Demonstration of Density Matrix Exponentiation Using a Superconducting Quantum Processor

M. Kjaergaard,[1,*,†] M. E. Schwartz,[2,†] A. Greene,[1,3] G. O. Samach,[1,2,3] A. Bengtsson,[1,4] M. O'Keeffe,[2] C. M. McNally,[1,3] J. Braumüller,[1] D. K. Kim,[2] P. Krantz,[1,‡] M. Marvian,[1,5] A. Melville,[2] B. M. Niedzielski,[2] Y. Sung,[1,3] R. Winik,[1] J. Yoder,[2] D. Rosenberg,[2] K. Obenland,[2] S. Lloyd,[1,5] T. P. Orlando,[1,3] I. Marvian,[6] S. Gustavsson,[1] and W. D. Oliver[1,2,7,3]

[1]*Research Laboratory of Electronics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA*

[2]*MIT Lincoln Laboratory, Lexington, Massachusetts 02421, USA*

[3]*Department of Electrical Engineering & Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA*

[4]*Microtechnology and Nanoscience, Chalmers University of Technology, Göteborg 412 96, Sweden*

[5]*Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA*

[6]*Departments of Physics & Electrical and Computer Engineering, Duke University, Durham, North Carolina 27708, USA*

[7]*Department of Physics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA*

(Received 1 March 2021; revised 1 October 2021; accepted 4 November 2021; published 7 January 2022)

Density matrix exponentiation (DME) is a general technique for using a quantum state ρ to enact the quantum operation $e^{-i\rho\theta}$ on a target system. It was first proposed in the context of quantum machine learning, but has since been shown to have broad applications in quantum metrology and computation. No experimental demonstration of DME has been performed thus far due to its demanding circuit depths and the need to efficiently generate multiple identical copies of ρ during the finite lifetime of the target system. In this work, we describe the first demonstration of the DME algorithm, which we accomplish using a superconducting quantum processor. Our demonstration relies on a 99.7% fidelity controlled-phase gate implemented using two tunable superconducting transmon qubits. We achieve a fidelity surpassing 90% at circuit depths exceeding 70 when comparing the output of the circuit executed on our quantum processor to a simulation assuming perfect operations and measurements.

DOI: 10.1103/PhysRevX.12.011005 Subject Areas: Quantum Information

I. INTRODUCTION

Density matrix exponentiation (DME) maps the density matrix of a first system (ρ) into the operator $U_{\mathrm{DME}} = e^{-i\rho\theta}$, which acts on a second system (σ). The first system ρ may be any single- or multiqubit density matrix, including mixed and entangled states, and need not be known to the experimenter; we refer to this state as an "instruction." We refer to the second system σ as the "data." Because of the no-cloning theorem [1], it is impossible to perfectly implement the mapping $\rho \mapsto U_{\mathrm{DME}}$ using just one copy of an unknown ρ. However, in Ref. [2], Lloyd *et al.* realized that by supplying N copies of ρ, each of which implements the operation $e^{-i\rho\theta/N}$, one can implement U_{DME} with exponentially fewer copies than would be required for tomographic reconstruction of ρ.

The DME algorithm has two remarkable properties: The circuit depth required to implement DME to a desired precision ϵ scales as $\mathcal{O}(\log(d)\theta^2/\epsilon)$ (where d is the dimension of ρ), and the number of copies of ρ required scales only as $\mathcal{O}(\theta^2/\epsilon)$, *independent* of d [2–4]. In comparison to the number of copies of ρ required for tomographic reconstruction to accuracy ϵ, this represents an exponential reduction in sample number [5]. The time- and sample-complexity speed-ups afforded by DME make it a key enabling protocol for a class of efficient algorithms relating to both quantum computation (using ρ to manipulate σ) and quantum metrology (using σ to study ρ).

The power of DME can be understood intuitively by a comparison of its function to that of instructions in classical

*mortenk@mit.edu

†These authors contributed equally to this work.

‡Present address: Microtechnology and Nanoscience, Chalmers University of Technology, Göteborg, Sweden.

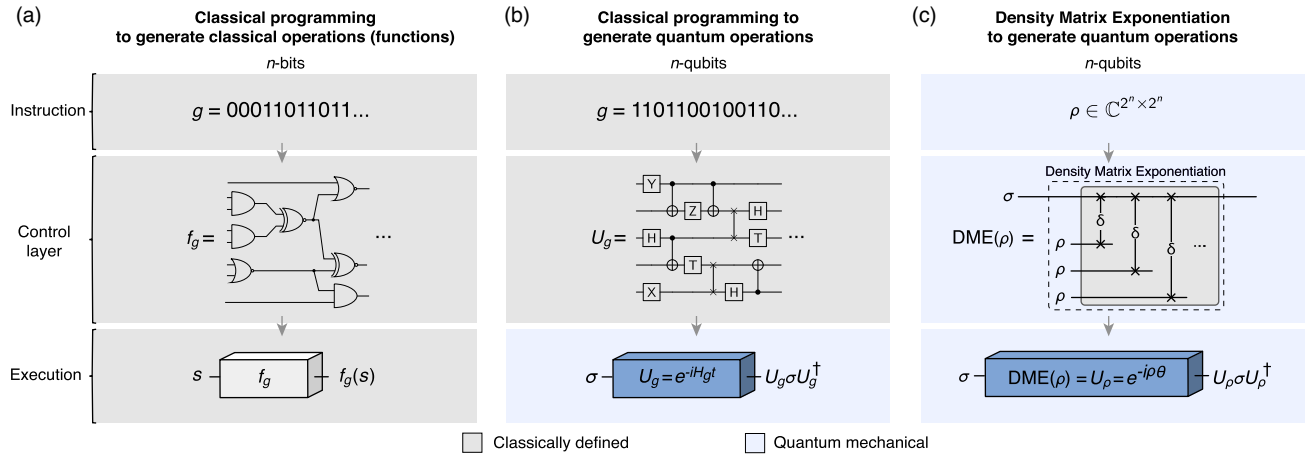


FIG. 1. Schematic representations of computing architectures. (a) Classical programming to generate classical operations (functions). Instructions are defined by a classical bit string g that uniquely determines a Boolean logic function f_g. The control layer executes the resulting circuit on data bits s to produce the output $f_g(s)$. (b) Classical programming to generate quantum operations. Instructions are defined by a classical bit string g that uniquely determines a unitary operation U_g. The gate sequence is applied to the quantum hardware (data qubits σ) to execute the unitary operation $U = \exp(-iH_g t)$, where H_g is the quantum circuit Hamiltonian, to produce the output $U_g \sigma U_g^\dagger$. If any components of the instruction string depend on information from previous steps in the algorithm, that information must be tomographically extracted from σ and then classically recompiled. (c) DME to generate quantum operations. In DME, the quantum operation is stored in the quantum instruction state ρ. Classical hardware generates N partial SWAP operations, a co-rotation enacted by a Heisenberg interaction between qubits σ and ρ (see text) along the same axis, over a small, classically chosen rotation angle $\delta = \theta/N$. These partial SWAP operations implement the unitary operation $U = e^{-i\rho\theta}$ without first requiring classical compilation of the information contained in ρ.

and quantum computing (Fig. 1). In a classical computer, both the instructions and the data are defined using classical bits and operations [Fig. 1(a)]. Information from one step in an algorithm can be easily passed forward from one step to the next, because the data and instructions are formally similar. In a typical quantum-computing application, the data are quantum, but the operations are determined using classical compilers [Fig. 1(b)]. This is efficient for applications that require no feed forward of information but can be prohibitively costly if the algorithm requires extracting more than classical information from the data. Feeding quantum data forward first requires that it be tomographically reconstructed, an exponentially costly process [5], so that it can be compiled into a new set of equivalent classical instructions. With DME, however, a fixed scaffolding of classically defined gates is used to implement an operation that is already encoded directly in a quantum instruction [Fig. 1(c)]. In this case, no tomographic reconstruction is needed to use this information in further processing steps, which can create significant savings for certain applications [3,4,6].

A number of known algorithms take advantage of the efficiency enabled by DME:

(i) Private quantum software execution. In this application, the action of an unknown (private) unitary U on an arbitrary quantum state may be efficiently emulated using a relatively small set of provided input-output relations $\{\rho_{\text{in}}\} \overset{U}{\mapsto} \{\rho_{\text{out}}\}$. Private quantum software execution utilizes a controlled-DME process for which pairs of ρ_{in} and ρ_{out} serve as instructions in successive steps. If provided with enough copies of ρ_{in} and ρ_{out}, an intercepting party could, in principle, perform tomographic reconstruction of all inputs and outputs and use this information to reconstruct the process that U implements. However, the number of copies of the states required for DME is far fewer than the number of copies required to perform full tomography: The "owner" of U can provide enough copies of the states to enable DME but not enough copies to perform tomography. In this way, the action of U can be emulated without compromising the privacy of U itself [3].

(ii) Hamiltonian simulation. DME has also been proposed as a subroutine in a variation of Hamiltonian simulation. Standard Hamiltonian or quantum simulation, for quantum chemistry, for example, is concerned with the complexity of implementing the operator e^{-iHt}. By presenting a Hamiltonian in the form of a density matrix, Kimmel *et al.* showed that DME enables a sample-optimal strategy for performing quantum simulation [4].

(iii) Quantum principal component analysis. By using DME to turn a state into an operator, quantum phase estimation can be implemented on that operator to extract the dominant eigenvalues and eigenvectors of ρ. This requires exponentially fewer copies of ρ than any other known methods and is the original application envisioned for DME [2,4,7].

(iv) Entanglement spectra. Pichler *et al.* [8] showed that if one is presented with a large entangled state—too large for standard tomographic protocols to practically reconstruct—its entanglement spectrum can still be revealed efficiently by a protocol using DME as the key subroutine. This leads to a novel and resource-efficient form of reduced-complexity benchmarking.

(v) Quantum machine learning. DME has been shown to play a role in providing quantum speed-up for quantum semidefinite programming [6] and to efficiently construct quantum support vector machines [9], albeit with demanding requirements on the input states.

In this work, we demonstrate the first experimental implementation of the DME algorithm using a superconducting quantum processor [10]. Specifically, we apply DME to a system comprising two superconducting qubits: a data qubit prepared in state σ and an instruction qubit prepared in state ρ. In Sec. II, we review the DME protocol in its original construction, which requires N physical copies of the instruction state. In Sec. III, we describe a technique we develop to efficiently refresh the instruction state using only a single physical instruction qubit [11]. This is a significant technical simplification that makes this demonstration possible. In Sec. IV, we demonstrate that our DME implementation indeed performs a rotation on the data qubit that is defined by the state of the instruction qubit. Finally, in Sec. V we explore the dynamics of the protocol as a function of N and benchmark its performance using state and process tomography.

II. THE DME PROTOCOL

DME implements the unitary operation $U_{\mathrm{DME}} = e^{-i\rho\theta}$ on quantum data according to an instruction ρ and a classically chosen angle θ. In concrete terms, if the data and instructions are single-qubit pure states σ and ρ, DME rotates σ by an angle θ about an axis defined by the Bloch

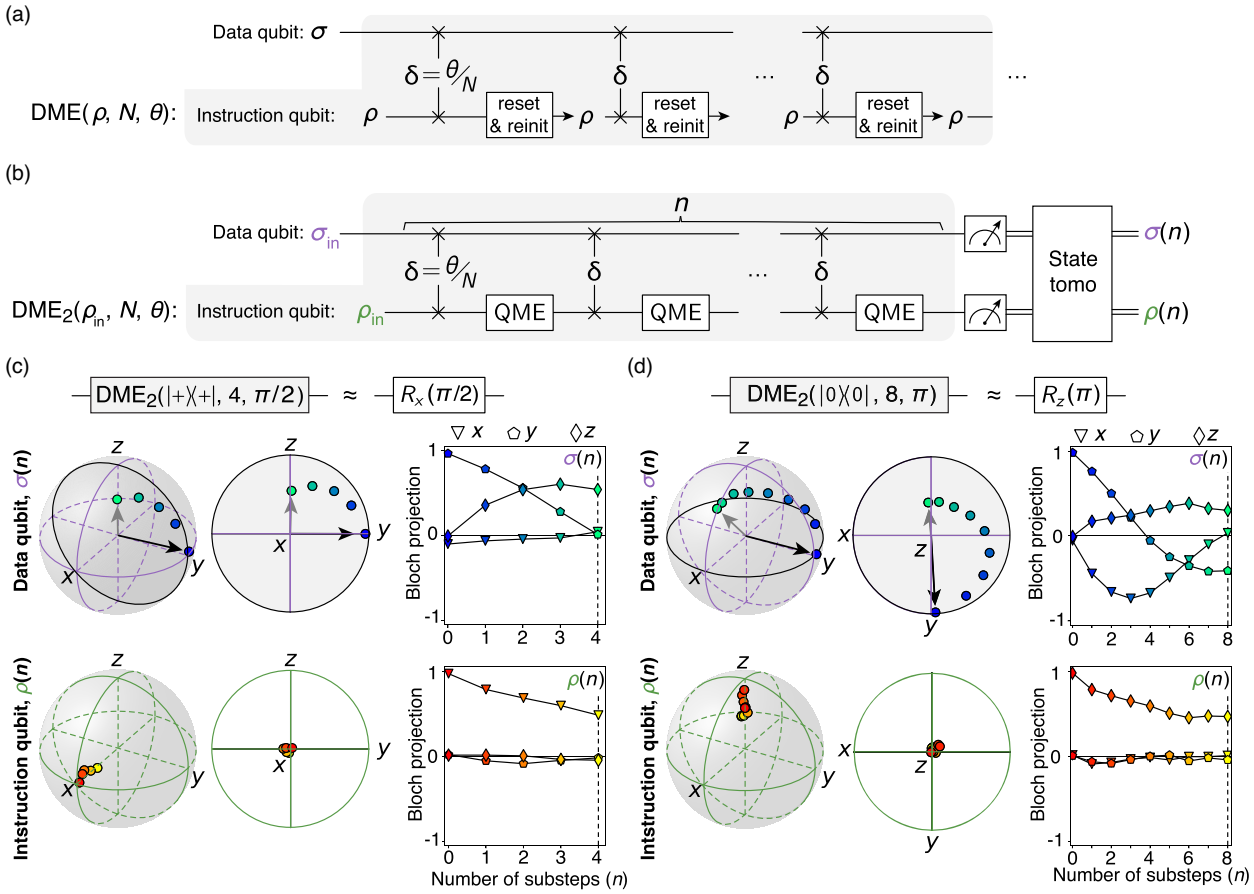


FIG. 2. Demonstration of the two-qubit effective density matrix exponentiation algorithm. (a) DME algorithm using active reset and reinitialization to reprepare the instruction state ρ after each δSWAP operation. (b) Implementation of DME_2 in which QME is used to approximately reinitialize the instruction qubit to ρ_{in} without active reset and repreparation (see text for details). The substep parameter n is stepped from 0 to N. In the experiment, we perform n rounds of $\delta\mathrm{SWAP} + \mathrm{QME}$, measure the two-qubit density matrix, and trace over each subsystem to extract the individual data and instruction qubit density matrices [$\sigma(n)$ and $\rho(n)$, respectively]. (c) Substeps of $\mathrm{DME}_2(|+\rangle\langle+|, 4, \pi/2)$ corresponding to $R_X(\pi/2)$ on the target qubit at the final step ($n = N$). Black lines are guides to the eye. (d) Substeps of $\mathrm{DME}_2(|0\rangle\langle0|, 8, \pi)$ corresponding to $R_Z(\pi)$ on the target qubit at $n = N$.

vector of ρ. More generally, σ and ρ are multiqubit states, and they need not be pure states.

The protocol that implements DME partitions U_{DME} into a sequence of N steps [Fig. 2(a)], each comprising a "partial SWAP" operation $\delta\mathrm{SWAP} \equiv e^{-i\mathrm{SWAP}\delta}$ [12] that is applied between σ and ρ [2]. The SWAP operation between two qubits can be enacted by a Heisenberg interaction,

$$\mathrm{SWAP}_{ij} = \frac{1}{2}\left(\mathbb{1}^{(i)}\mathbb{1}^{(j)} + \hat{\sigma}_x^{(i)}\hat{\sigma}_x^{(j)} + \hat{\sigma}_y^{(i)}\hat{\sigma}_y^{(j)} + \hat{\sigma}_z^{(i)}\hat{\sigma}_z^{(j)}\right).$$

The protocol relies on the relation

$$\begin{aligned}\mathrm{Tr}_\rho[e^{-i\mathrm{SWAP}\delta}\sigma \otimes \rho e^{i\mathrm{SWAP}\delta}] &= \sigma - i\delta[\rho,\sigma] + \mathcal{O}(\delta^2)\\ &= e^{-i\rho\delta}\sigma e^{i\rho\delta} + \mathcal{O}(\delta^2).\end{aligned}$$

That is, σ undergoes unitary evolution of the form $e^{-i\rho\delta}$ (to first order in δ) rotating by a small angle δ. By the reciprocity of SWAP operations, ρ undergoes a complementary unitary evolution about σ, leaving it in a state that differs from the original quantum instruction. As a result, the instruction qubits must be refreshed at each step to provide a new identical copy of the instruction state ρ. Choosing $\delta = \theta/N$ and repeating the δSWAP step N times [Fig. 2(a)] approximately yields the desired operator,

$$\mathrm{DME}(\rho, N, \theta) \to e^{-i\rho\theta} + \mathcal{O}\left(\frac{\theta^2}{N}\right). \tag{1}$$

We derive the error scaling in the Appendix H. This result is closely related to the Trotterization of noncommuting Hamiltonians to perform quantum simulations [13]. Similar to how dividing a quantum simulation into smaller steps reduces errors stemming from the Trotter approximation, partitioning DME into more steps with a smaller partial SWAP angle δ per step reduces the DME discretization error.

Of course, the trade-off for increased precision is a need for more copies of the quantum instructions. There are three general approaches to supplying the N copies of the instruction state ρ needed to execute DME:

(1) Teleport copies of the quantum instructions from a third party to the qubits comprising ρ.
(2) Identically prepare the instructions on N copies of ρ [hardware parallelization, as in Fig. 1(c)].
(3) Identically prepare the same set of qubits comprising ρ after each δSWAP [sequential preparation in time, Fig. 2(a)].

The first option may find use in, for example, private quantum software execution; however, it would add significant complexity to this first demonstration. The second option is the one envisioned in the original DME proposal [2], but it requires that all N copies of ρ must be able to couple to the data qubit (for example, by swapping fresh copies of ρ into the qubits that couple to σ or by using a system with a highly connected graph). In this work, we choose option 3: We refresh the same instruction qubit multiple times to avoid the need for teleportation or large numbers of instruction qubits and to allow us to easily vary N.

III. GENERATING N COPIES OF ρ

The most obvious approach for using one qubit to generate N copies of ρ [Fig. 2(a)] is to use measurement-conditioned active feedback to reset the instruction qubit to its ground state and then reprepare the instruction state ρ [14]. However, due to measurement infidelity and the decoherence that occurs during the relatively long duration of the requisite measurement, feedback, and preparation steps, the conventional active-reset approach would introduce an unacceptable level of errors for studying DME on current noisy quantum processors. We instead introduce an alternative approach [Fig. 2(b)] called quantum measurement emulation (QME) to minimize such errors and achieve the largest possible circuit depths. QME approximately reinitializes the instruction qubit in the time required for a single-qubit gate at the cost of a small loss in state purity.

QME is a probabilistic operation that mimics an ensemble-averaged qubit measurement [11]. For intuition, note that for a sufficiently small angle δ, the states of the two qubits are only slightly altered after a δSWAP operation. In this case, a projective measurement of the instruction qubit in the eigenbasis of ρ would reset the instruction qubit to its original state with high probability. Similarly, an ensemble-averaged measurement of many such identically prepared states (i.e., a measure-and-forget step) would reproduce the original ρ with only a slight depolarization. QME mimics this feature without actually performing a measurement by imposing a dephasing channel aligned to the axis of ρ.

The QME operation randomly applies either an identity gate ($\mathbb{1}$) or a π rotation in the instruction qubit eigenbasis according to a Bernoulli process with probability $p = 0.5$:

$$-\boxed{\mathrm{QME}_\nu}- = \begin{cases}\mathbb{1} & \text{with } p = 0.5\\ \mathrm{R}_\nu(\pi) & \text{with } 1-p = 0.5\end{cases} \tag{2}$$

where ν is a normalized vector parallel to the original instruction state. For instruction states aligned with the axes of the Bloch sphere, QME represents a probabilistic application of a Pauli gate. We incorporate QME into our circuit by interleaving δSWAP and QME operations [Fig. 2(b)]. The QME operations are randomized within each instantiation of the circuit, and in the same spirit as randomized compiling [15], the outcomes of multiple such randomized instantiations are averaged to mimic a single circuit with an active reset of ρ.

The choice of using QME as opposed to active feedback stems from a balanced consideration: Even using the fastest reset protocols currently reported for superconducting qubits, the time to reset a qubit is on the order of 500 ns [14]. Using QME, we are able to approximately reprepare the qubit state in the time of a single-qubit gate, in

our case, 30 ns. Technically, the trace distance between a state before and after a δSWAP interaction is approximately δ. Using QME to approximately reset the qubit reduces this to approximately δ^2. Thus, for small δ the state after QME is closer to its original quantization axis. However, the QME-enabled reset of ρ is approximate due to depolarization, which introduces an additional error term to the DME protocol (see Appendix I for derivation). For our two-qubit demonstration of DME with QME (which we denote DME_2), the total operation and error scaling is

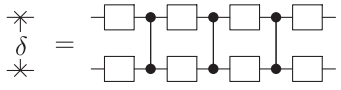
$$\mathrm{DME}_2(\rho, N, \theta) \to e^{-i\rho\theta} + \underbrace{\mathcal{O}\left(\frac{\theta^2}{N}\right)}_{\text{discretization}} + \underbrace{\mathcal{O}\left(\frac{\theta^2}{N}\right)}_{\text{QME}}. \quad (3)$$

Despite the added error term [cf. Eq. (1)], QME effectively supplies the requisite copies of ρ with less error than would be incurred with an active feedback approach in our system due to decoherence of the data qubit during the increased algorithmic runtime required for active feedback.

IV. IMPLEMENTING THE DME_2 ALGORITHM

We implement DME_2 using two frequency-tunable superconducting asymmetric transmon qubits [16,17] in an xmon layout [18]. Single-qubit gate fidelities on both qubits exceed 99.9% (see Appendixes A and B). We use the two-qubit controlled-phase (CZ) gate, which can be implemented by flux tuning one of the qubits near resonance with the $|11\rangle \leftrightarrow |20\rangle$ transition resulting in a phase shift of the $|11\rangle$ state [19,20]. To reduce leakage to the $|20\rangle$ state, we use a variant of an optimal waveform shape [21], which is symmetrized around the sweet spot of the qubit [22]. This technique results in a reduced impact of long timescale variation due to flux transients and reduces sensitivity to flux noise [22]. As a result, our CZ gate has a 99.7% fidelity measured with randomized benchmarking (see Appendix C for more details on gate calibration).

To realize the δSWAP operation, we use the open-source software CIRQ [23] to find an optimal decomposition of the δSWAP in terms of CZ and single-qubit gates. Any two-qubit unitary can be decomposed into single-qubit gates and CZ gates via [24]

$\delta\mathrm{SWAP} :=$  (4)

where each  represents (potentially different) general single-qubit gates that depend on the value of δ, and  is the CZ gate. While this is not the only way to decompose δSWAP, this particular choice allows us to rely solely on high-fidelity gates whose performance can be validated and efficiently optimized, a crucial part of calibrating for high-fidelity algorithmic performance (see Appendix D for details on compilation).

By relying solely on randomized benchmarking to calibrate our gates, we do not observe optimal performance of the algorithm. Because of the repetitive structure of DME_2, small remnant coherent errors accumulate during circuit execution. To remedy this effect, we develop a calibration routine to amplify the small coherent errors, allowing us to accurately determine their magnitude and correct them (see Appendix C for details on this process).

We now turn to demonstrating the DME_2 protocol with a known σ and ρ. In Figs. 2(c) and 2(d), we interrupt the algorithm after $n \leq N$ steps and perform state tomography (see Appendix E for details of the tomographic protocol) to visualize the evolution of the data qubit and instruction qubit as the DME_2 protocol proceeds. We use an initial state $\sigma_{\mathrm{in}} = |+i\rangle\langle+i|$ for the data qubit, and we introduce the notation $\mathrm{DME}_2(\rho_{\mathrm{in}}, N, \theta)$ to indicate the initial instruction state ρ_{in}, total number of steps N, and the phase rotation θ. Figure 2(c) shows an implementation of $\mathrm{DME}_2(|+\rangle\langle+|, 4, \pi/2)$. Since ρ is chosen to be x polarized, this instruction encodes the operation $R_X(\pi/2)$, a $\pi/2$ rotation about the x axis. Figure 2(d) shows an implementation of $\mathrm{DME}_2(|0\rangle\langle 0|, 8, \pi)$ encoding the instruction $R_Z(\pi)$, a π rotation about the z axis. In both cases, σ undergoes a rotation about an axis defined by ρ_{in}, which is visible in the step-by-step tomographic reconstruction of the data qubit state $\sigma(n)$. QME maintains the quantization axis direction of the instruction qubit state $\rho(n)$, albeit with gradual depolarization consistent with the effects of QME. The classically defined δSWAP operations are identical in these two cases; it is the change in the instruction state ρ_{in} that causes a different operation on the data qubit. Thus, the implemented quantum operation on σ is uniquely determined by the state of another quantum system, a demonstration of how the state ρ_{in} "instructs" the operation to be executed on the state. We emphasize that in this case, ρ is a known quantum state, but outside of this proof-of-principle demonstration it need not be. We only need to know how to generate ρ, we need not know its state.

V. BENCHMARKING THE DME_2 ALGORITHM

We now turn to benchmarking our implementation of the DME_2 algorithm. First, we execute DME_2 with a data state $\sigma_{\mathrm{in}} = |0\rangle\langle 0|$ and instruction state $\rho_{\mathrm{in}} = |+i\rangle\langle+i|$, while varying the total steps N (Fig. 3). This allows us to probe the interplay between discretization error (which decreases with N) and decoherence and noise-induced error (which increases with N). We use two angles $\theta = \pi$ and $\theta = \pi/2$ to elucidate the effects of changing the overall angle. For each experiment, we

(1) Perform the full algorithm $\mathrm{DME}_2(\rho_{\mathrm{in}}, N, \theta)$ with many QME randomizations.
(2) Tomographically reconstruct the density matrix $\sigma(N)$ for each N and average over QME randomization.
(3) Calculate the fidelity between $\sigma(N)$ and an ideal rotation with no discretization or processor error, as given by $\sigma_{\mathrm{ideal}} = e^{-i\rho_{\mathrm{in}}\theta}\sigma_{\mathrm{in}}e^{i\rho_{\mathrm{in}}\theta}$.

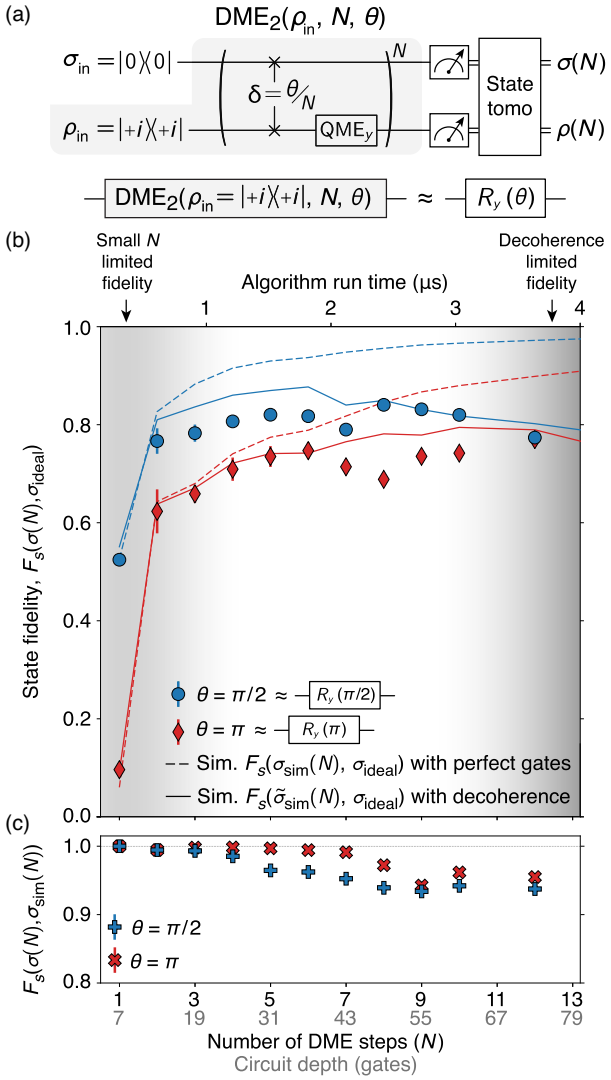


FIG. 3. Algorithm performance as a function of N. (a) Circuit schematic for $\mathrm{DME}_2(|+i\rangle\langle+i|, N, \theta)$. Data qubit is initialized in $\sigma_{\mathrm{in}} = |0\rangle\langle 0|$. (b) State fidelity ($F_s$) of the data qubit state σ to the ideal state $\sigma_{\mathrm{ideal}} = e^{-i\rho_{\mathrm{in}}\theta}\sigma_{\mathrm{in}}e^{i\rho_{\mathrm{in}}\theta}$ as a function of the total DME steps (N). The instruction qubit is initialized to the $|+i\rangle\langle+i|$ state, resulting in an ideal operation $e^{-i|+i\rangle\langle+i|\theta} = R_y(\theta)$. The x axis shows the number of $\delta\mathrm{SWAP} + \mathrm{QME}$ steps N (bottom, black), circuit depth (bottom, gray), and active circuit clock time (top). Data for $\theta = \pi$ ($\pi/2$) are shown with red/$\diamond$ (blue/$\circ$) markers. Dashed lines are the state fidelity between σ_{ideal} and a simulated output of the $\mathrm{DME}_2(|+i\rangle\langle+i|, N, \theta)$ circuit, assuming perfect gates [denoted $\sigma_{\mathrm{sim}}(N)$]. Solid lines are the same simulation as shown in dashed lines, but with amplitude-damping and depolarizing channels added, to simulate decoherence effects [denoted $\tilde{\sigma}_{\mathrm{sim}}(N)$]. (c) State fidelity of the measured $\sigma(N)$ to simulated output of the $\mathrm{DME}_2(|+i\rangle\langle+i|, N, \theta)$ circuit with perfect gates [$\sigma_{\mathrm{sim}}(N)$]. Error bars are determined from bootstrap analysis (detailed in Appendix F).

σ_{ideal} can equivalently be thought of as either resulting from a perfect unitary rotation or as the result of executing DME when implemented on an error-free processor with an infinite number of copies of the quantum instructions. In Fig. 3(b), we plot the state fidelity between the experimentally measured data qubit density matrix $\sigma(N)$ and σ_{ideal} calculated according to [25]

$$F_s(\sigma(N), \sigma_{\mathrm{ideal}}) = \mathrm{Tr}\left(\sqrt{\sqrt{\sigma(N)}\sigma_{\mathrm{ideal}}\sqrt{\sigma(N)}}\right)^2. \quad (5)$$

There are two sources of error we must consider in interpreting $F_s(\sigma(N), \sigma_{\mathrm{ideal}})$ as N is increased: the approximate nature of the algorithm, which causes errors that reduce with N, and imperfections in the quantum processor, which cause errors that increase with N. This tension is reflected in the immediate increase in $F_s(\sigma(N), \sigma_{\mathrm{ideal}})$ as a function of N due to diminishing discretization error, while the gradual reduction for higher N reflects decoherence and finite gate fidelities.

In order to quantitatively assess the role of discretization versus decoherence, we simulate the circuit in Fig. 3(a) assuming perfect gates and no decoherence, but including the discretization error arising from the approximate nature of DME_2. We sample every possible combination of QME gates for a DME_2 circuit of length N and simulate the application of each circuit to the experimentally measured $\rho_{\mathrm{in}} \otimes \sigma_{\mathrm{in}}$ (thus accounting for state-preparation errors). Figure 3(b) (dashed lines) show the result of calculating state fidelity between the noise-free simulation and the ideal rotation for both $\theta = \pi/2$ (blue curves) and $\theta = \pi$ (red curves).

To model the crossover from discretization-limited to noise-limited fidelity, we add amplitude-damping and dephasing channels to our simulation, using coherence parameters from independent measurements (see Appendix A). The fidelity between the simulated output states including decoherence effects [denoted $\tilde{\sigma}_{\mathrm{sim}}(N)$] and σ_{ideal} is plotted in Fig. 3(b) (solid lines) and shows good agreement with the experimental data, indicating we are mostly limited by decoherence effects and not coherent errors in the gates.

The simulated and experimental curves in Fig. 3 reflect the interplay between finite-N (discretization) error and processor (noise and decoherence) errors. At small N, the error is dominated by the approximate nature of DME_2 as given in Eq. (3). The error is greater for larger θ, consistent with error scaling as $\mathcal{O}(\theta^2/N)$. For large N, the discretization error improves and the processor's performance is instead limited by finite gate fidelity and decoherence; here, the curves for $\theta = \pi$ and $\theta = \pi/2$ begin to converge. The algorithm is at its most accurate for intermediate N, where discretization error is relatively low and the circuit is sufficiently free of compounding physical errors. This trade-off (improved performance with increasing circuit depth until gate fidelities become limiting) is a generic property of Trotterized quantum algorithms on noisy processors in the absence of error-correction protocols [26]. As a point of comparison for timescales, on the

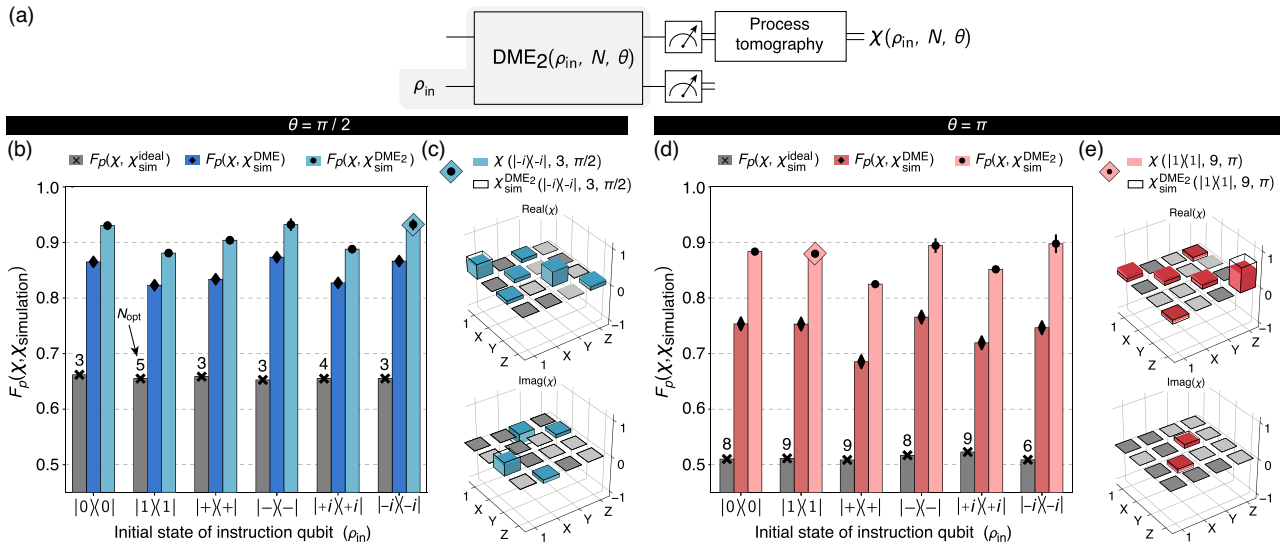


FIG. 4. Benchmarking process fidelity of DME_2 for $\theta = \pi/2$ and $\theta = \pi$. (a) Circuit schematic. Single-qubit process tomography is performed for a set of six instruction states ρ_{in} representing cardinal points of the Bloch sphere. (b),(d) Process fidelities between measured process maps and simulated processes for six instruction states and $\theta = \pi/2$ in (b) and $\theta = \pi$ in (d). Gray (×marker) denotes the fidelity $F_p(\chi, \chi_{\mathrm{sim}}^{\mathrm{ideal}})$ between the measured process map χ and a simulated ideal process $\chi_{\mathrm{sim}}^{\mathrm{ideal}}$, e.g., a rotation of angle θ around the axis given by the Bloch vector of ρ_{in}. The data are presented at $N = N_{\mathrm{opt}}$ determined as the step number at which the fidelity to $\chi_{\mathrm{sim}}^{\mathrm{ideal}}$ is maximized; N_{opt} is indicated by the number above each bar. Dark blue or red (◇ marker) indicates the fidelity $F_p(\chi, \chi_{\mathrm{sim}}^{\mathrm{DME}})$ between the measured process map and a simulation of the original DME algorithm (assuming perfect gates and no decoherence). Light blue or red (○ marker) shows the fidelity $F_p(\chi, \chi_{\mathrm{sim}}^{\mathrm{DME}})$ between the measured process map and a simulation of DME_2 (assuming perfect gates, no decoherence, and QME to reinitialize the instruction qubit at each step). (c) Representative process matrices for $\chi(|-i\rangle\langle -i|, 3, \pi/2)$ (blue) and a simulation of $\chi_{\mathrm{sim}}^{\mathrm{DME}_2}(|-i\rangle\langle -i|, 3, \pi/2)$ (wire frame). Colored process matrix elements indicate points with magnitude $\chi_{ij} > 0.02$; other elements are gray for clarity of scale. (e) Similar to (c) but for $\chi(|1\rangle\langle 1|, 9, \pi)$.

top axis of Fig. 3(b), we plot the wall-clock run-time of the circuit that is executed. A single DME_2 step, i.e., a δSWAP followed by a QME application, takes 335 ns (including settling times; see Appendix A for details).

Finally, to asses our experimental ability to accurately implement DME_2, we plot in Fig. 3(c) the fidelity between $\sigma(N)$ and the simulated output of the $\mathrm{DME}_2(|+i\rangle\langle +i|, N, \theta)$ circuit in the absence of noise $[\sigma_{\mathrm{sim}}(N)]$. To circuit depth 73, this fidelity exceeds 0.90.

We next perform quantum process tomography to assess the error budget of the DME_2 implementation independent of the target state. Here, we view DME_2 as a quantum channel Λ_{DME_2}, which in the operator-sum representation can be defined via a process matrix χ:

$$\Lambda_{\mathrm{DME}_2}(\rho_{\mathrm{in}}, N, \theta): \rho \mapsto \sum_{j,k=1}^{d^2} \chi_{jk}(\rho_{\mathrm{in}}, N, \theta)\sigma_j \rho \sigma_k. \quad (6)$$

From the process matrix, it is possible to define a process fidelity between two different processes [27]

$$F_p(\chi, \chi') = \mathrm{Tr}\left(\sqrt{\sqrt{\chi'}\chi\sqrt{\chi'}}\right)^2. \quad (7)$$

To calculate the process fidelity of our implemented algorithm, we experimentally reconstruct the full process matrix $\chi(\rho_{\mathrm{in}}, N, \theta)$ of the channel Λ_{DME_2}, for a set of instruction states (ρ_{in}) given by the cardinal points on the Bloch sphere. For each ρ_{in}, we sweep N to find the optimal point N_{opt} defined as the value of N which has the highest process fidelity to the ideal rotation $U_{\mathrm{ideal}} = e^{-i\rho_{\mathrm{in}}\theta}$. The mean N_{opt} for $\theta = \pi/2$ is 4 at circuit depth 25; for $\theta = \pi$, this increases to 8 at circuit depth 49, a direct reflection of the $\mathcal{O}(\theta^2/N)$ scaling of DME_2.

In Fig. 4, we plot the process fidelity of our DME_2 demonstration when compared to three different theoretical processes to elucidate the sources of error in our experimental implementation.

(i) $\chi_{\mathrm{sim}}^{\mathrm{ideal}}$: The process representing the ideal rotation U_{ideal}.

(ii) $\chi_{\mathrm{sim}}^{\mathrm{DME}}$: The process representing an implementation of DME in which the instruction state is removed and replaced by a perfect copy after every substep performed with perfect gates and no decoherence.

(iii) $\chi_{\mathrm{sim}}^{\mathrm{DME}_2}$: The process representing an implementation of DME_2, i.e., combining δSWAPs with QME, performed with perfect gates and no decoherence.

Comparing each of these processes to the experimentally extracted process elucidates different aspects of the DME algorithm, both generally and specific to the

implementation in this work. The data for $\theta = \pi/2$ are provided in Fig. 4(b) in blue, and the data for $\theta = \pi$ are provided in Fig. 4(c) in red.

The fidelity $F_p(\chi, \chi_{\text{sim}}^{\text{ideal}})$ is plotted in gray and is overall greater for $\pi/2$ than for π. This is qualitatively consistent with the $\mathcal{O}(\theta^2/N)$ scaling of the discretization error and holds across all cardinal settings of the instruction state. This process fidelity reflects the combination of errors arising from the discretized nature of density matrix exponentiation and the errors from imperfect gates and the approximate nature of QME.

The fidelity $F_p(\chi, \chi_{\text{sim}}^{\text{DME}})$ is shown in dark blue and red. The infidelity in this comparison reflects the physical errors arising from imperfect gates and the error from using QME to approximate the repreparation of ρ_{in}. The difference between $F_p(\chi, \chi_{\text{sim}}^{\text{ideal}})$ and $F_p(\chi, \chi_{\text{sim}}^{\text{DME}})$ is a reflection of a finite-N error intrinsic to the DME algorithm.

Finally, we wish to benchmark how well our quantum processor implements DME_2 by comparing to a simulation of DME_2 with perfect gate implementation (i.e., $\chi_{\text{sim}}^{\text{DME}_2}$). The fidelity $F_p(\chi, \chi_{\text{sim}}^{\text{DME}_2})$ is shown in light blue and red. This fidelity is the most direct metric for the performance of our processor and its ability to execute the algorithm as intended. The simulation of the output of DME_2 is performed by sampling over all QME randomizations and averaging their effect. The average process fidelity between experimental output of DME_2 and a simulation of the corresponding circuit without noise over all instruction settings is 0.91 for $\theta = \pi/2$ and 0.87 for $\theta = \pi$; this algorithmic fidelity is overall reduced for $\theta = \pi$ because N_{opt} occurs at deeper circuit depth incurring more gate errors and decoherence effects.

VI. SUMMARY AND OUTLOOK

Here, we use a superconducting quantum processor to demonstrate an implementation of the density matrix exponentiation algorithm [2]. Our implementation takes advantage of a 99.7% fidelity controlled-phase gate combined with a novel quantum measurement emulation technique to supply approximately reset copies of the instruction qubit required in the algorithm. We achieve state fidelities exceeding 0.9 at circuit depth of 73 sequential gates and process fidelities close to 0.9, independent of the setting of the instruction qubit, when comparing measured state- and process-tomographic data to a simulation of the same circuit assuming perfect gates and measurements. While for technical reasons we use pure states in DME_2, the original DME algorithm handles mixed states and efficiently extends to multiqubit systems requiring only the ability to perform controlled versions of the SWAP operation between pairs of data and instruction qubits [3,4,8,28].

While DME was originally proposed in the context of a specific quantum machine-learning algorithm, its implications may be far more profound: It represents a fundamentally different approach to quantum computer programming. Since DME can map a quantum state into a quantum operator, it opens the possibility of encoding quantum algorithms directly into quantum states and executing those algorithms on other quantum states. This idea has lead to the notion of "quantum software states" enabled by the use and efficiency of DME [4]. In classical computing, the data and the instructions are conceptually interchangeable, since they can both be fully described by bit strings. This data or instruction symmetry is known as homoiconicity. The "quantum software" paradigm offers an appealing quantum analog to classical homoiconicity: In fully quantum software, the data and the instructions are both stored in quantum states and thus interchangeable, mimicking homoiconicity from classical computation. This framework may lead to applications in the development of new efficient quantum algorithms, since no tomographic reconstruction will be needed to learn the quantum instructions. One challenge may be to develop programming frameworks for this approach to quantum computing.

This work also demonstrates that DME, and by extension other Trotterized algorithms and processes, is fundamentally best suited to the fault-tolerant (FT) quantum-computing era. On a "noisy intermediate-scale quantum" (NISQ) device [29], even with high-fidelity gates, we are able to perform only a handful of DME steps before being limited by decoherence and errors in the system. On NISQ devices, Trotterized algorithms are still important to study and perform in the near term because of the lessons they teach about quantum simulations, deep quantum circuits, and highly symmetric algorithms. However, their true power will be unleashed in the FT era in which system performance is no longer limited by physical errors. In a FT processor, algorithm performance is not constrained by the number of gates that can be performed during the lifetime of the qubits. Rather, performance will be limited by the scaling of the algorithms themselves (the number of logical qubits required and the associated circuit depths). This is the era in which DME will shine: The high dimensionality of the logical subspace on the one hand and the ability to increase N without concern for decoherence on the other, pushes the processor into a regime where the exponential scaling differences between DME and tomography are significant, and the ability to reduce error by increasing N becomes a relatively straightforward matter of state reset.

Conceptually, we expect that novel applications of DME will take advantage of the same phenomenon leveraged by hybrid classical-quantum algorithms in the NISQ era, such as the variational quantum eigensolver (VQE) [30]: One can often parametrize the gates required to generate a quantum state of interest, even when the system is too large to simulate the application of those gates and the generated quantum state is unknown. VQE and related algorithms

take advantage of this parametrization to prepare an unknown state and perform partial tomography on it to classically extract elements of interest. DME offers an even more powerful lever: By efficiently preparing the state and using active reset to reinitialize it many times over the course of a DME protocol, the full information contained in the instruction state can be used to perform operations and feed into an algorithm without its quantum information ever being revealed to the classical world. Several examples of algorithms that take this approach are described in the Introduction to this paper, but we suspect that the research community has just begun to scratch the surface of potential applications of quantum instructions. It is our hope that our demonstration of the feasibility of DME even in a NISQ system will spark development of even more such applications.

ACKNOWLEDGMENTS

The authors acknowledge feedback on the manuscript from Antti Vepsäläinen, Cyrus Hirjibehedin, Kyle Serniak, Thomas Hazard, and Steven Weber. M. K. acknowledges support from the Carlsberg Foundation during part of this work. A. G. acknowledges funding from the 2019 Google U.S./Canada Ph.D. Fellowship in Quantum Computing. I. M. acknowledges funding from NSF Grant No. FET-1910859. This research is funded in part by the U.S. Army Research Office Grant No. W911NF-18-1-0411 and the Assistant Secretary of Defense for Research & Engineering under Air Force Contract No. FA8721-05-C-0002. The opinions, interpretations, conclusions, and recommendations are those of the authors and are not necessarily endorsed by the United States Government.

M. K. and M. E. S. contributed equally to this work. M. K., M. S., A. G., G. S., A. B., C. M., and P. K. performed the experiments. M. K., M. S., A. G., G. S., M. O., C. M., and K. O. developed analytical tools and analyzed data. M. K., M. S., A. G., A. B., J. B., R. W., Y. S., and S. G. developed experimental control tools. M. K., M. O., C. M., and K. O performed algorithm simulations under realistic noise models. M. K., M. S., A. G., G. S., M. O., M. M., D. R., S. L., and I. M developed the DME_2 protocol. D. K., A. M., B. N., and J. Y. fabricated the devices. T. O., S. G., and W. O. provided experimental oversight and support. All authors contributed to the discussion of the results and the manuscript.

APPENDIX A: DEVICE PARAMETERS

The quantum processor used in this work has three asymmetric "xmon"-style qubits in a linear chain [16–18]. We use the two leftmost qubits in this protocol; the third is detuned and idles in its ground state. Figure 5(a) shows a schematic of the readout and control setup used to control the qubits. Figure 5(b) shows a scanning electron micrograph of a device identical to the one used in this work. In Table I, we summarize the parameters of the two qubits used for the experiments in the main text. The measured lifetime T_1 and Ramsey coherence time T_{2R} exhibit temporal fluctuations consistent with other reports [31,32].

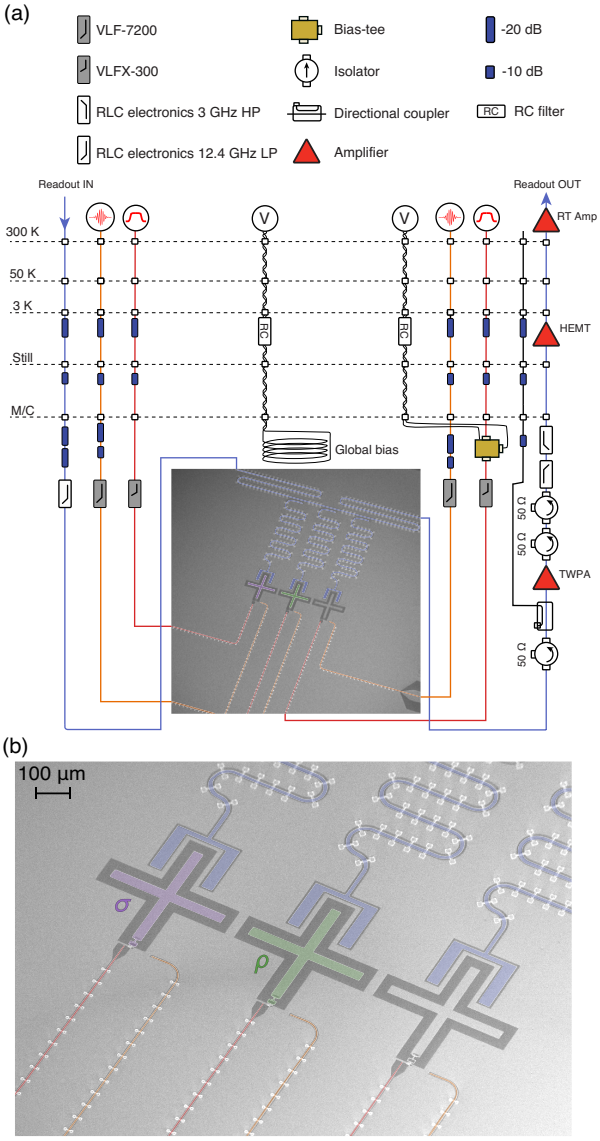


FIG. 5. (a) Schematic of readout and control wiring used for these experiments. The microwave line of qubit 3 is used to drive single-qubit gates on qubit 2. (b) SEM picture of identically fabricated device to the processor used in this work.

For a qubit undergoing frequency modulation (e.g., to implement the CZ gate), frequency-dependent T_1 (and T_{2R}) variations mean that the static coherence times do not necessarily set the relevant limiting timescale for the qubits [31]. To account for the frequency-dependent variations in coherence as the target qubit undergoes the CZ trajectory, we employ an *effective* T_1 (T_{2R}) parameter denoted $\tilde{T}_1$ ($\tilde{T}_{2R}$). These effective coherence times take into account any frequency-dependent variations of coherence as the

TABLE I. Parameters of the two qubits used in this work. See text for details of the definition of $\tilde{T}_1$ and $\tilde{T}_{2R}$.

Parameter	Qubit 1 (σ, target)	Qubit 2 (ρ, instruction)
Idling frequency, $\omega_i/2\pi$	4.748 GHz	4.225 GHz
Anharmonicity, $\eta/2\pi$	−175 MHz	−190 MHz
Coupling strength, $g/2\pi$	10.6 MHz	
Readout resonator frequency, $f_i/2\pi$	7.251 GHz	7.285 GHz
Junction asymmetry	1:5	1:10
Readout assignment fidelity	0.981	0.972
Relaxation time at idling point, T_1	23 μs	39 μs
Coherence time at idling point, T_{2R}	13 μs	25 μs
Effective relaxation time undergoing CZ trajectory, $\tilde{T}_1$	≈17 μs	(Same as idling)
Effective coherence time undergoing CZ trajectory, $\tilde{T}_{2R}$	≈5 μs	(Same as idling)
Single-qubit gate time, t_{1qb}	30 ns	30 ns
Two-qubit gate time, t_{CZ}	60 ns	

qubit frequency undergoes the trajectory to enact a CZ gate. The effective coherence times are used in simulations of the device performance during two-qubit gates. Since the frequency of qubit 2 is fixed during the CZ gate, its effective coherence times are identical to the idling coherence times.

Figure 6(a) shows an example measurement of $\tilde{T}_1$. We prepare the state $|10\rangle$ (an eigenstate of the CZ gate), apply n CZ gates in sequence, and measure the probability of staying in the $|10\rangle$ state. The exponential decay is fitted, and we find a characteristic number of gates $n_{\tilde{T}_1} \approx 264$. The CZ gate time is 60 ns, and we use a 5-ns spacing between each pulse, leading to an effective decay time $\tilde{T}_1 = n_{\tilde{T}_1} t_{CZ} \approx 17$ μs.

To measure the effective coherence time $\tilde{T}_{2R}$ [Fig. 6(b)], we prepare the $|+0\rangle$ state, apply n CZ gates, and apply a final $X_{\pi/2}$ pulse before measuring. Unlike a standard Ramsey measurement, in which we would idle between the $X_{\pi/2}$ pulses, here we perform back-to-back CZ gates, effectively aggregating decoherence effects over the full frequency range of the CZ gate. To ensure an oscillatory behavior, a small single-qubit phase error is added ($\phi_{q1} \neq 0$) equivalent to performing a detuned Ramsey experiment. Fitting an exponentially damped sine function gives a characteristic decay number $n_{\tilde{T}_{2R}} \approx 76$ CZ gates. We again estimate the effective coherence time as $\tilde{T}_{2R} = n_{T_{2R}} t_{CZ} \approx 5$ μs. Using the δSWAP decomposition in Eq. (4) with single-qubit gate times of 30 ns, two-qubit gate times of 60 ns, and a settling time of 5 ns between each pulse, we arrive at a single DME step taking 335 ns of wall-clock time. As a point of comparison to the algorithm run-time, the qubit coherence times are on the order of 10 to 40 μs (see Table I), but care must be taken when comparing these numbers, since one qubit is undergoing flux tuning, removing it from its sweet spot and reducing its effective T_2 time (see Fig. 6).

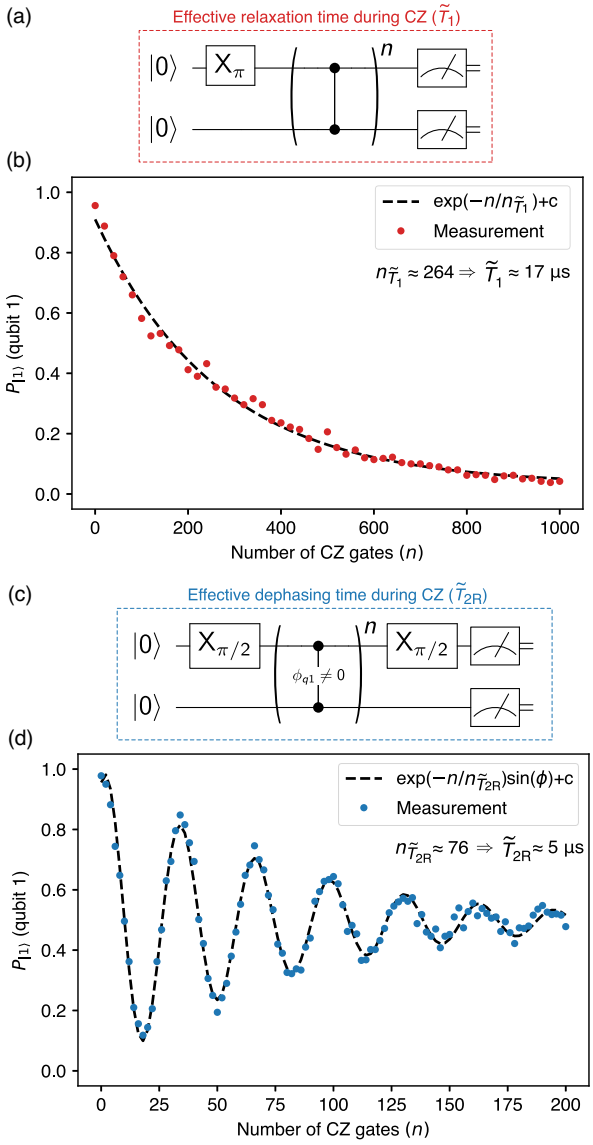


FIG. 6. (a) Measurement circuit to extract effective T_1-like decay time denoted $\tilde{T}_1$. (b) Probability of measuring qubit 1 in the excited state, as the number of CZ gates is increased. The number $n_{\tilde{T}_1}$ sets a characteristic gate number, which can be converted into a characteristic time, $\tilde{T}_1$. (c) Measurement circuit to extract effective T_{2R}-like decay time denoted $\tilde{T}_{2R}$. We essentially perform a Ramsey measurement but interleave CZ gates. (d) Probability of measuring qubit 1 in the excited state, as the number of CZ gates is increased. The number $n_{\tilde{T}_{2R}}$ gives the effective coherence time $\tilde{T}_{2R} \approx 5$ μs.

In Table II, we list the major experimental components used in the experiment.

TABLE II. Major experimental equipment used in the experiment. "Drive" is short for qubit drive; "pump" is microwave source used to activate a traveling-wave parametric amplifier.

Component	Manufacturer	Type
Control Chassis	Keysight	M9019A
AWG	Keysight	M3202A
ADC	Keysight	M3102A
LO/rf/drive/pump	Rohde & Schwartz	SGS100
Refrigerator	BlueFors	XLD
dc bias	Yokogawa	GS 200

APPENDIX B: GATE CHARACTERIZATION

The native gate set of our processor comprises microwave-driven single-qubit x and y rotations $R_X(\phi)$ and $R_Y(\phi)$, single-qubit virtual-z rotations $R_Z(\phi)$, and the two-qubit CZ gate [33]. In particular, we calibrate a numerically optimized 99.7% fidelity CZ gate [19,34] using the symmetrized "NetZero" optimal control waveform that reduces leakage and noise sensitivity [21,22,35].

We use a combination of metrics to quantify the quality of qubit operations during the algorithm. These techniques include single- and two-qubit randomized benchmarking (RB) as well as novel techniques for amplifying and correcting coherent errors. Figure 7 shows single-qubit Clifford randomized benchmarking of the single-qubit operations on both qubits 1 [Fig. 7(b)] and 2 [Fig. 7(c)]. Each trace averages 25 randomizations of the RB circuit [36]. The reference curves [circuit diagram in Fig. 7(a), gray dashed box] are fit to a function of the form

$$f(m) = Ap^m + B. \tag{B1}$$

For the one-qubit Clifford reference curve, we denote p by p_R. The average error per Clifford gate $\mathcal{C}$ can be calculated as

$$\epsilon_R = \frac{1}{2}(1 - p_R). \tag{B2}$$

The error associated with a specific single-qubit gate is extracted by performing interleaved randomized benchmarking (IRB). We fit the IRB data [circuit diagram in Fig. 7(a), red dashed box] for the relevant gate (denoted g) to Eq. (B1) (denoting by p_g the p value for gate g). Then, normalizing the error rate to the one-qubit Clifford reference [37],

$$\epsilon_g = \frac{1}{2}(1 - p_g/p_R). \tag{B3}$$

Using this procedure, we find an average Clifford gate fidelity ($F_R = 1 - \epsilon_R$) of 0.9987 for qubit 1 and 0.9987 for qubit 2. The average gate fidelity (i.e., $\bar{F} = \langle 1 - \epsilon \rangle_g$) over all single-qubit gates is 0.9991 for qubit 1 and 0.9994 for qubit 2.

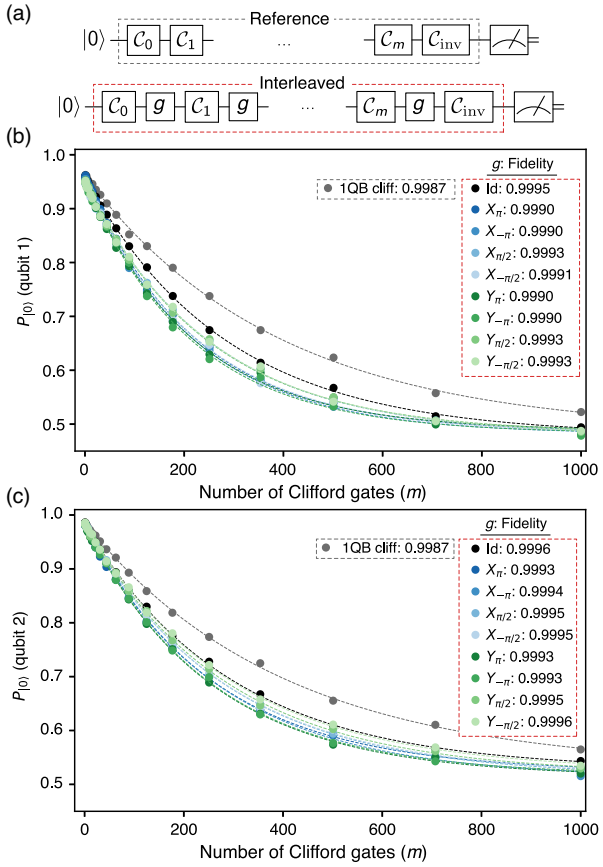


FIG. 7. (a) Circuit diagrams for measuring the reference curve (gray dashed box) and interleaved curve for a single-qubit gate g (red dashed box) relevant for Clifford randomized benchmarking for a single qubit. (b) [(c)] Results for reference (gray) and interleaved (varying colors for each gate) randomized benchmarking for qubit 1 [qubit 2].

In Fig. 8, we assess the two-qubit gate fidelity using randomized benchmarking. The protocol is identical to the single-qubit case, except we measure the probability of being in the $|00\rangle$ state after the sequence [36]. We use 48 randomizations for both reference and interleaved measurements [circuits shown in Fig. 8(a)]. In Fig. 8(b), we show the result of the RB and IRB measurements. The error bars are 1σ standard deviations of the output distribution of the 48 random circuits. The fit is again performed using Eq. (B1), and error margins are extracted using forward propagation of weights based on the standard deviation at each m to ensure accurate error bounds. This is achieved using the ABSOLUTE_SIGMA option of the PYTHON SCIPY.OPTIMIZE.CURVE_FIT function. The two-qubit Clifford reference error rate is calculated similarly to Eq. (B2) (with p being the two-qubit Clifford reference value denoted p_{2r}), but the error per Clifford gate is modified to

$$\epsilon_{2r} = \frac{3}{4}(1 - p_{2r}). \tag{B4}$$

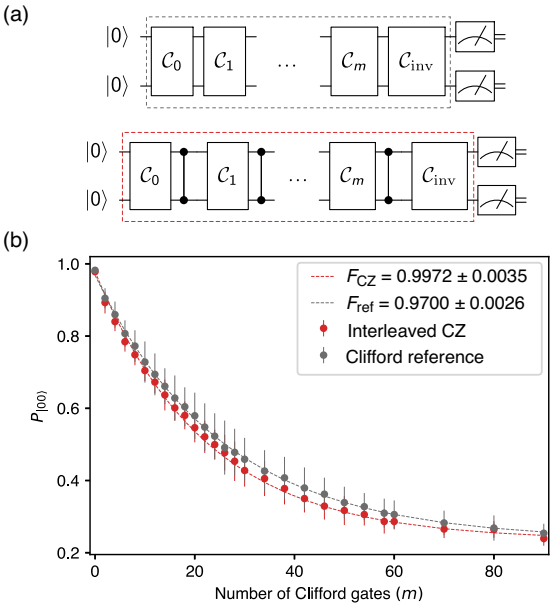


FIG. 8. (a) Gate sequences for measuring the two-qubit Clifford reference (gray dashed box) and interleaved CZ (red dashed box) RB numbers. (b) Example decay curve of $P_{|00\rangle}$ as the number of two-qubit Clifford gates (m) is increased. Each data point is averaged over $k = 48$ randomizations of the choice of Clifford gates. Error bars are 1σ standard deviations at each point from the 48 measurements, and fitting is performed using forward propagation of points weighted by their error bars.

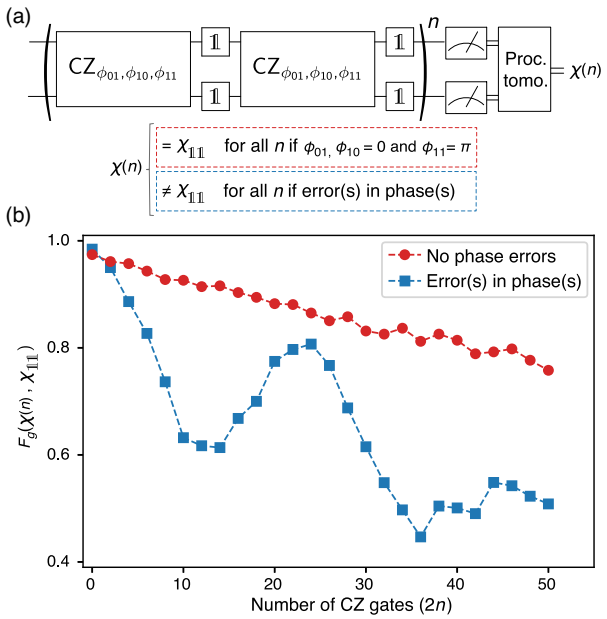


FIG. 9. (a) Gate sequence used to perform process tomography of a sequence of an even number of CZ gates to get the chi matrix $\chi(n)$ used to compare with the identity process map to infer coherent errors. The gate sequence will nominally implement $\chi_{\mathbb{1}\mathbb{1}}$ up to overall system decoherence (visible as the overall decrease of both the linear and oscillating measurements) if there are no phase errors in the $\mathrm{CZ}_{\phi_{01},\phi_{10},\phi_{11}}$ gate. (b) The gate fidelity $F_g(\chi(n), \chi_{\mathbb{1}\mathbb{1}})$ as the number of CZ gates ($2n$) is increased. With no phase errors in the CZ gate, F_g decreases monotonically. With a phase error in the CZ gate, F_g will oscillate with the period indicating the scale of the phase error.

Then, $\epsilon_{\rm CZ}$ is found by performing IRB and fitting the interleaved data to get $p_{\rm CZ}$ and normalizing to the 2QB reference error. Doing so, we find a CZ gate fidelity

$$F_{\rm CZ} = 1 - \epsilon_{\rm CZ} = 0.9972 \pm 0.0035. \tag{B5}$$

To achieve "last-mile" improvements in fidelity, we use numerical optimization techniques to fine-tune the parameters of the NetZero waveform, with the RB decay curve as a cost function [22,34].

APPENDIX C: COHERENT ERROR REDUCTION

As practitioners of quantum computing have explored more complex circuits at greater depth and with more underlying structure, it has become evident that RB is a limited metric for the performance of a gate (see, e.g., Refs. [38–40] and references therein). In particular, small coherent errors can cause disproportionately deleterious effects in algorithms with a repetitive structure (such as Trotterized algorithms), and RB is ill-suited to characterize such small coherent errors because it is designed to randomize over them.

To minimize the effects of coherent errors in the CZ gate, we implement a calibration technique which relies on process tomography of long strings of CZ gates (Fig. 9).

The general controlled-phase gate (denoted $\mathrm{CZ}_{\phi_{01},\phi_{10},\phi_{11}}$) is given by

$$\mathrm{CZ}_{\phi_{01},\phi_{10},\phi_{11}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & e^{-i\phi_{01}} & 0 & 0 \\ 0 & 0 & e^{-i\phi_{10}} & 0 \\ 0 & 0 & 0 & e^{-i\phi_{11}} \end{bmatrix}. \tag{C1}$$

If $\phi_{01} = \phi_{10} = 0$ and $\phi_{11} = \pi$, this produces the target CZ gate. However, for small deviations from these parameters it is still possible to achieve $\gtrsim$0.99 randomized benchmarking fidelities. Since small phase deviations can compound to form larger errors—specifically in algorithms with a repeating pattern like DME or quantum-error-correction protocols—we develop other calibration strategies to detect and correct such errors.

Our amplification protocol is comprised of implementing a circuit with two back-to-back blocks of $\mathrm{CZ}_{\phi_{01},\phi_{10},\phi_{11}}$ followed by identity gates on both qubits designed to mimic the presence of single-qubit gates, as shown in Fig. 9(a). If the CZ gate contains no phase errors, this sequence produces an identity operation, irrespective of the number (n) of such two-CZ blocks applied. We perform two-qubit

process tomography to extract the process matrix $\chi(n)$. We compare $\chi(n)$ to the process map of a two-qubit identity operation ($\chi_{\mathbb{1}\mathbb{1}}$) via the average gate fidelity $F_g(\chi(n),\chi_{\mathbb{1}\mathbb{1}})$, which is related to the process fidelity $F_p(\chi,\chi')$ [defined in Eq. (F2)] according to [41]

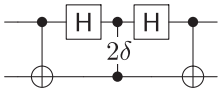
$$F_g(\chi,\chi') = \frac{dF_p(\chi,\chi')+1}{d+1}, \tag{C2}$$

where d is the dimensionality of the Hilbert space ($d=4$ in the case of a two-qubit gate).

Figure 9(b) shows the gate fidelity of a circuit optimized to remove phase errors from the CZ gate (red circles), and one in which a CZ gate *with* phase errors is used (blue squares). In the optimized case, the monotonic gate fidelity decay stems only from decoherence effects. However, in the presence of a coherent phase error, the gate fidelity oscillates with n. In this specific example, after roughly 25 CZ gates, the phase error has effectively rotated by 2π, corresponding to an approximate per-step error of $2\pi/25 \approx 0.08\pi$ in one of the phases. The evolution of the process maps is useful both practically (for achieving higher-performance gates) and scientifically (for understanding the limitations of RB). By examining the details of the process maps, we are able to infer in which of the parameters ϕ_{01}, ϕ_{10}, or ϕ_{11} the error appeared and to correct accordingly. This minor correction typically does not change the fidelity as measured with RB (except in the case of particularly egregious phase errors). From Fig. 9(b), it is also clear that process tomography of a single CZ instance does not reveal the coherent error: The first data point for the sequence with phase errors has nearly identical fidelity to the optimized gate. Both of these facts are consistent with a growing understanding that RB may not be the optimal approach to identifying and correcting coherent errors in single- and multiqubit gates. Finally, the identity gates are inserted between the CZ gates to as closely as possible mimic the generic optimal gate sequence of a two-qubit algorithm, without exploiting any specific structure of an algorithm.

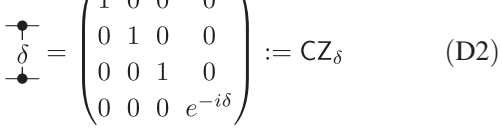
APPENDIX D: COMPILATION

As we mention in the main text, we implement δSWAP using single-qubit gates and the entangling CZ gate; see Eq. (5). The open-source software package CIRQ [23] is used to determine the appropriate single-qubit gate parameters for a given δSWAP. A conceptually transparent approach to generating a δSWAP uses the decomposition

$\delta\text{SWAP} :=$ 

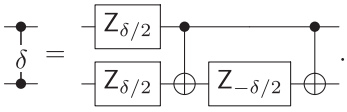
(D1)

where



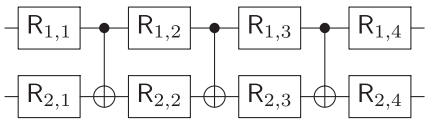
$:= \text{CZ}_\delta$ (D2)

is a partial CZ gate, and  is the CNOT gate with qubit 2 as the target. The CZ_δ gate can in turn be compiled using an additional decomposition



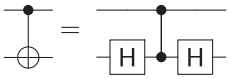
. (D3)

However, such an approach would introduce two CZ gates for each CZ_δ gate, adding significant circuit depth overhead. We use a more generalized and gate-efficient approach, relying on the fact that any two-qubit gate can generically be decomposed into a circuit with the structure [24,42]

$U_{2\text{QB}} =$ 

. (D4)

Here, $R_{i,j}$ is a single-qubit gate acting on qubit i at moment j in the circuit. By using the identity



(D5)

and absorbing the Hadamard gates (H) into the neighboring single-qubit gates, the circuit in Eq. (D4) becomes identical to the circuit in Eq. (4).

We use the open-source software CIRQ [23] to determine the settings of the single-qubit gates for each value of δ. The single-qubit rotations around the x, y axes are decomposed according to $R_Z(-\varphi)R_X(\theta)R_Z(\varphi)$ (the PHASEDXPOWGATE in CIRQ), and the R_Z rotations are performed virtually [43]. The δSWAP is implemented using the SWAPPOWGATE function in CIRQ (the SWAPPOWGATE has a factor of 2 difference, relative to our definition of δSWAP). Thus, we are able to compose a unique composite gate sequence for each δSWAP relying only on high-fidelity single- and two-qubit gates.

Figure 10 shows the full compilation protocol. To construct the full $\text{DME}(\rho,N,\theta)$ circuit, we append N copies of the compiled δSWAP gate using $\delta=\theta/N$, interleaving the requisite QME_ν on qubit 2 (the instruction qubit, ρ) to emulate the effect of measurements. Rows 1 and 2 show the generic structure and gate decomposition of our implementation of DME_2. The final layer of single-qubit gates in the δSWAPδSWAP at step n can be recompiled together with the QME_ν and the first layer of single-qubit gates in the δSWAP at step $n+1$. We use CIRQ to slice out

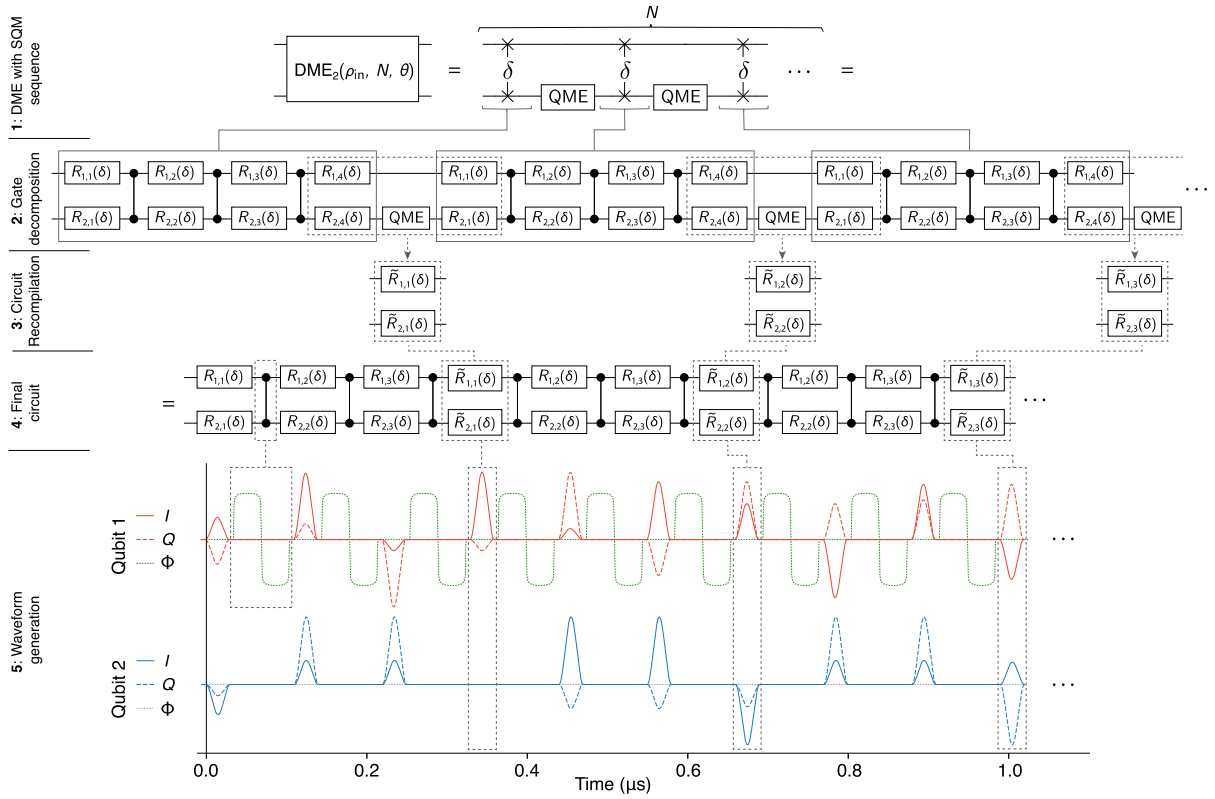


FIG. 10. Details of δSWAP and DME compilation. Row 1: the density matrix exponentiation algorithm implemented using partial SWAP operations and the simulated quantum measurement (QME) gate. Row 2: decomposing each δSWAP according to Eq. (D4). Each substep at this step requires eight layers of gates (seven for δSWAP decomposition and one for QME). Row 3: the three layers of single-qubit gates stemming from the end of the δSWAP of step n, followed by QME, and the first layer of single-qubit gates in δSWAP of step $n+1$ can be recompiled into a single layer. Row 4: the recompiled gates are reinserted into the algorithm result in the optimal structure of exactly one CZ gate, followed by a single layer of single-qubit gates. Row 5: example waveform output to the I, Q (x, y) ports and the flux-tuning pulse (labeled Φ) implementing the NetZero waveform used to implement the CZ gate [21,22].

these three layers (row 2) of single-qubit gates, recompile them into a single layer (row 3), and reinsert them (row 4). Finally, in row 5 we show an example waveform output from our signal generation software, implementing the first $n = 3$ steps in an $N = 5$ DME_2 program.

Our compilation relies upon a restricted set of gates that are readily characterized and numerically optimized. The final compiled circuit has a regular structure (each CZ gate is followed by exactly one layer of single-qubit gates), amenable to generic tuneup protocols for reducing coherent error buildup. These features enable it to achieve high algorithmic fidelity at significant circuit depth.

APPENDIX E: STATE AND PROCESS TOMOGRAPHY

Quantum state tomography is performed by taking advantage of independent single-shot readout of all four computational states $\{00, 01, 10, 11\}$. We first calibrate the measurement operators by building a matrix $\bar{\bar{\beta}}$ that maps the two-qubit Pauli matrices $\hat{\sigma}_{\mathbb{1}\mathbb{1}}, \hat{\sigma}_{\mathbb{1}Z}, \hat{\sigma}_{Z\mathbb{1}}$, and $\hat{\sigma}_{ZZ}$ onto the measurement probabilities p_{ij}:

$$\vec{\mathrm{p}} = \bar{\bar{\beta}}\,\vec{\sigma}, \tag{E1}$$

where

$$\vec{\mathrm{p}} \equiv \begin{pmatrix} p_{00} \\ p_{01} \\ p_{10} \\ p_{11} \end{pmatrix} \quad \text{and} \quad \vec{\sigma} \equiv \begin{pmatrix} \hat{\sigma}_{\mathbb{1}\mathbb{1}} \\ \hat{\sigma}_{\mathbb{1}Z} \\ \hat{\sigma}_{Z\mathbb{1}} \\ \hat{\sigma}_{ZZ} \end{pmatrix}. \tag{E2}$$

The $\bar{\bar{\beta}}$ matrix is calibrated using techniques drawn from Ref. [44]; a full motivation and derivation of the technique can be found there. For a measurement of $\vec{\mathrm{p}}$ with perfect fidelity and no qubit decay during measurements, all components of $\bar{\bar{\beta}}$ have amplitude 0.25; deviations from this amplitude correspond to a calibration of such measurement errors. We begin by calibrating the single-qubit $\bar{\bar{\beta}}$ matrices, namely,

$$\begin{pmatrix} p_0 \\ p_1 \end{pmatrix} = \begin{pmatrix} \beta_{\mathbb{1}}^0 & \beta_Z^0 \\ \beta_{\mathbb{1}}^1 & \beta_Z^1 \end{pmatrix} \begin{pmatrix} \hat{\sigma}_{\mathbb{1}} \\ \hat{\sigma}_Z \end{pmatrix} \tag{E3}$$

by fitting Rabi oscillations in p_0 and p_1 for each qubit. Because the two-qubit probability vector $\vec{p}$ is generated from correlations between single-qubit measurements, the two-qubit $\bar{\bar{\beta}}$ matrix is given by the tensor product of the single-qubit matrices, e.g., $\bar{\bar{\beta}} = \bar{\bar{\beta}}_1 \otimes \bar{\bar{\beta}}_2$.

An arbitrary 4×4 matrix including a two-qubit density matrix ρ may be mapped onto the Pauli basis according to

$$\rho = \sum_{i,j=\{\mathbb{1},X,Y,Z\}} c_{ij}\hat{\sigma}_{ij}. \tag{E4}$$

The general 4×4 matrix of this form has 16 degrees of freedom; trace normalization of a physical density matrix reduces this to 15. The native readout gives us access to the components of ρ contained in $\hat{\sigma}_Z$. We gain information about the other components by performing one of nine premeasurement rotations drawn from

$$R = R_1 \otimes R_2, \tag{E5}$$

where

$$R_{1,2} = \begin{cases} R_Y(-\frac{\pi}{2}) & \text{mapping } \hat{\sigma}_X \mapsto \hat{\sigma}_Z, \\ R_X(\frac{\pi}{2}) & \text{mapping } \hat{\sigma}_Y \mapsto \hat{\sigma}_Z, \\ \mathbb{1} & \text{mapping } \hat{\sigma}_Z \mapsto \hat{\sigma}_Z. \end{cases} \tag{E6}$$

For data in Fig. 2 (Figs. 3 and 4), we perform 2000 (500) single-shot measurements for each tomographic rotation in order to ensure accurate estimates of $\vec{p}$. Each of the nine rotation-and-measurement pairings provides four linearly independent measurements of a form similar to Eq. (E1), for a total of 36 equations that overspecify 15 degrees of freedom. We perform maximum-likelihood estimation [45] to derive the positive semidefinite Hermitian matrix that is most consistent with our combined measurement results.

Single-qubit density matrices in Figs. 2 and 3 are extracted by performing partial traces over the two-qubit density matrix calculated using the approach described above; the data in Fig. 4 are drawn from single-qubit tomography performed on the target qubit using a similar protocol.

Single-qubit quantum process tomography, as presented in Fig. 4, is performed using standard techniques [42]. The target qubit is sequentially prepared in four input states

$$\sigma_{\text{in}} = \{|0\rangle\langle 0|, |1\rangle\langle 1|, |+\rangle\langle +|, |i\rangle\langle i|\}, \tag{E7}$$

which span the single-qubit Hilbert space. These prepared states are then passed through the process $\text{DME}_2(\rho_{\text{in}}, N, \theta)$, and single-qubit state tomography is performed to extract the set of mappings $\{\sigma_{\text{in}} \xrightarrow{\text{DME}_2(\rho_{\text{in}},N,\theta)} \sigma(N)\}$. Linear combinations of these mappings provide the process map χ that reveals the effect of the quantum channel on an arbitrary input density matrix. We then employ techniques developed in Ref. [46] to efficiently project χ onto the closest completely positive and trace-preserving (CPTP) mapping χ_{CPTP} ensuring physicality of the process.

APPENDIX F: BOOTSTRAP ERROR ANALYSIS

We employ bootstrapping techniques to derive the uncertainty bounds in Figs. 3 and 4. In principle, one could simply take a sample of many QME randomizations and calculate the mean and uncertainty within that dataset. However, those error bars are not representative of the error in the DME_2 protocol; rather, they represent the uncertainty of a protocol in which only a single QME randomization is used to perform DME_2. As a result, these error bars are unphysically large, particularly at small N where the protocol chooses from one of only a few paths that have very different outcomes.

The true uncertainty of the DME_2 protocol is captured by (i) accumulating enough QME samples to ensure sufficient randomizations, (ii) building density or process matrices from the average outcome of all these randomizations, and then (iii) repeating this process many times with different randomizations to estimate the uncertainty. This is precisely what bootstrapping accomplishes [47].

The following describes the protocol for extracting boostrapped averages and uncertainties for Fig. 3. For each data point representing a unique setting of $\text{DME}_2(\rho_{\text{in}}, N, \theta)$, we employ the following protocol:

(1) For a given instantiation of the QME gates, execute $\text{DME}_2(\rho_{\text{in}}, N, \theta)$ and perform two-qubit state tomography.
(2) For r_{QME} different instantiations of QME gates, repeat step 1 to accumulate the experimental density matrices from which bootstrapped samples will be drawn.
(3) Using sample with replacement, select n_{samp} samples from the r_{QME} datasets and average the density matrices together. This represents a single bootstrapped density matrix.
(4) Perform a partial trace over the instruction qubit to extract the reduced density matrix of the target system.
(5) Calculate the state fidelity to the states of interest.
(6) Repeat steps 3–5 a total of N_{samp} times to extract mean fidelities and 1σ uncertainties.

State fidelity is calculated according to [25]

$$F_s(\sigma, \sigma') = \text{Tr}\left(\sqrt{\sqrt{\sigma'}\sigma\sqrt{\sigma'}}\right)^2. \tag{F1}$$

The bootstrapping protocol for generating process maps and process fidelities in Fig. 4 is similar to that used for state tomography, but we lay it out here explicitly for completeness.

(1) For a given instantiation of the QME gates, prepare the target input states $\{\sigma_{\text{in}}\}$, apply $\text{DME}_2(\rho_{\text{in}}, N, \theta)$, and perform single-qubit state tomography to generate the mappings $\{\sigma_{\text{in}} \mapsto \sigma(N)\}$ required for process tomography.
(2) For r_{QME} different instantiations of QME gates, repeat step 1 to produce a set of $4 \times r_{\text{QME}}$ single-qubit density matrices.
(3) For each of the four σ_{in}, select an independent sample with replacement of n_{samp} σ_{out} instances and average together, leaving four averaged mappings $\{\sigma_{\text{in}} \mapsto \sigma_{\text{out}}\}$.
(4) Calculate the process matrix using the averaged mappings $\sigma_{\text{in}} \mapsto \sigma(N)$. This represents a single bootstrapped process matrix.
(5) Calculate the process fidelity to the process of interest.
(6) Repeat steps 3–5 a total of N_{samp} times to extract mean fidelities and 1σ uncertainties.

The process fidelity between two χ matrices is given by [27]

$$F_p(\chi, \chi') = \text{Tr}\left(\sqrt{\sqrt{\chi'}\chi\sqrt{\chi'}}\right)^2. \tag{F2}$$

In Fig. 3, we collect $r_{\text{QME}} = 295$ circuit randomizations; in Fig. 4, we collect $r_{\text{QME}} = 105$ circuit randomizations. In both cases, we use $n_{\text{samp}} = 100$ and $N_{\text{samp}} = 50$. The number of QME randomizations used for process tomography is limited by experimental time due to the significant additional experimental overhead required for process tomography in comparison to state tomography, and due to the fact that in Fig. 4 we characterize processes for six settings of ρ. The bootstrap sample size n_{samp} and number of bootstrap samples N_{samp} are chosen somewhat arbitrarily, as in all bootstrapping implementations, but are designed to ensure that each bootstrapped sample approaches a central limit with respect to the underlying QME randomization. A graphical representation of the convergence under QME randomizations is shown in Fig. 12; more details are provided in Appendix J.

APPENDIX G: CIRCUIT SIMULATION WITH NOISE

In order to show the qualitative consistency between the data in Fig. 3 and a model of coherence-limited implementation of the DME_2 protocol, we simulate the randomized DME_2 circuits with added decoherence. We input a DME_2 circuit generated by CIRQ to a software tool that adds decoherence (amplitude damping and dephasing) channels corresponding to the identity for duration(s) of the preceding one- or two-qubit gate. An example of this procedure is shown in Fig. 11.

The channel $\mathcal{E}$ that composes amplitude damping and dephasing is given by

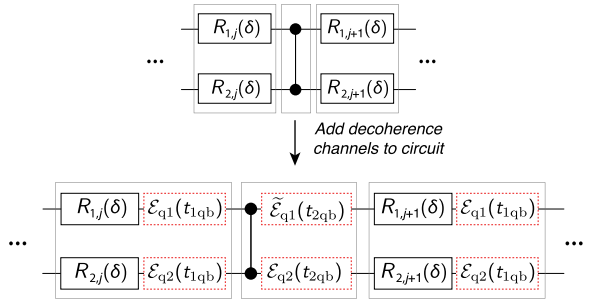


FIG. 11. Instrumenting the DME_2 circuit for simulation of decoherence-induced errors.

$$\mathcal{E}_{qk}(t_{1\text{QB}}): \rho_{qk} \mapsto \sum_{\substack{i=1,2 \\ j=1,2,3}} A_{i,\Gamma_1}(t_{1\text{QB}}) D_{j,\Gamma_\phi}(t_{1\text{QB}}) \rho_{qk} \times D^\dagger_{j,\Gamma_\phi}(t_{1\text{QB}}) A^\dagger_{i,\Gamma_1}(t_{1\text{QB}}), \tag{G1}$$

where $A_{i,\Gamma_1}(t)$ is the amplitude-damping process (with $\Gamma_1 = 1/T_1$), and $D_{j,\Gamma_\phi}(t)$ is the dephasing process ($\Gamma_\phi = 1/T_{2R} - 1/2T_1$), $\Gamma_{1,qk}$ and $\Gamma_{\phi,qk}$ are the appropriate coherence parameters for qubit k, and t is the time of the preceeding single- or two-qubit gate on that qubit. The amplitude-damping and dephasing Krauss operators are given by

$$A_{1,\Gamma_1}(t) = \begin{pmatrix} 1 & 0 \\ 0 & e^{-\Gamma_{1,qk}t/2} \end{pmatrix}, \tag{G2}$$

$$A_{2,\Gamma_1}(t) = \begin{pmatrix} 0 & \sqrt{1 - e^{-\Gamma_{1,qk}t}} \\ 0 & 0 \end{pmatrix}, \tag{G3}$$

$$D_{1,\Gamma_\phi}(t) = \begin{pmatrix} e^{-\Gamma_{\phi,qk}t/2} & 0 \\ 0 & e^{-\Gamma_{\phi,qk}t/2} \end{pmatrix}, \tag{G4}$$

$$D_{2,\Gamma_\phi}(t) = \begin{pmatrix} \sqrt{1 - e^{-\Gamma_{\phi,qk}t}} & 0 \\ 0 & 0 \end{pmatrix}, \tag{G5}$$

$$D_{3,\Gamma_\phi}(t) = \begin{pmatrix} 0 & 0 \\ 0 & \sqrt{1 - e^{-\Gamma_{\phi,qk}t}} \end{pmatrix}. \tag{G6}$$

The channel $\tilde{\mathcal{E}}$ is defined similarly to $\mathcal{E}$, but decoherence rates in the process definitions are replaced with their *effective* coherence parameters. The channel $\tilde{\mathcal{E}}$ thus accounts for the modified coherence properties as qubit 1 undergoes the CZ trajectory (see Fig. 6).

Each instrumented circuit yields a QME-dependent density matrix representing the simulated finite-coherence circuit output for that QME realization. These density matrices are averaged over all 2^N QME realizations (for a DME_2 circuit with N steps), thus producing the noisy simulated two-qubit DME_2 output state denoted "Sim. $F_s(\sigma, \sigma_{\text{ideal}})$ with decoherence" and plotted as a solid line

in Fig. 3(b). For the simulations presented, we use parameters $T_1 = 20\ \mu$s, $T_{2R} = 10\ \mu$s for both qubits, and effective coherence times for qubit 1 of $\tilde{T}_1 = 10\ \mu$s and $\tilde{T}_{2R} = 5\ \mu$s during the channel $\tilde{\mathcal{E}}$. These parameters are qualitatively consistent with, but overall reduced from, the measured parameters in Table I. This difference may indicate additional coherent errors not captured by this model (e.g., from residual $\hat{\sigma}_Z\hat{\sigma}_Z$ interaction or leakage out of the computational subspace).

APPENDIX H: ALGORITHMIC ERROR IN DME

In this section, we show that the algorithmic error in $\mathrm{DME}(\rho, N, \theta)$ (the version of DME in which the instruction state is refreshed with a new, perfect copy after each Trotter step) may be modeled as an amplitude-damping channel and derive its scaling with the parameters of the algorithm. We do so first for a specific instruction state and then generalize to an arbitrary instruction. Throughout, we use $\hat{\sigma}_i$ to indicate the corresponding Pauli matrix.

Suppose that we have instruction and target qubits initially in states ρ and σ, respectively, and apply the operation $e^{-i\mathrm{SWAP}\delta}$ to the joint state $\rho \otimes \sigma$. We first consider the special case in which $\rho = |0\rangle\langle 0|$ and then show how this generalizes to an arbitrary state. The effect of the δSWAP on the target qubit is given by the quantum channel

$$\mathcal{E}^{\rho=|0\rangle\langle 0|}_{\delta\mathsf{SWAP}}(\sigma) = \mathrm{Tr}_\rho(e^{-i\mathsf{SWAP}\delta}[\sigma \otimes |0\rangle\langle 0|]e^{i\mathsf{SWAP}\delta}). \tag{H1}$$

Next, we use the fact that

$$e^{i\mathsf{SWAP}\delta} = \cos(\delta)\hat{\sigma}_{\mathbb{1}\mathbb{1}} + i\sin(\delta)\mathsf{SWAP}, \tag{H2}$$

which follows from the fact that $\mathrm{SWAP}^2 = \hat{\sigma}_{\mathbb{1}\mathbb{1}}$ where $\hat{\sigma}_{\mathbb{1}\mathbb{1}}$ is the two-qubit identity matrix. Using this together with the identity $\mathrm{Tr}_\rho[\mathrm{SWAP}(X \otimes Y)] = YX$ (where Tr_ρ is a partial trace over the second subsystem), we find

$$\begin{aligned}\mathcal{E}^{\rho=|0\rangle\langle 0|}_{\delta\mathsf{SWAP}}(\sigma) = {}& \cos^2(\delta)\sigma + i\cos(\delta)\sin(\delta)[\sigma, |0\rangle\langle 0|] \\ & + \sin^2(\delta)|0\rangle\langle 0|.\end{aligned} \tag{H3}$$

Using the matrix representation of σ in the $\{|0\rangle, |1\rangle\}$ basis, we find that σ transforms as

$$\begin{pmatrix} \sigma'_{00} & \sigma'_{01} \\ \sigma'_{10} & \sigma'_{11} \end{pmatrix} = \begin{pmatrix} \sigma_{00} + \sigma_{11}\sin^2(\delta) & \cos\delta e^{-i\delta}\sigma_{01} \\ \cos\delta e^{+i\delta}\sigma_{10} & \sigma_{11}\cos^2(\delta) \end{pmatrix}, \tag{H4}$$

where $\sigma_{ij} = \langle i|\sigma|j\rangle$ as measured in the $\{|0\rangle, |1\rangle\}$ basis. The channel that implements this transformation has a simple interpretation as the composition of a rotation and an amplitude decay.

Let

$$\mathcal{U}^{\rho=|0\rangle\langle 0|}_\delta(\cdot) = e^{-i\delta|0\rangle\langle 0|}(\cdot)e^{i\delta|0\rangle\langle 0|} = e^{-i(\delta/2)\hat{\sigma}_Z}(\cdot)e^{+i(\delta/2)\hat{\sigma}_Z} \tag{H5}$$

be the superoperator corresponding to the unitary $e^{-i\delta|0\rangle\langle 0|}$, or equivalently, the superoperator corresponding to the rotation by angle δ around the z axis. Also, let $\mathcal{A}_p$ be the amplitude-damping channel described by the Kraus decomposition

$$\mathcal{A}_p(\sigma) = A_1\sigma A_1^\dagger + A_2\sigma A_2^\dagger, \tag{H6}$$

where

$$A_1 = \begin{pmatrix} 1 & 0 \\ 0 & \sqrt{1-p} \end{pmatrix}, \qquad A_2 = \begin{pmatrix} 0 & \sqrt{p} \\ 0 & 0 \end{pmatrix}. \tag{H7}$$

This amplitude-damping channel describes the process in which the system in state $|1\rangle$ decays to state $|0\rangle$ with probability p. It can be shown that the amplitude-damping channel satisfies the condition

$$\mathcal{A}_p \circ \mathcal{U}_\delta = \mathcal{U}_\delta \circ \mathcal{A}_p \tag{H8}$$

for all $\theta \in [0, 2\pi)$. This equality implies that the action of this channel is invariant under rotations around the z axis.

Then, using Eq. (H4) one can show that

$$\mathcal{E}^{\rho=|0\rangle\langle 0|}_{\delta\mathsf{SWAP}}(\sigma) = \mathcal{A}_{\sin^2(\delta)} \circ \mathcal{U}_\delta(\sigma) = \mathcal{U}_\delta \circ \mathcal{A}_{\sin^2(\delta)}(\sigma). \tag{H9}$$

The overall effect of one Trotter step of DME_N can therefore be understood as the following: (i) applying the unitary $e^{-i\delta|0\rangle\langle 0|}$ to the system σ, followed by (ii) applying the amplitude-damping channel $\mathcal{A}_{\sin^2\delta}$ to the system σ. Note that because of the condition in Eq. (H8), by flipping the order of steps (i) and (ii), we get the same final state.

Now suppose we repeat the above operation N times. That is, we prepare the instruction qubit in state $\rho = |0\rangle\langle 0|$, couple it to σ via the unitary $e^{-i\mathrm{SWAP}\delta}$, then discard the instruction qubit and prepare it again in state $|0\rangle\langle 0|$, and repeat the above procedure with N different copies of ρ. Then, using Eq. (H8) one can show that, given an initial state σ, the final state of the target system will be

$$\begin{aligned}[\mathcal{E}^{\rho=|0\rangle\langle 0|}_{\delta\mathsf{SWAP}}]^N(\sigma) &= [\mathcal{A}_{\sin^2(\delta)} \circ \mathcal{U}_\delta]^N(\sigma) \\ &= \mathcal{A}^N_{\sin^2(\delta)} \circ \mathcal{U}_{N\delta}(\sigma).\end{aligned} \tag{H10}$$

Since amplitude-damping channels are closed under composition, we see that

$$\mathcal{A}^N_{\sin^2(\delta)} = \mathcal{A}_{1-\cos^{2N}(\delta)}. \tag{H11}$$

Therefore, the overall effect on the target system is equivalent to applying the perfect unitary $e^{-iN\delta|0\rangle\langle 0|}$, and then applying the amplitude-damping channel $\mathcal{A}_{1-\cos^{2N}(\delta)}$.

Now, suppose in the above procedure, instead of state $|,\rangle\langle 0|$ we prepare the instruction qubit in state $|\phi\rangle\langle\phi| = V|0\rangle\langle 0|V^\dagger$, where V is an arbitrary unitary. Then, using the fact that $\mathrm{SWAP}(V\otimes V') = (V'\otimes V)\mathrm{SWAP}$, one can show that the overall effect of this transformation on the target system can be described as a unitary rotation $e^{-iN\delta|\phi\rangle\langle\phi|}$ followed by an amplitude-damping channel in the basis defined by state $|\phi\rangle$ and its orthogonal state.

To translate explicitly to the language of the main text, let $\delta = \theta/N$ and $\rho = |\phi\rangle\langle\phi|$, and use the above procedure to implement the unitary $e^{-i\rho\theta}$ on the target system σ, using N copies of the instruction state ρ. From Eq. (H11), we find that the overall error in this procedure is determined by the probability $p_N = 1-\cos^{2N}(\delta)$. Then, for $\delta \in (0, 2\pi]$ and $N \gg 1$, we have

$$p_N = 1-\cos^{2N}\left(\frac{\theta}{N}\right) \approx 1 - e^{-\frac{\theta^2}{N}} \approx \frac{\theta^2}{N} \quad \text{for large } N. \tag{H12}$$

In the limit of large N, this corresponds to an algorithmic error for the DME_N algorithm of $\mathcal{O}(\theta^2/N)$, as quoted in the main text.

APPENDIX I: ALGORITHMIC ERROR DUE TO QME

Here we provide an intuitive picture for the QME operation as well as a formal proof of the modified algorithmic error bound in Eq. (3) of the main paper.

We build the intuition for this section by returning to the concrete example from Appendix H, i.e., the instruction qubit prepared in $\rho = |0\rangle\langle 0|$. We also suppose that the target qubit is prepared in an orthogonal state, say, $\sigma = |+i\rangle\langle +i|$ (which is an eigenstate of the Pauli matrix $\hat{\sigma}_Y$). Since δSWAP is a symmetric operation by the logic in Appendix H, the state of ρ following a small δSWAP interaction is given by a rotation about the y axis followed by an amplitude-damping channel (which we neglect for the moment). In this case, the state of the instruction qubit becomes

$$\rho' = \begin{pmatrix} \cos^2(\delta) & -\cos(\delta)\sin(\delta) \\ -\cos(\delta)\sin(\delta) & \sin^2(\delta) \end{pmatrix}. \tag{I1}$$

The trace distance between ρ and ρ' is of order $|\delta|$. However, if we measure and forget the state of the instruction qubit in the basis of its original quantization axis (i.e., the z basis), the coherent off-diagonal components of the density matrix are dephased, and we are left with

$$\rho'' = \begin{pmatrix} \cos^2(\delta) & 0 \\ 0 & \sin^2(\delta) \end{pmatrix}. \tag{I2}$$

The trace distance between ρ'' and ρ is of order δ^2. Because DME operates in the $\delta \ll 1$ regime, we have $\delta^2 \ll \delta$. Measuring and forgetting therefore leaves the instruction qubit in a slightly perturbed state that is closer to that of the initial state ρ.

The intuition developed for $\rho = |0\rangle\langle 0|$ extends naturally to an arbitrary initial state $\rho = |\nu_\parallel\rangle\langle\nu_\parallel|$ in a basis defined by $\nu = \{|\nu_\parallel\rangle, |\nu_\perp\rangle\}$. A small arbitrary rotation will result in the state

$$\begin{aligned}\rho' = {} & \cos^2(\beta)|\nu_\parallel\rangle\langle\nu_\parallel| + \sin^2(\beta)|\nu_\perp\rangle\langle\nu_\perp| \\ & + \cos(\beta)\sin(\beta)(e^{i\phi}|\nu_\parallel\rangle\langle\nu_\perp| + e^{-i\phi}|\nu_\perp\rangle\langle\nu_\parallel|),\end{aligned} \tag{I3}$$

where β and ϕ generically parametrize the rotation. A measurement in the basis ν dephases the off-diagonal elements in this basis, leaving

$$\rho'' = \cos^2(\beta)|\nu_\parallel\rangle\langle\nu_\parallel| + \sin^2(\beta)|\nu_\perp\rangle\langle\nu_\perp|, \tag{I4}$$

which is closer than ρ' to ρ by a factor of $|\beta|$.

Performing a physical measurement along an arbitrary axis ν generically would require (i) rotating ν onto the z axis, (ii) performing a projective readout, and (iii) rotating back to the original axis. All of these steps require finite clock time: Single-qubit gates (measurements) typically require tens (hundreds) of nanoseconds to complete. We would like to avoid this significant experimental overhead while still maintaining the ability to partially restore the instruction qubit to its initial state. Instead of physically performing the measurement, we can apply the unitaries $\{\hat{\sigma}_\mathbb{1}, \hat{\sigma}_\nu\}$ with equal probabilities, where $\hat{\sigma}_\nu = \hat{n}_\parallel \cdot (\hat{\sigma}_X, \hat{\sigma}_Y, \hat{\sigma}_Z)$ and $\hat{n}_\parallel$ is a unit vector parallel to ρ. Such protocols may be equivalently thought of as an approach to turning a coherent error into an incoherent error along a known axis. This protocol is the QME operation used in the main paper.

When averaged over many iterations, the randomized QME operation dephases the system in the ν basis, just as in Eqs. (I3) and (I4). Assuming the instruction qubit is initially in state ρ', it turns out that the resulting state is the same for measurement and random gate application, i.e.,

$$\begin{aligned}&\frac{1}{2}(|\nu_\parallel\rangle\langle\nu_\parallel|\rho'|\nu_\parallel\rangle\langle\nu_\parallel| + |\nu_\perp\rangle\langle\nu_\perp|\rho'|\nu_\perp\rangle\langle\nu_\perp|) \\ &\quad = \frac{\hat{\sigma}_\mathbb{1}\rho'\hat{\sigma}_\mathbb{1} + \hat{\sigma}_\nu\rho'\hat{\sigma}_\nu}{2} = \frac{1}{2\pi}\int_0^{2\pi} d\gamma e^{-i\gamma\hat{\sigma}_\nu}\rho' e^{i\gamma\hat{\sigma}_\nu}.\end{aligned} \tag{I5}$$

These three terms represent, respectively, measuring and forgetting, random gate application, and phase randomization. Their equivalence can be understood more formally from the standpoint of the stochastic master equation, to which Ref. [48] provides an accessible introduction.

This approach is also related to the quantum Zeno effect, in which persistent measurement along an axis of interest "pins" the qubit state to that axis by continuously dephasing any rotations away from it [49].

Finally, we calculate the additional error introduced to the DME algorithm by the use of QME. For this, we return to the specific case where $\rho = |0\rangle\langle 0|$ (though this also generalizes to arbitrary ρ). As in Appendix H, we apply the unitary $e^{-i\mathrm{SWAP}\delta}$ to the joint state $\sigma \otimes |0\rangle\langle 0|$, and then randomly apply one of the unitaries $\{\hat{\sigma}_{\mathbb{1}}, \hat{\sigma}_Z\}$ to the instruction qubit. Then, it can be shown that the total state of instruction and target qubit is given by

$$\begin{aligned}&\frac{1}{2}(e^{-i\mathrm{SWAP}\delta}[\sigma \otimes |0\rangle\langle 0|]e^{i\mathrm{SWAP}\delta} \\ &\quad + (\hat{\sigma}_{\mathbb{1}} \otimes \hat{\sigma}_Z)e^{-i\mathrm{SWAP}\delta}[\sigma \otimes |0\rangle\langle 0|]e^{i\mathrm{SWAP}\delta}(\hat{\sigma}_{\mathbb{1}} \otimes \hat{\sigma}_Z)) \\ &= \underbrace{\mathcal{E}^{\rho=|0\rangle\langle 0|}_{\delta\mathrm{SWAP}}(\sigma) \otimes |0\rangle\langle 0|}_{\mathrm{DME}} - \underbrace{\sin^2(\delta)\langle 1|\sigma|1\rangle[|0\rangle\langle 0| \otimes \hat{\sigma}_Z]}_{\mathrm{QME\ error}},\end{aligned}$$

where $\mathcal{E}^{\rho=|0\rangle\langle 0|}_{\delta\mathrm{SWAP}}(\sigma)$ is the quantum channel defined in Eq. (H9). Note that the first term $\mathcal{E}^{\rho=|0\rangle\langle 0|}_{\delta\mathrm{SWAP}}(\sigma) \otimes |0\rangle\langle 0|$ is exactly the desired state which can be used for the next round of DME. On the other hand, the second term $\sin^2(\delta)\langle 1|\sigma|1\rangle[|0\rangle\langle 0| \otimes \hat{\sigma}_Z]$ can be treated as an error. To find the contribution of this term in the total error, we use the fact that the trace norm is nonincreasing under any trace-preserving quantum operation $\mathcal{F}$: $\|\mathcal{F}(X)\|_{\mathrm{tr}} \leq \|X\|_{\mathrm{tr}}$, where $\|\cdot\|_{\mathrm{tr}}$ is the trace norm, i.e., the sum of the absolute value of the eigenvalues of the operator.

For the second term in Eq. (I6), we have

$$\begin{aligned}&\|\sin^2(\delta)\langle 1|\sigma|1\rangle[|0\rangle\langle 0| \otimes \hat{\sigma}_Z]\|_{\mathrm{tr}} \\ &\quad = 2\sin^2(\delta)\langle 1|\sigma|1\rangle \leq 2\sin^2(\delta). \qquad \text{(I6)}\end{aligned}$$

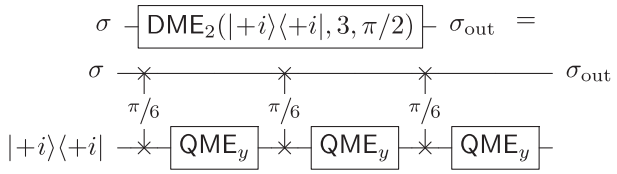
Therefore, the additional error introduced by each application of QME is bounded by $2\sin^2(\delta)$. Repeating this process N times, and using the triangle inequality for the trace norm, we find that the distance between the final total system state and the state produced by DME is bounded by $2N\sin^2(\delta)$. Choosing $\delta = \theta/N$, we find that the overall additional error introduced by the use of QME is bounded by

$$2N\sin^2(\delta) = 2N\sin^2\left(\frac{\theta}{N}\right) \leq \frac{2\theta^2}{N}. \qquad \text{(I7)}$$

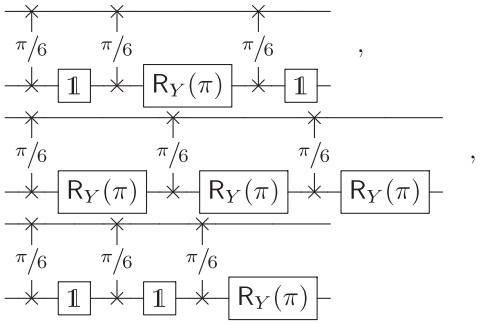
The right-hand side of Eq. (I7) is the QME-induced-error contribution cited in the main text.

APPENDIX J: QUANTIFYING THE IMPACTS OF FINITE QME RANDOMIZATIONS

To properly implement the probabilistic nature of the QME operation, we instantiate each DME_2 circuit a number of times. Consider as an example the $N = 3$ version of the DME_2 circuit from Fig. 3,



In this case, each QME presents a random choice between applying $R_Y(\pi)$ or 1 at each occurrence. For an N step DME_2 there are 2^N configurations of QME gates. In the experiment, it is infeasible to sample all 2^N realizations, and instead we sample a smaller number denoted r. The circuits below show $r = 3$ random example realizations of the circuit,



In the experiment, a total r_{QME} of circuits are executed, providing a sample from which we can extract average properties. The generic process for extracting average properties over r instantiations is sketched in Fig. 12.

From the datasets used in the main paper, we can also explore algorithmic behavior as the randomizations of QME increase toward the central limit. In Figs. 12(b)–12(d), we plot three relevant figures of merit as a function of r and N for the $\theta = \pi$ dataset of Fig. 3 in the main text. Figure 12 shows the evolution of the state fidelity of the output state as a function of r. For all values of N, we observe that after approximately 50 randomizations, the effect of introducing more circuits with random choices of QME gates does not significantly alter the result. Figure 12(c) shows the concurrence of the two-qubit density matrix, a measurement of bipartite entanglement in the system [50]. After just a few randomizations $r > 10$, concurrence goes to zero, indicating that (quantum) correlations are suppressed, as expected. There may also be classical correlations between the σ and ρ subsystems. In Fig. 12(d), we therefore plot the mutual information $I(\sigma, \rho)$ between each subsystem, where

$$I_\Omega(\sigma, \rho) = S(\mathrm{Tr}_\sigma(\Omega)) + S(\mathrm{Tr}_\rho(\Omega)) - S(\Omega) \qquad \text{(J1)}$$

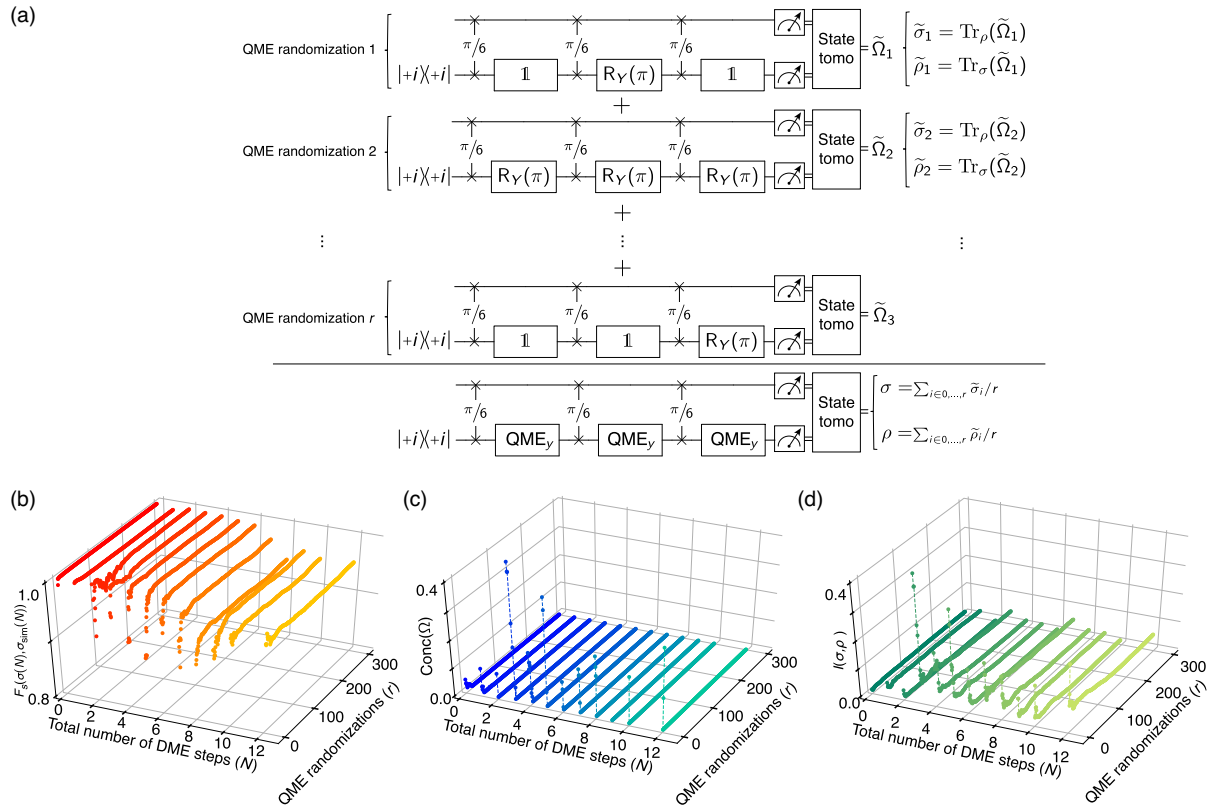


FIG. 12. (a) Schematic definition of experimental execution of a DME protocol using QME operations (i.e., DME_2). (b) The state fidelity between the measured output state and the result of ideal gates implementing DME, as the number of QME randomizations are increased. (c) Concurrence in the two-qubit density matrix Ω (the combined state of the system) for increasing number of QME randomizations. (d) The mutual information between the two subsystems σ and ρ, as more randomizations of QME are used.

is the mutual information, and $S(\Omega) = -\mathrm{Tr}(\Omega \ln \Omega)$ is the von Neumann entropy of the density matrix Ω. Here we again observe that after $r > 10$, any correlations between the subsystems are effectively removed.

[1] W. K. Wooters and W. H. Zurek, *A Single Quantum Cannot Be Cloned*, Nature (London) **299**, 802 (1982).

[2] S. Lloyd, M. Mohseni, and P. Rebentrost, *Quantum Principal Component Analysis*, Nat. Phys. **10**, 631 (2014).

[3] I. Marvian and S. Lloyd, *Universal Quantum Emulator*, arXiv:1606.02734.

[4] S. Kimmel, C. Y.-Y. Lin, G. H. Low, M. Ozols, and T. J. Yoder, *Hamiltonian Simulation with Optimal Sample Complexity*, npj Quantum Inf. **3**, 13 (2017).

[5] J. Haah, A. W. Harrow, Z. Ji, X. Wu, and N. Yu, *Sample-Optimal Tomography of Quantum States*, IEEE Trans. Inf. Theory **63**, 5628 (2017).

[6] F. G. S. L. Brando *et al.*, *Quantum SDP Solvers: Large Speed-Ups, Optimality, and Applications to Quantum Learning*, arXiv:1710.02581.

[7] Z. Li *et al.*, *Resonant Quantum Principal Component Analysis*, arXiv:2104.02476.

[8] H. Pichler, G. Zhu, A. Seif, P. Zoller, and M. Hafezi, *Measurement Protocol for the Entanglement Spectrum of Cold Atoms*, Phys. Rev. X **6**, 041033 (2016).

[9] P. Rebentrost, M. Mohseni, and S. Lloyd, *Quantum Support Vector Machine for Big Data Classification*, Phys. Rev. Lett. **113**, 130503 (2014).

[10] M. Kjaergaard, M. E. Schwartz, J. Braumüller, P. Krantz, J. I.-J. Wang, S. Gustavsson, and W. D. Oliver, *Superconducting Qubits: Current State of Play*, Annu. Rev. Condens. Matter Phys. **11**, 369 (2020).

[11] A. Greene *et al.*, *Error Mitigation via Stabilizer Measurement Emulation*, arXiv:2102.05767.

[12] Y. Salathe, M. Mondal, M. Oppliger, J. Heinsoo, P. Kurpiers, A. Potocnik, A. Mezzacapo, U. LasHeras, L. Lamata, E. Solano, S. Filipp, and A. Wallraff, *Digital Quantum Simulation of Spin Models with Circuit Quantum Electrodynamics*, Phys. Rev. X **5**, 021027 (2015).

[13] S. Lloyd, *Universal Quantum Simulators*, Science **273**, 1073 (1996).

[14] P. Magnard, P. Kurpiers, B. Royer, T. Walter, J. C. Besse, S. Gasparinetti, M. Pechal, J. Heinsoo, S. Storz, A. Blais, and A. Wallraff, *Fast and Unconditional All-Microwave Reset of a Superconducting Qubit*, Phys. Rev. Lett. **121**, 060502 (2018).

[15] J. J. Wallman and J. Emerson, *Noise Tailoring for Scalable Quantum Computation via Randomized Compiling*, Phys. Rev. A **94**, 052325 (2016).

[16] J. Koch, T. M. Yu, J. Gambetta, A. A. Houck, D. I. Schuster, J. Majer, A. Blais, M. H. Devoret, S. M. Girvin, and R. J. Schoelkopf, *Charge-Insensitive Qubit Design Derived from the Cooper Pair Box*, Phys. Rev. A **76**, 042319 (2007).

[17] M. D. Hutchings, J. B. Hertzberg, Y. Liu, N. T. Bronn, G. A. Keefe, M. Brink, J. M. Chow, and B. L. T. Plourde, *Tunable Superconducting Qubits with Flux-Independent Coherence*, Phys. Rev. Applied **8**, 044003 (2017).

[18] R. Barends, J. Kelly, A. Megrant, D. Sank, E. Jeffrey, Y. Chen, Y. Yin, B. Chiaro, J. Mutus, C. Neill *et al.*, *Coherent Josephson Qubit Suitable for Scalable Quantum Integrated Circuits*, Phys. Rev. Lett. **111**, 080502 (2013).

[19] F. W. Strauch, P. R. Johnson, A. J. Dragt, C. J. Lobb, J. R. Anderson, and F. C. Wellstood, *Quantum Logic Gates for Coupled Superconducting Phase Qubits*, Phys. Rev. Lett. **91**, 167005 (2003).

[20] L. DiCarlo *et al.*, *Demonstration of Two-Qubit Algorithms with a Superconducting Quantum Processor*, Nature (London) **460**, 240 (2009).

[21] J. M. Martinis and M. R. Geller, *Fast Adiabatic Qubit Gates Using Only Sigmaz Control*, Phys. Rev. A **90**, 022307 (2014).

[22] M. A. Rol, F. Battistel, F. K. Malinowski, C. C. Bultink, B. M. Tarasinski, R. Vollmer, N. Haider, N. Muthusubramanian, A. Bruno, B. M. Terhal, and L. DiCarlo, *Fast, High-Fidelity Conditional-Phase Gate Exploiting Leakage Interference in Weakly Anharmonic Superconducting Qubits*, Phys. Rev. Lett. **123**, 120502 (2019).

[23] CIRQ-v0.5.0, a PYTHON library for writing, manipulating, and optimizing quantum circuits and running them against quantum computers and simulators, https://github.com/quantumlib/cirq.

[24] F. Vatan and C. Williams, *Optimal Quantum Circuits for General Two-Qubit Gates*, Phys. Rev. A **69**, 032315 (2004).

[25] M. M. Wilde, *Quantum Information Theory*, 2nd ed. (Cambridge University Press, Cambridge, England, 2017).

[26] R. Rines, K. Obenland, and I. Chuang, *Empirical Determination of the Simulation Capacity of a Near-Term Quantum Computer*, arXiv:1905.10724.

[27] A. N. Korotkov, *Error Matrices in Quantum Process Tomography*, arXiv:1309.6405.

[28] Y. Y. Gao, B. J. Lester, K. S. Chou, L. Frunzio, M. H. Devoret, L. Jiang, S. M. Girvin, and R. J. Schoelkopf, *Entanglement of Bosonic Modes through an Engineered Exchange Interaction*, Nature (London) **566**, 509 (2019).

[29] J. Preskill, *Quantum Computing in the NISQ Era and Beyond*, Quantum **2**, 79 (2018).

[30] A. Kandala *et al.*, *Hardware-Efficient Variational Quantum Eigensolver for Small Molecules and Quantum Magnets*, Nature (London) **549**, 242 (2017).

[31] P. V. Klimov, J. Kelly, Z. Chen, M. Neeley, A. Megrant, B. Burkett, R. Barends, K. Arya, B. Chiaro, Y. Chen *et al.*, *Fluctuations of Energy-Relaxation Times in Superconducting Qubits*, Phys. Rev. Lett. **121**, 090502 (2018).

[32] J. J. Burnett, A. Bengtsson, M. Scigliuzzo, D. Niepce, M. Kudra, P. Delsing, and J. Bylander, *Decoherence Benchmarking of Superconducting Qubits*, npj Quantum Inf. **5**, 54 (2019).

[33] 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. **6**, 021318 (2019).

[34] J. Kelly, R. Barends, B. Campbell, Y. Chen, Z. Chen, B. Chiaro, A. Dunsworth, A. G. Fowler, I. C. Hoi, E. Jeffrey *et al.*, *Optimal Quantum Control Using Randomized Benchmarking*, Phys. Rev. Lett. **112**, 240504 (2014).

[35] W. D. Oliver, Y. Yu, J. C. Lee, K. K. Berggren, L. S. Levitov, and T. P. Orlando, *Mach-Zehnder Interferometry in a Strongly Driven Superconducting Qubit*, Science **310**, 1653 (2005).

[36] R. Barends *et al.*, *Superconducting Quantum Circuits at the Surface Code Threshold for Fault Tolerance*, Nature (London) **508**, 500 (2014).

[37] A. Chen, Ph.D. thesis, University of California, Santa Barbara, 2018.

[38] J. Wallman, C. Granade, R. Harper, and S. T. Flammia, *Estimating the Coherence of Noise*, New J. Phys. **17**, 113020 (2015).

[39] T. Proctor, K. Rudinger, K. Young, M. Sarovar, and R. Blume-Kohout, *What Randomized Benchmarking Actually Measures*, Phys. Rev. Lett. **119**, 130502 (2017).

[40] J. J. Wallman, *Randomized Benchmarking with Gate-Dependent Noise*, Quantum **2**, 47 (2018).

[41] M. A. Nielsen, *A Simple Formula for the Average Gate Fidelity of a Quantum Dynamical Operation*, Phys. Lett. A **303**, 249 (2002).

[42] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*, 10th anniversary ed. (Cambridge University Press, New York, 2011).

[43] D. C. McKay, C. J. Wood, S. Sheldon, J. M. Chow, and J. M. Gambetta, *Efficient Z Gates for Quantum Computing*, Phys. Rev. A **96**, 022330 (2017).

[44] J. M. Chow, L. DiCarlo, J. M. Gambetta, A. Nunnenkamp, L. S. Bishop, L. Frunzio, M. H. Devoret, S. M. Girvin, and R. J. Schoelkopf, *Detecting Highly Entangled States with a Joint Qubit Readout*, Phys. Rev. A **81**, 062325 (2010).

[45] K. Banaszek, G. M. D'Ariano, M. G. A. Paris, and M. F. Sacchi, *Maximum-Likelihood Estimation of the Density Matrix*, Phys. Rev. A **61**, 010304(R) (1999).

[46] G. C. Knee, E. Bolduc, J. Leach, and E. M. Gauger, *Quantum Process Tomography via Completely Positive and Trace-Preserving Projection*, Phys. Rev. A **98**, 062336 (2018).

[47] B. Efron, *Bootstrap Methods: Another Look at the Jackknife*, Ann. Stat. **7**, 1 (1979).

[48] K. Jacobs and D. A. Steck, *A Straightforward Introduction to Continuous Quantum Measurement*, Contemp. Phys. **47**, 279 (2006).

[49] W. M. Itano, D. J. Heinzen, J. Bollinger, and D. Wineland, *Quantum Zeno Effect*, Phys. Rev. A **41**, 2295 (1990).

[50] W. K. Wootters, *Entanglement of Formation of an Arbitrary State of Two Qubits*, Phys. Rev. Lett. **80**, 2245 (1998).