Pulse-Engineered Controlled-V Gate and Its Applications on Superconducting Quantum Device

TAKAHIKO SATOH[1,2] , **SHUN OOMURA[1,2], MICHIHIKO SUGAWARA[1,2], AND NAOKI YAMAMOTO[1,2]**

[1]Quantum Computing Center, Keio University, Yokohama 223-8522, Japan
[2]Graduate School of Science and Technology, Keio University, Yokohama 223-8522, Japan

Corresponding author: Takahiko Satoh (e-mail: satoh@sfc.wide.ad.jp).

ABSTRACT In this article, we demonstrate that, by employing the 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 gate (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 (CV 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 article, 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: 1) OpenPulse [1] and 2) quantum assembly language (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 the circuit design. As a result, OpenPulse can reduce the execution time through the optimal pulse design for various type of quantum gates [2], [3]; also, it can be applied to generate a new gate specific to a particular physical simulation [4]. Recently, a computational framework has been proposed to aid such synthesis problems [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 [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 [8]. The point of taking this fixed gate set is that, because it contains only one 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 [9]–[13], with the help of the echo scheme and the cancellation pulse technique [14]. However, the error rate of CX gate is still much higher than that of single-qubit gates [15], due to the longer pulse length (gate time) than that of single-qubit gates and the effects of cross-talk [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.
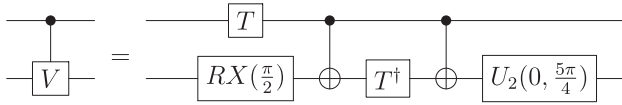
FIGURE 1. Circuit diagram of cv gate in QASM-based implementation.

The abovementioned issue may be resolved by adding some two-qubit gates to the default universal gate set composed of single-qubit gates and cx gate. In this article, we take the controlled-V (cv) gate whose matrix representation in the computational basis is given by

$$
CV = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1+i}{2} & \frac{1-i}{2} \\ 0 & 0 & \frac{1-i}{2} & \frac{1+i}{2} \end{bmatrix} \tag{1}
$$

which readily leads to the relation $CV^2 = CX$. Note that, in the cx-based default implementation of cv gate on QASM, one needs two cx gates to create the two-qubit interaction process, as shown in Fig. 1. 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 six cx gates to implement if cx is only given to us, but it can be implemented using two cx and three cv gates if cv gate is further available [6], [19], [20].

The rest of this article is organized as follows. In Section II, we describe how to implement cv gate using Open-Pulse, 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 that of the default cx-based one. Actually, we show the experimental demonstration to generate $\sqrt{\text{iSWAP}}$ and $\sqrt{\text{SWAP}}$ 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. CR INTERACTION
On IBM quantum devices, the 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]:

$$
H_{\text{CR}} = \sum_{P=I,X,Y,Z} \frac{\omega_{ZP}(A, \phi)}{2} Z \otimes P
$$
$$
+ \sum_{Q=X,Y,Z} \frac{\omega_{IQ}(A, \phi)}{2} I \otimes Q \tag{2}
$$

where the qubit ordering is control $\otimes$ target and ω_{ZP} and ω_{IQ} represent the interaction strengths, which are the functions of the amplitude A and the phase ϕ of the microwave pulse, respectively. 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

$$
U_{\text{CR}} = \exp(-itH_{\text{CR}}). \tag{3}
$$

B. PULSE-ENGINEERED CX AND CV GATES
Let us define the general two-qubit unitary operator

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

where D and E are the arbitrary single-qubit operators. With this notation, the cx gate is represented as [10]

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

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 the $Z \otimes X$ 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 $\tilde{H}_{ZX}$ 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 $\tilde{H}_{ZX}$ by setting the interaction strength in terms of the pulse duration t as $\theta = \omega_{ZX}(A, \phi)t/\pi$.

$$
[ZX]^\theta = \tilde{U}_{ZX} = \exp\left(-i\pi t \tilde{H}_{ZX}\right)
$$
$$
\tilde{H}_{ZX} = \frac{\omega_{ZX}(A, \phi)}{2} Z \otimes X. \tag{6}
$$

For the cx gate case, the two-qubit interaction time t_{CX} should be $t_{\text{CX}} = \pi/2\,\omega_{ZX}(A, \phi)$ to realize $\theta = -1/2$. In the
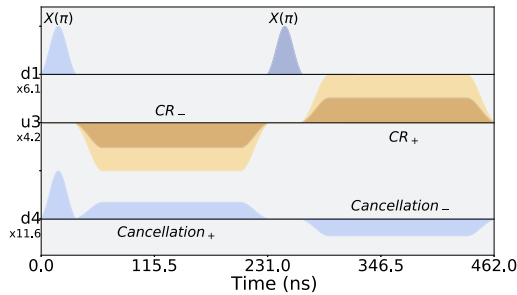
FIGURE 2. Pulse schedule of cx gate for the control qubit 1 and the target qubit 4, implemented on IBM Q Toronto. Here, d1 and d4 denote the drive channels for local operations of qubits 1 and 4, respectively, while u3 is the control channel for CR pulse responsible for the two qubit interactions. CR_+ and CR_- are the CR pulse shapes implementing $[ZX]^{1/2}$ on u3; d4 is the channel serving as the cancellation pulse. Two π-pulses on d1 placed before and after the first CR pulse are used to realize the echo scheme. The first Gaussian pulse on d4 corresponds to $[IX]$ in (5), whereas $[ZI]$ is implemented without actual pulse-irradiation.

following, from (5) and the fact that IX, ZX, and ZI commute with each other, one can see that cv gate is decomposed as

$$\mathrm{CV} = [ZI]^{1/4}[ZX]^{-1/4}[IX]^{1/4}. \tag{7}$$

In this article, we directly implement $[ZX]^{-1/4}$ part using OpenPulse, without decomposing this gate into multiple cx gates. As expected from (6), the interaction strength θ of the two qubits interaction part $[ZX]^\theta$ is proportional to the duration of CR pulse, as far as the effective Hamiltonian stands. Thus, we can create $[ZX]^{-1/4}$ by taking the duration of the CR pulse t_{CV} as

$$t_{\mathrm{CV}} = \frac{\pi}{4\omega_{ZX}(A, \phi)} \tag{8}$$

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 article, 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 $[ZI]^{1/4}$, respectively; moreover, the CR pulse duration is changed to the value corresponding to $[ZX]^{1/4}$.

C. EXPERIMENTAL ENVIRONMENT
In this article, we used the zeroth, first, and fourth 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, respectively. Each experiment demonstrated in this article was conducted 8192 times (meaning that 8192 measurements were performed for each circuit). There exist measurement errors that accidentally flips the detected
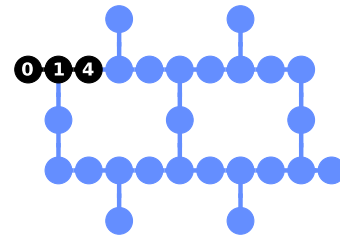

FIGURE 3. Coupling map of ibmq_toronto processor. We used qubits 0, 1, and 4 for the experiments.

TABLE I Qubit Performance of Ibmq_toronto Processor

Qubit	$\sqrt{X}$ gate error	Readout error
0	2.78e-4	5.69e-2
1	6.97e-4	4.05e-2
4	3.42e-4	7.30e-2

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 I. Also, the two-qubit cx gate errors are 1.065% and 1.5969% for the 0 and 1 qubits pair and 1 and 4 qubits pair, respectively.

D. EXPERIMENTAL RESULTS
We implemented the gate (6) with several values of the pulse duration τ_d of the two CR pulses, which correspond to CR_- and CR_+, as shown in Fig. 2, from 45.5 to 161 ns. For each τ_d, we test the following trial cv gate:

$$\mathrm{CV_{trial}}(\tau_d) = [ZI]^{1/4}[ZX]^{\theta(\tau_d)}[IX]^{1/4} \tag{9}$$

where $\theta(\tau_d) = -\tau_d/4t_{\mathrm{CV}}$. Note that the duration for realizing the cx gate is 196 ns (see Appendix B for details); hence, from the relation $\mathrm{CV^2 = CX}$, ideally τ_d would be identical to $\tau_{\mathrm{CV}} = 98 = 196/2$ 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.

Fig. 4 shows the gate fidelity of the trial cv gate (9) as a function of the CR pulse duration τ_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, $[ZX]^\theta$ can be analytically calculated using (4) and $\theta(\tau_d)$ linearly increases with respect to τ_d. Also, for reference, the gate fidelity of the cv gate implemented in the QASM format (denoted as QASM cv) is 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
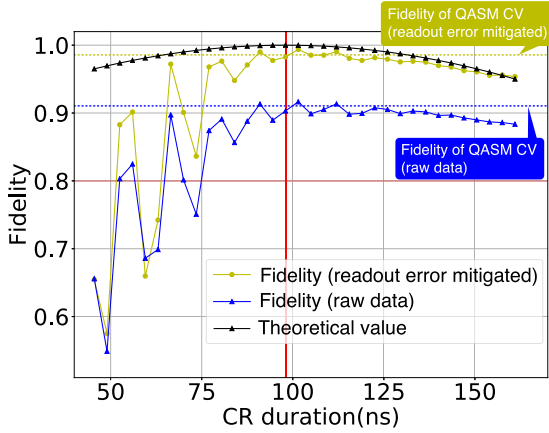
FIGURE 4. Gate fidelity of the trial cv gate (9) to the ideal cv gate, as a function of the duration of CR pulse. Red vertical line denotes half duration of CR pulse in cx pulse schedule. The black line represents the theoretically calculated gate fidelity between the exact cv gate and the trial cv gate (9). Here, the physical control and target qubit is the zeroth and the first one depicted in Fig. 3, respectively.

FIGURE 5. (a) Pulse sequence for cv gate in the QASM implementation. (b) Pulse sequence for cv gate implemented by OpenPulse with the CR pulse duration 101.5 ns.

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 two cx gates.

Fig. 5 shows the actual pulse sequence of cv gate implemented in (a) the default QASM format with two cx gates (see Fig. 1) and in (b) the 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 in (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 $\sqrt{\text{SWAP}}$ and $\sqrt{\text{iSWAP}}$ gates 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.
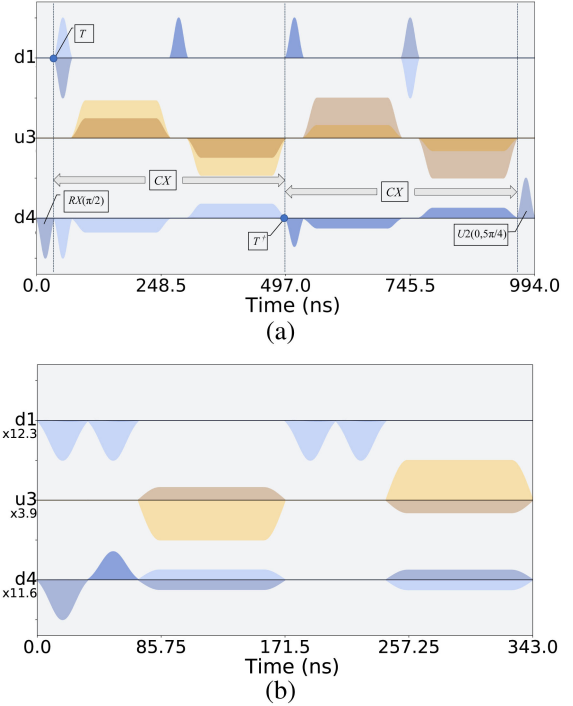
A. CARTAN DECOMPOSITION
The Cartan decomposition proves that an arbitrary two-qubit unitary operation $U \in SU(4)$ can be represented in the form

$$U = k_1 \exp\left\{\frac{i}{2}(aX \otimes X + bY \otimes Y + cZ \otimes Z)\right\} k_2 \quad (10)$$

where $k_1, k_2 \in SU(2) \otimes SU(2)$ are the 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 the Weyl chamber that provides a clear view of geometric structure of the set of all nonlocal 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]. As shown in Fig. 6(b), there 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 $\sqrt{\text{SWAP}}$. Note from (5) that cx is locally equivalent to $[ZX]^{-1/2}$, which is further locally equivalent to $[XX]^{-1/2}$, 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 $[a, b, c]$ that
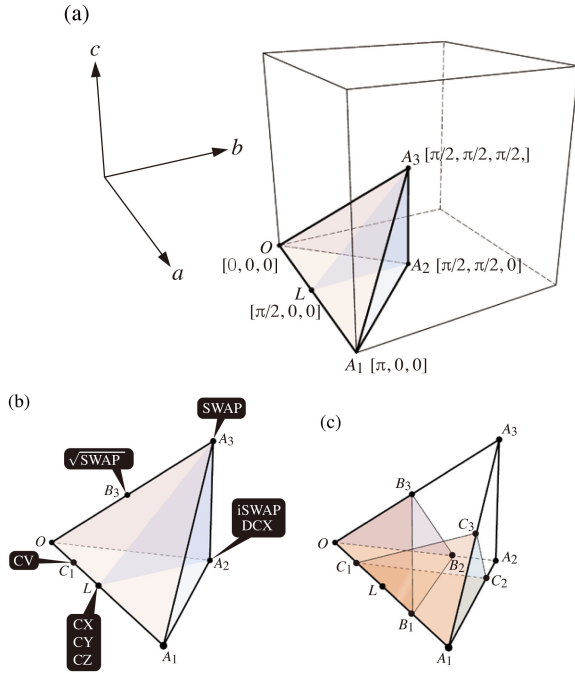
FIGURE 6. (a) Weyl chamber (tetrahedron $OA_1A_2A_3$) contains all the locally equivalent class of two-qubit operations, with the exception of points on its base (see the caption of Fig. 7). (b) Five important points in the Weyl chamber; at each point typical locally equivalent gates are indicated. (c) Colored area, i.e., the union of tetrahedra $OB_1B_2B_3$ and $A_1C_1C_2C_3$ in the Weyl chamber, shows the set of two-qubit gate realized with three cv gates. All the points in the figures are defined as $B_1 = [3\pi/4, 0, 0]$, $B_2 = [3\pi/8, 3\pi/8, 0]$, $B_3 = [\pi/4, \pi/4, \pi/4]$, $C_1 = [\pi/4, 0, 0]$, $C_2 = [5\pi/8, 3\pi/8, 0]$, and $C_3 = [3\pi/4, \pi/4, \pi/4]$.

satisfies the following condition:

$$0 \leq a + b + c \leq n\gamma, \quad a - b - c \geq \pi - n\gamma. \quad (11)$$

This equation implies that $n = 3$ operations of cx (or any of locally equivalent gate to $[\pi/2, 0, 0]$) with appropriate local gates can span the entire area of the Weyl chamber, i.e., tetrahedron $OA_1A_2A_3$; that is, as is well-known, three cx gates 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:

$$0 \leq a + b \leq 2\gamma, \quad a - b \geq \pi - 2\gamma. \quad (12)$$

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. 7).

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, (12) with $\gamma = \pi/4$ indicates that two cv gates can generate any unitary gate represented by the point in the locally equivalent areas OLB and A_1LC, as shown in Fig. 7 [28]. These areas are included in the triangle OA_1A_2. Hence, there exist gates such that two cx gates can generate, while
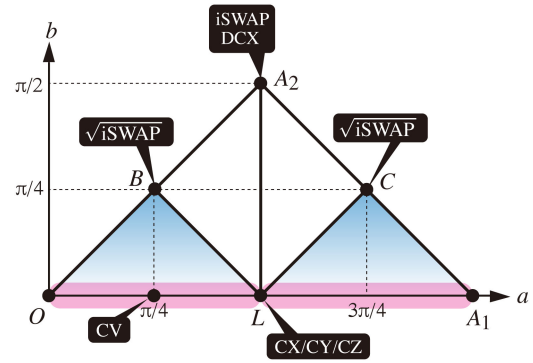

FIGURE 7. Area of two-qubit gates generated with two cx or cv gates. Two cx gates can generate any gate represented by the point in the triangle area OA_1A_2, which is the base of Weyl chamber. The blue region represents the set of gates that two cv gates can generate. The triangle areas OLB and A_1LC are locally equivalent; in particular, $B = [\pi/4, \pi/4, 0]$ and $C = [3\pi/4, \pi/4, 0]$ are equivalent points corresponding to $\sqrt{\text{iSWAP}}$ gate. The lines OL and LA_1 (overlaid with pink belt) correspond to the set of controlled-U gates.

two cv gates cannot, such as double-cx (DCX) gate, i.e., a two-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, (11) with $n = 3$ and $\gamma = \pi/4$ elucidates the set of two-qubit gates that can be generated with three cv gates, which is shown in the colored area in Fig. 6(c). We can expect the same advantage as the two cv case, in implementing some two-qubit gates contained in this area via three pulse-engineered cv gates.

C. EFFICIENT IMPLEMENTATION OF $\sqrt{ISWAP}$ AND $\sqrt{SWAP}$ VIA PULSE-ENGINEERED CV GATES
Here, we show an experimental demonstration to implement the following two two-qubit gates via the pulse-engineered cv gates. That is, we consider $\sqrt{\text{iSWAP}}$ gate represented by the point $B = [\pi/4, \pi/4, 0]$ in Fig. 7 as

$$\sqrt{iSWAP} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{\sqrt{2}} & \frac{i}{\sqrt{2}} & 0 \\ 0 & \frac{i}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (13)$$
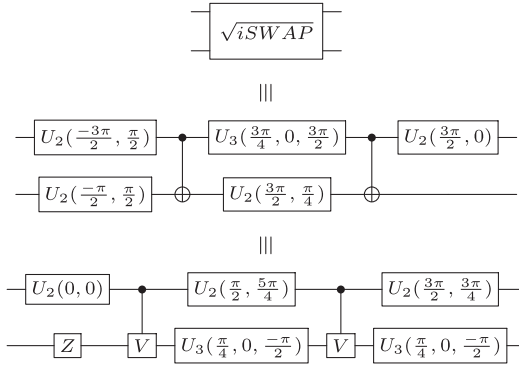
FIGURE 8. Circuit diagram for $\sqrt{\text{iSWAP}}_{\text{CX}}$ gate (middle) and $\sqrt{\text{iSWAP}}_{\text{CV}}$ gate (lower).
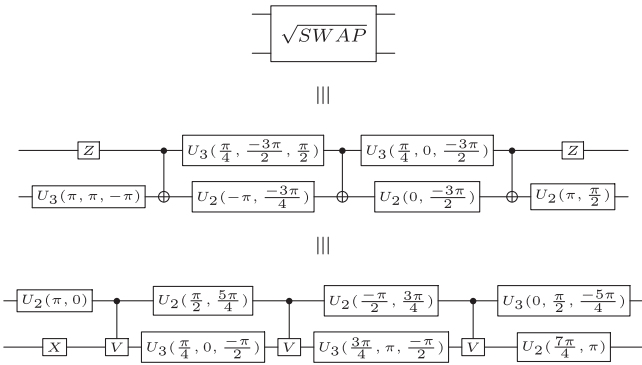
FIGURE 9. Circuit diagram for $\sqrt{\text{SWAP}}_{\text{CX}}$ gate (middle) and $\sqrt{\text{SWAP}}_{\text{CV}}$ gate (lower).

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

$$\sqrt{\text{SWAP}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1+i}{2} & \frac{1-i}{2} & 0 \\ 0 & \frac{1-i}{2} & \frac{1+i}{2} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{14}$$

Each of these gates together with some single-qubit gates can construct a universal gate set.

Recall that we cannot determine the Cartan decomposition (10) uniquely, for any two-qubit unitary matrix U. Thus, we used the decomposition algorithm "TwoQubitBasisDecomposer" implemented in Qiskit [22]. Fig. 8 shows two types of decomposed gate layout of $\sqrt{\text{iSWAP}}$ based on CX (middle) and CV (lower): 1) $\sqrt{\text{iSWAP}}_{\text{CX}}$; and 2) $\sqrt{\text{iSWAP}}_{\text{CV}}$, 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:

$$U_2(\phi, \lambda) = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & -e^{i\lambda} \\ e^{i\phi} & e^{i(\phi+\lambda)} \end{bmatrix} \tag{15}$$

$$U_3(\theta, \phi, \lambda) = \begin{bmatrix} \cos(\theta/2) & -e^{i\lambda}\sin(\theta/2) \\ e^{i\phi}\sin(\theta/2) & e^{i(\phi+\lambda)}\cos(\theta/2) \end{bmatrix}. \tag{16}$$
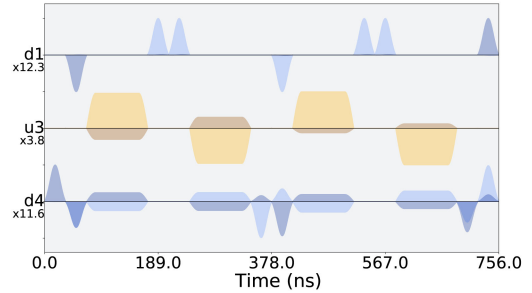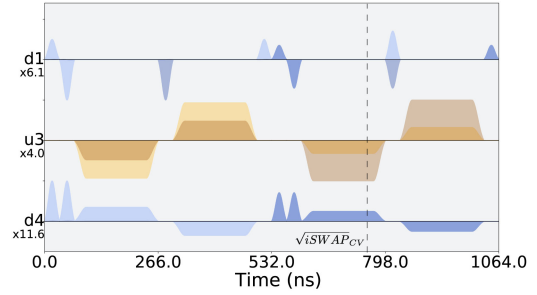

FIGURE 10. Pulse sequences of $\sqrt{\text{iSWAP}}_{\text{CX}}$ (upper) and $\sqrt{\text{iSWAP}}_{\text{CV}}$ (lower) corresponding to the circuit diagram in Fig. 8. Dashed line denotes the total gate time of $\sqrt{\text{iSWAP}}_{\text{CV}}$.
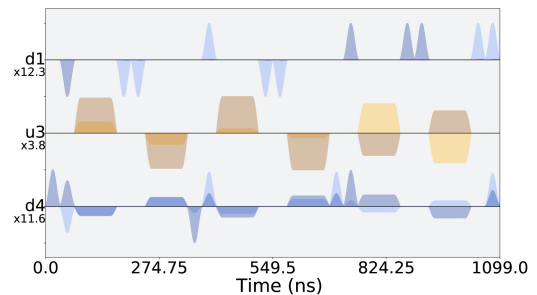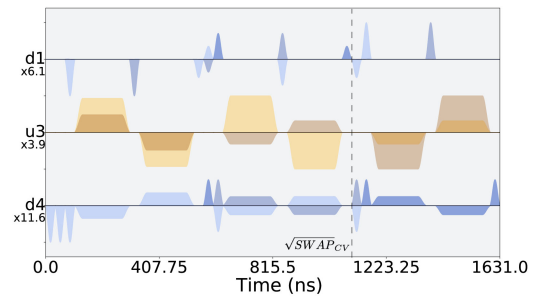

FIGURE 11. Pulse sequences of $\sqrt{\text{SWAP}}_{\text{CX}}$ (upper) and $\sqrt{\text{SWAP}}_{\text{CV}}$ (lower) corresponding to the circuit diagram in Fig. 9.

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 II.

Fig. 10 shows that the total gate time of $\sqrt{\text{iSWAP}}_{\text{CV}}$ is shortened by 308 ns compared to that of $\sqrt{\text{iSWAP}}_{\text{CX}}$. Also, from Fig. 11, we find that the total gate time of $\sqrt{\text{SWAP}}_{\text{CV}}$ 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 II, together

TABLE II Gate Fidelity and Gate Time of √iSWAP and √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)
$\sqrt{iSWAP}_{CX}$	2	–	0.9765	1064
$\sqrt{iSWAP}_{CV}$	–	2	0.9852	756
$\sqrt{SWAP}_{CX}$	3	–	0.9604	1631
$\sqrt{SWAP}_{CV}$	–	3	0.9818	1099

with the gate time; $\sqrt{iSWAP}$ ($\sqrt{SWAP}$) 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 $\sqrt{iSWAP}$ or $\sqrt{SWAP}$ 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.

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-cx gate). The idea presented here is applicable to the general multiqubit Toffoli gate appearing in many long-term algorithms, such as the QRAM database [30] and the diffusion operator in the Grover's search algorithm [31].

A. GATE IMPLEMENTATION FOR LINEARLY COUPLED THREE QUBITS

If three qubits are fully connected, then we can construct Toffoli gate using six cx gates (and some single-qubit gates), while the combination of three cv and two 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 increase.

   Here, we consider two different construction of the Toffoli gate with and without cv gates: 1) TOF$_{CV}$ gate; and 2) TOF$_{CX}$ gate, as 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 three cx and three 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 six cx gates as well as TOF$_{CX}$. Note that the 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 the Toffoli gate. However, the pure Toffoli composed of TOF$_{CV}$ and subsequent SWAP gate needs six cx and three cv gates, meaning that it still can be realized with shorter gate time than TOF$_{CX}$by the pulse engineering of cv.
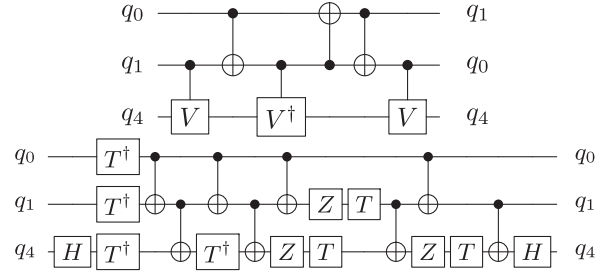

FIGURE 12. Toffoli gate for linearly coupled three qubits. (Upper) TOF$_{CV}$composed of CV and CX gates. (Lower) TOF$_{CX}$composed of only CX gates [32]. Note that the former exchanges q_0 and q_1, while maintaining the functionality of the Toffoli gate.
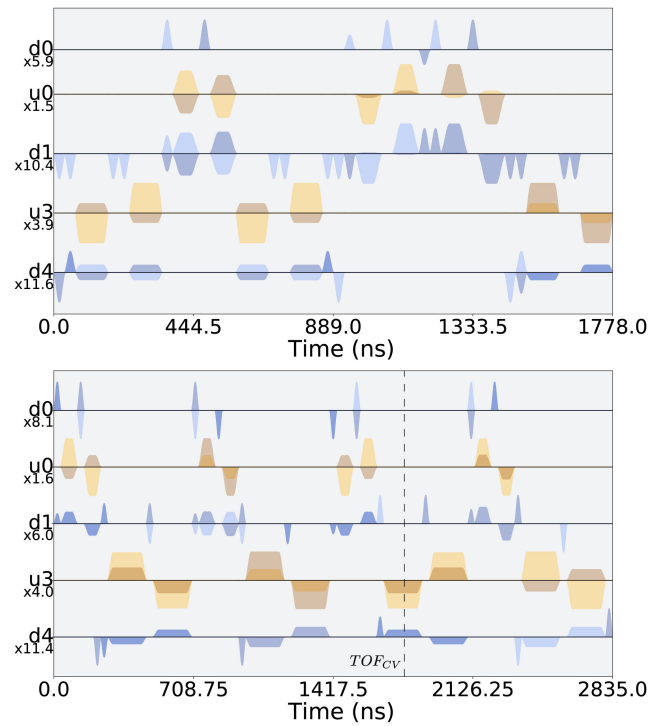

FIGURE 13. Pulse sequences of TOF$_{CV}$ composed of cx and cv gates (upper) and that of TOF$_{CX}$ composed of only cx gates (lower). In addition to the channels used in the previous sections, DriveChannel d0 of the control qubit and ControlChannel u0 as interaction of qubits 0 and 1 are used. The gate time of TOF$_{CV}$ and TOF$_{CX}$ are 1778 and 2835 ns, respectively.

B. EXPERIMENTAL RESULTS

We conducted an experiment to compare the actual performance of TOF$_{CV}$(three cx and three cv) to TOF$_{CX}$(eight cx), where the pulse-engineered cv is used in the former, on the 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 and 2835 ns for TOF$_{CV}$ and TOF$_{CX}$, respectively, suggesting that TOF$_{CV}$ would have better precision than TOF$_{CX}$.
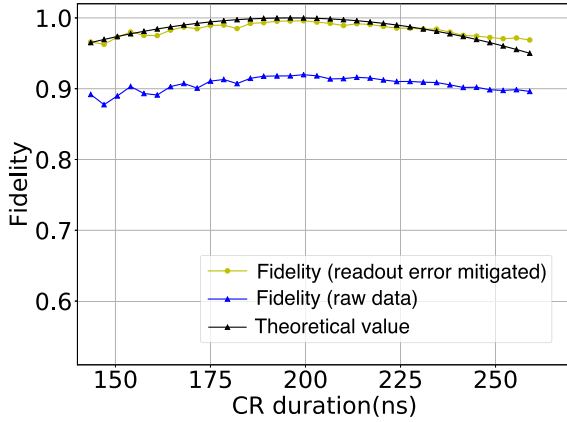
FIGURE 14. Gate fidelity of the trial cx gate (17) to the ideal cx gate as a function of the duration of CR pulse.

TABLE III Fidelity F_s for the Two Types of Toffoli Gates

Input states	TOF_{CX}	TOF_{CV}
$|000\rangle$	0.9287	0.9773
$|001\rangle$	0.9538	0.9711
$|010\rangle$	0.8886	0.9318
$|011\rangle$	0.9344	0.9450
$|100\rangle$	0.8697	0.9259
$|101\rangle$	0.9113	0.9481
$|110\rangle$	0.9167	0.9284
$|111\rangle$	0.9217	0.9089
$|+10\rangle$	0.8454	0.9462
$|1+0\rangle$	0.9046	0.9341
$|++1\rangle$	0.8860	0.9513
$|--1\rangle$	0.8603	0.9408
Average	0.9018	0.9424

Since QPT requires an excessive number of experiments, we have adopted the quantum state tomography and calculated the state fidelity [33]

$$F_s(\rho_{\exp}, \rho_{\text{ide}}) = Tr[\sqrt{\sqrt{\rho_{\exp}}\rho_{\text{ide}}\sqrt{\rho_{\exp}}}]^2 \qquad (17)$$

where ρ_{ide} denotes the ideal target density matrix and $\rho_{\exp}$ denotes the reconstructed density matrix in the experiment using the state tomography. As the input state to the Toffoli gate, we prepared 12 states listed in Table III, 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 III gives 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 given in Table 1, simply because the gate length becomes longer.

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

article, 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 two 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 (see Fig. 4) indicates that the optimal pulse length does not change in each calibration. The gate-time improvement in the circuit design, which eventually leads to the gate-fidelity improvement, was demonstrated with $\sqrt{\text{SWAP}}$, $\sqrt{\text{iSWAP}}$, and Toffoli gates. Note that the gate fidelity improvement were not totally great (0.66% improvement for the cv implementation, 0.87% for $\sqrt{\text{iSWAP}}$, and 2.14% for $\sqrt{\text{SWAP}}$), and this may be due to the presence of ZZ interactions that cannot be counteracted by the echo scheme [14], which 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, which 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.

A. GAUSSIAN SQUARE PULSE ENVELOPE

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

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

where A is the maximum amplitude and σ^2 is the variance of the Gaussian part. Note that the overall pulse length or the duration is defined as

$$\tau_d = 2\tau_r + \tau_w. \qquad (18)$$

B. OPTIMAL PULSE DURATION OF CX GATE

Fig. 1 shows the pulse schedule for implementing cx gate, where the CR pulse duration is 196 ns and accordingly the total gate time is 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 qubits 1 and 4. We

evaluated the following trial CX gate with changing the duration $\tau_d \in [144, 259]$ ns:

$$CX_{\text{trial}}(\tau_d) = [ZI]^{1/2}[ZX]^{\theta(\tau_d)}[IX]^{1/2} \qquad (19)$$

where $\theta(\tau_d) = -\tau_d/2\tau_{\text{CX}}$ with the nominal value $\tau_{\text{CX}} = 196$ 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., $\tau_d = \tau_{\text{CX}} = 196$ ns, which achieves the perfect gate fidelity.

ACKNOWLEDGMENT

REFERENCES

[1] D. C. McKay *et al.*, "Qiskit backend specifications for Open-QASM and OpenPulse experiments," 2018, *arXiv:1809.03452*, doi: 10.48550/arXiv.1809.03452.

[2] P. Gokhale, A. Javadi-Abhari, N. Earnest, Y. Shi, and F. T. Chong, "Optimized quantum compilation for near-term algorithms with OpenPulse," in *Proc. 53rd Annu. IEEE/ACM Int. Symp. Microarchitecture*, 2020, pp. 186–200, doi: 10.1109/MICRO50266.2020.00027.

[3] N. Earnest, C. Tornow, and D. J. Egger, "Pulse-efficient circuit transpilation for quantum applications on cross-resonance-based hardware," *Phys. Rev. Res.*, vol. 3, no. 4, 2021, Art. no. 043088, doi: 10.1103/PhysRevResearch.3.043088.

[4] J. P. Stenger, N. T. Bronn, D. J. Egger, and D. Pekker, "Simulating the dynamics of braiding of majorana zero modes using an IBM quantum computer," *Phys. Rev. Res.*, vol. 3, no. 3, 2021, Art. no. 033171, doi: 10.1103/PhysRevResearch.3.033171.

[5] T. Nguyen and A. McCaskey, "Enabling pulse-level programming, compilation, and execution in XACC," *IEEE Trans. Comput.*, vol. 71, no. 3, pp. 547–558, Mar. 2022, doi: 10.1109/TC.2021.3057166.

[6] A. Barenco *et al.*, "Elementary gates for quantum computation," *Phys. Rev. A*, vol. 52, no. 5, 1995, Art. no. 3457, doi: 10.1103/PhysRevA.52.3457.

[7] J.-L. Brylinski and R. Brylinski, "Universal quantum gates," in *Math. Quantum Computat.*, pp. 101–116.

[8] A. Zulehner and R. Wille, "Compiling SU(4) quantum circuits to IBM QX architectures," in *Proc. 24th Asia South Pacific Des. Automat. Conf.*, 2019, pp. 185–190, doi: 10.1145/3287624.3287704.

[9] S. Kirchhoff *et al.*, "Optimized cross-resonance gate for coupled transmon systems," *Phys. Rev. A*, vol. 97, no. 4, 2018, Art. no. 042348, doi: 10.1103/PhysRevA.97.042348.

[10] C. Rigetti and M. Devoret, "Fully microwave-tunable universal gates in superconducting qubits with linear couplings and fixed transition frequencies," *Phys. Rev. B*, vol. 81, no. 13, 2010, Art. no. 134507, doi: 10.1103/PhysRevB.81.134507.

[11] E. Magesan and J. M. Gambetta, "Effective Hamiltonian models of the cross-resonance gate," *Phys. Rev. A*, vol. 101, no. 5, 2020, Art. no. 052308, doi: 10.1103/PhysRevA.101.052308.

[12] N. Sundaresan, 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, 2020, Art. no. 020318, doi: 10.1103/PRXQuantum.1.020318.

[13] P. Krantz, M. Kjaergaard, F. Yan, T. P. Orlando, S. Gustavsson, and W. D. Oliver, "A quantum engineer's guide to superconducting qubits," *Appl. Phys. Rev.*, vol. 6, no. 2, 2019, Art. no. 021318, doi: 10.1063/1.5089550.

[14] S. Sheldon, E. Magesan, J. M. Chow, and J. M. Gambetta, "Procedure for systematically tuning up cross-talk in the cross-resonance gate," *Phys. Rev. A*, vol. 93, no. 6, 2016, Art. no. 060302, doi: 10.1103/PhysRevA.93.060302.

[15] IBM, "IBM quantum experience." [Online]. Available: https://quantum-computing.ibm.com/

[16] P. Murali, D. C. McKay, M. Martonosi, and A. Javadi-Abhari, "Software mitigation of crosstalk on noisy intermediate-scale quantum computers," in *Proc. 25th Int. Conf. Architectural Support Program. Lang. Operating Syst.*, 2020, pp. 1001–1016, doi: 10.1145/3373376.3378477.

[17] P. Mundada, G. Zhang, T. Hazard, and A. Houck, "Suppression of qubit crosstalk in a tunable coupling superconducting circuit," *Phys. Rev. Appl.*, vol. 12, no. 5, 2019, Art. no. 054023, doi: 10.1103/PhysRevApplied.12.054023.

[18] M. Sarovar, T. Proctor, K. Rudinger, K. Young, E. Nielsen, and R. Blume-Kohout, "Detecting crosstalk errors in quantum information processors," *Quantum*, vol. 4, 2020, Art. no. 321, doi: 10.22331/q-2020-09-11-321.

[19] D. P. DiVincenzo, "Quantum gates and circuits," *Proc. Roy. Soc. London. Ser. A: Mathematical, Phys. Eng. Sci.*, vol. 454, no. 1969, pp. 261–276, 1998.

[20] S. Garion *et al.*, "Experimental implementation of non-clifford interleaved randomized benchmarking with a controlled-*s* gate," *Phys. Rev. Res.*, vol. 3, Mar. 2021, Art. no. 013204, doi: 10.1103/PhysRevResearch.3.013204.

[21] T. Alexander *et al.*, "Qiskit pulse: Programming quantum computers through the cloud with pulses," *Quantum Sci. Technol.*, vol. 5, no. 4, Aug. 2020, Art. no. 044006, doi: 10.1088/2058-9565/aba404.

[22] G. Aleksandrowicz *et al.*, "Qiskit: An open-source framework for quantum computing," 2019, doi: 10.5281/zenodo.2562110.

[23] C. J. Wood, J. D. Biamonte, and D. G. Cory, "Tensor networks and graphical calculus for open quantum systems," *Quant. Inf. Comp.*, vol. 15, pp. 579–811, 2015, doi: 10.26421/QIC15.9-10-3.

[24] B. Schumacher, "Sending entanglement through noisy quantum channels," *Phys. Rev. A*, vol. 54, no. 4, 1996, Art. no. 2614, doi: 10.1103/PhysRevA.54.2614.

[25] T. F. Havel, "Robust procedures for converting among Lindblad, Kraus and matrix representations of quantum dynamical semigroups," *J. Math. Phys.*, vol. 44, no. 2, pp. 534–557, 2003, doi: 10.1063/1.1518555.

[26] J. Zhang, J. Vala, S. Sastry, and K. B. Whaley, "Geometric theory of nonlocal two-qubit operations," *Phys. Rev. A*, vol. 67, no. 4, 2003, Art. no. 042313, doi: 10.1103/PhysRevA.67.042313.

[27] G. Vidal and C. M. Dawson, "Universal quantum circuit for two-qubit transformations with three controlled-not gates," *Phys. Rev. A*, vol. 69, no. 1, 2004, Art. no. 010301, doi: 10.1103/PhysRevA.69.010301.

[28] J. Zhang, J. Vala, S. Sastry, and K. B. Whaley, "Optimal quantum circuit synthesis from controlled-unitary gates," *Phys. Rev. A*, vol. 69, no. 4, 2004, Art. no. 042309, doi: 10.1103/PhysRevA.69.042309.

[29] A. W. Cross, L. S. Bishop, J. A. Smolin, and J. M. Gambetta, "Open quantum assembly language," 2017, *arXiv:1707.03429*, doi: 10.48550/arXiv.1707.03429.

[30] V. Giovannetti, S. Lloyd, and L. Maccone, "Quantum random access memory," *Phys. Rev. Lett.*, vol. 100, no. 16, 2008, Art. no. 160501, doi: 10.1103/PhysRevLett.100.160501.

[31] L. K. Grover, "Quantum computers can search arbitrarily large databases by a single query," *Phys. Rev. Lett.*, vol. 79, no. 23, 1997, Art. no. 4709, doi: 10.1103/PhysRevLett.79.4709.

[32] J. Gwinner, M. Briański, W. Burkot, Ł. Czerwiński, and V. Hlembotskyi, "Benchmarking 16-element quantum search algorithms on superconducting quantum processors," 2020, *arXiv:2007.06539*, doi: 10.48550/arXiv.2007.06539.

[33] E. Magesan, R. Blume-Kohout, and J. Emerson, "Gate fidelity fluctuations and quantum process invariants," *Phys. Rev. A*, vol. 84, no. 1, 2011, Art. no. 012309, doi: 10.1103/PhysRevA.84.012309.

[34] A. Kandala *et al.*, "Demonstration of a high-fidelity CNOT for fixed-frequency transmons with engineered ZZ suppression," *Phys. Rev. Lett.*, vol. 127, no. 13, Sep. 2021, Art. no. 130501, doi: 10.1103/PhysRevLett.127.130501.

[35] N. Sundaresan, 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, Dec. 2020, Art. no. 020318, doi: 10.1103/PRXQuantum.1.020318.

[36] B. K. Mitchell *et al.*, "Hardware-efficient microwave-activated tunable coupling between superconducting qubits," *Phys. Rev. Lett.*, vol. 127, no. 20, 2021, Art. no. 200502, doi: 10.1103/PhysRevLett.127.200502.

[37] K. Wei *et al.*, "Quantum crosstalk cancellation for fast entangling gates and improved multi-qubit performance," 2021, *arXiv:2106.00675*, doi: 10.48550/arXiv.2106.00675.

Takahiko Satoh studied at Keio University, Tokyo, Japan, and the University of Tokyo (UT), Tokyo, Japan, in 2015. He received the B.A. degree in environmental information, in 2010, the M.S. degree in computer science, in 2012, and the Ph.D. degree in computer science from UT, in 2016.

He is currently a Project Assistant Professor with Quantum Computing Center, Keio University, Yokohama, Japan. His research interests include quantum computing and quantum networking, particularly quantum network coding, NISQ algorithm design, and quantum internet security.

Dr. Satoh is a Member of the Physical Society of Japan.

Shun Oomura received the B.S. degree in engineering and the M.S. degree in engineering, in 2019 and 2021, respectively, from Keio University, Tokyo, Japan, where he is currently studying with Sony Semiconductor Solutions Corporation.

His research interests include quantum computing and physical control of them, and he is interested in improving the NISQ performance.

Michihiko Sugawara received the undergraduate degree in chemistry and the M.S. and Ph.D. degrees from Tohoku University, Sendai, Japan, in 1988, 1990, and 1993, respectively.

In 1992, he was a JSPS Postdoctoral Fellow with Tohoku University. In June 1994, he was a Postdoctoral Fellow with Arizona State University, then he moved to Institute of Atomic and Molecular Science, Taiwan, in November 1994. In 1995, he was a Research Associate and Assistant Professor with the Department of Chemistry, Keio University. He is currently a Project Associate Professor with Quantum Computing Center, Keio University, Yokohama, Japan. His research interests include coherent control of quantum dynamics, theoretical design of control field, numerical calculation of vibrational eigenstates of various molecules, and wave packet dynamics.

Naoki Yamamoto received the B.S. degree in engineering and the M.S. and Ph.D. degrees in information physics and computing from The University of Tokyo, Tokyo, Japan, in 1999, 2001, and 2004, respectively.

From 2003 to 2007, he was a Research Fellow with the Japan Society for the Promotion of Science. From 2004 to 2007, he was a Postdoctoral Fellow with the California Institute of Technology. From 2007 to 2008, he was a Postdoctoral Fellow with Australian National University. He is currently a Professor with the Department of Applied Physics and Physico-Informatics, Keio University, Yokohama, Japan, where he is also the Chair of Quantum Computing Center. His research focuses on quantum computation and control.