-qudit operations.

In Section 3.5, we use similar optimal control techniques to implement a single-ququart operation on an experimental device, showing that our methods and assumptions are realistic given a well-characterized machine.

3.4 Properties of Qubit Gates on Ququarts

Our gate set and mixed-radix architecture provides real advantages over typical qubit-only versions. Within each ququart, we have a pair of encoded qubits between which gates are 5x faster and 10x higher fidelity than qubit-only schemes. By using a single computation device, the total amount of control hardware required is reduced (at most by half). Additionally, we have much higher connectivity between qubits once they are encoded in ququarts. In a ququart-qubit pair, there are three computational qubits directly connected to one another. Between two ququarts, there are four fully connected computational qubits. This is higher relative connectivity compared to industry standards for superconducting: lines, grids, and heavy hex architectures. Improved connectivity reduces expensive qubit movement operations. These increased connections are demonstrated in Figure 3.

Compression is not without its downsides. In Table 1, we see mixed-radix and ququart gates take much longer than qubit based gates. Pulses must be more carefully designed, and leakage between states is more prominent, resulting in the longer gates times. Each increasing energy level has a shorter coherence time scaling with $1/k$ where k is the energy level. Shorter decoherence, combined with longer gate times, means using mixed-radix and ququart based gates is a delicate balancing act between increasing fidelity due to gate execution while not increasing error due to decoherence.

3.5 Experimental Demonstration of Single-Ququart Control

Driven by advantages found in theoretical studies [46], experimental researchers have explored the implementation of these higher-dimensional systems, leading to realizations of qutrit devices which manipulate the third energy level [17, 22, 26, 41, 50, 57]. These works show that including higher levels is possible although challenging due to higher susceptibility to noise and lower coherence times.

Motivated by the findings for ququart-specific applications we have been studying control of four energy levels in experiment on a physical device. We extend the capabilities of one qudit of the superconducting transmon device presented in [33, 34, 50] to include the fourth state. We implement two-qubit Randomized Benchmarking (RB) [38] on this single ququart following our encoding scheme.

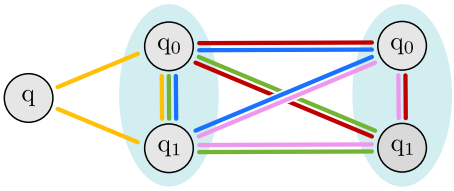

Figure 3: Visualization of connectivity advantages in qubit-ququart systems. Encoding qubits in ququarts (light blue) enables triangle connectivity between triples of qubits, where two of which are encoded in the same ququart and one appears either in a bare qubit or encoded in a neighboring ququart.

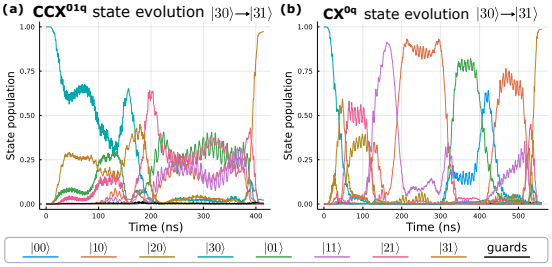

Figure 4: Visualization comparing the evolution of a $|3\rangle$-controlled X gate in a mixed-radix environment for a CCX gate in (a) and a CX gate in (b).

RB is a common method to characterize the average Clifford gate fidelity. This is achieved by executing Clifford circuits of varying depth, which perform the identity operation in the ideal case, and measuring the probabilities of the system returning to the ground state (survival probabilities). The fidelity can be extracted from exponential regression. The RB circuits are generated using Qiskit [1].

We additionally implement Interleaved Randomized Benchmarking (IRB) [39] to specifically find the fidelity of the single-ququart gate $H \otimes H$, which performs a Hadamard gate on each encoded qubit in parallel, used by the compiler below. The gate control pulse is designed using similar optimal control methods as discussed in Section 3.3 adapted to this experimental device.

Results from this work are shown in Fig. 2. We find an average Clifford gate fidelity of $F_{\text{RB}} \approx 95.8\%$ from normal RB while interleaving with the $H \otimes H$ gate yields $F_{\text{IRB}} \approx 92.1\%$ fidelity per operation. From that the specific gate fidelity, $F_{HH} \approx 96.0\%$, can be extracted. This first study shows that ququarts can be realized in experiment and optimal control yields high-quality pulses to manipulate their state. At the time of this writing we are not aware of any comparable demonstration. We are convinced that the fidelities can be improved with more carefully engineered ququart devices and more sophisticated pulse design methods.

4 THE QUANTUM WALTZ: THREE QUBIT GATES ON QUQUARTS

Three-qubit gates are widely used in arithmetic operations, such as the Cuccaro adder [13] and multi-controlled-CNOT [7], as smaller pieces in larger quantum algorithms such as [23]. While QAOA and VQE see more use in the current quantum algorithm space, some QAOA based algorithms still use three-qubit gates [25]. Additionally, some error correction schemes make heavy use of three-qubit gates [58]. Current hardware platforms typically decompose these gates. Using higher radix we can reduce gate times mitigating the issue of reduced coherence times of higher-energy levels enabling more efficient execution of quantum circuits containing these gates.

4.1 Connectivity Advantage

The set of two qubit gates laid out in Section 3.2 are enough to universally perform general qubit computation on ququarts [37, 44], but simply compiling to two-qubit gates would not take advantage of the flexibility of this abstraction. When we encode two qubits in a ququart, we virtually increase the connectivity between qubits, see Figure 3. Each of the encoded qubits in a ququart is connected to an adjacent qubit, or both of the encoded qubits in an adjacent ququart. As highlighted by each of the different colors this creates many triangle subgraphs between encoded qubits . Triangle subgraphs are uncommon in current hardware due to the increased probability of crosstalk [14, 42]. But, triangle-based interactions are common in many different circuits that use three-qubit gates. Here, we increase the number of virtual connections without increasing number of physical connections to create four interactions between encoded qubits.

It is not fundamentally harder to interact three or four qubits worth of information than two qubits worth with a single operation on ququarts. These gates are equivalent to either mixed-radix or full-ququart gates. For example, if we have a fully encoded ququart next to a bare qubit and perform a Toffoli gate targeting the qubit, it is equivalent to a $|3\rangle$-controlled X gate on the qubit. This is computationally simpler than the several $|1\rangle$- and $|3\rangle$-controlled X required in the decomposition and can be seen in the state evolutions in Figure 4. This gate implementation gives superconducting qubits more natural access to the native multi-qubit gates, avoids decompositions that add extra gates and performs three-qubit interactions between two physical quantum devices, reducing the complexity of implementing such a three-qubit pulse across three devices and two couplers. Used in conjunction with the previously generated one- and two-qubit gates, we can more efficiently perform circuits that include three-qubit gates.

4.2 Generated Pulses

4.2.1 Multi-control Gates. Native three-qubit gates on two physical units have the potential to offer a significant improvement in gate fidelity and execution time. In Table 2, we show pulse durations of the three-qubit Toffoli gate in several mixed-radix and full-ququart configurations. These gates were synthesized using the same fidelity targets and pulse generation techniques as the two-qubit gates, a higher fidelity than if decomposed with many gates of the same target fidelity. After synthesizing the different configurations of Toffoli gates, we find that there is a substantial difference in the gate duration depending on which qubits are controls and which is the target.

Table 2: Mixed-Radix and Full-Ququart Three-Qubit Gate Durations

(a) Mixed-Radix (ns)		(b) Full-Ququart (ns)			
CCX^{q01}	619	$CCX^{01,0}$	536	$CCX^{01,1}$	552
CCX^{1q0}	697	$CCX^{0,01}$	785	$CCX^{0,10}$	785
CCX^{01q}	412	$CCX^{1,10}$	785	$CCX^{1,01}$	680
CCZ^{01q}	264	$CCZ^{01,0}$	232	$CCZ^{01,1}$	310
$CSWAP^{01q}$	684	$CSWAP^{01,0}$	680	$CSWAP^{01,1}$	744
$CSWAP^{10q}$	762	$CSWAP^{10,0}$	758	$CSWAP^{10,1}$	822
$CSWAP^{q01}$	444	$CSWAP^{0,01}$	510	$CSWAP^{1,01}$	432

Consider the mixed-radix example where both control qubits are encoded in the same ququart, and the target qubit is in the bare qubit, or the CCX^{01q} gate, seen in Figure 5a. This configuration is about two-thirds the time of the CCX^{0q1}, seen in Figure 5b, where the control qubits are split across the bare qubit and the ququart. The reason for this difference is twofold. The first follows from the two-qubit only gates. Gates which use the ququart as a control and the qubit as a target are generally faster, the pulse only induces population changes between the $|0\rangle$ and $|1\rangle$ state of the qubit, rather than between $|0\rangle$ and $|1\rangle$, and $|2\rangle$ and $|3\rangle$ of the ququart. The second is that the entire ququart acts as the control, only changing the state of the bare qubit if the ququart is in the $|3\rangle$ state. In the split-control case, the ququart must control on both the $|2\rangle$ and $|3\rangle$ state.

The same concept of separation of controls and targets follows for the full-ququart Toffoli gates as well. Regardless of whether the target qubit is in the first or second encoding of the ququart, it is substantially faster to keep the controls encoded in the *same* ququart with the target encoded in a separate ququart.

4.2.2 Target-Independent Gates. Separating the controls and targets into different devices yields more efficient gate execution; however, compiling circuits to conform to this configuration is unnecessarily constraining. Instead, we consider a situation where all multi-qudit gates are *target-independent* and only affect the global state when all three qubits are in $|1\rangle$. For example, the Toffoli gate, or CCX, is locally equivalent to CCZ which is target-independent, as seen in Figure 6c.

When pulses are synthesized, CCZ is much more efficient as seen in Table 2, remarkably on par with the speed of the qubit only gates. In addition, we only need to define three configurations: $CCZ^{q,01}, CCZ^{0,01}, CCZ^{1,01}$, as opposed to the nine possible CCX configurations, reducing computational overhead. We postulate the short duration of these gates is because CCZ only changes the phase of the entire three-qubit state rather than the population. This makes the CCZ a valuable tool when compiling three-qubit gates.

4.2.3 Multi-target Gates. We also consider gates that use one control qubit to affect the state of some number of other qubits, for example the CSWAP. With our methods we synthesize gates to the same fidelity targets as before and show their times in Table 2. We find benefits when separating the control qubit from the target qubits as depicted in Figure 5c versus Figure 5d. When both targets

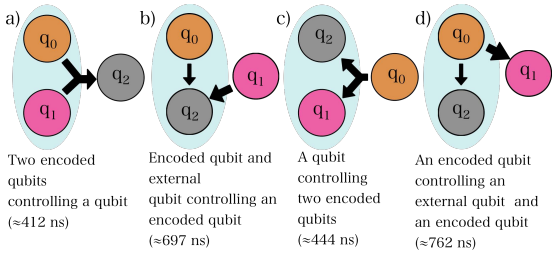

Figure 5: Examples of mixed-radix two-control and two-target gates. a) A configuration where both controls are encoded in the ququart and the target is mapped to a qubit. b) A configuration where the controls are split across the qubit and the ququart and the target is encoded in the ququart. c) A configuration where both targets are encoded in the ququart and the control is mapped to the qubit. d) A configuration where the targets are split across the qubit and the ququart and the control is encoded in the ququart.

are encoded the same ququart, we limit the state changes to be between $|1\rangle$ and $|2\rangle$ in that ququart.

5 COMPILATION STRATEGIES

5.1 Using Three-Qubit Gates

In our qubits-on-ququarts compilation strategy, we expand the physical connectivity graph between the ququarts on a given architecture and treat each ququart as two connected qubits. Each qubit in the expanded ququart is fully connected to the qubits in the neighboring ququarts as shown in Figure 3. We call this new graph the interaction graph; it maintains a mapping of where circuit qubits are mapped to on this graph. When one or fewer of the qubits in the expanded ququart is mapped to, the entire ququart is in a qubit state. Otherwise, it is considered to be in the ququart state.

To execute three-qubit gates, circuit qubits must be routed into a connected subgraph of the interaction graph, e.g. for $CCZ(q_0, q_1, q_2)$ requires $q_0 \sim q_1$ and $q_1 \sim q_2$ but it is not guaranteed that $q_0 \sim q_2$, where $\sim$ defines adjacency. We develop a compiler optimization which appropriately performs routing and gate selection based on this adjacency and use of higher dimension. While we are able to perform any configuration of three qubit gates directly in mixed-radix or full-ququarts scenarios, we take care to use best configurations to minimize time in the less stable $|2\rangle$ or $|3\rangle$ states.

5.1.1 Qubit-Only. In a qubit-only regime we can use a decomposition into eight CX operations [53]. This decomposition has the flexibility of being target-independent from a compilation standpoint. This is an expensive compilation, requiring eight two-qubit gates and 14 one-qubit gates. But, it does not use the less stable $|2\rangle$ and $|3\rangle$ state. Alternatively, we can use a directly-optimized three-qubit pulse sequence. QOC software failed to find a solution for a direct CCZ operation, so we synthesize a pulse implementing the *i*Toffoli gate using a three-qubit version of our quantum optimal control software that only uses the first two levels of the qubits and use the decomposition shown in Figure 6d inspired by [31] to execute a complete Toffoli gate.

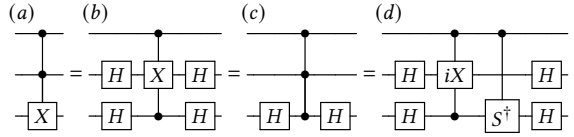

Figure 6: Different decompositions for the Toffoli Gate. a) is the base Toffoli circuit. b) is Toffoli circuit with a swapped second control and target from the original. By surrounding the control and the target with Hadamards, we perform the same operation. c) The Toffoli gate constructed from a CCZ gate which can be used as a Toffoli by surrounding the target with Hadamard gates. d) The Toffoli gate constructed from an iToffoli gate, which requires an controlled $S^\dagger$ gate in addition to Hadamard gates.

5.1.2 Intermediate Mixed-Radix. We also permit *temporary* use of the higher energy levels to perform an operation. By performing an encoding gate (ENC) followed by the three-qubit gate and a final decode ($ENC^\dagger$) operation, we get temporary access to full connectivity to perform fast three-qubit gates.

The compiler should opt to encode qubits of similar *type*, i.e. both controls together or both targets together. Let $U(q_0, q_1, q_2)$ be the operation with q_0, q_1 the same (either both controls or both targets). In some cases, encoding is simple because the routing strategy (prior work) results in $q_0 \sim q_1$ as in 5(a). However, it may fail to do this by default and we may have $q_0 \sim q_2$ and $q_1 \sim q_2$ as in 5(b).

We have three options to compile to a favorable configuration. First, we could enforce the ideal relationship through additional gates by adding an additional SWAP(q_0, q_2). Second, in the special case where $U = X$ Toffoli we can change which pair is the same type with Hadamard gates as in Figure 6b to use the most efficient implementation; we call this *re-targeting*. Third, if U permits, we transform U into U' so that q_0, q_1, q_2 are all the same type; for example we transform CCX to CCZ so each operand is a "control," Figure 6c. While the additional re-targeting or transformation gates add both error and duration, they enable the shortest duration version of U to be used for an overall net increase in fidelity. We consider the special cases of $U \in \{CCV|V \in SU(2)\}$, i.e the set of locally equivalent gates to CCX. We leave the generalized case to future work in circuit synthesis.

5.1.3 Full Ququart. Mixed-radix three-qubit gate strategies apply for full-ququart compilation as well. However, the router by default, described below, does not distinguish control or target. When executing three-qubit gates, we ensure only qubits of the same type are encoded if it does not require an extra swap operation.

5.2 Mapping and Routing

Our compilation for encoded qubits on ququart architectures is similar to previous compilation strategies for qubits as seen in many previous works [12, 15, 43] and adapts them to three-qubit gates on ququart architectures. However, unlike these prior works, we take into account the varying fidelities and durations of internal ququart versus mixed-radix versus full-ququart inter-ququart gates, similar to [35].

The first step is to decompose the operations in the circuit to native gates supported by the device. Our compiler handles the native execution of three-qubit gates, we decompose to the CX, CCX, CCZ or CSWAP along with a parameterized single-qubit rotation gate.

Qubits are mapped onto the interaction graph with the goal of maximizing locality. We assign a weight between each pair of qubits in the original circuit according to: $w(i,j) = \sum_{t \in C} o(i,j,t)/t$, where the sum is over each time step t in the circuit C and $o(i,j,t) = 1$ if qubits i, j interact in time step t and 0 otherwise. This weight includes lookahead functionality by weighting future interactions (larger t) smaller. The first qubit is mapped according to which has greatest total weight to all other qubits:
$\underset{i}{\text{argmax}} W(i) = \sum_{j \in Q_c/\{i\}} w(i,j)$. This qubit is placed in the first encoded location of the center-most qudit on the connection graph. For each other qubit, we choose the circuit qubit that has the greatest W with respect to the placed qubits. For each adjacent qubit, n, to the placed qubits, we compute $\sum_{j \in Q_P} w(i,j) d(n, \varphi(j))$ where φ is the mapping of circuit qubits to physical qubits, and d is a specialized fidelity function between the qubits estimating the possibility of error along the communication path. We then map the qubit to the minimizing location.

When routing, we track the circuit qubits on the interaction graph and use SWAP gates until the interacting qubits are adjacent. We attempt to disrupt advantageous qubit layouts as little as possible by using adaptive weights that change as operations are scheduled based on [6]. This strategy attempts to keep qubits interacting in the near future close to one another where the disruption of each potential SWAP between circuit qubits i, j is calculated by $D(i,j) = \sum_{k \in Q_c} w(i,k)(d(\varphi(i),\varphi(k)) - d(\varphi(j),\varphi(k))) + w(j,k)(d(\varphi(j),\varphi(k)) - d(\varphi(i),\varphi(k)))$. However, rather than using simple distances, we use the same specialized distance metric incorporating the previous function d. We choose the SWAP candidate that minimizes this value while always moving the qubit closer to the other qubits it needs to interact with. To generalize to three-qubit based routing we modify the cost function to $C(i) = \sum_{j \in Q_o/i} D(i, \varphi^{-1}(n))(d(\varphi(i),\varphi(j)) - d(n,\varphi(j))$ where Q_o is now a set of all operands.

It would be reasonably simple to extend this compiler design to accomodate k-qubits on n-d-level-qudits, where we pack each qudit with $log_2(d)$ qubits, and ensure that there is no way to move any one qubit closer to another in a fully connected set of qubits. However, we only explore three-qubit gates on a maximum of two, four-level devices, or three, two-level devices in this work. This is for design and practical reasons. From a design point of view, our gate set and compiler are intended to be used after a circuit has been translated into qubit-based gates. Compiling natively to higher-radix qudit operations would require a much larger set of basis gates than the qubit-based set we use here. Additionally, there is not a standard set of four-or-more qubit gates that are typically used in circuits, meaning there would have to be some arbitrary decomposition to four-qubit gates, rather than three qubits. Choosing a basis gate set is a time intensive process and has to be done selectively [20]. We therefore expand the normal one and two-qubit framework used in many compilers to include the most commonly used three-qubit gates as it is the most common multi-qubit gate.

It should be noted that all translation to higher-radix devices occurs during this compilation step. The general programmer still writes a program in terms of qubits. The compiler translates the program into the correct sequence of qubit-on-ququart operations to perform the same computation. In the case of full-ququart operation, the measured state would be decoded according to the compression strategy.

6 EVALUATION

6.1 Circuits

We examine five three-qubit based circuits that can be parameterized by number of qubits with different constructions. The first is the Generalized Toffoli (CNU) circuit [5], which flips the state of a target qubit if all the controls are one. This circuit uses exclusively Toffoli gate based decomposition and is highly parallel. The Cuccaro Adder [13] is nearly entirely serialized using $2n + 2$ qubits with a mix of three-, two- and single-qubit gates to add two n-bit numbers. Third is a QRAM circuit which uses primarily CSWAP gates to retrieve data from or move data into a set of qubits [21]. The fourth is a Select circuit, which is a preparation mechanism used in Quantum Phase Estimation (QPE) [36]. It performs a particular Pauli operation on n qubits for each potential 2^m states of m index qubits [3]. For our case, the choice of Pauli string does not affect compilation. To keep the fidelity of circuit simulation within comparable bounds, we only select on two random values rather than all of the potential 2^m values the index qubits could be in. The fifth is a purely synthetic circuit to study relative strength of our architecture on potential distributions of CX versus CCX gates.

6.2 Baselines, Hardware Topology and Error

We compare against two strategies. The first is a compilation that routes the circuit with three-qubit gates, before decomposing them to one- and two-qubit gates only. This is in line with current practices for most compilation pipelines. The second baseline does not decompose to these smaller gates. Instead, the *i*Toffoli-based decomposition is executed directly on qubits similar to [31]. This is a more challenging gate to synthesize, as discussed previously. For simulation, this gate has a 99% fidelity and with 912 ns duration determined via the same quantum optimal control strategies as the mixed-radix and full-ququart gates, and the $CX^\dagger$ gate has a duration if 126 ns. Additionally, we use the Hadamard-based retargeting technique to ensure that we are applying the Toffoli gate to the correct qubit without an extra SWAP. This allows us to always use the demonstrated *i*Toffoli gate where the target qubit is the center of three connected qubits.

We consider the same underlying hardware topology for each comparison point - a 2D mesh. This type of grid architecture has relative density on the upper end of realized superconducting connectivity graphs, reflective of Google's Sycamore chip [2] and more dense than IBM's heavy-hex [18]. We consider a grid design with dimensions $\lceil\sqrt{n}\rceil \times \frac{n}{\lceil\sqrt{n}\rceil}$ with nearest neighbor connectivity.

We use a realistic T1 time from an IBM device of 163.45μs [27]. Higher energy levels decohere more quickly. In theory, each state decays at a rate of $o(1/k)$ where k is the energy level as discussed in [59]. We therefore use 81.73μs and 54.15μs as the T1 times for the $|2\rangle$ and $|3\rangle$ states. As any transmon technically has access to these higher energy states, we do not expect that a device designed to access these higher-energy states will reduce the base T1 time.

6.3 Circuit Estimation

We use two metrics to estimate the fidelity of a circuit without simulation to extrapolate how compiled circuits may perform by comparing simulation to estimation. The first is the product of all of the gate success rates in the circuit, called the gate expected probability of success (gate EPS). Since there are multiple classes of multi-qubit gates, some of which have higher fidelity than others, we use the product of these success rates.

Second, we model decoherence as an exponential decay where the probability of no decoherence is $\prod_{k=1}^{3} \exp(k * t_k/T_1)$ where t_k is the time the qubit spends in state k. When we construct the circuit we keep track of how long each qudit exists in the $|1\rangle$ or $|3\rangle$ state as the maximum state and calculate the probability of not decohering over the course of the execution for each qudit. The product of the expected success of each qudit is the EPS due to coherence for the entire circuit. When multiplied by the gate EPS, we have the EPS for the entire circuit.

6.4 Circuit Simulation

Since access to ququart devices at this scale are limited, we must use simulation to evaluate the performance of our approach. We use the trajectory method [8] for improved scalability compared to full density simulation. This work simulates circuits of up to 24 qubits (or, equivalently, 12 ququarts). For this work, for each circuit, we generate at least 1000 random quantum states and for each we simulate once and compute the average fidelity over all random states. We emphasize the use of random *quantum* states as classical inputs are not always affected by quantum errors.

In the past, prior work on simulation of qudit systems neglects the realistic duration differences between gates which results in drastically different usage patterns and simply injecting errors on a moment-to-moment basis can skew results. For example, in this work our direct-to-pulse compilation of CCX and CCZ gates have significantly different execution times. We modify the trajectory method simulation slightly to account for this difference. Rather than inserting many idle gates during each time step, before each gate, we insert one idle gate using the exact time that qudit has been idle. This is a more accurate representation of from which state these qudits could be decohering.

6.5 Noise Model for Qudit Systems

For qubits we consider both symmetric depolarizing and amplitude damping errors. There are four possible single-bit channels: no error (I), bit flip errors (X), phase flip errors (Z) and bit and phase flip errors ($Y = ZX$). In simulation each error channel is drawn with probability $p/3$. Two-qubit errors are given as the product of single-qubit errors, e.g. $X \otimes X$ for a bit flip on both interacting qubits; there are 16 possible channels of this type so each error occurs with probability $p/15$ and no error ($I \otimes I$) occurs with probability $1 - 15p$.

For a general qudit system, we consider a generalized form of these errors. The "bit-flip" type gates become $X_{+1 \bmod d}$ and the "phase-flip" errors become $Z_d = \text{diag}(1, \exp\{\omega\}, \exp\{\omega^2\}, ...,$

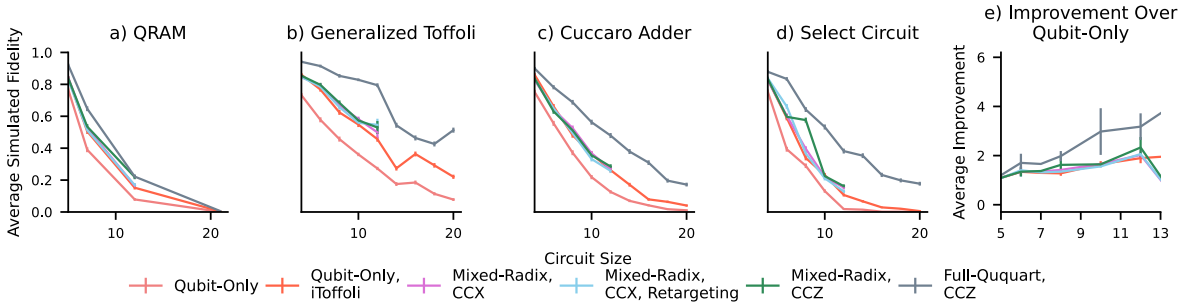

Figure 7: Simulated results for QRAM, Generalized Toffoli, Cuccaro Adder and Select Circuit from 5 to 21 qubits with different mixed-radix and full-ququart compilation strategies. The mixed-radix strategies do not have complete error bars due to the requirement to simulate a four-level system for every qubit which would require more than 86 GB of memory per circuit in our simulation framework. The final graph is the average fidelity improvement for each compilation method over the qubit-only compilation method as the size of the circuit increases.

$\exp\left\{\omega^{d-1}\right\}$) where ω^j is the $j-th$ root of unity. The product of $\{I, X_{+1\bmod d}, ..., X_{+1\bmod d}^{d-1}\}$ and $\{I, Z_d, Z_d^2, ..., Z_d^{d-1}\}$ is a basis for all $d \times d$ Pauli matrices which allows us to construct a general symmetric qudit depolarizing channel. This explains the expected increase in error for using qudit systems: For a two-qubit gate the chance of *no* error is $1-15p$ while for a ququart this chance diminishes to $1-255p$ let alone possible differences in p [40].

Amplitude damping for qubits can be described as non-unitary transformations on the quantum state with operators $K_0 = \text{diag}(1, \sqrt{1-\lambda_1})$ and $K_1 = \sqrt{\lambda_1} e_{0,1}$. Here $e_{i,j}$ refers to a matrix with all 0's except for a 1 in the i-th row and j-th column and is of appropriate dimension. In the general qudit case we have $K_0 = \text{diag}(1, \sqrt{1-\lambda_1}, \sqrt{1-\lambda_2}, ...\sqrt{1-\lambda_d})$, $K_1 = \sqrt{\lambda_1} e_{0,1}$, ... $K_d = \sqrt{\lambda_d} e_{0,d-1}$. Since we primarily focus on a superconducting system in this study we take $\lambda_m = 1 - \exp\{-m\Delta t/T_1\}$ where Δt is the idling duration and T_1 is the coherence time of the qubit [29].

In this work we are also concerned with the manipulation of mixed-radix systems. When drawing an error for such a system, for example a qubit-ququart interaction, we consider only relevant errors for the respective participant. For instance, a two-qudit error is drawn from $P_2 \otimes P_4$ and not from $P_4 \otimes P_4$ (where P_d is the set of d-dimensional Paulis, exactly the set of potential errors described above). Similarly, for two-qubit gates on encoded qubits, we consider only single *ququart* errors since gates on encoded systems are equivalent to single-ququart gates.

7 RESULTS

When we are able to perform native implementations of three-qubit gates via ququarts, we significantly reduce the number of gates that need to be executed, reducing failure rate. However, the reduction in time to execute these gates may not be enough to overcome the reduced coherence time of higher radix states. In Figure 7a-d, we examine the simulation fidelities for three-qubit compilation strategies across different sized circuits using Toffoli gate based decompositions. Each point represents the average fidelity of 1000+ different initial states run once, with randomly inserted error. The error bars are the standard error, which is the standard deviation of the all the trials divided by the square root of the number of trials. The mixed-radix compilation schemes stop at 12 qubits due to memory-based computational limitations. While mixed-radix circuits start in an all-qubit state, we must model them as if they are entirely on ququarts, since we must be able to model the higher levels at all times. This restricts the number of physical devices we are able to simulate in this scheme to 12 ququarts.

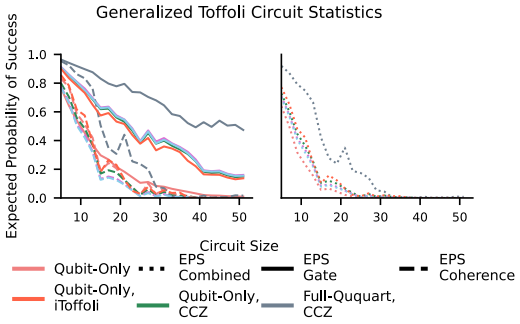

Figure 8: EPS statistics for the generalized Toffoli circuit. We show the gate and coherence EPS on the left and the product EPS on the right.

The first difference is that all of our mixed-radix and full-ququart compilation strategies exceed the fidelity of our baseline two-qubit gate, qubit-only compilation scheme. From a pure gate error perspective, this should not be unexpected, each of these schemes greatly reduces the number of gates required to execute the same operation. Figure 8 demonstrates how the EPS for gate error is substantially improved by using three two-ququart gates gates or one two-ququart gate for full-ququart computation. And, as hoped, the simulation finds that the idle time potentially spent in the $|2\rangle$ and $|3\rangle$ state does not outweigh the benefits of the using fewer gates and the shorter circuit duration. The shorter duration of the gates counteracts the increased decoherence rate of the ququarts. Figure 8 also demonstrates the same point. The coherence EPS between all of the mixed-radix strategies and the qubit-only baseline are nearly the same, and is improved for full-ququart strategies. The general trend of our simulation results is mirrored in Figure 8 where the

total EPS of the circuit is shown. As the EPS trends match what we find in our simulated results, we are able to infer that the scaling of simulation will match the scaling of the EPS results. While we only show examples of the generalized Toffoli circuit, the results are similar for other circuits. However, we note that the mixed-radix strategies only marginally outperform or match the simulated results of the qubit-only *i*Toffoli based strategy. This makes sense when we examine the *i*Toffoli based decomposition. We must insert an extra SWAP gate to perform the corrective Controlled-S gate, resulting in a similar number of gates, and duration, for both decompositions. Additionally, this SWAP may result in extra corrective SWAPs later on. The extra communication disrupts the layout of the circuit further than was intended, and can require extra gates.

Digging into the difference between the higher-radix strategies, we find that the mixed-radix strategies are all relatively similar to one another, with some additional separation as the size of the circuit increases. We first compare the mixed-radix Hadamard corrected CCX gates, shown in light blue, to the mixed-radix gates without this correction, shown in pink. While there is some cancellation between the single qubit gates, the extra serialization and marginal gate error of the correction gates is a drawback to using this correction strategy based on the simulated results. The reduction in time from the better configuration of the CCX gate is not always enough to overcome these additional costs. If we instead use CCZ decomposition, shown in green, we consistently achieve the same, or better, fidelity, especially as the size of the circuit increases and the reduction from CCZ gates is more pronounced. In these cases, the benefits found by using shorter target-independent gates from the start rather than retargeting improves the fidelity of the circuit in this mixed-radix regime. In Figure 7e we find that the mixed-radix gates achieve 2x better fidelities for circuit size 12, which is a significant improvement over two-qubit gate computation. This alone would be a important optimization for three-qubit gate based circuits.

We find that the ququart compilation scheme, shown in grey, has higher fidelity improvement, up to 3x reductions as seen in Figure 7e, and 50% improvement over the *i*Toffoli baseline and mixed-radix strategy. The reasoning behind this is two-fold. The first is that we no longer need to encode and decode gates before each three-qubit gate in this scheme reducing gate error. Gate reduction is important, and this reduces the number of gates. The second is reduction of communication. With the higher connectivity at all times, we reduce the qubit communication required to perform certain gates. Both of these factors add to the reduction in time, keeping the full-ququart based circuits under the coherence limits and maintaining higher circuit fidelity. We further reduce the overall circuit time by using faster, target-independent CZ gates in place of CX.

There are cases where full-ququart compilation does not outperform mixed-radix compilation to the same degree. For instance, the QRAM circuit. There are more than double the CX gates as Toffolis in this circuit. The serialization induced by ququarts with slower two-qubit gates reduces the effectiveness of ququarts. Additionally, these benchmarks are only kernels of computations that could be used within the context of larger circuits. In such cases, we will not have the benefit of a perfect mapping to start. This would not affect the improvement in fidelity from the qubit only to the mixed-radix strategies, but the effort to encode the qubits into a full-ququart regime before execution may outweigh the benefits.

7.1 Special Gate Case Study: CSWAP

As detailed in Section 4.2.3 we could instead decompose to a different-three qubit gate in the circuit. In the case of QRAM, this is the CSWAP gate. In Figure 9a, we explore the differences in fidelity when we use CSWAP gate alongside the original results using CCZ gates. A CSWAP can be constructed from two CX gates and one CCX gate, but cannot be re-targeted in the same way. Regardless, in the mixed-radix state, by orienting the CSWAP such that the targets are separate from the controls when possible and like qubits are with like, we see improvements over the CCZ decomposition. In fact it is able to beat the full-ququart CCZ compilation in some cases because of the reduced number of CX gates. While we can always attempt to encode the qubits favorably in a mixed-radix environment, this is not as natural a change when compiling for ququarts, and could lead to bad configurations if we solely focus on the disruption of qubits on ququarts. If we focus on the CSWAP in a full-ququart regime, the basic version shown in blue, and instead use the strategy that places the targets in the same ququart, shown in bright pink, we find even more improvement to our full-ququart encoding regime. This further indicates the importance using the best decomposition possible by separating the targets and the controls of certain gates.

7.2 Sensitivity to Ququart Gate Error Rate

While we synthesized our gates using a realistic Hamiltonian, it is still more difficult to physically realize gates that access higher energy levels. In Figure 9b we explore how the simulated fidelity changes as the error on ququart and mixed-radix gate increases for an 11-qubit Cuccaro Adder. Both strategies see a very fast drop off as the gate error increases, crossing over the qubit-only baseline fidelity when the ququart error rate is between two and four times worse than qubit gates for mixed-radix compilation (97% fidelity), and between four and six times worse for full-ququart compilation (94%). We also find that the *i*Toffoli strategy outperforms the full-ququart strategy at three times worse ququart gates than qubit gates as well. While these are still high fidelity targets, it does indicate that we do not need our three-qubit gates exceed the fidelity of two-qubit gates for these strategies to be successful.

7.3 Sensitivity to Ququart Coherence Error Rate

In this work we selected the expected theoretical decrease in coherence time as the grounding for most of our simulation experiments. However, physical realizations don't always meet reality. Accessing higher energy levels may prove to be more costly in terms of coherence time due to lack of control, or it may be less of an issue as has been found by some testbeds when accessing the qutrit state [9]. In Figure 9c we demonstrate the effects of changing the rate that the $|2\rangle$ and $|3\rangle$ levels decohere for an 12-qubit QRAM circuit. The main detail to note is that as the rate increases, the distance between mixed-radix and full-ququart fidelities decreases until mixed-radix becomes higher fidelity. Mixed-radix gates do not spend as much time in the higher level states, so as machines are developed and

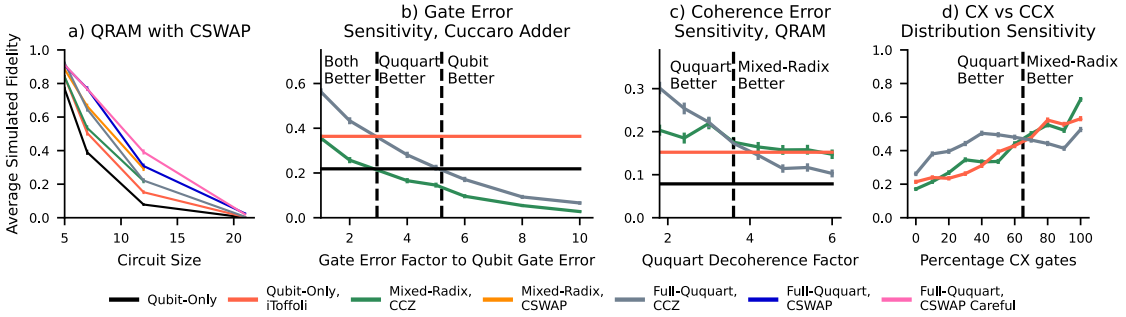

Figure 9: The results of several sensitivity studies. a) Sensitivity in simulation by using CSWAP gates in different orientations instead of decomposing to Toffoli gates. b) Changes in CCZ compilation strategies' fidelities as gate error ququarts increases. c) Changes in CCZ compilation strategies' fidelities as coherence error for the $|2\rangle$ and $|3\rangle$ level states changes. d) Differences in fidelities between mixed-radix and full-ququart compilation strategies as the distribution of CX gates to CCX gates in a circuit changes. In all graphs The black line represents the qubit-only fully-decomposed compilation method. The red line represents the qubit-only iToffoli-based decomposition. Below those points mixed-radix or full-ququart methods are more error prone than using only qubits. Please note the different scaling on the y-axis.

these level are more unstable, it may be better to avoid using full ququart encodings for larger circuits.

7.4 Ratio of Three Qubit Gates to Two Qubit Gates

It may not always be the case that the number of three qubit gates greatly exceeds the number of two qubit gates. Future applications may have a higher mix of two-qubit gates to three-qubit gates, or may only require a few three qubit gates to perform the desired operation. In Figure 9d, we example how the fidelity of different mixes of two-qubit to three-qubit gates is effected by compilation using a full-ququart strategy versus a mixed-radix strategy for an 11-qubit circuit. As the ratio of two-qubit to three-qubit gates increases, it becomes less and less profitable to use a full ququart encoding. At 60% CX gates it becomes more profitable to remain in the mixed-radix regime. Using CX gates on ququarts requires more serialization, since we cannot perform two separate operations on qubits encoded in the same ququart. This increases the time, and we start seeing the effects of reduced coherence times. This changes the calculus about when mixed-radix is better than full-ququart compilation. In cases where we don't need to use as many three qubit gates, it does not make as much sense to use ququarts for the entirety of the circuit. While this indicates that quantum circuits that only use two-qubit gates do not benefit from this encoding scheme, we can use resynthesis tools [59] to automatically insert three-qubit gates into the circuit, such as in [45]. However, resynthesis can introduce additional error as a perfect direct translation is not always possible and is better explored in a future work. We also include the *i*Toffoli strategy in this analysis as well. We find that it matches the mixed-radix strategy, further solidifying that these strategies have similar performance characteristics.

8 RELATED WORK

While this work is the first we are aware of to explore ququart-based execution, there have been studies using existing superconducting qubit technology to execute native three qubit gates. In particular, Kim et al. [31] and Gokhale et al. [21] have explored driving two connections between three qubits in a line to perform three qubit gates in a superconducting architecture. Gokhale developed a technique to execute two CX gates in parallel in a single CXX gate. These gates did not find improved fidelity, but achieved similar goals of faster parallel gate execution than serial execution as described in this work. It should be noted that this was done without explicit calibration for this sort of operation. While this work may seem similar through the application of multiqubit gates across many qubits, it mainly focuses on non-superconducting devices, which are able to make use of a global operator gate. It touches on performing three-qubit gates on superconducting devices, but is unable to generate gates that are more successful than the serialized decomposition. Our work focuses more on superconducting devices and direct synthesis, and must also contend with the issue of communication.

Kim et al. [31] developed a 98.2% fidelity iToffoli gate between three superconducting qubits. In this case, both controls induce a state change in a center qubit by driving the connections between the qubits. This gate is performed very quickly with a gate duration of 392 ns. While an impressive result, it is difficult to compare this work to our own as the device has a substantially different Hamiltonian than this work assumes. Additionally, this required significant manual calibration between each of the three qubits, a process which may not scale well to larger systems. This work is the basis for our *i*Toffoli baseline, which we found to be similar in performance to the mixed-radix strategy. However, the computational complexity of generating these pulses is much higher, requiring additional optimizable controls and a larger simulated Hilbert space (when taking into account the simulation of additional "guard" energy levels). Additionally, the mixed-radix scheme presented here only requires calibration between each pair of qudits, which is similar to the processes already in use on quantum computers. The *i*Toffoli scheme would require calibration between each

trio of qubits, which would add a significant additional overhead. This is also the case for [21].

There have also been several physical realizations of the *i*Toffoli gate that use the *i*SWAP gate, a CPHASE gate, and a reverse *i*SWAP gate using the $|2\rangle$ state to change the state of a qubit in [16, 26]. Galda et. al. [17] explores using qutrits on IBM's Jakarta device to implement a Toffoli gate with 78% fidelity. This is similar to our work, using the more accessible $|2\rangle$ state to perform a three-qubit gate. These are conceptually similar to the encode, mixed-radix Toffoli, and decode scheme that was laid out in this work. We believe that a machine specifically designed with qudits in mind could enable much higher fidelities for similar experiments.

9 CONCLUSION

The architecturally imposed requirement to decompose more complex three-qubit gates into component one- and two-qubit gates is an extreme hurdle for realizing quantum computing. Decomposing these gates increases both the number of error-prone gates that need to be executed, and the execution time of the circuit on devices with short coherence times. However, many architectures have access to higher level states beyond the traditional two-level system. While more prone to decoherence and error, this extra computational space can be used to compress quantum data, encoding two qubits into one physical device called a ququart.

This work takes advantage of increased connectivity and interaction potential when we have encoded qubits into a four-level system. Encoding qubits in this way allows for the interaction of three to four qubits across a single physical connection, and we synthesize a library or efficient three-qubit gates via optimal control that take advantage of this virtual connectivity and are much faster and higher fidelity than performing the decomposition of a three-qubit gate. We also demonstrate the viability of this encoding scheme and gate set via the execution of a $H \otimes H$ gate on real superconducting hardware. We then use these gates to develop compilation strategies, the quantum waltz, that use the most efficient configurations of three-qubit gates on mixed-radix and full-ququart systems to produce circuits that achieve 2x and 3x better simulated fidelities in mixed-radix and full-ququart environments, respectively compared to two-qubit based strategies. We also demonstrate that ququart-based gates are a viable alternative to *i*Toffoli based three-qubit pulse strategies with potential practical upsides. Despite the difficulty of accessing and performing operations on higher level states, this efficient implementation of three-qubit gates provides worthwhile benefits for quantum computation.

Mixed-radix and full-ququart implementations of three-qubit gates makes ququart computation an invaluable piece of the quantum computing repertoire. It is more flexible than previous hand optimized circuits to improve circuit execution via higher radix devices, does not require the use of quaternary-based logic, and can be selectively applied to certain sections of larger circuits. Realized implementations of these gates provide a massive opportunity to improve near-term execution of quantum circuits and expand the capabilities of quantum computers.

ACKNOWLEDGMENTS

This work is funded in part by EPiQC, an NSF Expedition in Computing, under award CCF-1730449; in part by STAQ under award NSF Phy-1818914; in part by NSF award 2110860; in part by the US Department of Energy Office of Advanced Scientific Computing Research, Accelerated Research for Quantum Computing Program; and in part by the NSF Quantum Leap Challenge Institute for Hybrid Quantum Architectures and Networks (NSF Award 2016136) and in part based upon work supported by the U.S. Department of Energy, Office of Science, National Quantum Information Science Research Centers. FTC is Chief Scientist for Quantum Software at ColdQuanta and an advisor to Quantum Circuits, Inc.

We would like to thank Casey Duckering for his input in early discussion of compiler development for ququarts. We would like to thank Stefanie Günther and N. Anders Petersson for valuable advice on using the quantum optimal control software packages Juqbox and Quandary.

REFERENCES

[1] MD SAJID ANIS, Abby-Mitchell, Héctor Abraham, AduOffei, Rochisha Agarwal, Gabriele Agliardi, Merav Aharoni, Vishnu Ajith, Ismail Yunus Akhalwaya, Gadi Aleksandrowicz, Thomas Alexander, Matthew Amy, Sashwat Anagolum, Anthony-Gandon, Eli Arbel, Abraham Asfaw, Anish Athalye, Artur Avkhadiev, Carlos Azaustre, PRATHAMESH BHOLE, Abhik Banerjee, Santanu Banerjee, Will Bang, Aman Bansal, Panagiotis Barkoutsos, Ashish Barnawal, George Barron, George S. Barron, Luciano Bello, Yael Ben-Haim, M. Chandler Bennett, Daniel Bevenius, Dhruv Bhatnagar, Prakhar Bhatnagar, Arjun Bhobe, Paolo Bianchini, Lev S. Bishop, Carsten Blank, Sorin Bolos, Soham Bopardikar, Samuel Bosch, Sebastian Brandhofer, Brandon, Sergey Bravyi, Nick Bronn, Bryce-Fuller, David Bucher, Artemiy Burov, Fran Cabrera, Padraic Calpin, Lauren Capelluto, Jorge Carballo, Ginés Carrascal, Adam Carriker, Ivan Carvalho, Adrian Chen, Chun-Fu Chen, Edward Chen, Jielun (Chris) Chen, Richard Chen, Franck Chevallier, Kartik Chinda, Rathish Cholarajan, Jerry M. Chow, Spencer Churchill, CisterMoke, Christian Claus, Christian Clauss, Caleb Clothier, Romilly Cocking, Ryan Cocuzzo, Jordan Connor, Filipe Correa, Zachary Crockett, Abigail J. Cross, Andrew W. Cross, Simon Cross, Juan Cruz-Benito, Chris Culver, Antonio D. Córcoles-Gonzales, Navaneeth D, Sean Dague, Tareq El Dandachi, Animesh N. Dangwal, Jonathan Daniel, Marcus Daniels, Matthieu Dartiailh, Abdón Rodríguez Davila, Faisal Debouni, Anton Dekusar, Amol Deshmukh, Mohit Deshpande, Delton Ding, Jun Doi, Eli M. Dow, Patrick Downing, Eric Drechsler, Eugene Dumitrescu, Karel Dumon, Ivan Duran, Kareem EL-Safty, Eric Eastman, Grant Eberle, Amir Ebrahimi, Pieter Eendebak, Daniel Egger, ElePT, Emilio, Alberto Espiricueta, Mark Everitt, Davide Facoetti, Farida, Paco Martín Fernández, Samuele Ferracin, Davide Ferrari, Axel Hernández Ferrera, Romain Fouilland, Albert Frisch, Andreas Fuhrer, Bryce Fuller, MELVIN GEORGE, Julien Gacon, Borja Godoy Gago, Claudio Gambella, Jay M. Gambetta, Adhisha Gammanpila, Luis Garcia, Tanya Garg, Shelly Garion, James R. Garrison, Jim Garrison, Tim Gates, Hristo Georgiev, Leron Gil, Austin Gilliam, Aditya Giridharan, Glen, Juan Gomez-Mosquera, Gonzalo, Salvador de la Puente González, Jesse Gorzinski, Ian Gould, Donny Greenberg, Dmitry Grinko, Wen Guan, Dani Guijo, John A. Gunnels, Harshit Gupta, Naman Gupta, Jakob M. Günther, Mikael Haglund, Isabel Haide, Ikko Hamamura, Omar Costa Hamido, Frank Harkins, Kevin Hartman, Areeq Hasan, Vojtech Havlicek, Joe Hellmers, \Lukasz Herok, Stefan Hillmich, Hiroshi Horii, Connor Howington, Shaohan Hu, Wei Hu, Chih-Han Huang, Junye Huang, Rolf Huisman, Haruki Imai, Takashi Imamichi, Kazuaki Ishizaki, Ishwor, Raban Iten, Toshinari Itoko, Alexander Ivrii, Ali Javadi, Ali Javadi-Abhari, Wahaj Javed, Qian Jianhua, Madhav Jivrajani, Kiran Johns, Scott Johnstun, Jonathan-Shoemaker, JosDenmark, JoshDumo, John Judge, Tal Kachmann, Akshay Kale, Naoki Kanazawa, Jessica Kane, Kang-Bae, Annanay Kapila, Anton Karazeev, Paul Kassebaum, Tobias Kehrer, Josh Kelso, Scott Kelso, Hugo van Kemenade, Vismai Khanderao, Spencer King, Yuri Kobayashi, Kovi11Day, Arseny Kovyrshin, Rajiv Krishnakumar, Pradeep Krishnamurthy, Vivek Krishnan, Kevin Krsulich, Prasad Kumkar, Gawel Kus, Ryan LaRose, Enrique Lacal, Raphaël Lambert, Haggai Landa, John Lapeyre, Joe Latone, Scott Lawrence, Christina Lee, Gushu Li, Tan Jun Liang, Jake Lishman, Dennis Liu, Peng Liu, Lolcroc, Abhishek K. M, Liam Madden, Yunho Maeng, Saurav Maheshkar, Kahan Majmudar, Aleksei Malyshev, Mohamed El Mandouh, Joshua Manela, Manjula, Jakub Marecek, Manoel Marques, Kunal Marwaha, Dmitri Maslov, Pawe\l Maszota, Dolph Mathews, Atsushi Matsuo, Farai Mazhandu, Doug McClure, Maureen McElaney, Cameron McGarry, David McKay, Dan McPherson, Srujan Meesala, Dekel Meirom, Corey Mendell, Thomas Metcalfe, Martin Mevissen,

Andrew Meyer, Antonio Mezzacapo, Rohit Midha, Daniel Miller, Hannah Miller, Zlatko Minev, Abby Mitchell, Nikolaj Moll, Alejandro Montanez, Gabriel Monteiro, Michael Duane Mooring, Renier Morales, Niall Moran, David Morcuende, Seif Mostafa, Mario Motta, Romain Moyard, Prakash Murali, Daiki Murata, Jan Müggenburg, Tristan NEMOZ, David Nadlinger, Ken Nakanishi, Giacomo Nannicini, Paul Nation, Edwin Navarro, Yehuda Naveh, Scott Wyman Neagle, Patrick Neuweiler, Aziz Ngoueya, Thien Nguyen, Johan Nicander, Nick-Singstock, Pradeep Niroula, Hassi Norlen, NuoWenLei, Lee James O'Riordan, Oluwatobi Ogunbayo, Pauline Ollitrault, Tamiya Onodera, Raul Otaolea, Steven Oud, Dan Padilha, Hanhee Paik, Soham Pal, Yuchen Pang, Ashish Panigrahi, Vincent R. Pascuzzi, Simone Perriello, Eric Peterson, Anna Phan, Kuba Pilch, Francesco Piro, Marco Pistoia, Christophe Piveteau, Julia Plewa, Pierre Pocreau, Alejandro Pozas-Kerstjens, Rafa\l Pracht, Milos Prokop, Viktor Prutyanov, Sumit Puri, Daniel Puzzuoli, Pythonix, Jesús Pérez, Quant02, Quintiii, Rafey Iqbal Rahman, Arun Raja, Roshan Rajeev, Isha Rajput, Nipun Ramagiri, Anirudh Rao, Rudy Raymond, Oliver Reardon-Smith, Rafael Martín-Cuevas Redondo, Max Reuter, Julia Rice, Matt Riedemann, Rietesh, Drew Risinger, Pedro Rivero, Marcello La Rocca, Diego M. Rodríguez, RohithKarur, Ben Rosand, Max Rossmannek, Mingi Ryu, Tharrmashastha SAPV, Nahum Rosa Cruz Sa, Arijit Saha, Abdullah Ash Saki, Sankalp Sanand, Martin Sandberg, Hirmay Sandesara, Ritvik Sapra, Hayk Sargsyan, Aniruddha Sarkar, Ninad Sathaye, Niko Savola, Bruno Schmitt, Chris Schnabel, Zachary Schoenfeld, Travis L. Scholten, Eddie Schoute, Mark Schulterbrandt, Joachim Schwarm, James Seaward, Sergi, Ismael Faro Sertage, Kanav Setia, Freya Shah, Nathan Shammah, Will Shanks, Rohan Sharma, Yunong Shi, Jonathan Shoemaker, Adenilton Silva, Andrea Simonetto, Deeksha Singh, Divyanshu Singh, Parmeet Singh, Phattharaporn Singkanipa, Yukio Siraichi, Siri, Jesús Sistos, Iskandar Sitdikov, Seyon Sivarajah, Slavikmew, Magnus Berg Sletfjerding, John A. Smolin, Mathias Soeken, Igor Olegovich Sokolov, Igor Sokolov, Vicente P. Soloviev, SooluThomas, Starfish, Dominik Steenken, Matt Stypulkoski, Adrien Suau, Shaojun Sun, Kevin J. Sung, Makoto Suwama, Oskar S\lowik, Hitomi Takahashi, Tanvesh Takawale, Ivano Tavernelli, Charles Taylor, Pete Taylour, Soolu Thomas, Kevin Tian, Mathieu Tillet, Maddy Tod, Miroslav Tomasik, Caroline Tornow, Enrique de la Torre, Juan Luis Sánchez Toural, Kenso Trabing, Matthew Treinish, Dimitar Trenev, TrishaPe, Felix Truger, Georgios Tsilimigkounakis, Davindra Tulsi, Doğukan Tuna, Wes Turner, Yotam Vaknin, Carmen Recio Valcarce, Francois Varchon, Adish Vartak, Almudena Carrera Vazquez, Prajjwal Vijaywargiya, Victor Villar, Bhargav Vishnu, Desiree Vogt-Lee, Christophe Vuillot, James Weaver, Johannes Weidenfeller, Rafal Wieczorek, Jonathan A. Wildstrom, Jessica Wilson, Erick Winston, WinterSoldier, Jack J. Woehr, Stefan Woerner, Ryan Woo, Christopher J. Wood, Ryan Wood, Steve Wood, James Wootton, Matt Wright, Lucy Xing, Jintao YU, Bo Yang, Unchun Yang, Jimmy Yao, Daniyar Yeralin, Ryota Yonekura, David Yonge-Mallo, Ryuhei Yoshida, Richard Young, Jessie Yu, Lebin Yu, Yuma-Nakamura, Christopher Zachow, Laura Zdanski, Helena Zhang, Iulia Zidaru, Bastian Zimmermann, Christa Zoufal, aeddins-ibm, alexzhang13, b63, bartekbartlomiej, bcamorrison, brandhsn, chetmurthy, deeplokhande, dekel.meirom, dime10, dlasecki, ehchen, ewinston, fanizzamarco, fs1132429, gadial, galeinston, georgezhou20, georgios-ts, gruu, hhorii, hhyap, hykavitha, itoko, jeppevinkel, jessica-angel7, jezerjojo14, jliu45, johannesgreiner, jscott2, klinvill, krutik2966, ma5x, michelle4654, msuwama, nico-lgrs, nrhawkins, ntgiwsvp, ordmoj, sagar pahwa, pritamsinha2304, rithikaadiga, ryancocuzzo, saktar-unr, saswati-qiskit, septembrr, sethmerkel, sg495, shaashwat, smturro2, sternparky, strickroman, tigerjack, tsura-crisaldo, upsideon, vadebayo49, welien, willhbang, wmurphy-collabstar, yang.luh, and Mantas Čepulkovskis. 2021. Qiskit: An Open-source Framework for Quantum Computing. https://doi.org/10.5281/zenodo.2573505

[2] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C. Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando G. S. L. Brandao, David A. Buell, Brian Burkett, Yu Chen, Zijun Chen, Ben Chiaro, Roberto Collins, William Courtney, Andrew Dunsworth, Edward Farhi, Brooks Foxen, Austin Fowler, Craig Gidney, Marissa Giustina, Rob Graff, Keith Guerin, Steve Habegger, Matthew P. Harrigan, Michael J. Hartmann, Alan Ho, Markus Hoffmann, Trent Huang, Travis S. Humble, Sergei V. Isakov, Evan Jeffrey, Zhang Jiang, Dvir Kafri, Kostyantyn Kechedzhi, Julian Kelly, Paul V. Klimov, Sergey Knysh, Alexander Korotkov, Fedor Kostritsa, David Landhuis, Mike Lindmark, Erik Lucero, Dmitry Lyakh, Salvatore Mandrà, Jarrod R. McClean, Matthew McEwen, Anthony Megrant, Xiao Mi, Kristel Michielsen, Masoud Mohseni, Josh Mutus, Ofer Naaman, Matthew Neeley, Charles Neill, Murphy Yuezhen Niu, Eric Ostby, Andre Petukhov, John C. Platt, Chris Quintana, Eleanor G. Rieffel, Pedram Roushan, Nicholas C. Rubin, Daniel Sank, Kevin J. Satzinger, Vadim Smelyanskiy, Kevin J. Sung, Matthew D. Trevithick, Amit Vainsencher, Benjamin Villalonga, Theodore White, Z. Jamie Yao, Ping Yeh, Adam Zalcman, Hartmut Neven, and John M. Martinis. 2019. Quantum supremacy using a programmable superconducting processor. *Nature* 574, 7779 (Oct. 2019), 505–510. https://doi.org/10.1038/s41586-019-1666-5

[3] Ryan Babbush, Craig Gidney, Dominic W. Berry, Nathan Wiebe, Jarrod McClean, Alexandru Paler, Austin Fowler, and Hartmut Neven. 2018. Encoding Electronic Spectra in Quantum Circuits with Linear T Complexity. *Physical Review X* 8, 4 (Oct. 2018). https://doi.org/10.1103/physrevx.8.041015 Publisher: American Physical Society (APS).

[4] Jonathan M Baker, Casey Duckering, and Frederic T Chong. 2020. Efficient quantum circuit decompositions via intermediate qudits. In *2020 IEEE 50th International Symposium on Multiple-Valued Logic (ISMVL)*. IEEE, 303–308.

[5] Jonathan M. Baker, Casey Duckering, Alexander Hoover, and Frederic T. Chong. 2019. Decomposing Quantum Generalized Toffoli with an Arbitrary Number of Ancilla. https://doi.org/10.48550/ARXIV.1904.01671

[6] Jonathan M. Baker, Casey Duckering, Alexander Hoover, and Frederic T. Chong. 2020. Time-sliced quantum circuit partitioning for modular architectures. In *Proceedings of the 17th ACM International Conference on Computing Frontiers*. ACM. https://doi.org/10.1145/3387902.3392617

[7] Adriano Barenco, Charles H. Bennett, Richard Cleve, David P. DiVincenzo, Norman Margolus, Peter Shor, Tycho Sleator, John A. Smolin, and Harald Weinfurter. 1995. Elementary gates for quantum computation. *Physical Review A* 52, 5 (Nov. 1995), 3457–3467. https://doi.org/10.1103/PhysRevA.52.3457

[8] Todd A. Brun. 2002. A simple model of quantum trajectories. *American Journal of Physics* 70, 7 (2002), 719–737. https://doi.org/10.1119/1.1475328 _eprint: https://doi.org/10.1119/1.1475328.

[9] Alba Cervera-Lierta, Mario Krenn, Alá n Aspuru-Guzik, and Alexey Galda. 2022. Experimental High-Dimensional Greenberger-Horne-Zeilinger Entanglement with Superconducting Transmon Qutrits. *Physical Review Applied* 17, 2 (Feb. 2022). https://doi.org/10.1103/physrevapplied.17.024062 Publisher: American Physical Society (APS).

[10] Peter Chapman. 2020. Scaling IonQ's Quantum Computers: The Roadmap. https://ionq.com/posts/december-09-2020-scaling-quantum-computer-roadmap

[11] Yulin Chi, Jieshan Huang, Zhanchuan Zhang, Jun Mao, Zinan Zhou, Xiaojiong Chen, Chonghao Zhai, Jueming Bao, Tianxiang Dai, Huihong Yuan, Ming Zhang, Daoxin Dai, Bo Tang, Yan Yang, Zhihua Li, Yunhong Ding, Leif K. Oxenløwe, Mark G. Thompson, Jeremy L. O'Brien, Yan Li, Qihuang Gong, and Jianwei Wang. 2022. A programmable qudit-based quantum processor. *Nature Communications* 13, 1 (Dec. 2022), 1166. https://doi.org/10.1038/s41467-022-28767-x

[12] Alexander Cowtan, Silas Dilkes, Ross Duncan, Alexandre Krajenbrink, Will Simmons, and Seyon Sivarajah. 2019. On the qubit routing problem. *arXiv preprint arXiv:1902.08091* (2019).

[13] Steven A. Cuccaro, Thomas G. Draper, Samuel A. Kutin, and David Petrie Moulton. 2004. A new quantum ripple-carry addition circuit. https://doi.org/10.48550/ARXIV.QUANT-PH/0410184

[14] Yongshan Ding, Pranav Gokhale, Sophia Fuhui Lin, Richard Rines, Thomas Propson, and Frederic T. Chong. 2020. Systematic Crosstalk Mitigation for Superconducting Qubits via Frequency-Aware Compilation. In *2020 53rd Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*. IEEE. https://doi.org/10.1109/micro50266.2020.00028

[15] Casey Duckering, Jonathan M. Baker, Andrew Litteken, and Frederic T. Chong. 2021. Orchestrated trios: compiling for efficient communication in Quantum programs with 3-Qubit gates. In *Proceedings of the 26th ACM International Conference on Architectural Support for Programming Languages and Operating Systems*. ACM. https://doi.org/10.1145/3445814.3446718

[16] A. Fedorov, L. Steffen, M. Baur, M. P. da Silva, and A. Wallraff. 2011. Implementation of a Toffoli gate with superconducting circuits. *Nature* 481, 7380 (Dec. 2011), 170–172. https://doi.org/10.1038/nature10713 Publisher: Springer Science and Business Media LLC.

[17] Alexey Galda, Michael Cubeddu, Naoki Kanazawa, Prineha Narang, and Nathan Earnest-Noble. 2021. Implementing a Ternary Decomposition of the Toffoli Gate on Fixed-Frequency Transmon Qutrits. https://doi.org/10.48550/arXiv.2109.00558 arXiv:2109.00558 [quant-ph].

[18] Jay Gambetta. 2022. Expanding the IBM Quantum roadmap to anticipate the future of quantum-centric supercomputing. https://research.ibm.com/blog/ibm-quantum-roadmap-2025?social_post=6953094465&linkId=164428745

[19] Pranav Gokhale, Jonathan M. Baker, Casey Duckering, Natalie C. Brown, Kenneth R. Brown, and Frederic T. Chong. 2019. Asymptotic improvements to quantum circuits via qutrits. In *Proceedings of the 46th International Symposium on Computer Architecture*. ACM. https://doi.org/10.1145/3307650.3322253

[20] Pranav Gokhale, Ali Javadi-Abhari, Nathan Earnest, Yunong Shi, and Frederic T. Chong. 2020. Optimized Quantum Compilation for Near-Term Algorithms with OpenPulse. http://arxiv.org/abs/2004.11205 arXiv:2004.11205 [quant-ph].

[21] Pranav Gokhale, Samantha Koretsky, Shilin Huang, Swarnadeep Majumder, Andrew Drucker, Kenneth R. Brown, and Frederic T. Chong. 2020. Quantum Fan-out: Circuit Optimizations and Technology Modeling. https://doi.org/10.48550/ARXIV.2007.04246

[22] Noah Goss, Alexis Morvan, Brian Marinelli, Bradley K. Mitchell, Long B. Nguyen, Ravi K. Naik, Larry Chen, Christian Jünger, John Mark Kreikebaum, David I. Santiago, Joel J. Wallman, and Irfan Siddiqi. 2022. High-Fidelity Qutrit Entangling Gates for Superconducting Circuits. http://arxiv.org/abs/2206.07216 arXiv:2206.07216 [cond-mat, physics:quant-ph].

[23] Lov K. Grover. 1996. *A fast quantum mechanical algorithm for database search*. Technical Report arXiv:quant-ph/9605043. arXiv. https://doi.org/10.48550/arXiv.quant-ph/9605043 arXiv:quant-ph/9605043 type: article.

[24] Stefanie Günther, N. Anders Petersson, and Jonathan L. DuBois. 2021. Quantum Optimal Control for Pure-State Preparation Using One Initial State. *arXiv:2106.09148 [quant-ph]* (Aug. 2021). http://arxiv.org/abs/2106.09148 arXiv:

2106.09148.

[25] Stuart Hadfield, Zhihui Wang, Bryan O'Gorman, Eleanor G. Rieffel, Davide Venturelli, and Rupak Biswas. 2019. From the Quantum Approximate Optimization Algorithm to a Quantum Alternating Operator Ansatz. *Algorithms* 12, 2 (Feb. 2019), 34. https://doi.org/10.3390/a12020034 arXiv:1709.03489 [quant-ph].

[26] Alexander D. Hill, Mark J. Hodson, Nicolas Didier, and Matthew J. Reagor. 2021. Realization of arbitrary doubly-controlled quantum phase gates. https://doi.org/10.48550/ARXIV.2108.01652

[27] IBM. [n. d.]. IBM Quantum. https://quantum-computing.ibm.com/ Publication Title: IBM Quantum.

[28] S. S. Ivanov, H. S. Tonchev, and N. V. Vitanov. 2012. Time-efficient implementation of quantum search with qudits. *Physical Review A* 85, 6 (June 2012), 062321. https://doi.org/10.1103/PhysRevA.85.062321

[29] N. Khammassi, I. Ashraf, X. Fu, C.G. Almudever, and K. Bertels. 2017. QX: A high-performance quantum computer simulation platform. In *Design, Automation & Test in Europe Conference & Exhibition (DATE), 2017*. 464–469. https://doi.org/10.23919/DATE.2017.7927034

[30] Navin Khaneja, Timo Reiss, Cindie Kehlet, Thomas Schulte-Herbrüggen, and Steffen J. Glaser. 2005. Optimal control of coupled spin dynamics: design of NMR pulse sequences by gradient ascent algorithms. *Journal of Magnetic Resonance* 172, 2 (Feb. 2005), 296–305. https://doi.org/10.1016/j.jmr.2004.11.004

[31] Yosep Kim, Alexis Morvan, Long B. Nguyen, Ravi K. Naik, Christian Jünger, Larry Chen, John Mark Kreikebaum, David I. Santiago, and Irfan Siddiqi. 2022. High-fidelity three-qubit iToffoli gate for fixed-frequency superconducting qubits. *Nature Physics* 18, 7 (May 2022), 783–788. https://doi.org/10.1038/s41567-022-01590-3 Publisher: Springer Science and Business Media LLC.

[32] Jens Koch, Terri M. Yu, Jay Gambetta, A. A. Houck, D. I. Schuster, J. Majer, Alexandre Blais, M. H. Devoret, S. M. Girvin, and R. J. Schoelkopf. 2007. Charge-insensitive qubit design derived from the Cooper pair box. *Phys. Rev. A* 76, 4 (Oct. 2007), 042319. https://doi.org/10.1103/PhysRevA.76.042319 Publisher: American Physical Society.

[33] Ziqian Li, Tanay Roy, David Rodriguez Perez, Kan-Heng Lee, Eliot Kapit, and David I. Schuster. 2023. Autonomous error correction of a single logical qubit using two transmons. https://doi.org/10.48550/arXiv.2302.06707

[34] Ziqian Li, Tanay Roy, David Rodríguez Pérez, David I. Schuster, and Eliot Kapit. 2023. Hardware efficient autonomous error correction with linear couplers in superconducting circuits. https://doi.org/10.48550/arXiv.2303.01110

[35] Andrew Litteken, Jonathan M. Baker, and Frederic T. Chong. 2022. Communication Trade Offs in Intermediate Qudit Circuits. In *2022 IEEE 52nd International Symposium on Multiple-Valued Logic (ISMVL)*. IEEE, Dallas, TX, USA, 43–49. https://doi.org/10.1109/ISMVL52857.2022.00014

[36] Guang Hao Low and Isaac L. Chuang. 2019. Hamiltonian Simulation by Qubitization. *Quantum* 3 (July 2019), 163. https://doi.org/10.22331/q-2019-07-12-163 Publisher: Verein zur Forderung des Open Access Publizierens in den Quantenwissenschaften.

[37] MingXing Luo and XiaoJun Wang. 2014. Universal quantum computation with qudits. *Science China Physics, Mechanics & Astronomy* 57, 9 (Sept. 2014), 1712–1717. https://doi.org/10.1007/s11433-014-5551-9

[38] Easwar Magesan, J. M. Gambetta, and Joseph Emerson. 2011. Scalable and Robust Randomized Benchmarking of Quantum Processes. *Physical Review Letters* 106, 18 (May 2011), 180504. https://doi.org/10.1103/PhysRevLett.106.180504 Publisher: American Physical Society.

[39] Easwar Magesan, Jay M. Gambetta, B. R. Johnson, Colm A. Ryan, Jerry M. Chow, Seth T. Merkel, Marcus P. da Silva, George A. Keefe, Mary B. Rothwell, Thomas A. Ohki, Mark B. Ketchen, and M. Steffen. 2012. Efficient Measurement of Quantum Gate Error by Interleaved Randomized Benchmarking. *Physical Review Letters* 109, 8 (Aug. 2012), 080505. https://doi.org/10.1103/PhysRevLett.109.080505 Publisher: American Physical Society.

[40] Daniel Miller, Timo Holz, Hermann Kampermann, and Dagmar Bruß. 2018. Propagation of generalized Pauli errors in qudit Clifford circuits. *Physical Review A* 98, 5 (Nov. 2018). https://doi.org/10.1103/physreva.98.052316 Publisher: American Physical Society (APS).

[41] A. Morvan, V. V. Ramasesh, M. S. Blok, J. M. Kreikebaum, K. O'Brien, L. Chen, B. K. Mitchell, R. K. Naik, D. I. Santiago, and I. Siddiqi. 2021. Qutrit Randomized Benchmarking. *Physical Review Letters* 126, 21 (May 2021), 210504. https://doi.org/10.1103/PhysRevLett.126.210504

[42] Pranav Mundada, Gengyan Zhang, Thomas Hazard, and Andrew Houck. 2019. Suppression of Qubit Crosstalk in a Tunable Coupling Superconducting Circuit. *Physical Review Applied* 12, 5 (Nov. 2019). https://doi.org/10.1103/physrevapplied.12.054023 Publisher: American Physical Society (APS).

[43] Prakash Murali, Jonathan M. Baker, Ali Javadi Abhari, Frederic T. Chong, and Margaret Martonosi. 2019. Noise-Adaptive Compiler Mappings for Noisy Intermediate-Scale Quantum Computers. https://doi.org/10.48550/ARXIV.1901.11054

[44] Michael A. Nielsen and Isaac L. Chuang. 2011. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press. https://www.amazon.com/Quantum-Computation-Information-10th-Anniversary/dp/1107002176?SubscriptionId=AKIAIOBINVZYXZQZ2U3A&tag=chimbori05-20&linkCode=xm2&camp=2025&creative=165953&creativeASIN=1107002176

[45] Tirthak Patel, Daniel Silver, and Devesh Tiwari. 2022. Geyser: A Compilation Framework for Quantum Computing with Neutral Atoms. In *Proceedings of the 49th Annual International Symposium on Computer Architecture (ISCA '22)*. Association for Computing Machinery, New York, NY, USA, 383–395. https://doi.org/10.1145/3470496.3527428 event-place: New York, New York.

[46] Archimedes Pavlidis and Emmanuel Floratos. 2021. Arithmetic Circuits for Multilevel Qudits Based on Quantum Fourier Transform. *Physical Review A* 103, 3 (March 2021), 032417. https://doi.org/10.1103/PhysRevA.103.032417 arXiv:1707.08834 [quant-ph].

[47] N. Anders Petersson and Fortino Garcia. 2021. Optimal Control of Closed Quantum Systems via B-Splines with Carrier Waves. *arXiv:2106.14310 [quant-ph]* (June 2021). http://arxiv.org/abs/2106.14310 arXiv: 2106.14310.

[48] N. Anders Petersson, Fortino M. Garcia, Austin E. Copeland, Ylva L. Rydin, and Jonathan L. DuBois. 2020. Discrete Adjoints for Accurate Numerical Optimization with Application to Quantum Control. *arXiv:2001.01013 [quant-ph]* (Nov. 2020). http://arxiv.org/abs/2001.01013 arXiv: 2001.01013.

[49] John Preskill. 2018. Quantum Computing in the NISQ era and beyond. *Quantum* 2 (Aug. 2018), 79. https://doi.org/10.22331/q-2018-08-06-79 Publisher: Verein zur Forderung des Open Access Publizierens in den Quantenwissenschaften.

[50] Tanay Roy, Ziqian Li, Eliot Kapit, and David I. Schuster. 2022. Realization of two-qutrit quantum algorithms on a programmable superconducting processor. http://arxiv.org/abs/2211.06523 arXiv:2211.06523 [quant-ph].

[51] Lennart Maximilian Seifert, Jason Chadwick, Andrew Litteken, Frederic T. Chong, and Jonathan M. Baker. 2022. *Time-Efficient Qudit Gates through Incremental Pulse Re-seeding*. Technical Report arXiv:2206.14975. arXiv. http://arxiv.org/abs/2206.14975 arXiv:2206.14975 [quant-ph] type: article.

[52] Sarah Sheldon, Easwar Magesan, Jerry M. Chow, and Jay M. Gambetta. 2016. Procedure for systematically tuning up cross-talk in the cross-resonance gate. *Physical Review A* 93, 6 (June 2016), 060302. https://doi.org/10.1103/PhysRevA.93.060302

[53] Vivek V. Shende and Igor L. Markov. 2008. On the CNOT-cost of TOFFOLI gates. (2008). https://doi.org/10.48550/ARXIV.0803.2316 Publisher: arXiv.

[54] Shlomo E. Sklarz and David J. Tannor. 2002. Loading a Bose-Einstein condensate onto an optical lattice: An application of optimal control theory to the nonlinear Schr\"odinger equation. *Physical Review A* 66, 5 (Nov. 2002), 053619. https://doi.org/10.1103/PhysRevA.66.053619 Publisher: American Physical Society.

[55] Asma Taheri Monfared, Majid Haghparast, and Kamalika Datta. 2019. Quaternary Quantum/Reversible Half-Adder, Full-Adder, Parallel Adder and Parallel Adder/Subtractor Circuits. *International Journal of Theoretical Physics* 58, 7 (July 2019), 2184–2199. https://doi.org/10.1007/s10773-019-04108-5

[56] Yuchen Wang, Zixuan Hu, Barry C. Sanders, and Sabre Kais. 2020. Qudits and High-Dimensional Quantum Computing. *Frontiers in Physics* 8 (2020). https://doi.org/10.3389/fphy.2020.589504

[57] Xian Wu, S. L. Tomarken, N. Anders Petersson, L. A. Martinez, Yaniv J. Rosen, and Jonathan L. DuBois. 2020. High-fidelity software-defined quantum logic on a superconducting qudit. *Physical Review Letters* 125, 17 (Oct. 2020), 170502. https://doi.org/10.1103/PhysRevLett.125.170502 arXiv:2005.13165 [quant-ph].

[58] Theodore J. Yoder. 2017. Universal fault-tolerant quantum computation with Bacon-Shor codes. http://arxiv.org/abs/1705.01686 arXiv:1705.01686 [quant-ph].

[59] Ed Younis, Costin C Iancu, Wim Lavrijsen, Marc Davis, Ethan Smith, and USDOE. 2021. Berkeley Quantum Synthesis Toolkit (BQSKit) v1. https://doi.org/10.11578/dc.20210603.2