Demonstration of quantum volume 64 on a superconducting quantum computing system

Petar Jurcevic*, Ali Javadi-Abhari†, Lev S. Bishop*, Isaac Lauer*, Daniela F. Bogorin*, Markus Brink*, Lauren Capelluto*, Oktay Günlük*, Toshinari Itoko†, Naoki Kanazawa†, Abhinav Kandala*, George A. Keefe*, Kevin Krsulich*, William Landers*, Eric P. Lewandowski*, Douglas T. McClure*, Giacomo Nannicini*, Adinath Narasgond*, Hasan M. Nayfeh*, Emily Pritchett*, Mary Beth Rothe†, Srikanth Srinivasan*, Neeraja Sundaresan*, Cindy Wang*, Ken X. Wei*, Christopher J. Wood*, Jeng-Bang Yau*, Eric J. Zhang*, Oliver E. Dial†, Jerry M. Chow*, Jay M. Gambetta*

*IBM Quantum, IBM T.J. Watson Research Center, Yorktown Heights, NY 10598, USA
†IBM Quantum, IBM Research Tokyo, 19-21 Nihonbashi Hakozaki-cho, Chuo-ku, Tokyo, 103-8510, Japan

Abstract—We improve the quality of quantum circuits on superconducting quantum computing systems, as measured by the quantum volume, with a combination of dynamical decoupling, compiler optimizations, shorter two-qubit gates, and excited state promoted readout. This result shows that the path to larger quantum volume systems requires the simultaneous increase of coherence, control gate fidelities, measurement fidelities, and smarter software which takes into account hardware details, thereby demonstrating the need to continue to co-design the software and hardware stack for the foreseeable future.

I. INTRODUCTION

Quantum computing is a new kind of computing, using the same physical rules that atoms follow in order to manipulate information. At this fundamental level, quantum computers execute quantum circuits — like a classical computer’s logical circuits — but now using the physical phenomena of superposition, entanglement, and interference to implement mathematical calculations that are out of reach for even our most advanced supercomputers.

As we progress towards machines capable of implementing circuits with a quantum advantage, meaning certain information processing tasks can be performed more efficiently or cost effectively than with classical circuits, quantum volume (QV) [1] serves as a holistic benchmark for quantum systems indicating the size of the quantum circuits that can be run on them. Sensitive to improvements in many aspects of device performance, quantum volume includes gate errors, measurement errors, the quality of the circuit compiler, and spectator errors. In Ref. [1] and later in Ref. [2], QV16 was measured on ibmq_montreal and a Honeywell quantum system, respectively. In Ref. [3] QV8 was measured for the Rigetti Aspen-4 quantum system. We recently increased ibmq_johannesburg to QV32 [4] by improving our physical understanding of the two-qubit cross-resonance gate and using rotary echo pulses to reduce gate and spectator errors. Finally in unpublished work Honeywell has claimed to measure QV64 [5].

Here we demonstrate an increase in the quantum volume of an IBM quantum system by improving the Qiskit compiler [6], implementing excited state promoted (ESP) readout, shorter two-qubit gates, and adding dynamic decoupling to the idle qubits. These last three demonstrate the need for timing and pulse control in cloud quantum systems [7]. While individually not one of these improvements is enough to allow ibmq_montreal to reach QV64, when combined we achieve QV64 with a heavy output probability (HOP) of $0.701 \pm 0.031 ( > 2/3 \pm 2\sigma )$ with a confidence interval of $98.744\% ( z = 2.25 )$, see Fig. 2(a). The quantum volume test requires exceeding 2/3 HOP by a $97.725\% ( z = 2 )$ confidence interval.

In Section II we give an overview of the ibmq_montreal device, which is a 27-qubit IBM Quantum Falcon processor; in Section III we discuss the improvements to the compiler; in Section IV we discuss the dynamical decoupling protocol; in Section V we discuss the faster implementation of the direct CNOT gate which extends the improved pulse control of [4]; in Section VI we discuss the improvement in measurement fidelity by using a control pulse to promote the excited state to a higher level before measurement [8]. Finally in Section VII we conclude the paper.

II. QUANTUM SYSTEM - ibmq_montreal

The device studied in this work is from the recent series of IBM Quantum Falcon processors, which consist of 27 qubits arranged in a lattice designed for a distance-3 hybrid Bacon-Shor-surface code [9]. A photo of this processor is shown in Fig. 1(a), and a schematic of its connectivity is shown in Fig. 1(b). A high connectivity layout, such as ‘all-to-all’, is preferable for random quantum circuits (such as QV circuits) in order to minimize the average qubit-qubit distance; however, additional edges in the connectivity increase the chance of frequency collision, cross-talk, and spectator errors. The IBM Quantum Falcon processor is a compromise, preserving a connectivity efficient for a logical qubit while simultaneously reducing detrimental effects of collisions and cross-talk without excessive insertion of SWAP gates to emulate ‘all-to-all’ connectivity. In these and related systems, by using the techniques described in [4], we have
measured a QV of 32 on the last 7 deployed systems [10] demonstrating the reliability of this architecture.

In this paper we achieve QV64 on ibmq_montreal, which is one of the latest deployed IBM Quantum Falcon processors. The quantum volume circuits were run on a line of six qubits, Q16-Q19-Q22-Q25-Q24-Q23 (orange shaded qubits in Fig. 1(b)). Individual qubit properties are shown in Fig 2(b) with the following average values: $T_1 = 113\,\mu s$, $T_2 = 122\,\mu s$, error per single-qubit gate $3.8 \times 10^{-4}$, error per two-qubit gate $6.4 \times 10^{-3}$, and single-qubit readout assignment error $6.0 \times 10^{-3}$. Gate errors were measured with simultaneous single-qubit and individual two-qubit randomized benchmarking [11].

The qubits are fixed-frequency transmons with frequencies $\approx 5\,\text{GHz}$. Single-qubit gates are driven resonantly with a microwave pulse of duration $\tau_{\text{eq}} = 21.33\,\text{ns}$. A DRAG pulse envelope [12] corrects $\sigma_z$-errors and signal dispersion due to wiring. Two-qubit gates are based on a cross-resonance scheme [13], [14], [15] with a target rotary pulse [4] and an additional offset pulse-shape on the target for implementing a direct (echoless) CNOT as described later. Two-qubit gate lengths are $\tau_{\text{eq}} = 199 – 309\,\text{ns}$.

III. COMPILER

Circuit compilation is a substantial part of quantum computation. Here we report improvements in the state-of-the-art Qiskit compiler to achieve reductions in the number of gates which results in circuits with shorter depths. The compilation of a quantum volume circuit for a superconducting processor can be roughly broken down into two stages. The first stage is to map the circuit to the hardware’s qubit connectivity constraints. At the conclusion of this step, each circuit will consist of a series of SU(4) gates on the available links, as well as the overhead of routing qubit information on the physical fabric, usually in the form of SWAPs. The second step consists of local expansions to the native gates of the hardware and optimizations. We introduce new compiler passes to improve both stages, and leverage existing passes in the Qiskit compiler throughout to achieve further reductions where possible: approximate synthesis, commutative cancellation, and peephole optimization of single-qubit and two-qubit chains of gates.

It is worth noting that the particular passes reported here have general utility beyond QV. Qubit mapping and routing is ubiquitous in compiling for limited-connectivity architectures, and SU(4) synthesis has broad use in peephole optimization of sequential two-qubit gates.

a) Qubit layout and routing via Binary Integer Programming: We formulate qubit layout and routing as a binary integer programming (BIP) problem, which we are able to solve to optimality. We choose as the cost function, $C$, the effective fidelity, modeled as the product of the fidelity of all the implemented gates:

$$C = K^d \prod_{j \in G} F^\text{best}_j \prod_{j \in S} \prod_{i=0}^{3} F^\beta_i,$$

(1)

where $K$ is a factor penalizing circuits with high depth $d$; $G [G]$ are the set of gates that are mapped directly [mapped with mirroring – combining SWAP with a gate]; and $S$ is the set of added SWAP gates. Here, $F_\beta$ is the gate fidelity of the available entangling gate (which must be applied 3 times to implement SWAP). $F^\text{best}_j [F^\beta_i]$ is the modeled fidelity of the best approximation to the target unitary making $i = 0, \ldots, 3$ uses of the entangling gate

$$F^\text{best}_j = \max \{ F^\text{avg}_{i,j}(F_\beta) \},$$

(2)

$$F^\beta_i = \max \{ F^\text{avg}_{i,j}(F_\beta) \},$$

(3)

and $F^\text{avg}_{i,j}$ is the average gate fidelity due to approximating the $j$-th gate with $i$ uses of the entangling gate [1, Appendix B].

The freedom to implement either a gate or its mirror

![Image](image-url)
allows elimination of many explicit SWAP gates, and by restricting the number of candidate SWAP insertion sites we are able to reduce the size of the BIP problem such that it can be solved to optimality in around one second per circuit, using optimization software such as CPLEX [16]. Figure 3 shows the performance of this BIP pass in comparison to the state-of-the art SABRE algorithm [17] available in Qiskit, showing substantial improvement in both the mean and maximum number of uses of the entangling gate.

b) Pulse-efficient SU(4) decomposition: The `ibmq_montreal` device has the following native gate set for achieving universal quantum computation: Ctrl-X (CX), Sqrt-X (SX) and Phase($\theta$). The CX gate itself can be implemented directly or be created using an Echo Cross-Resonance (ECR) pulse[18] (c.f. Section V). The Phase gate can be achieved with zero time and error [19]. We refer to any gate that is one pulse (i.e. equivalent to an SX by a pre-/post-phase) as a single-qubit (SQ) gate (e.g. Hadamard). A generic single-qubit operation (U) can be achieved with at most 2 SQ pulses.

Given the CX, SQ or ECR, SQ set of native pulses, we aim to minimize them during the expansion of each SU(4) and SWAP. A second goal is to expand them in a way that creates further opportunities for optimization. It is known that any SU(4) can be implemented using at most 3 CX gates [20], and 2 CX gates suffice for many useful approximations (e.g. at 99% fidelity) [1] (cf. Figure 4(a)). ECR is locally equivalent to CX, so it has the same requirements. While the question of “optimal” SU(4) decomposition has been extensively studied, the optimality criteria has usually been the number of 2-qubit gates [20], [21]. To extract ultimate performance, we are also interested in minimizing the total number of pulses and the duration.

Our approach is based on three strategies:

1. Circuit simplification to reduce redundant pulses: starting from a Qiskit synthesis of an arbitrary SU(4), we apply repeated circuit identities to the result to reduce its cost. This gives us a constructive SU(4) decomposition, depicted in Figure 4(b), which is optimal in the number of pulses (by a simple parameter counting argument). This decomposition has another advantage, in that 8 out of 10 single-qubit pulses are placed on the outside of the structure. Given that 2 SQ pulses suffice for any aggregate single-qubit operation, this creates an opportunity for merging with preceding and following layers of SU(4) in the circuit. One surprising consequence of this decomposition is that for the special case of a SWAP operation, the decomposition is locally less efficient than a textbook expansion; however globally it is more efficient as it creates more opportunities for cancellation (Figure 4(c)). We arrive at similar pulse-efficient decompositions targeting the ECR gate, and also for approximated SU(4)s that use 2 CX instead of 3 (omitted for brevity).

2. Decomposition in the natural gate direction: While the device software is easily capable of implementing a CX gate in both directions, in reality there is a preferred gate direction in terms of speed and error on the hardware. The other direction is achieved by local pre- and post-rotations.

The same is true for ECR gates. By querying the device for its natural direction, we can expand each SU(4) and SWAP in the correct direction in the compiler, avoiding further cost down the road. To synthesize a general SU(4) when the logical and physical directions are mismatched, we employ a trick of double mirroring (adding SWAPs before and after the SU(4)). The doubly-mirrored SU(4) implements a different operator, where the middle two rows and middle two columns are swapped. We perform a pulse-efficient synthesis on the doubly-mirrored operator, but apply it in the circuit with the reverse order of qubits. This will ensure the original operator is implemented, but also now with the correct physical gate direction (Figure 4). Double-mirroring creates a locally equivalent gate, so any approximation to the original SU(4) still holds with the same error bounds.

3. Decomposition to native gate: If a direct CX is not available, we compile to the fundamental two-qubit interaction available. In the case of ECR, this saves us the extra single-qubit pulses involved in creating a CX. This demonstrates the benefit of removing simplifying abstraction barriers in the exposed gate set to gain efficiency in compiling [22], [23], [24].

IV. DYNAMICAL DECOUPLING

When quantum circuits are mapped to physical hardware, not all physical gates can be performed simultaneously. Gate execution-times can vary significantly, not only between single- and two-qubit gates, but also between individual qubits and qubit-pairs. In addition, architecture-specific gate schemes and connectivity determine which and how many gates can be executed in parallel.

Fig. 3. Comparison of QV64 circuits transpiled to a line connectivity with (a), (b) the state-of-the-art Qiskit compiler and (c), (d) an improved transpilation method based on BIP and additional gate cancellations, see text. (a) and (c) show the same random example circuit, mapped with “SABRE” and mapped with BIP, respectively. The purple boxes represent the random SU(4) and SWAPs are indicated in blue. (b) and (d) show statistics of 2000 circuits using both methods. CX count improvements are due to improved mapping, and S-Q (single-qubit) count improvements the result of pulse-efficient compilation. Both contribute to shorter durations. We assume basis gate fidelity $F_i = 0.99$ for the approximate SU(4) expansion in all cases. If the native gate is ECR (rather than direct-CX), we get additional 7% reduction in mean duration by targeting the native gate and absorbing local pre-rotations.
be a single Hahn echo-pulse [28], refocusing the low-frequency noise spectrum acting on a unitary. Various decoupling sequences have been proposed [29], [30], [31], some with self-correcting properties [32], [33], others with non-equidistant temporal spacing [34], and hybrids combining both [35], [36], [37], in order to optimize the effective filter function. Recently, dynamical decoupling has been shown to improve single-qubit states and an entangled two-qubit state on a Rigetti and IBM quantum computer [38].

For the successful QV64 measurement presented here, we used the sequence $\tau^{i,q}/2 - X_p - \tau^{i,q} - X_m - \tau^{i,q}/2$, with delays $\tau^{i,q} = (T_{idle} - 2 * T_{X,p/m})/2$, where $T_{idle}$ is the $i$th idle length on qubit $q$, and $T_{X,p/m}$ is the duration of one echo pulse with $X_{p,m}$ being a $\pi$-pulse around x-axis with positive/negative sense of rotation.

Figure 6 shows a comparison of identical QV-circuits run with (DD) and without (Idle) dynamical decoupling. Dynamical decoupling with $X_p - X_m$ sequences improves 72.8% of all circuits in this run, i.e. $\text{HOP}_{\text{DD}} > \text{HOP}_{\text{Idle}}$, with an average HOP increase of 0.0178. The interplay between various DD sequences and random circuits, such as QV circuits, is an open research focus.

V. DIRECT CX GATE

Even with state-of-the-art compiling, QV64 circuits consist of a total of 57 two-qubit gates and 146 single-qubit gates on average. Any improvement in gate speed can significantly reduce the circuit duration compared to the coherence times. However, the optimal gate speed for running a circuit is in general not the speed that maximizes the fidelity of the individual gates. In particular, qubits experience idle times in a multi-qubit circuit (see Section IV), and the fidelity of the identity operation during these idle times is not captured in the single-qubit or two-qubit randomized benchmarking fidelities often used to characterize quantum systems. Finding the optimal trade-off between individual gate fidelity and circuit fidelity is currently open research, in addition to characterizing which errors are enhanced by driving gates faster. Here we focus on techniques to reduce two-qubit gate durations, but note that small increases in the speed of either single- or two-qubit gates can significantly impact the performance of QV64 circuits.

As mentioned in Section III, an immediate way to “speed up” two-qubit gates is to incorporate into the circuit compilation any pre-/post-single-qubit rotations needed to get from the native ECR gate to a CX or CNOT. We compare the standard echoed cross-resonance gate ECR CX, shown at the top of Fig. 7(a), to an ECR gate in which single qubit rotations are compiled separately, reducing the two-qubit gate duration to only the entangling portion of the gate. The errors of ECR CX and ECR, measured by two-qubit randomized benchmarking, are shown in Fig. 7(b) as a function of the two-qubit gate duration.

Two-qubit gates can be further sped up by finding high-fidelity alternatives to the echo pulse sequence, effectively...
removing another single-qubit gate from the total two-qubit gate duration. We compare an example of a “direct” echo-free CX pulse sequence, shown at the bottom of Fig. 7(a), to ECR and ECR CX. This sequence demonstrates an improvement over previous direct CNOT attempts [15] by leveraging our understanding of target rotary pulsing [4]. The resonant drive of the target is implemented as the sum of two parts, an active cancellation tone and a target rotary tone that are symmetric and antisymmetric over the CR pulse, respectively. The active cancellation tone cancels IX terms in the native CR Hamiltonian and any IY terms due to classical crosstalk, while the target rotary pulse can be used to reduce unwanted ZZ and ZY.

The impact of reducing the total gate duration is clearly evidenced by a reduction of two-qubit gate error, as shown in Fig. 7(b). All gate sequences – ECR CX, ECR, and direct CX – experience a sudden loss of fidelity with increasing pulse amplitude, but the direct CX experiences this break down at a much shorter gate time. We note that reducing the gate duration below that which minimizes two-gate error as measured by randomized benchmarking can increase the HOP of a QV circuit, showing the importance of balancing circuit optimization with gate optimization. For our successful demonstration of QV64 we used a direct CX gate duration of 199 ns, which is shorter than that which minimizes the two-qubit gate error.

VI. STATE INITIALIZATION AND READOUT

Qubit-state initialization to a fiducial simple state and qubit-specific measurement are two out of five (plus two) necessary DiVincenzo criteria for quantum computation [39]. While certain metrics are designed specifically to be insensitive to “state preparation and measurement” (SPAM) errors, e.g. randomized benchmarking [40], [41] and gate set tomography [42], [43], quantum volume was developed as a holistic system measure and hence is sensitive to SPAM-errors.

In its simplest form, qubit initialization or reset is done passively by waiting multiple $T_1$ relaxation times before every new computational cycle in order to let the qubit thermalize with its surrounding bath. With ever-increasing coherence times, thermal relaxation protocols impractically limit the computational repetition rate. Various active reset schemes have been proposed and experimentally demonstrated [44], [45]. IBM Quantum systems implement a similar unconditional reset scheme [46]. By measuring the readout matrix (Fig. 8(a)) we can infer a reset error of
We reached a quantum volume of 64 through a combination of four factors: improvements of the Qiskit compiler, refinements to the two-qubit gate and its calibration, addition of dynamical decoupling to mitigate noise affecting idle qubits, and introduction of excited state promoted readout. The last three techniques were developed by having lower-in-the-stack access to how the pulses and gates that comprise quantum circuits are defined before being sent to control the qubits. Furthermore, we note that optimizing the fidelity of quantum circuits is not equivalent to optimizing the gates and confirms the need for circuit benchmarks like quantum volume. This type of hardware-aware approach to make improvements to circuit performance is a hallmark of the current era of noisy quantum systems which we expect to continue until we can achieve error rates in the range of $10^{-4}$.

ACKNOWLEDGEMENT

We thank all those that contributed to the hardware system delivery and implementation, including the IBM Microelectronics Research Laboratory and Central Scientific Services teams as well as the worldwide team who designed, built and tested the custom control electronics. We further acknowledge all the work from the broader IBM Quantum team who helped this effort across the full stack.

REFERENCES

[1] A.W. Cross, L.S. Bishop, S. Sheldon, P.D. Nation, and J.M. Gambetta. Validating quantum computers using randomized model circuits. Phys. Rev. A, 100:032328, Sep 2019.
[2] J. M. Pino, J. M. Dreiling, C. Figgatt, J. P. Gaebler, S. A. Moses, M. S. Allman, M. Baldwin, C. H. and Foss-Feig, D. Hayes, K. Mayer, C. Ryan-Anderson, and B. N. Neyenhuis. Demonstration of the qcd trapped-ion quantum computer architecture. arXiv:2003.01293, 2020.
[3] Peter J Karalekas, Nikolas A Tzekz, Eric C Peterson, Colm A Ryan, Marcus P da Silva, and Robert S Smith. A quantum-classical cloud platform optimized for variational hybrid algorithms. Quantum Science and Technology, 5(2):024003, apr 2020.
[4] Neereja Sundaresan, Isaac Lauer, Emily Pritchett, Easwar Magesan, Petar Jurcevic, and Jay M. Gambetta. Reducing unitary and spectator errors in cross resonance with optimized rotary echoes. arXiv:2007.02925, 2020.
[5] Bob Yirka. Honeywell claims to have built the highest-performing quantum computer available. https://phys.org/news/2020-06-honeywell-built-highest-performing-quantum.html, 2020.
[6] Qiskit: An open-source framework for quantum computing. https://qiskit.org/, 2020.
[7] David C McKay, Thomas Alexander, Luciano Bello, Michael J Biercuk, Lev Bishop, Jiayin Chen, Jerry M Chow, Antonio D Corcoles, Daniel Egger, Stefan Filipp, et al. Qiskit backend specifications for openqasm and openpulse experiments. arXiv preprint arXiv:1809.03452, 2018.
[8] F. Mallet, F. Ong, A. Palacios-Laloy, F. Nguyen, P. Bertet, D. Vion, and D. Esteve. Single-shot qubit readout in circuit quantum electrodynamics. Nature Phys. 5:791−795, 2009.
[9] Christopher Chamberland, Guanyu Zhu, Theodore J. Yoder, Jared B. Hertzberg, and Andrew W. Cross. Topological and subsystem codes on low-degree graphs with flag qubits. Phys. Rev. A, 100:032328, Sep 2019.
[10] IBM Quantum Experience. https://quantum-computing.ibm.com/, 2020.
[11] J.M. Gambetta, A. D. Corcoles, S. T. Merkel, B. R. Johnson, J.A. Smolin, J.M. Chow, C.A. Ryan, C. Rigetti, S. Poletto, T.A. Ohki, M.B. Ketchen, and M. Steffen. Characterization of addressability by simultaneous randomized benchmarking. Phys. Rev. Lett., 109:240504, Dec 2012.
[12] F. Motzoi, J. M. Gambetta, P. Rebentrost, and F. K. Wilhelm. Simple pulses for elimination of leakage in weakly nonlinear qubits. Phys. Rev. Lett., 113:110501, Sep 2009.
[13] G.S. Paraoanu. Microwave-induced coupling of superconducting qubits. Phys. Rev. B, 74:140504, Oct 2006.
