Pulse-engineered Controlled-V gate and its applications on superconducting quantum device

TAKAHIKO SATOH\textsuperscript{1,2}, SHUN OOMURA\textsuperscript{1,2}, MICHIHIKO SUGAWARA\textsuperscript{1,2}, AND NAOKI YAMAMOTO\textsuperscript{1,2}

\textsuperscript{1}Quantum Computing Center, Keio University, Hiyoshi 3-14-1, Kohoku, Yokohama 223-8522, Japan
\textsuperscript{2}Graduate School of Science and Technology, Keio University, Hiyoshi 3-14-1, Kohoku, Yokohama 223-8522, Japan

Corresponding author: Takahiko Satoh (email: satoh@sfc.wide.ad.jp).

This work was supported by MEXT Quantum Leap Flagship Program Grant Number JPMXS0118067285 and JPMXS0120319794.

ABSTRACT In this paper, we demonstrate that, by employing OpenPulse design kit for IBM superconducting quantum devices, the controlled-V gate (CV gate) can be implemented in about half the gate time to the controlled-X (CX or CNOT gate) and consequently 65.5\% reduced gate time compared to the CX-based implementation of CV. Then, based on the theory of Cartan decomposition, we characterize the set of all two-qubit gates implemented with only two or three CV gates; using pulse-engineered CV gates enables us to implement these gates with shorter gate time and possibly better gate fidelity than the CX-based one, as actually demonstrated in two examples. Moreover, we showcase the improvement of linearly-coupled three-qubit Toffoli gate, by implementing it with the pulse-engineered CV gate, both in gate time and the averaged output-state fidelity. These results imply the importance of our CV gate implementation technique, which, as an additional option for the basis gate set design, may shorten the overall computation time and consequently improve the precision of several quantum algorithms executed on a real device.

INDEX TERMS Controlled-V gate, IBM Quantum device, OpenPulse

I. INTRODUCTION

There are several type of platforms for implementing quantum computer, such as superconducting, ion, and optical devices. In this paper, we study the problem of reducing the circuit depth (total gate time) in the superconducting quantum device provided by IBM (called IBM Quantum), where Qiskit serves as the software development environment. Qiskit has two representation languages for designing quantum programs: OpenPulse\textsuperscript{1} and QASM.

OpenPulse is a language for specifying and physically controlling the pulse level of a target quantum gate, that enables introducing a large freedom in circuit design. As a result, OpenPulse can reduce the execution time through optimal pulse design for various type of quantum gates\textsuperscript[2,3]; also it can be applied to generate a new gate specific to a particular physical simulation\textsuperscript[4]. Recently, a computational framework has been proposed to aid such synthesis problems\textsuperscript[5].

QASM is the language for the circuit design with several quantum gates. Physically, each gate is decomposed into a set of precisely calibrated gates chosen from the universal quantum gate set\textsuperscript[6,7]. The universal gate set used in IBM Quantum is composed of single-qubit gates and the Controlled-X (CX, or often called CNOT) gate\textsuperscript[8]. The point of taking this fixed gate set is that, because it contains only 1 two-qubit interaction gate (i.e., CX gate), the calibration process is relatively easy. In particular, CX gate can be implemented precisely via the cross resonance (CR) Hamiltonian\textsuperscript[9–13], with the help of the echo scheme and the cancellation pulse technique\textsuperscript[14]. However, the error rate of CX gate is still much higher than that of single-qubit gates\textsuperscript[15], due to the longer pulse length (gate time) than that of single-qubit gates and the effects of cross-talk\textsuperscript[16–18]. Hence, if a quantum algorithm must be realized on a circuit with unnecessarily many CX gates due to the QASM constraint, the accuracy of circuit will significantly decrease.

The above-mentioned issue may be resolved by adding some two-qubits gates to the default universal gate set composed of single-qubit gates and CX gate. In this work, we take the Controlled-V (CV) gate whose matrix representation
in the computational basis is given by

\[ CV = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1+i}{2} & \frac{1-1}{2} \\ 0 & 0 & \frac{1+i}{2} & \frac{1+i}{2} \end{bmatrix}, \] (1)

which readily leads to the relation \( CV^2 = CX \). Note that, in the CX-based default implementation of CV gate on QASM, one needs 2 CX gates to create the two-qubits interaction process as shown in Fig. 1.

![FIGURE 1. Circuit diagram of CV gate in QASM-based implementation.](image)

The main reason for choosing CV gate is its potential ability to reduce the gate time in several QASM-based quantum algorithms. The point is that, by using OpenPulse, we can effectively implement CV gate by just halving the pulse length of the CR pulse used for generating CX gate, as suggested by the relation \( CV^2 = CX \). That is, the gate time of pulse-engineered CV gate is half that of CX gate, while the QASM-based CV gate shown in Fig. 1 needs the gate time at least twice that of CX. Therefore, if some CX gates on a quantum circuit can be replaced with the same or less number of CV gates, the total gate time of this circuit is reduced and thereby the accuracy of the circuit will be improved. A typical example is the Toffoli gate; it needs at least 6 CX gates to implement if CX is only given to us, but it can be implemented using 2 CX and 3 CV gates if CV gate is further available [6], [19], [20].

This paper is organized as follows. In Section II, we describe how to implement CV gate using OpenPulse and then show the experimental result; the gate time of the pulse-engineered CV gate is shortened by 65.5% and the gate fidelity is improved by 0.66%, compared to the default QASM-based implementation of CV gate. In Section III, we first use the theory of Cartan decomposition to characterize the set of all two-qubit gates implemented with only two or three CV gates; because the pulse-engineered CV gate can be implemented with shorter gate time, those two-qubit gates can also be implemented with shorter gate time and possibly better gate fidelity than the default CX-based one. Actually, we show the experimental demonstration to generate \( \sqrt{iSWAP} \) and \( \sqrt{SWAP} \) using the pulse-engineered CV gates and confirm that, in both cases, the gate fidelity is improved thanks to the shorter gate time. In Section IV, we showcase an efficient method for implementing a linearly-coupled three-qubit Toffoli gate using the pulse-engineered CV gate.

II. PULSE-ENGINEERED CV GATE

A. CROSS RESONANCE INTERACTION

On IBM Quantum devices, the cross resonance (CR) interaction is used to couple two qubits [10], by irradiating the control qubit with a microwave pulse at the transition frequency of the target qubit. The microwave pulse has a Gaussian-square type envelope in the default setup; see Appendix A. Under some approximation, we obtain the following model CR Hamiltonian [11], [12], [14], [20]:

\[ H_{CR} = \sum_{P=I,XYZ} \frac{\omega_{ZP}(A,\phi)}{2} Z \otimes P + \sum_{Q=XYZ} \frac{\omega_{IQ}(A,\phi)}{2} I \otimes Q, \] (2)

where the qubit ordering is control-target. \( \omega_{ZP} \) and \( \omega_{IQ} \) represent the interaction strength, which are functions of the amplitude \( A \) and the phase \( \phi \) of the microwave pulse. Note that the CR Hamiltonian is valid under the condition that the microwave pulse with transition frequency of the target qubit is irradiated to the control qubit. In the absence of noise, the two qubits are driven by the unitary operator

\[ U_{CR} = \exp(-itH_{CR}). \] (3)

B. PULSE-ENGINEERED CX AND CV GATES

Let us define the general two-qubit unitary operator

\[ [DE]^\theta = \exp \left( -i\frac{\theta}{2} D \otimes E \right), \] (4)

where \( D \) and \( E \) are arbitrary single-qubit operators. With this notation, the CX gate is represented as [10]:

\[ CX = [ZI]^{1/2}[ZX]^{-1/2}[IX]^{1/2}. \] (5)

That is, the two-qubit operation required to form the CX gate can only be served by the \( Z \otimes X \) Hamiltonian. However, the CR Hamiltonian [2] contains terms other than \( Z \otimes X \) term, which thus should be eliminated by some means for implementing the CX gate via the CR Hamiltonian. This goal can be achieved, by using the echo sequence pulse scheme and applying a direct cancellation pulse on the target qubit as illustrated in Fig. 2; in other words, these techniques are effectively used to generate the unitary evolution driven by the effective Hamiltonian, \( \tilde{H}_{ZZ} \), composed of only the \( Z \otimes X \) term [12], [21]. In general, one can implement the unitary operator \( [ZX]^\theta \) driven by the effective Hamiltonian \( \tilde{H}_{ZZ} \), by setting the interaction strength in terms of the pulse duration \( t \) as \( \theta = \omega_{ZX}(A,\phi)t/\pi \):

\[ [ZX]^\theta = \tilde{U}_{ZX} = \exp(-i\pi t\tilde{H}_{ZX}), \]

\[ \tilde{H}_{ZX} = \frac{\omega_{ZX}(A,\phi)}{2} Z \otimes X. \] (6)

For the CX gate case, the two-qubit interaction time \( t_{CX} \) should be \( t_{CX} = \pi/2 \omega_{ZX}(A,\phi) \) to realize \( \theta = -1/2 \).

Next, from Eq. (5) and the fact that \( IX, ZX, \) and \( ZI \) commute with each other, one can see that CV gate is decomposed as

\[ CV = [ZI]^{1/4}[ZX]^{-1/4}[IX]^{1/4}. \] (7)
In the present work, we directly implement $[ZX]^{-1/4}$ part using OpenPulse, without decomposing this gate into multiple CX gates. As expected from Eq. (6), the interaction strength $\theta$ of the two-qubits interaction part $[ZX]^0$ is proportional to the duration of CR pulse, as far as the effective strength $\tau$ of multiple CX gates. As expected from Eq. (6), the interaction part $[ZX]^{-1/4}$ by taking the duration of the CR pulse $t_{CV}$ as

$$t_{CV} = \frac{\pi}{4\omega_{ZX}(A, \phi)},$$

which is half the value of calibrated CX gate’s CR pulse duration. The CR pulse envelope is a GaussianSquare pulse, i.e. a square pulse with Gaussian-shaped rising and falling edges [21] (see also Appendix A).

Note that, in all experimental demonstration shown in this paper, we keep the basic structure of the pulse schedule and amplitude parameters, for the combined CR and cancellation pulses in Fig. 2 unchanged, whereas we replace the local gate parameters for $[IX]^{1/2}$ and $[ZI]^{1/2}$ in the CX pulse definition with those of $[IX]^{1/4}$ and $[ZI]^{1/4}$; moreover, the CR pulse duration is changed to the value corresponding to $[ZX]^{-1/4}$.

C. EXPERIMENTAL ENVIRONMENT

In the present work, we used the 0th, 1st, and 4th qubits of ibmq_toronto, as shown in Fig. 3. Single-qubit gate operations on qubits 0, 1, and 4 are realized by the microwave irradiation to the drive-channel, $d0$, $d1$, and $d4$, respectively, whereas the CR-pulses for the two-qubit interactions between qubits 0 and 1, and qubits 1 and 4 are applied to the control channels, $u0$ and $u3$. Each experiment demonstrated in this paper was conducted 8192 times (meaning that 8192 measurement was performed for each circuit). There exist measurement errors that accidentally flips the detected bit; we applied the readout error-mitigation technique [22] to fix this error. We list the single-qubit gate error and the readout error of the device in Table 1. Also the two-qubit CX gate errors are 1.065% and 1.5969% for the 0-1 qubits pair and 1-4 qubits pair, respectively.

D. EXPERIMENTAL RESULTS

We implemented the gate (6) with several values of the pulse duration $\tau_d$ of the two CR pulses, which correspond to CR− and CR+ shown in Fig. 2 from 45.5 ns to 161 ns. For each $\tau_d$ we test the following trial CV gate:

$$CV_{trial}(\tau_d) = [ZI]^{1/4}[ZX]^{\theta(\tau_d)}[IX]^{1/4},$$

where $\theta(\tau_d) = -\tau_d/4t_{CV}$. Note that the duration for realizing the CX gate is 196 ns (see Appendix B for details); hence, from the relation $CV^2 = CX$, ideally $\tau_d$ would be identical to $\tau_{CV} = 98 = 196/2$ ns to realize CV gate. We make this duration adjustment only for the flat-top part, and the Gaussian flanks are fixed. We applied the quantum process tomography (QPT) to construct the trial CV gate, to evaluate its gate fidelity $F_p$ to the ideal CV gate [23]–[25]. Note that we can use interleaved randomized benchmarking [20] or randomized_benchmarking function in the Qiskit libraries [22], to estimate the gate fidelity.

Figure 4 shows the gate fidelity of the trial CV gate (9) as a function of the CR pulse duration $\tau_d$, with and without the readout mitigation; these are the averages of three experimental results conducted three different days. The black line represents the theoretically calculated gate fidelity between the exact CV gate and the trial CV gate (9), as a function of the CR duration; in the latter, $[ZX]^{1/4}$ can be analytically calculated using Eq. (4), and $\theta(\tau_d)$ linearly increases with respect to $\tau_d$. Also for reference, the gate fidelity of the CV gate implemented in the QASM format (denoted as QASM CV) are shown. First, note that the readout error-mitigation works well and gives better fidelity values compared to the raw (unmitigated) results. The mitigated fidelity of CV gate implemented with OpenPulse (denoted as Pulse CV) takes the maximum value 99.23% (averaged value for three different days) at the CR duration $\tau_d = 101.5$ ns, which
is close to the expected value $\tau_{CV} = 98$ ns, i.e., half the duration of CR pulse of the calibrated CX gate. Throughout all three different experiments, the maximum value is taken at 101.5 ns, which indicates that the optimal pulse duration is robust against calibration change. Another important finding is that the maximum value 99.23% is 0.66% higher than that of the CV gate fidelity achieved via the default QASM-based implementation using 2 CX gates.

Figure 5 shows the actual pulse sequence of CV gate implemented in (a) the default QASM format with 2 CX gates (see Fig. 1) and (b) OpenPulse with the optimal pulse duration 101.5 ns. The total gate time of CV gate is 994 ns for the former, while it is 343 ns for the latter. Hence the present OpenPulse-based implementation achieves 65.5% reduction in the total gate time of CV gate, compared to the default one (a), in addition to 0.66% improvement in the gate fidelity.

III. TWO-QUBIT GATE DESIGN WITH CV GATES

Arbitrary two-qubit gates can be implemented with three CX gates [26], [27]. However, generating two-qubit interactions only with CX gates can unnecessarily prolong the gate time. In this section, we study the set of two-qubit gates that can be configured with up to three CV gates instead of the same number of CX gates, based on the theory of Cartan decomposition. In particular, we consider $\sqrt{SWAP}$ gate and $\sqrt{iSWAP}$ gate as examples; they can be implemented with three and two CV gates, respectively, and thus the resulting gate-time is obviously shortened compared to the default CX-based implementations. We have also experimentally confirmed that the gate fidelity of those CV-based gates is superior to that of the CX-based one.

A. CARTAN DECOMPOSITION

The Cartan decomposition proves that an arbitrary two-qubit unitary operation $U \in SU(4)$ can be represented in the form

$$U = k_1 \exp \left( \frac{i}{2} (aX \otimes X + bY \otimes Y + cZ \otimes Z) \right) k_2,$$

(10)

where $k_1, k_2 \in SU(2) \otimes SU(2)$ are local single-qubit operations. When two-qubit unitaries $U$ and $V$ are connected through $U = k_1 V k_2$, we call that $U$ and $V$ are locally equivalent.

The Cartan decomposition is directly used to construct Weyl chamber that provides a clear view of geometric structure of the set of all non-local two-qubit gates. The Weyl chamber is illustrated as the tetrahedron $OA_1A_2A_3$ in Fig. 6(a); the point $[a, b, c]$ represents a locally equivalent class of two-qubit gate [26], [28]. Shown in Fig. 6(b) are particularly important points corresponding to familiar two-qubit gates, $L = [\pi/2, 0, 0]$ for $\{CX, CY, CZ\}$, $A_2 = [\pi/2, \pi/2, 0]$ for $\{DCX, iSWAP\}$, $A_3 = [\pi/2, \pi/2, \pi/2]$ for $\{SWAP\}$, and $B_3 = [\pi/4, \pi/4, \pi/4]$ for $\sqrt{SWAP}$. Note from Eq. (5) that $\text{CX}$ is locally equivalent to $[ZX]^{-1/2}$, which is further locally equivalent to $[XZ]^{-1/2}$ and thus identified by $L = [\pi/2, 0, 0]$. From this view, it is clear that CV corresponds to $C_1 = [\pi/4, 0, 0]$. A particularly useful result provided by this geometric picture is that $n$ ($\geq 3$) times repetition of $[\gamma, 0, 0]$ with $\gamma \in (0, \pi/2]$ can create an arbitrary two-qubit gate $[a, b, c]$. 

![Image 36x751 to 127x770]
can generate arbitrary two-qubit unitary gates. Similarly, by using two $[\gamma, 0, 0]$ gates, we can create arbitrary two-qubit gate $[a, b, 0]$ that satisfies the following condition:

$$0 \leq a + b \leq 2\gamma, \quad a - b \geq \pi - 2\gamma.$$  

Thus, two CX gates can generate any two-qubit gate represented by the point inside the triangle $OA_1A_2$, which corresponds to the base of the Weyl chamber (see Fig. 6).

### B. CONFIGURABLE CV-BASED TWO-QUBIT GATES

We can now characterize the set of two-qubit gates generated by two or three operations of CV gate represented by $C_1 = [\pi/4, 0, 0]$.

First, Eq. (12) with $\gamma = \pi/4$ indicates that 2 CV gates can generate any unitary gate represented by the point in the locally equivalent areas $OLB$ and $A_1LC$ illustrated in Fig. 7(23). These areas are included in the triangle $OA_1A_2$. Hence, there exist gates such that 2 CX gates can generate while 2 CV gates cannot, such as DCX (Double-CX gate, i.e., a 2-qubit gate composed of two back-to-back CX gates with alternate controls) or equivalently iSWAP represented by $A_2 = [\pi/2, \pi/2, 0]$. However, there are still many useful two-qubit gate in $OLB$ and $A_1LC$, and it is thus important to have the pulse-engineered CV gate for generating those gates with significantly shorter time and possibly better gate fidelity than the case using the default QASM-based implementation with only CX. For example, the controlled-$U$ gate plays an essential role in several quantum algorithms such as Quantum Fourier Transform; fortunately, an arbitrary controlled-$U$ gate is specified by the point $[\gamma, 0, 0]$ on the line $OL$ or $A_1L$ and thus can be generated using two CV gates.

Second, Eq. (11) with $n = 3$ and $\gamma = \pi/4$ elucidates the set of two-qubit gates that can be generated with 3 CV gates, which is depicted in the colored area in Fig. 6(c). We can expect the same advantage as the 2 CV case, in implementing some two-qubit gates contained in this area via three pulse-engineered CV gates.

### C. EFFICIENT IMPLEMENTATION OF $\sqrt{\text{SWAP}}$ AND $\sqrt{\text{SWAP}}$ VIA PULSE-ENGINEERED CV GATES

Here we show an experimental demonstration to implement the following 2 two-qubit gates via the pulse-engineered CV gates. That is, we consider $\sqrt{\text{SWAP}}$ gate represented by the point $B = [\pi/4, \pi/4, 0]$ in Fig. 7.

$$\sqrt{\text{SWAP}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1+i}{\sqrt{2}} & \frac{1+i}{\sqrt{2}} & 0 \\ 0 & \frac{1-i}{\sqrt{2}} & \frac{1-i}{\sqrt{2}} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$  

and $\sqrt{\text{SWAP}}$ gate represented by the point $B_3 = [\pi/4, \pi/4, \pi/4]$ in Fig. 6.

$$\sqrt{\text{SWAP}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1+i}{2} & \frac{1+i}{2} & 0 \\ 0 & \frac{1-i}{2} & \frac{1-i}{2} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$
Each of these gates together with some single-qubit gates can construct a universal gate set.

Recall that we cannot determine the Cartan decomposition uniquely, for any two-qubit unitary matrix \( U \). Thus, we used the decomposition algorithm 'TwoQubitBasisDecomposer' implemented in Qiskit [22]. Figure 8 shows two types of decomposed gate layout of \( \sqrt{\text{SWAP}} \) based on CX (middle) and CV (lower), which we call \( \sqrt{\text{SWAP}}_{\text{CX}} \) and \( \sqrt{\text{SWAP}}_{\text{CV}} \), respectively. Also the case of \( \sqrt{\text{SWAP}} \) is shown in Fig. 9 where the CX- and CV-based decompositions are called \( \sqrt{\text{SWAP}}_{\text{CX}} \) and \( \sqrt{\text{SWAP}}_{\text{CV}} \), respectively. Here, \( U_{2}(\phi, \lambda) \) and \( U_{3}(\theta, \phi, \lambda) \) are the single qubit gates in the QASM language [29], defined as follows:

\[
U_{2}(\phi, \lambda) = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & e^{i \lambda} \\ e^{i \phi} & e^{i(\phi+\lambda)} \end{bmatrix}, \quad (15)
\]

\[
U_{3}(\theta, \phi, \lambda) = \begin{bmatrix} \cos(\theta/2) & -e^{i \lambda} \sin(\theta/2) \\ e^{i \phi} \sin(\theta/2) & e^{i(\phi+\lambda)} \cos(\theta/2) \end{bmatrix} \quad (16)
\]

The pulse schedules corresponding to these four decomposed circuits are shown in Figs. 10 and 11, where the pulse for CX and CV were implemented with the optimized CR gates in the QASM language [29], defined as follows:

\[ U_{2}(\phi, \lambda) = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & e^{i \lambda} \\ e^{i \phi} & e^{i(\phi+\lambda)} \end{bmatrix}, \quad (15) \]

\[ U_{3}(\theta, \phi, \lambda) = \begin{bmatrix} \cos(\theta/2) & -e^{i \lambda} \sin(\theta/2) \\ e^{i \phi} \sin(\theta/2) & e^{i(\phi+\lambda)} \cos(\theta/2) \end{bmatrix} \quad (16) \]

The pulse schedules corresponding to these four decomposed circuits are shown in Figs. 10 and 11, where the pulse for CX and CV were implemented with the optimized CR duration time identified in Section III.

**FIGURE 8.** Circuit diagram for \( \sqrt{\text{SWAP}}_{\text{CX}} \) gate (middle) and \( \sqrt{\text{SWAP}}_{\text{CV}} \) gate (lower).

**FIGURE 9.** Circuit diagram for \( \sqrt{\text{SWAP}}_{\text{CX}} \) gate (middle) and \( \sqrt{\text{SWAP}}_{\text{CV}} \) gate (lower).

\( \sqrt{\text{SWAP}}_{\text{CV}} \) is 532 ns shorter than that of \( \sqrt{\text{SWAP}}_{\text{CX}} \). We compute the gate fidelities of these four gates to their ideal correspondence, using QPT. The results are summarized in Table 2 together with the gate time; \( \sqrt{i \text{SWAP}} \) (\( \sqrt{\text{SWAP}} \)) gate with CV gates achieves the better fidelity by 0.87 (2.14) % compared with the default CX-based implementation. This might be thanks to the shortened gate time realized via the pulse-engineered CV gate. Note that, when \( \sqrt{i \text{SWAP}} \) or \( \sqrt{\text{SWAP}} \) is involved in some larger
quantum circuits, the gate-time advantage of the CV-based implementation may lead to significant improvement in the fidelity of those circuit.

### TABLE 2. Gate fidelity and gate time of \( \sqrt{\text{SWAP}} \) and \( \sqrt{\text{SWAP'}} \), implemented with the default CX gate or the pulse-engineered CV gate. For each input, we performed 8192 shots and calculated the gate fidelity \( F_p \) with the use of read-out error mitigation.

| Gate               | #CX | #CV | Fidelity \( (F_p) \) | Gate time (ns) |
|--------------------|-----|-----|----------------------|---------------|
| \( \sqrt{\text{SWAP}} \) | 2   | –   | 0.9765               | 1064          |
| \( \sqrt{\text{SWAP'}} \) | –   | 2   | 0.9852               | 756           |
| \( \sqrt{\text{SWAP}} \) | 3   | –   | 0.9604               | 1631          |
| \( \sqrt{\text{SWAP'}} \) | –   | 3   | 0.9818               | 1099          |

### IV. HIGH-SPEED AND HIGH-PRECISION TOFFOLI GATE WITH CV GATES

The pulse-engineered CV gate can be applied to improve the speed and precision of bigger size gates beyond the two-qubit case. As a demonstration, here we study the three-qubit Toffoli gate (or the Controlled-Controlled-X gate). The idea presented here is applicable to the general multi-qubit Toffoli gate appearing in many long-term algorithms such as QRAM database [30] and the diffusion operator in Grover’s search algorithm [31].

#### A. GATE IMPLEMENTATION FOR LINEARLY-COUPL ED THREE QUBITS

If three qubits are fully connected, then we can construct Toffoli gate using 6 CX gates (and some single-qubit gates), while the combination of 3 CV and 2 CX gates also constructs Toffoli gate; hence the pulse-engineered CV gate enables reducing the total gate time. However, the standard structure of the current IBM Quantum devices is of the linear coupling form of qubits, in which case the number of necessary gates increases.

Here we consider two different construction of Toffoli gate with and without CV gates, \( TOF_{CV} \) and \( TOF_{CX} \) gate shown in Fig. 12 note that \( q_j \) \((j = 0,1,4)\) represents the \( j \)th qubit of ibmq_tronto device shown in Fig. 3 and thus \( q_0 \) and \( q_4 \) are not directly connected. \( TOF_{CV} \) gate has 3 CX and 3 CV gates; hence, with the use of pulse-engineered CV gate, the total gate time of \( TOF_{CV} \) becomes shorter than that of the textbook Toffoli with 6 CX gates as well as \( TOF_{CX} \). Note that the \( \text{SWAP} \) gate is built in there to connect \( q_0 \) and \( q_4 \); consequently, \( TOF_{CV} \) exchanges \( q_0 \) and \( q_1 \), while maintaining the functionality of Toffoli gate. However, the pure Toffoli composed of \( TOF_{CV} \) and subsequent \( \text{SWAP} \) gate needs 6 CX and 3 CV gates, meaning that it still can be realized with shorter gate time than \( TOF_{CX} \) by the pulse engineering of CV.

#### B. EXPERIMENTAL RESULTS

We conducted an experiment to compare the actual performance of \( TOF_{CV} \) (3 CX and 3 CV) to \( TOF_{CX} \) (8 CX), where the pulse-engineered CV is used in the former, on ibmq_tronto processor shown in Fig. 3. The pulse sequences corresponding to these Toffoli gates are depicted in Fig. 13. As expected, the total gate time are 1778 ns and 2835 ns for \( TOF_{CV} \) and \( TOF_{CX} \), respectively, suggesting that \( TOF_{CV} \) would have better precision than \( TOF_{CX} \).
gate, we prepared 12 states listed in Table 3, where $|\pm\rangle = (|0\rangle \pm |1\rangle)/\sqrt{2}$. We performed 8192 shots (measurements) for each initial state and calculated $F_{s}(\rho_{\exp}, \rho_{\text{ide}})$. Table 3 summarizes the results, showing the superiority of TOF$_{CV}$ for all input states except $|111\rangle$. As a result, TOF$_{CV}$ has 4.06% higher average fidelity than TOF$_{CX}$. This is a bigger superiority of the CV-based gate over the conventional CX-based one, compared to the previous case shown in Table 1, simply because the gate length becomes longer.

| Input states | TOF$_{CX}$ | TOF$_{CV}$ |
|--------------|------------|------------|
| [000]        | 0.9287     | 0.9773     |
| [001]        | 0.9538     | 0.9711     |
| [010]        | 0.8886     | 0.9318     |
| [011]        | 0.9344     | 0.9450     |
| [100]        | 0.8809     | 0.9259     |
| [101]        | 0.9113     | 0.9481     |
| [110]        | 0.9167     | 0.9284     |
| [111]        | 0.9217     | 0.9089     |
| [+10]        | 0.8545     | 0.9462     |
| [+11]        | 0.9046     | 0.9341     |
| [+10]        | 0.8860     | 0.9513     |
| [+11]        | 0.8603     | 0.9408     |
| Average      | 0.9018     | 0.9424     |

### V. CONCLUSION

Using only CX gates for entangling qubits in quantum computation is now a de facto standard. IBM Quantum is no exception. While this approach is less burdensome for calibration, it has the disadvantage that some gate/circuit structure become redundant. To resolve this issue, in this paper we proposed using CV gates in addition to the default gate set; actually OpenPulse allows us to realize CV gate with shorter gate time than that of CX gate as well as the default CV gate composed of 2 CX gates. The parameters of the corresponding CR Hamiltonian for realizing such pulse-engineered CV gate are the same as those of the CX gate, except for the pulse length and some local gate parameters, meaning that the calibration burden is not significant. In particular, the result of Section II-C (Fig. 4) indicates that the optimal pulse length does not change in each calibration. The gate-time improvement in circuit design, which eventually leads to the gate-fidelity improvement, has been demonstrated with $\sqrt{SWAP}$, $\sqrt{SWAP}$, and Toffoli gates. Note that the gate fidelity improvement were not totally great (0.66% improvement for the CV implementation, 0.87% for $\sqrt{SWAP}$, and 2.14% for $\sqrt{SWAP}$), and this may be due to the presence of ZZ interactions that cannot be counteracted by the echo scheme [14] that was employed in our method. Suppression of the ZZ interactions [34]–[37] would allow us to further improve the gate performance.

In summary, from the practicality and feasibility viewpoint, we believe that the new gate set that contains the proposed pulse-engineered CV gate can be used to effectively reduce the redundancy of several quantum circuits, thereby realize shorter gate time in total, and eventually improve several quantum algorithms. Actually, to investigate a wider range of applications, we plan to execute comparative verification of the proposed method on a bigger-size circuit or a near-term quantum algorithm.

### APPENDIX.

#### A. GAUSSIAN SQUARE PULSE ENVELOPE

In all experiments we employed the Gaussian-Square pulse composed of the constant-amplitude part of length (width) $\tau_w$ and Gaussian-formed rising and falling edges of length $\tau_r$. The overall pulse waveform $f(t)$, as a function of time $t$, is thus given by

$$f(t) = \begin{cases} A \exp\left(-\frac{1}{2\sigma^2}(t - \frac{\tau_r}{2})^2\right), & 0 \leq t < \tau_r \\ A \left(\tau_r \leq t < \tau_r + \tau_w\right) \\ A \exp\left(-\frac{1}{2\sigma^2}(t - \frac{\tau_r + \tau_w}{2})^2\right), & \tau_r + \tau_w \leq t < \tau_d, \end{cases}$$

where $A$ is the maximum amplitude and $\sigma^2$ is the variance of the Gaussian part, respectively. Note that the overall pulse length or the duration is defined as

$$\tau_d = 2\tau_r + \tau_w.$$  

#### B. OPTIMAL PULSE DURATION OF CX GATE

Figure 1 shows the pulse schedule for implementing CX gate, where the CR pulse duration is 196 ns and accordingly the total gate time 462 ns; this is actually the best value that achieves the maximum gate fidelity. Here we show the detail of the OpenPulse experiment to identify this optimal duration.

The experiment was conducted in the same setting described in Section II-C, with the use of qubit 1 and 4. We evaluated the following trial CX gate with changing the duration $\tau_d \in [144, 259]$ ns:

$$CX_{\text{trial}}(\tau_d) = [ZI]^{1/2}[ZX]^\theta(\tau_d)[IY]^{1/2}$$  

where $\theta(\tau_d) = -\tau_d/2\tau_{CX}$ with the nominal value $\tau_{CX} = 196$ ns. The yellow and blue lines in Fig. 14 depict the gate fidelity between the pulse-engineered $CX_{\text{trial}}(\tau_d)$ and the ideal CX gate, with and without the readout error mitigation respectively. The black dotted line depicts the gate fidelity between the theoretical $CX_{\text{trial}}(\tau_d)$ and the ideal CX gate. The figure thus shows that the optimal duration is exactly the nominal value, i.e., $\tau_d = \tau_{CX} = 196$ ns, which achieves the perfect gate fidelity.

### ACKNOWLEDGMENT

The results presented in this paper were obtained in part using an IBM Q quantum computing system as part of the IBM Network. The views expressed are those of the authors and do not reflect the official policy or position of IBM or the IBM Q team.

### REFERENCES

[1] D. C. McKay, T. Alexander, L. Bello, M. J. Biercuk, L. Bishop, J. Chen, J. M. Chow, A. D. Córcoles, D. Egger, S. Filipp et al., “Qiskit backend specifications for openqasm and openpulse experiments,” arXiv preprint arXiv:1809.03452, 2018.
P. Gokhale, A. Javadi-Abhari, N. Earnest, Y. Shi, and F. T. Chong, “Optimized quantum compilation for near-term algorithms with open-pulse,” in 2020 53rd Annual IEEE/ACM International Symposium on Microarchitecture (MICRO). IEEE, 2020, pp. 186–200.

J. P. Stenger, N. T. Bronn, D. J. Egger, and D. P. Pekker, “Simulating the dynamics of braiding of majorana zero modes using an ibm quantum computer,” Physical Review Research, vol. 3, no. 4, p. 043088, 2021.

T. Nguyen and A. Mccaskey, “Enabling pulse-level programming, compilation, and execution in xacc,” IEEE Transactions on Computers, 2021.

A. Barenco, C. H. Bennett, R. Cleve, D. P. DiVincenzo, N. Margolus, P. Shor, T. Sleator, J. A. Smolin, and H. Weinfuter, “Elementary gates for quantum computation,” Physical Review A, vol. 52, no. 5, p. 3457, 1995.

J-L. Brylinski and R. Brylinski, “Universal quantum gates,” Mathematics of quantum computation, vol. 79, 2002.

A. Zulehner and R. Wille, “Compiling su (4) quantum circuits to ibm qx architectures,” in Proceedings of the 24th Asia and South Pacific Design Automation Conference, 2019, pp. 185–190.

S. Kirchhoff, T. Kessler, P. J. Liebermann, E. Assémat, S. Machnes, F. Motzoi, and F. K. Wilhelm, “Optimized cross-resonance gate for coupled transmon systems,” Physical Review A, vol. 97, no. 4, p. 042348, 2018.

C. Rigetti and M. Devoret, “Fully microwave-tunable universal gates in superconducting qubits with linear couplings and fixed transition frequencies,” Physical Review B, vol. 81, no. 13, p. 134507, 2010.

E. Magesan and J. M. Gambetta, “Effective hamiltonian models of the cross-resonance gate,” Physical Review A, vol. 101, no. 5, p. 052308, 2020.

N. Sundaesran, I. Lauer, E. Pritchett, E. Magesan, P. Jurcevic, and J. M. Gambetta, “Reducing unitary and spectator errors in cross resonance with optimized rotary echoes,” PRX Quantum, vol. 1, no. 2, p. 020318, 2020.

P. Krantz, M. Kjaergaard, F. Yan, T. P. Orlando, S. Gustavsson, and W. D. Oliver, “A quantum engineer’s guide to superconducting qubits,” Applied Physics Reviews, vol. 6, no. 2, p. 021318, 2019.

S. Sheldon, E. Magesan, J. M. Chow, and J. M. Gambetta, “Procedure for systematically tuning up cross-talk in the cross-resonance gate,” Physical Review A, vol. 93, no. 6, p. 060302, 2016.

IBM, “IBM quantum experience.”

P. Murali, D. C. McKay, M. Martinoni, and A. Javadi-Abhari, “Software mitigation of crosstalk on noisy intermediate-scale quantum computers,” in Proceedings of the Twenty-Fifth International Conference on Architectural Support for Programming Languages and Operating Systems, 2020, pp. 1001–1016.

P. Mundada, G. Zhang, T. Hazard, and A. Houck, “Suppression of qubit crosstalk in a tunable coupling superconducting circuit,” Physical Review Applied, vol. 12, no. 5, p. 054023, 2019.

FIGURE 14. Gate fidelity of the trial CX gate (17) to the ideal CX gate, as a function of the duration of CR pulse.